THE ECLECTIC POWER COMPANY

USER MANUAL FOR
HUGGINS' EMAIL FORM SCRIPT

Publication #5-01-063

Rev. 28.Sep.2013 for Version 2.2.5

James S. Huggins
The Eclectic Power Company

Huggins' Email Form Script
Publication Number 5-01-063, revised 28.Sep.2013 for Version 2.2.5

NB: Small revisions to the script (e.g., bug fixes) may not necessitate a new version of the documentation. Check the website to determine current status of scripts and documentation.

To obtain the current version of the script and documentation, go to
www.JamesSHuggins.com/huggins-email-form-script

Printed and bound by Eclectic Power Press, an imprint of The Eclectic Power Company.

Eclectic Power Press
11150 Beamer Road, Suite 109-F3P
Houston, Texas 77089-2331
www.EclecticPowerPress.com

Limits of Liability and Disclaimer of Warranty
This publication is distributed with the understanding that neither the publisher nor the author is engaged in rendering legal, accounting, investment, financial or other professional services. Readers are advised to see proper licensed professionals for all such services. Investments in software, marketing techniques and business is risky. You could lose your money.

Spam Practices and Privacy
Neither the author nor publisher endorse or condone spam email, search engine spam techniques or personal privacy abuse. Both the author and publisher specifically recommend against all such practices.

What If I Don't Want to Read This Huge Manual?

Many computer users, particularly more advanced computer users, don't want to read the documentation.

Several have written to tell me they **never** read the manual.

Are you like this?

If you are, then **just for you** I have rewritten, and reorganized the manual to reduce the amount you need to read and the time it will take.

Very Important

You do **not** have to read the **whole** manual.

You can stop when you get to the tiny notice section titled "You Have Finished the Quick Start"; that section is on page 52.

But please start here and keep going for at least this small first part.
Do **not** skip anything.

At least once each week I help someone online or on the phone who is having problems because they skipped stuff.

I have worked hard to make this as fast as possible to get started. It will pay off if you follow these directions and work thru the process.

For New Users

1. **Read and work — not *scan* — the front part of this manual.**

Begin here and continue through the section titled "You Have Finished the Quick Start" that begins on page 52.

Do **not** skip anything. **Do** the exercises.

These sections will move you **quickly** through testing your web host and then setting up a simple but highly functional form for your site. Many people find that it is "enough" and they only need to slightly modify this work to achieve the form on their site.

Skipping sections will **not** speed you up. It **will** slow you down.

2. **Then, scan the entire manual, front to back.**

After you get your feet wet, and get a basic form working, take the time to scan the entire manual, front to back. Scanning the entire manual will give you the details to fill in the concepts, and will help you understand what additional options are available to you.

IF YOU ALREADY HAVE A WORKING VERSION

1. **Note: Only do this if you already have this form script working on your website.**

Do *not* take these shortcuts just because you are an experienced web-person.

If this is your first implementation of this script, follow the section "For New Users", above.

2. **See the New Version Notes.**

See A Few Notes on this New Version on page 53.

3. **See the thoughts on future versions.**

See Pending Changes on page 56.

4. **Then, if you want, *scan* the entire manual, front to back.**

WHATEVER YOU DO, DON'T MODIFY THE PROGRAM

I have worked hard to make the program "parameter driven". If you change the program itself (instead of just changing the parameters), you will get to make those same changes every single time I make and release a new version!

This is not a legal issue; it is a practical one. You have the legal right to modify if you wish. But, so far, **no one** has shown me anything that **requires** a modification.

You should be able to get the program to do it all just by changing the parameters.

Then when I release a better/improved version you won't have to make the changes all over again.

If you believe that is **not** true, please write me before you head down the wrong path:

<p align="center">JHuggins@JamesSHuggins.com
or
JHuggins@EclecticPower.com.</p>

CHANGES, FIXES AND ENHANCEMENTS

From time to time I make changes to the script and to the documentation of the script. Some of these changes are for problem fixes. Version 1.2, for example, was a response to user notice of a problem working on some servers with some configurations of PHP. Other changes may be enhancements. And still other changes may be just documentation changes, correcting typos and improving the information.

It is critical for you and other Users of Huggins' Email Form Script to know promptly about these changes. For this reason, I created the *HEFS Bulletin* to keep users informed.

HEFS BULLETIN

The *HEFS Bulletin* is a "self-hosted" ezine, powered by Gammadyne Mailer.[1]

I publish it ***very*** irregularly, as needed, to alert subscribers to changes, fixes and enhancements to Huggins Email Form Script, and related documentation and support files.

SUBSCRIBING TO *HEFS BULLETIN*

To subscribe to *HEFS Bulletin* go to:

www.JamesSHuggins.com/hefs-bulletin

YOU NEED TO SUBSCRIBE, RIGHT NOW

If you haven't done it already, please do it right now, before you do anything else. Subscribe to this publication. It is critical that you subscribe in order to ensure that you receive critical updates. Please. Do this now. Right now. Not 10 minutes from now. Right now. Please.

Go here and subscribe:

www.JamesSHuggins.com/hefs-bulletin

---

[1] For more information see Gammadyne Mailer on page 223.

THOROUGH? YES. SIMPLE? YES. SHORT? NO!

This is not a short manual.

In fact one user wrote me to complain.

> You made it anything but simple, and just too much reading stuff and
> repeating the same reading stuff. What a pain. I remember a class I took some
> time ago, my instructor favorite saying was "keep it simple stupid".
> Even your emails go on and on and on.

So, let me just admit it now.

This is **not** a short manual. However, I believe it is a **thorough** manual.

In addition, I have worked to make it also be a **simple** manual.

Thorough? Yes. Simple? Yes. Short? No!

SUGGESTIONS ARE WELCOME.

That said, I'm open to all suggestions. Help me make it better. Just email me:

@JamesSHuggins.co JHuggins m
or
JHuggins@EclecticPower.com.

OVERVIEW OF THIS MANUAL

WHAT IS THIS MANUAL (EBOOK)?

This manual (ebook) provides documentation for a PHP[2] script known as Huggins' Email Form Script. This script processes data from web forms and sends emails.

This documentation provides the:

- Background and purpose of the script
- Full documentation of the script
- Licensing information

It also includes:

- Sample setup for a simple Web Form
- A step-by-step process to using some of the more advanced script features
- Extensive information on each available parameter
- Full information on each error message
- A discussion of web spiders and spambots
- A FAQ
- Troubleshooting information
- Upgrade information
- License information
- Testimonials
- A list of relevant web resources
- A glossary

WHAT THIS MANUAL (EBOOK) IS NOT

This manual (ebook) does not provide extensive tutorial information on either HTML or on HTML Forms. Instead, this manual (ebook) presumes that the reader is familiar with the underlying fundamentals of creating a Web Form.

---

[2] PHP is one of the programming languages used to write scripts like this. Other languages for scripts include Perl (.pl) and ASP (.asp).

For those readers unfamiliar with Web Forms, I have written a *separate* manual (ebook) tutorial entitled *Creating HTML Email Forms*.

This *separate* manual (ebook) tutorial on creating forms is available in two ways:

- By purchase ($20.00[3]) from JSH.us/chef or JamesSHuggins.com/creating-html-email-forms
- For free, to students of Cricket's Tech Training Class at JSH.us/cricket-tech-training or tech.groups.yahoo.com/group/tech-training[4]

WHAT IS HUGGINS' EMAIL FORM SCRIPT?

When you go to a webpage and fill out a Web Form and then click "Submit", some software on the web server must process the data you typed in and do something with that data.

Sometimes that data is sent to someone (e.g., the site owner) as an email. These types of "contact forms" are frequently used on websites to avoid putting in a real email address. One reason is that web spiders (also called spambots) travel the web looking for email addresses to add to spam lists. Using a Web Form to send the email can hide the email address from these spiders and reduce spam.

Sometimes that data is also emailed back to the person who filled out the Web Form as a confirmation or echo just to show that the data was processed, to confirm (echo) the information entered on the Web Form, and to provide additional information (e.g., that the Form User can expect a reply within a day or two).

Huggins' Email Form Script supports these functions.

DISTRIBUTION

The Version 2.2.5 distribution of this manual (ebook) and software is available from:

www.JamesSHuggins.com/huggins-email-form-script

---

[3] The price may have changed. This price is effective 28.Sep.2013. Be sure to see the website for the current pricing.

[4] Cricket's group may have "gone away". I'm researching it now.

CONTENTS

NB: I recommend you check your distribution to be certain it contains these contents.

The Version 2.2.5 distribution includes the following files (listed in alphabetical order):

- **0 READ ME FIRST V2.2.5.txt**
  This is the "Read Me First" file. Always read the "Read Me First" file in any distribution for the latest information. The primary documentation may not yet have been updated with information in the "Read Me First" file.

- **0 READ ME SECOND V2.2.5.txt**
  This "Read Me Second" file is my not completely successful attempt to get people to sign up for the HEFS Bulletin. Please, please, please sign up for the bulletin. This is very important.

- **0 READ ME THIRD V2.2.5.txt**
  Some people immediately start modifying, or trying to modify, the program. This is not the idea. You should be able to do all this without modifying the program. This notice encourages you not to modify the program.

- **check-info.php.txt**
  This text file contains PHP code to have your server display PHP installation/configuration information."

  - ✓ **NB**: the file is distributed as a .txt file. Please rename and remove the final ".txt" giving a new name of "check-info.php".

- **HEFS Acknowledgment Link V2.2.5.txt**
  This text file contains HTML for an acknowledgement link named you can paste on your web page. See also *HEFS Acknowledgement Notice V2.2.5.txt* below.

- **HEFS Acknowledgment Notice V2.2.5.txt**
  This text file contains HTML you can place into your web page to document your use of the HEFS Huggins' Email Form Script. See also *HEFS Acknowledgement Link V2.2.5.txt* above.

- **hefs-example-1.htm.txt**
  This text file contains HTML which I use for a page on my website: http://www.jamesshuggins.com/h/hefs/hefs-example-1.htm
  I include this only so you will have it if you want it. I do not use this for anything directly in this documentation.

- **HEFS Form Statement and Parms for HTML - V2.2.5.txt, HEFS Form Statement and Parms for XHTML - V2.2.5.txt**
  These text files contain the HTML and XHTML form statement and hidden field parameters you can copy/paste into your form. The two versions are for the HTML and XHTML versions. These two are equivalent except that the one for XHTML includes the closing slashes required by that format and the one for HTML does not. These files are not used "as is", but instead provide code to be used through a copy/paste and then modified.

- **hefs-mail-test-v2.2.5.htm.txt**
  This text file contains HTML for a minimal PHP Test page. We use this to help you test your server.

  - ✓ **NB**: the file is distributed as a .txt file. Please rename and remove the final ".txt".

hefs-simple-form-page-1a for HTML V2.2.5.htm.txt, hefs-simple-form-page-1b for HTML V2.2.5.htm.txt, hefs-simple-form-page-2a for XHTML V2.2.5.htm.txt hefs-simple-form-page-2b for XHTML V2.2.5.htm.txt
These are simple, "ready to use" pages using either HTML or XHTML. These are used to help you get started quickly. These four are equivalent except that the "1" files for HTML do not include the closing slashes required by the XHTML format, while the "2" files for XHTML do include the closing slashes. In addition the "a" files provide for first and last name while the "b" files only provide for a single unified name. These files are not used "as is", but instead provide an initial starting point for you.

- **hefs-simple-form-page-3a for HTML-with-CSS V2.2.5.htm.txt, hefs-simple-form-page-3b for HTML-with-CSS V2.2.5.htm.txt, hefs-simple-form-page-4a for XHTML-with-CSS V2.2.5.htm.txt, hefs-simple-form-page-4b for XHTML-with-CSS V2.2.5.htm.txt,**
  These pages are based on contributions to the community by Jonathan Lewis (http://www.LewisPhillips.com). They also use concepts/code from the article at http://www.sitepoint.com/style-web-forms-css . I encourage a review of this article. They add CSS to my pages to give you other options. These four are equivalent except that the "3" files for HTML do not include the closing slashes required by the XHTML format, while the "4" files for XHTML do include the closing slashes. In addition the "a" files provide for first and last name while the "b" files only provide for a single unified name. These files are not used "as is", but instead provide an initial starting point for you.

- **hefs-simple-form-page-5a for HTML V2.2.5.htm.txt,**
  hefs-simple-form-page-5b for HTML-with-CSS V2.2.5.htm.txt,
  hefs-simple-form-page-6a for XHTML V2.2.5.htm.txt
  hefs-simple-form-page-6b for XHTML-with-CSS V2.2.5.htm.txt
  These pages are **also** based on contributions to the community by Jonathan Lewis (http://www.LewisPhillips.com). They also use concepts/code from the article at http://www.sitepoint.com/style-web-forms-css . I encourage a review of this article. They add CSS to my pages to give you other options. These four are equivalent except that the "5" files for HTML do not include the closing slashes required by the XHTML format, while the "6" files for XHTML do include the closing slashes. In addition the "a" files provide for first and last name while the "b" files only provide for a single unified name. These files are not used "as is", but instead provide an initial starting point for you.

- **Host Issues for Huggins' Email Form Script V2.2.5 9-01-060.pdf**
  This PDF file is an excerpt of the "Host Issues" chapter of the User Manual and was created to permit individuals to quickly explore the host issues without needing to download and cope with the whole User Manual.

- **huggins-email-form-script-v2.2.5.php5.txt**
  This is the actual script … a PHP file.

  ✓ **NB**: the file is distributed with a .txt suffix. Please rename by removing the final ".txt".

- **pqy-v2.2.5.php.txt**
  This text file contains PHP code for a minimal PHP Test Script. We use this to help you test your server.

  ✓ **NB:** the file is distributed as a .txt file. Please rename by removing the final ".txt".

- **Simple Form Page Samples.pdf**
  This PDF file contains displays of the 12 different sample html files.

- **User Manual for Huggins' Email Form Script V2.2.5 5-01-063.pdf**
  This PDF file contains the documentation (this manual).

NOTES ON THE SAMPLE HTML

The sample HTML *in this manual* is not "strict", and is not XHTML. As such, there are no self-closing tags. If your page uses XHTML *you* should add the self-closing tags.

The script package contains two different files:

- "HEFS Form Statement and Parameters V2.2.5.txt"
- "HEFS Form Statement and Parameters V2.2.5 for XHTML.txt"

These two different files contain versions for HTML and HTML that may be copied into your form page.

YOUR FEEDBACK IS WELCOME

This manual (ebook) is an evolutionary document.

If you believe I could improve this manual (ebook), I welcome your feedback.

Please send your feedback to:

<div align="center">

JHuggins@JamesSHuggins.com

or

JHuggins@EclecticPower.com.

</div>

KEEPING UP TO DATE

Please subscribe to the *HEFS Bulletin* to keep up to date about bugs, fixes and enhancements. For more information see information about *Tracking Fixes and Enhancements* on page 199 189Special Terms Used in This Manual (Ebook). Or just go to JSH.us/hefs-bulletin.

The *HEFS Bulletin* is the only way I have of continually communicating these important bulletins to the script users.

KEY LICENSE TERMS

LICENSE

Huggins' Email Form Script is made available through the GNU General Public License Version 3. To view a copy of this license, visit http://www.gnu.org/licenses/gpl.html.

This is a free license and permits use on commercial websites. It restricts incorporation into some commercial products.

In addition, the specific license for this product requires acknowledgement of the use on the web page.

This documentation is made available through the Creative Commons Attribution-No Derivative Works 3.0 Unported License (3.0) (CreativeCommons.org/licenses/by-nd/3.0). This is a free license but restricts development of derivative works.

This manual (ebook) explains the license in more detail in the section named *The License* beginning on page 200.

ACKNOWLEDGING YOUR USE OF THE SCRIPT

In return for the free use of Huggins' Email Form Script, there are three (3) teeny tiny requirements:

1. Include a visible linked acknowledgement on the Form Page. This acknowledgement should be substantially like:

```
<a title="Huggins' Email Form Script" target="_blank"
href="http://JSH.us/hefs">This form uses Huggins' Email Form Script</a>
```

NB: the Acknowledgement Link file ("HEFS Acknowledgement Link V2.2.5.txt") includes this HTML in an easy to use, copy/paste format to make this easier.[5]

---

[5] The naming scheme of V2.2.5 references the version number of the script. The version number is "A.B.C" where "A" is the primary release number, "B" is the secondary release number and "C" is the tertiary release number. Typically, minor bug fixes update the tertiary release number. Small enhancements and changes update the secondary release number. And major changes update the primary release number. All the files in a download package use a similar numbering scheme to show that they are all part of the package. In general, you should always download and use a complete package of files with similar names. This ensures that the files all work together.

The fonts, sizes, colors and other similar characteristics should be set to compliment the site. You may modify the text or add text at your discretion provided the meaning is substantially the same.

Feel free to adjust the specifics to compliment your page. The only objective is to help someone else find and use this script.

Special Notice 1: the target="_blank" attribute has been deprecated in standard HTML.I continue to use it because (a) it is still broadly supported in the browsers and (b) I don't approve of the deprecation; BUT if you are a standards stickler, just take that out

Special Notice 2: I've been asked whether the phrase "This form uses Huggins' Email Form Script" could be shortened to "Huggins' Email Form Script". Sure. I'm not a stickler for that level of detail. The goal is to help others find the script so this is clearly another acceptable option. Just do the right thing; I'm a pretty agreeable guy.

Placing this linked acknowledgement on the Form Page will help to promote Huggins' Email Form Script. I sincerely appreciate this consideration.

2. Include the following acknowledgement in the HTML code of the Form Page itself:

```
<head>
<!-- Form Credit: This email form uses Huggins' Email Form Script;
   To learn more visit "http://JSH.us/hefs".-->


note: the <head> is just to show where to put this so be careful not to
duplicate that
```

NB: the Acknowledgement Notice file ("HEFS Acknowledgement Notice V2.2.5.txt") includes this HTML in an easy to use, copy/paste format to make this easier. In addition, the sample HTML files contain this notice.

3. Allow the form script to place the bottom of page notices on any error pages produced. To see an example of what I am talking about go to http://clearlaketutoring.com/contact-form.htm and submit the form not filled out to generate an error page.

WHAT IS THIS SCRIPT FOR?

WHAT DO EMAIL FORMS DO?

Email Web Forms on websites have common characteristics:

- They accept input from the Form User
- They assemble this input into emails which they send to predefined addresses
- They may send the Form User a confirmation email

Email Forms are frequently used on websites:

- To provide a way for site visitors to contact the site owner without the use of email,
- To provide structure in the communication, ensuring, or at least encouraging, Form Users to provide the information needed, and
- To protect the site owner's email address from harvesting by web spiders (spambots)

There are other kinds of Web Forms as well. These may, for example,

- Update a database, or
- Process a transaction (e.g., paying a bill through your bank, or ordering a product from a web store).

Huggins' Email Form Script only addresses Web Forms which send emails.

EMAIL FORM PROBLEMS

Achieving these email functions requires either that the Form Designer rely on a predefined script or, alternatively, use a scripting tool like PHP to construct his own script and/or pages.

FrontPage[6] offers FrontPage Server Extensions to process these types of Web Forms.

However, not all FrontPage developers use these extensions and many people use Dreamweaver or some tool other than FrontPage. In addition, Microsoft's successor to FrontPage, Expression Web, does not offer these extensions and does not have "built in" support for processing forms.

---

[6] I *was* a Microsoft MVP for the FrontPage product. I am also a user of the Microsoft Expression Web product. Details are JSH.us/h/web1/microsoft-mvp.htm

Many of the scripts which are available for this type of processing expose the email addresses to web spiders which harvest the email addresses for spam and to spambots which hijack the form/script to send spam.[7]

And, finally, existing scripts may not provide the flexibility the Form Designer requires such as the ability to:

- Format each email,
- Create multiple formats of emails,
- Email multiple people,
- Protect the email addresses from web spiders,
- Protect the form from spambots and
- Validate the input data.

INSPIRATION FOR AND HISTORY OF THE HUGGINS' EMAIL FORM SCRIPT

I wrote Huggins' Email Form Script to address some of these problems.

The inspiration was a discussion of similar problems encountered by fellow FrontPage users in August and September 2005 and discussed on some email lists, including one FrontPage list I co-moderated[8], another FrontPage list[9] on which I was active (and for which I became a co-moderator in Feb 2007) and also lists related to an SEO training class I co-moderated.[10]

To solve these problems I developed and released Version 1.0 of the script in Nov.2005. While it was discussions on FrontPage lists which prompted the original development, I wrote the script to be completely independent of FrontPage or any other proprietary authoring system.

---

[7] Web spiders (also called web crawlers) follow links around the web, looking at web pages for email addresses. These spiders not only find the obvious email addresses, they also have been programmed to identify many of the common attempts to hide email addresses. The issue for Web Forms is that many of the available scripts do not attempt to hide the email addresses in *any* way. Instead, they simply place the email address plainly in a "hidden field" in the form. Web spiders find such email addresses in a heartbeat. Another type of web spider (web crawler) looks for unprotected forms/scripts it can use to send spam directly. One way it does this is by injecting HTML and header fields like "CC:" and "BCC" to get the script to send the spam for it. For more information this, see en.wikipedia.org/wiki/Web_spider and en.wikipedia.org/wiki/Spambot.

[8] See tech.groups.yahoo.com/group/fplist or JSH.us/fplist

[9] See tech.groups.yahoo.com/group/frontpage or JSH.us/frontpagelist2

[10] See www.gnc-web-creations.com/seo-optimization.htm or JSH.us/cricketseo

After I released Version 1.0 of the script in Nov.2005, users began making suggestions for the next version. In Sep.2006 I began making some of those changes. That series of changes resulted in fixing some bugs and in substantial enhancements to the script. Version 2.0 was released in Apr.2007.

Work on the script continued with more enhancements and improvements in Version 2.1.0 released for BETA in Feb.2008. This initial BETA had lots of problems. Personal schedules delayed correction until May.2008 when I released BETA 2 and then in Jun.2008 released BETA 3. The "final" Version 2.1.0 was released 05.Jul.2008. There were a couple small changes. Then, on 09.Nov.2008 I released a change (2.2.0) that renamed a variable. On 26.Dec.2008 I released a new version including a more robust template to help and a minor change in one of the error messages, to give Version 2.2.1 and Manual #53.

On 08.May.2009 I released Version 2.2.2 and Manual #55. This included some documentation revisions for clarity and an improved error message C005: Sub-Parameter Must Be Yes/No, or Maybe Some Other Value on page 167.

On 01.Aug.2010 I released Version 2.2.3 and Manual #59. This included some documentation revisions for clarity and a file name change to reduce confusion.

On 17.Aug.2012 I released Version 2.2.4 and Manual #60. Primarily, this release was intended to release the "fix" necessitated by changes in PHP 5.3.0. These changes have been "tested" for almost 2 years as different hosts have updated and different users have asked me for the fix. In addition, there are minor documentation updates, primarily to improve readability.

Lastly, 28.Sep.2012, I released Version 2.2.5 and Manual #63. Primarily, this release was intended allow me to update some comments within the code to reflect external changes and to provide a better basis for the other changes that have been "in the works" for too long.

GENERAL OBJECTIVES OF THE HUGGINS' EMAIL FORM SCRIPT

Before I began Huggins' Email Form Script, and again when I began to modify it heavily, I set out the objectives for the script:

- Be usable *without knowing PHP* (the underlying scripting language)
- Be usable without making modifications to the script
- Be usable with a minimal knowledge of how Web Forms work
- Be usable just by following the directions
- Hide all email addresses from spiders
- Protect the form/script from spambots
- Allow two different emails to be sent to "static" addresses

- Allow an optional confirmation email sent to the Form User
- Allow the Form Designer some control over the layout of all emails
- Allow each of the three emails to have a different format
- Allow creation/update of a CSV file[11] containing all form data
- Allow the Form Designer to specify the destination URL to display after processing
- Provide validation of the input data

FEATURES OF THE HUGGINS' EMAIL FORM SCRIPT

As the script evolved, it transformed these general objectives into a specific list of features:

- Sends an email containing the values submitted through the Web Form by the Form User.
- Uses "Hidden Field" Parameters to specify all options.
- Checks parameter specifications for configuration errors to help Form Designers correctly specify the parameters (e.g., to be sure that referenced fields actually exist in the form.)
- Sends the email to multiple email addresses.
- "Hides" the email addresses to stop web spider email harvesting.
- Provides multiple protections against abuse by spambots, including:
  - max size checking,
  - ability to require a non-default selection from a drop down list,
  - checking for injection exploits spambots use to repurpose such forms,
  - requiring a field to be left empty and
  - use of a pseudo-captcha[12] check to require the Form User to input a valid check value".
- Uses the email address and name(s) specified by the Form User as the "from address" for each email sent.
- Sends (optionally) a confirmation email back to the Form User, using the email address and names(s) specified by the Form User as the "to address".

---

[11] See en.wikipedia.org/wiki/Comma-separated_values. A CSV file can be easily processed with Microsoft Excel.

[12] See en.wikipedia.org/wiki/Captcha. Pseudo-captcha is my term for the "common sense question" approach described at en.wikipedia.org/wiki/Captcha#Attempts_at_more_accessible_CAPTCHAs. While not highly secure, they are proving to be "secure enough" for this application.

- Allows reporting of values for the IP Address, Browser Name and Referring Page.
- Uses optional field labels (names for the field other than the field name) to identify a field and report its value in an email.
- Offers three (3) different general email formats and allows customizations within each format.
- Can send two different formats of email to two different groups of email addresses.
- Permits further formatting of each email, including:
  - subject line,
  - text at the top and bottom of each email,
  - blank lines and label lines, and
  - different reporting layout for the data.
- Provides an extensive collection of possible field edits, including:
  - Email: checks a field to be sure it is a properly formatted email address and that the domain specified is configured for email;
  - Comparison: ensuring that two different fields match (e.g., as a way to validate email addresses, account numbers, etc.);
  - Equal: checks a field against a list of permitted values;
  - Not Equal: checks a field against a list of prohibited values;
  - Integer: checks a field to be sure an integer value is entered, and that the value is between minimum and maximum values;
  - Number: checks a field to be sure a number is entered, and that the value is between minimum and maximum values;
  - Text: checks a field to be sure only permitted characters are in the field;
  - Checkbox Group: checks a group of checkboxes to ensure that at least some number (min) were checked and that at most some number (max) were checked
  - Drop Down: requires that the Form User select a choice from a Drop Down field; (NB: this edit helps prevent spambots);
  - Attraction: ensures that a field is left blank (called attraction because it attracts spambots to put data in the field);
  - Pseudo-Captcha: ensures a Form User to enters the correct answer to a simple question to prevent spambots;
  - Size: checks the size of each field entered against the min and max size specified; (NB: checking max helps prevent spambots);
- Lists all identified errors on a separate error page.
- Allows customization of the error page:
  - Title,

- Primary Heading,
- Secondary Heading, and
- an Opening and Closing text line, and
- the Error Messages
- Supports creation of a template error page so that the error page will "be like" other pages on the website;
- Allows optional display of "Tracking Information" on the error page to facilitate debugging (e.g., Script Version, Script Name and Location, Referring Page, Date and Time).
- Goes to a parameter specified URL after processing.

REQUIREMENTS TO USE THE SCRIPT

GENERAL REQUIREMENTS

To use Huggins' Email Form Script, your web host must provide the ability to process PHP. [13] In addition, it must support at least PHP 5.1.0, released 24-Nov-2005[14] to use **all** features of the script.

It may be possible to use the script without PHP 5.1.0, but I do not support this and given that support for PHP 4 has been discontinued[15], I strongly encourage an upgrade to PHP 5. If you find **any** host that provides **only** PHP 4 please let me know[16].

In addition, the web host must not restrict the ability of PHP to send emails.[17]

ACKNOWLEDGING YOUR USE OF THE SCRIPT

In return for the free use of Huggins' Email Form Script, there are two teeny tiny requirements:

1. Please include a visible linked acknowledgement on the Form Page. This acknowledgement should be substantially like:

```
<a title="Huggins' Email Form Script" target="_blank"
href="http://www.JamesSHuggins.com/huggins-email-form-script">
This form uses Huggins' Email Form Script</a>
```

NB: the Acknowledgement Link file ("HEFS Acknowledgement Link V2.2.5.txt") includes this HTML in an easy to use, copy/paste format to make this easier.[18] In addition, this acknowledgement is included in the "full" sample HTML files.

---

[13] PHP is one of the programming languages used to write scripts like this. Other languages for scripts include Perl (.pl) and ASP (.asp). See en.wikipedia.org/wiki/PHP.

[14] PHP version 5.1.0 was released **years** ago … 24.Nov.2005 (see www.php.net/releases). Prior versions of this script could run on PHP 4.0.5 for Unix/Linux/Apache and 4.3.0 for Windows, but new features of this version require 5.1.0. Support for PHP 4 was *officially* discontinued 2007-12-31 (see www.php.net/releases). I strongly encourage you to be absolutely certain you know for certain what version your host runs and what version your particular server runs. To find out for sure, see Checking Your Host on page 23.

[15] Support for PHP 4 was *officially* discontinued 2007-12-31 (see www.php.net/releases). See also www.internetnews.com/dev-news/article.php/3725291.

[16] Just email me at JHuggins@JamesSHuggins.com.

[17] It appears that some web hosts install PHP but restrict the ability of PHP to send emails. It appears that some concerns about spam prompt this action. If your host does this, I strongly encourage you to **get a new host!** And while you are at it, please let me know at JHuggins@JamesSHuggins.com

The fonts, sizes, colors and other similar characteristics should be set to compliment the site. You may modify the text or add text at your discretion provided the meaning is substantially the same.

Feel free to adjust the specifics to compliment your page. The only objective is to help someone else find and use this script.

Special Notice 1: the target="_blank" attribute has been deprecated in standard HTML.I continue to use it because (a) it is still broadly supported in the browsers and (b) I don't approve of the deprecation; BUT if you are a standards stickler, just take that out

Special Notice 2: I've been asked whether the phrase "This form uses Huggins' Email Form Script" could be shortened to "Huggins' Email Form Script". Sure. I'm not a stickler for that level of detail. The goal is to help others find the script so this is clearly another acceptable option.

Placing this linked acknowledgement on the Form Page will help to promote Huggins' Email Form Script. I sincerely appreciate this consideration.

2. Include the following acknowledgement in the HTML code of the Form Page itself:

```
<head>
<!-- Form Credit: This email form uses Huggins' Email Form Script;
   To learn more visit "http://www.JamesSHuggins.com/huggins-email-form-
script".-->


note: remove this line; the <head> is just to show where to put this so be
careful not to duplicate that
```

NB: the Acknowledgement Notice file ("HEFS Acknowledgement Notice V2.2.5.txt") includes this HTML in an easy to use, copy/paste format to make this easier. In addition, this acknowledgement is included in the "full" sample HTML files

---

[18] The naming scheme of V2.2.5 references the version number of the script. The version number is "A.B.C" where "A" is the primary release number, "B" is the secondary release number and "C" is the tertiary release number. Typically, minor bug fixes update the tertiary release number. Small enhancements and changes update the secondary release number. And major changes update the primary release number. All the files in a download package use a similar numbering scheme to show that they are all part of the package. In general, you should always download and use a complete package of files with similar names. This ensures that the files all work together.

WILL HUGGINS' EMAIL FORM SCRIPT WORK ON ALL WEB SERVERS AND WEB HOSTS?

No. It won't. It ought to. But it doesn't.

Not all web hosts implement PHP. Not all web hosts implement PHP well. And even worse, some web hosts restrict the very core functionality of PHP which this script needs[19]

I talk more about this in the next section (Checking Your Host) on page *23*. I **strongly** recommend that you follow the server tests in that section to be sure your web host can properly process PHP code, especially if you are using this script on a Windows Server.

UNIX/LINUX OR WINDOWS?

This PHP script has been tested on both Unix/Linux and Windows Servers and works in both environments.

However, beta testers have had more problems with Windows Servers than with Unix/Linux servers. For example, two web hosts, EasyCGI.com and, most recently, M6.net, a host that focuses on Windows Servers, were both unable to provide a satisfactory Windows environment. This problem is described in more detail in the section titled *Host Issues* on page 189.

My general recommendation is, "only use a windows server if you need to, are **sure** you need to, and can explain why you need to".

KEEPING UP TO DATE

While not strictly a "requirement", I strongly recommend[20] that you also subscribe to the *HEFS Bulletin* to keep up to date about fixes and enhancements. For more information see *Tracking Fixes and Enhancements* on page 199.

THE PHP5 SUFFIX

The scripts hosted on my server used to use a suffix of .php5.

This was because my server supported both PHP 4 and PHP 5.

The ".php5" suffix forced my server to always use PHP 5.

I no longer use the ".php5" suffix and only include this as a note.

---

[19] For example, some restrict the ability of PHP scripts to effectively send email.

[20] Yes. I know I'm beating this point into the ground … deeply. But I've had experience with people who did not subscribe and then wondered why they didn't receive update information. So please try to forgive my obsessive harping on this.

CHECKING YOUR HOST

DO THIS FIRST: CHECK YOUR HOST

Before you start performing any tests, I first suggest you review this list

- EasyCGI.com
- BlueRazor.com
- M6.net
- Gate.com
- GoDaddy.com
- FastHosts.co.uk

If your host is one of these, first see my section titled *Host Issues* on page 189.

IF YOUR HOST DOES NOT "FAIL" YET …

Before attempting to install Huggins' Email Form Script you should determine the answers to the following questions:

1. Can your host/server support PHP?
2. Does your particular account/package support PHP?
3. What level of PHP does your host/server and account/package support?
4. Does your host/server impose *any* technical restrictions on the functionality of the PHP "mail" command?

In the sections which follow I help to answer these questions.

A QUICK CHECK

There is a very easy way to check PHP on *many* servers.

1. Rename the file in the package named check-info.php5.txt to check-info.php, or
2. If you don't have the file, just make a file named check-info.php using Notepad, then
3. Insert the following line in it:
   <? phpinfo(); ?>
4. Upload check-info.php to your server.
5. Execute this file in your domain. That is, in your browser, type:

```
http://www.mydomain.com/check-info.php

(NB: you must replace www.mydomain.com with your website address.)
```

If you have PHP easily available, you will then see a PHP configuration information page.

Remember … This version of the script requires that your web host must provide must support at least PHP 5.1.0, released 24-Nov-2005.[21]

If you do not see this page, your host/server or account/package may require special considerations to use PHP or may not support PHP at all.

This first check is **not** sufficient, however. You must **also** perform the second check.

A SECOND, MORE EXTENSIVE CHECK – CHECKING EMAILS

In the effort to assist some beta testers with installation and use of the Huggins' Email Form Script, I developed more extensive pages and scripts to help test the host.

These scripts recognize that the Huggins' Email Form Script has, at its core, two simple functions:

1. Accept input from a Web Form through the Form Post action, and
2. Send an email.

I have created two *teeny tiny* files. These are:

1. hefs-mail-test-v2.2.5.htm
2. hefs-min-test-v2.2.5.php5

These two files, used together, provide an extremely simple, *minimal* test of these basic PHP functions.

HEFS-MAIL-TEST-V2.2.5.HTM

This tiny web page is a Web Form with four Form Fields:

1. From Name
2. From Email
3. To Name
4. To Email

It also has a submit button that invokes pqy-v2.2.5.php.

---

[21] Prior versions could run on PHP 4.0.5 for Unix/Linux/Apache and 4.3.0 for Windows, but new features of this version require 5.1.0. If you are not sure what your host runs, see Checking Your Host on page 23.

PQY-V2.2.5.PHP

This tiny php page sends three emails to test the PHP email function on the web host.

1. Email #1 uses both the From Name/From Email and the To Name/To Email to send the test.

2. Email #2 uses only the From Email and To Email to send the test.

3. Email #3 uses only the To Email and does not specify a From Email address. The "from address" may be blank or may be a "default" specified by your system.

CONTENTS OF HEFS-MAIL-TEST-V2.2.5.HTM

```
<html>
<head>
</head>
<body>
<form action=" pqy-v2.2.5.php" method="post" name="minimaltest">
<p>From Name<br>
<input type="text" name="fromname" size="40"></p>
<p>From Email<br>
<input type="text" name="fromemail" size="40"></p>
<p>To Name<br>
<input type="text" name="toname" size="40"></p>
<p>To Email<br>
<input type="text" name="toemail" size="40"></p>
<p><input type="submit" value="Submit"> </p>
</form>
</body>
</html>
```

CONTENTS OF PQY-V2.2.5.PHP

```
<?

error_reporting(E_ALL);

$ToName    = trim(stripslashes(strip_tags($_POST["toname"])));
$ToEmail   = trim(stripslashes(strip_tags($_POST["toemail"])));

$FromName  = trim(stripslashes(strip_tags($_POST["fromname"])));
$FromEmail = trim(stripslashes(strip_tags($_POST["fromemail"])));



//   TEST 1

$email = "$ToName <$ToEmail>";
$from  = "$FromName <$FromEmail>";

$subject="Testing HEFS Minimal Mail Functionality #1 V2.2.5";
$message="HEFS Minimal Mail Functionality for Test #1 is Working!";

if(mail("$email","$subject","$message","From: $from")){
    echo "email #1 sent !!! </br>";
}else{
    echo "email #1 error !!! </br></br>";
    echo "This failure is serious, but MAY not be fatal.</br>";
    echo "IF emails #2 and #3 are sent and received, there MAY be a fix for
this error</br>";
    echo "Be sure to read the section in the manual on the test results for
this test and</br>";
    echo "also read about the parameter
PleaseForgiveMyUseOfAWindowsServer.</br>";
    echo "I added this special variable in order to let the script
function</br>";
    echo "even on broken Windows servers. </br>";
    echo "But remember, in order for this to be fixable,</br>";
    echo "emails #2 and #3 must be sent and received.</br>";
    echo "If they are not, do not pass go and do not collect $200.</br>";
}



//   TEST 2

$email = "$ToEmail";
$from  = "$FromEmail";

$subject="Testing HEFS Minimal Mail Functionality #2 V2.2.5";
$message="HEFS Minimal Mail Functionality for Test #2 is Working!";

if(mail("$email","$subject","$message","From: $from")){
    echo "email #2 sent !!! </br>";
}else{
    echo "email #2 error !!! </br></br>";
    echo "Do not proceed without resolving this issue.</br>";
    echo "You must resolve this issue or the script will not function on your
host.</br>";
```

```
    echo "This is not a script issue per se, and may be fixable by your
host.</br>";
    echo "For example, your host may have turned off</br>";
    echo "some feature and may just need to turn it on for you.</br>";
    echo "On the other hand, some hosts intentionally cripple this
function.</br>";
    echo "If you cannot resolve this issue, do not proceed.</br>";
    echo "Unless you can resolve this issue, the script will not work.</br>";
}



//   TEST 3

$email = "$ToEmail";

$subject="Testing HEFS Minimal Mail Functionality #3 V2.2.5";
$message="HEFS Minimal Mail Functionality for Test #3 is Working!";

if(mail("$email","$subject","$message")){
    echo "email #3 sent !!! </br>";
}else{
    echo "email #3 error !!! </br></br>";
    echo "Do not proceed without resolving this issue.</br>";
    echo "You must resolve this issue or the script will not function on your
host.</br>";
    echo "This is not a script issue per se, and may be fixable by your
host.</br>";
    echo "For example, your host may have turned off</br>";
    echo "some feature and may just need to turn it on for you.</br>";
    echo "On the other hand, some hosts intentionally cripple this
function.</br>";
    echo "If you cannot resolve this issue, do not proceed.</br>";
    echo "Unless you can resolve this issue, the script will not work.</br>";
}
```

HOW TO CONDUCT THE TEST

1. Download the two test files. These files are named hefs-mail-test-v2.2.5.htm.txt and pqy-v2.2.5.php.txt.

2. Rename the two files, removing the ".txt" suffix.

3. Install pqy-v2.2.5.php on your host.

4. In hefs-mail-test-v2.2.5.htm, change (if necessary) the reference to pqy-v2.2.5.php. For example, if you had to put it into a special PHP folder, be sure to reference that folder.

5. Using a browser, go to the hefs-mail-test-v2.2.5.htm page on your site.

6. Fill in the fields and click submit.

7. See if the results indicate the email was sent.

8. See if the email was actually sent (i.e., did you get the email).

TEST GUIDELINES

- In *all* tests, make the "from" and "to" names different *and also* make the "from" and "to" email addresses different.

- Conduct at least one test using a "to" email address that is not on your web host. (e.g., a Gmail.com address, a Yahoo.com address or an email address hosted on another web host).

- Conduct at least one test using a "from" email address that is not on your web host. (e.g., a Gmail.com address, a Yahoo.com address or an email address hosted on another web host).

- If possible, conduct at least one test using both a "from" and also "to" email address that is not on your web host. (e.g., a Gmail.com address, a Yahoo.com address or an email address hosted on another web host).

TEST PURPOSES

- Email Test #1:
  tests whether an email can be sent using a "to" address and from address consisting of both name and email address:[22]

    From: Dan Foreman <Dan@xxx.com>
    To: Mary Jones <Mary@yyy.com>

---

[22] On many Windows servers, Microsoft has not made this industry standard available to PHP. The result is that this test will fail, as will the script. Version 2.2.3 introduced the parameter named FormPleaseForgiveMyUseOfAWindowsServer. This parameter forces the script to not try this and instead to use format of Email Test #2. For more information see FormPleaseForgiveMyUseOfAWindowsServer on page 123.

- Email Test #2:

  tests whether an email can be sent using a "to" address and "from" address consisting of only email address:

      From: Dan@xxx.com
      To: Mary@yyy.com

- Email Test #3:

  tests whether an email can be sent using a "to" address consisting of only email address without any "from" address:

      From:
      To: Mary@yyy.com

TEST RESULTS

For each test you should receive **three** different emails. (The subjects of the emails will identify them as #1, #2 and #3.)

If you do not receive Test Email #1, and have a Windows Server, you may need to use the parameter named *FormPleaseForgiveMyUseOfAWindowsServer* that will force the script to not try this and instead to use format of Email Test #2.[23] If you do not know why Test Email #1 is not received, I recommend you contact support at your host to find an answer.

If you do not receive Test Email #2 and Test Email #3, there may be an issue in getting Huggins' Email Form Script to function on your chosen host. I recommend you contact support at your host to ask why you did not receive the emails.

Note that Test Email #3 is **not** required, and if it fails that is **not** a severe limitation. However, the failure of 2 and especially 1 may be showstopping.

If you encounter unknown problems with the use of my script, consider setting up the test and then contacting tech support at your host. In one case, when faced with the test as evidence, the host admitted that they have crippled the email functionality required by the script which prompted the user to contact me looking for a better host. I was able to recommend one that solved the problem and was even cheaper.

Also, feel free to contact me to see if I can possibly assist.

---

[23] See FormPleaseForgiveMyUseOfAWindowsServer on page 123

BOTTOM LINE

If you cannot get the mail test to work using hefs-mail-test-v2.2.5.htm and pqy-v2.2.5.php, you will **not** be able to get my script to work. This is important because the failures may not be obvious. It just won't work. Do **not** skip these tests. Do **not** proceed unless you resolve these tests.

Please feel free to contact me about any problems here:

<div align="center">

JHuggins@JamesSHuggins.com

or

JHuggins@EclecticPower.com.

</div>

QUICK START FOR A SIMPLE FORM

A QUICK APPROACH TO A SIMPLE FORM

Up until Version 2.2.1 (26.Dec 2008) it had been necessary for every user to create their form from scratch. While this has worked, some users requested that I create and offer a simple form, and a process for getting it working as fast as possible.

I polled my users asking for guidance and as a result have created a simple form that has proven to help users get started.

Using this predefined webpage, you have a way to move quickly in these general steps (details follow):

1. Do not attempt to test until you get to the test point.
2. Download the Script Distribution Package
3. Test Your Server
4. Use the predefined Web Form Page I give you and create a page on your site
5. Make changes to this web form page
6. Test your work using the script on my server
7. Determine Where You Will Place the PHP Script
8. Upload the Script to Your Server
9. Change the Form Statement
10. Test It Out
11. Make Your Own "Thank You" Page
12. Subscribe to HEFS Bulletin
13. Learn How to Do More

1. DO NOT ATTEMPT TO TEST UNTIL YOU GET TO THE TEST POINT

The script will fail until you create/correct enough parameters for it to work. Do not get overly eager. Do not try to test early. For sure do not test before step 5I.

2. DOWNLOAD THE SCRIPT DISTRIBUTION PACKAGE

Distribution for the Huggins' Email Form Script can be obtained from

www.JamesSHuggins.com/huggins-email-form-script

I recommend that you create a special folder to hold all these files for easy reference and access.

The Version 2.2.5 distribution of this software includes:

- **0 READ ME FIRST V2.2.5.txt**
  This is the "Read Me First" file. Always read the "Read Me First" file in any distribution for the latest information. The primary documentation may not yet have been updated with information in the "Read Me First" file.

- **0 READ ME SECOND V2.2.5.txt**
  This "Read Me Second" file is my not completely successful attempt to get people to sign up for the HEFS Bulletin. Please, please, please sign up for the bulletin. This is very important.

- **0 READ ME THIRD V2.2.5.txt**
  Some people immediately start modifying, or trying to modify, the program. This is not the idea. You should be able to do all this without modifying the program. This notice encourages you not to modify the program.

- **check-info.php.txt**
  This text file contains PHP code to have your server display PHP installation/configuration information."

  - ✓ **NB**: the file is distributed as a .txt file. Please rename and remove the final ".txt" giving a new name of "check-info.php".

- **HEFS Acknowledgment Link V2.2.5.txt**
  This text file contains HTML for an acknowledgement link named you can paste on your web page. See also *HEFS Acknowledgement Notice V2.2.5.txt* below.

- **HEFS Acknowledgment Notice V2.2.5.txt**
  This text file contains HTML you can place into your web page to document your use of the HEFS Huggins' Email Form Script. See also *HEFS Acknowledgement Link V2.2.5.txt* above.

- **hefs-example-1.htm.txt**
  This text file contains HTML which I use for a page on my website:
  http://www.jamesshuggins.com/h/hefs/hefs-example-1.htm
  I include this only so you will have it if you want it. I do not use this for anything directly in this documentation.

- **HEFS Form Statement and Parms for HTML - V2.2.5.txt,
  HEFS Form Statement and Parms for XHTML - V2.2.5.txt**
  These text files contain the HTML and XHTML form statement and
  hidden field parameters you can copy/paste into your form. The two
  versions are for the HTML and XHTML versions. These two are
  equivalent except that the one for XHTML includes the closing
  slashes required by that format and the one for HTML does not.
  These files are not used "as is", but instead provide code to be used
  through a copy/paste and then modified.

- **hefs-mail-test-v2.2.5.htm.txt**
  This text file contains HTML for a minimal PHP Test page. We use
  this to help you test your server.

  ✓ **NB**: the file is distributed as a .txt file. Please rename and remove the final
    ".txt".

**hefs-simple-form-page-1a for HTML V2.2.5.htm.txt,
  hefs-simple-form-page-1b for HTML V2.2.5.htm.txt,
  hefs-simple-form-page-2a for XHTML V2.2.5.htm.txt
  hefs-simple-form-page-2b for XHTML V2.2.5.htm.txt**
  These are simple, "ready to use" pages using either HTML or XHTML.
  These are used to help you get started quickly. These four are
  equivalent except that the "1" files for HTML do not include the
  closing slashes required by the XHTML format, while the "2" files for
  XHTML do include the closing slashes. In addition the "a" files
  provide for first and last name while the "b" files only provide for a
  single unified name. These files are not used "as is", but instead
  provide an initial starting point for you.

- **hefs-simple-form-page-3a for HTML-with-CSS V2.2.5.htm.txt,
  hefs-simple-form-page-3b for HTML-with-CSS V2.2.5.htm.txt,
  hefs-simple-form-page-4a for XHTML-with-CSS V2.2.5.htm.txt,
  hefs-simple-form-page-4b for XHTML-with-CSS V2.2.5.htm.txt,**
  These pages are based on contributions to the community by
  Jonathan Lewis (http://www.LewisPhillips.com). They also use
  concepts/code from the article at http://www.sitepoint.com/style-web-forms-css . I encourage a review of this article. They add CSS to
  my pages to give you other options. These four are equivalent except
  that the "3" files for HTML do not include the closing slashes
  required by the XHTML format, while the "4" files for XHTML do
  include the closing slashes. In addition the "a" files provide for first
  and last name while the "b" files only provide for a single unified
  name. These files are not used "as is", but instead provide an initial
  starting point for you.

- **hefs-simple-form-page-5a for HTML V2.2.5.htm.txt,
  hefs-simple-form-page-5b for HTML-with-CSS V2.2.5.htm.txt,
  hefs-simple-form-page-6a for XHTML V2.2.5.htm.txt
  hefs-simple-form-page-6b for XHTML-with-CSS V2.2.5.htm.txt**
  These pages are **also** based on contributions to the community by Jonathan Lewis (http://www.LewisPhillips.com). They also use concepts/code from the article at http://www.sitepoint.com/style-web-forms-css . I encourage a review of this article. They add CSS to my pages to give you other options. These four are equivalent except that the "5" files for HTML do not include the closing slashes required by the XHTML format, while the "6" files for XHTML do include the closing slashes. In addition the "a" files provide for first and last name while the "b" files only provide for a single unified name. These files are not used "as is", but instead provide an initial starting point for you.

- **Host Issues for Huggins' Email Form Script V2.2.5 9-01-060.pdf**
  This PDF file is an excerpt of the "Host Issues" chapter of the User Manual and was created to permit individuals to quickly explore the host issues without needing to download and cope with the whole User Manual.

- **huggins-email-form-script-v2.2.5.php5.txt**
  This is the actual script … a PHP file.
  - ✓ **NB**: the file is distributed with a .txt suffix. Please rename by removing the final ".txt".

- **pqy-v2.2.5.php.txt**
  This text file contains PHP code for a minimal PHP Test Script. We use this to help you test your server.
  - ✓ **NB:** the file is distributed as a .txt file. Please rename by removing the final ".txt".

- **Simple Form Page Samples.pdf**
  This PDF file contains displays of the 12 different sample html files.

- **User Manual for Huggins' Email Form Script V2.2.5 5-01-063.pdf**
  This PDF file contains the documentation (this manual).

3. TEST YOUR SERVER

Before you start anything, test your server. See Checking Your Host on page 23. This will quickly tell you whether your host supports PHP5 and whether it has turned off any of the features you need to make this script work.

4. CHOOSE FROM AMONG THE PREDEFINED WEB FORM PAGES I GIVE YOU AND CREATE A PAGE ON YOUR SITE

I provide many different Web Form Pages as templates. The general name format is **"hefs-simple-form-page-NX ..."** as follows for "N" and "a"

"N" can be:
1: for fundamental basic HTML
2: for HTML with CSS
3: for XHTML
4: for XHTML with CSS
5: for HTML with CSS and some "more advanced" code
6: for XHTML with CSS and some "more advanced" code

"X: can be:
a: for a form that asks for first name and last name
b: for the same form, except that it asks only for a "full name".

You can see samples of all these pages within the PDF document named **Simple Form Page Samples.pdf**

Once you have decided which form to use[24], copy it to your development environment as the basis, being sure to remove the ".txt" suffix.

NB: *The examples in this section of the document use the HTML format without the closing slashes. If you are primarily using XHTML on your website then the samples in the XHTML format are better for you. Be sure to use the correct version to start with: either the HTML or the XHTML.*

5. MAKE CHANGES TO THIS WEB FORM PAGE

Now we'll make changes to this web form page to customize it for you.

---

[24] To choose, you may wish to rename **all** the samples by removing the ".txt" suffix and look at the result of each one.

In the steps that follow I'll walk you thru the HTML one statement at a time to guide you with changes. I'll take you thru the change and also point you to the place in this manual that gives you more information.

Recommendation: name your Form Fields using only lower-case letters and numbers. In my experience, that reduces errors.

5A. THE "FOR OFFICE USE ONLY" FIELD

```
<p>For office use only.<br>Do not put anything in this field.
<input type="text" name="email2" size="60"></p>
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

The form includes a field named "**email2**" and labeled "For office use only.". This field is used to help trap spam. Do *not* change or delete this field. Leave this field there.

NB: This instruction was added because some users didn't think they needed this and then deleted it. Do *not* delete it.

5B. THE FORM STATEMENT

```
<form action="http://JamesSHuggins.com/p/huggins-email-form-script-v2.2.5-
k9y9p8r5.php5"
      method="post"
      name="example1">
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

The Form statement starts the definition of the form on the page. It says what to do when the user clicks the "submit button", and it gives the form a name. The Form statement in the example points to the email form script *on my site*. You can use this to test, but don't use it for production. You can and should change it *later* to point to the script on your site. We will use the script on my site only for your *initial* testing. You can change the "name" ("example1") to anything you would like.

5C. FORMECHOUSER

```
<input name="FormEchoUser" type="hidden" value="yes">
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

This parameter tells the script whether to generate and send an email back to the person who fills out the form. It is set to "yes". The script will send an email back to the person who fills out the form. See *FormEchoUser* on page 72.We will **not** change this parameter.

NB: This example HTML does not use XHTML and does not use the closing slashes. If you are primarily using XHTML on your website then the other sample format is better. Be sure to use the correct version to start with. Either the HTML or the XHTML.

5D. FORMEMAILFIELDLIST

```
<input name="FormEmailFieldList" type="hidden"
value="email_address, confirm_email_address">
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

This parameter tells the script what form field contains the email address of the submitter: **email_address**. It also tells the script what form field contains the confirmation email address: **confirm_email_address**. See *FormEmailFieldList* on page 75. We will **not** change this parameter.

5E. FORMFIELDNAMELABELPLUSLIST

```
<input name="FormFieldNameLabelPlusList" type="hidden" value=
            "first_name, First Name, Y, 0, 60 |
             last_name, Last Name, Y, 0, 60 |
             email_address, Email Address, Y, 0, 60 |
             confirm_email_address, Confirm Email Address, N, 0, 60 |
             themessage, Message, Y, 0, 0 |
             test78, Question about summer, N, 0, 10 |
             email2, For Office Use Only, N, 0, 0 |
             blankline,
             ip, IP Address, N, 0, 0 |
             browser, Browser Information, N, 0, 0 |
             referer, Referer Page, N, 0, 0">
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

This parameter specifies the names of all the fields in the form, their "label" to use for reporting, whether or not they are required, and their minimum and maximum lengths. See *FormFieldNameLabelPlusList* on page 114. We will **not** change this parameter.

5F. FORMFIELDNAMEEDITLIST

```
<input name="FormFieldNameEditList" type="hidden" value=
              "test78, captcha, hot, ~ |
              email2, attraction, ~"
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

This parameter specifies special edits for the script to perform on certain fields. In the example above, it says that the field named test78 is a pseudo-captcha[25] field. It requires that the user enter "hot". And it says that the field named email 2 is an attraction field. This requires that the user not enter anything. See *FormFieldNameEditList* on page 98. We will **not** change this parameter.

5G. FORMNAMEFIELDLIST

```
<input name="FormNameFieldList" type="hidden"
value="first_name, last_name"
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

This parameter identifies which fields comprise the name of the submitter. In this form two fields are used for the name of the submitter: **first_name** and **last_name**. See *FormNameFieldList* on page 119. We will **not** change this parameter.

5H. FORMNEXTURL

```
<input name="FormNextURL" type="hidden"
value="http://JSH.us/h/hefs/huggins-email-form-script-thank-you.htm">
```

NB: the script it not yet ready for testing. Wait at least till after step 5I.

This parameter tells the script where to go "next". In this case it is set to go to JSH.us/h/hefs/huggins-email-form-script-thank-you.htm. You should eventually create your own landing page and **then** you should change this to go there. When you change it, use the long form of the URL as I did in the example. Spell it out completely. It will save you some trouble. See *FormNextURL* on page 120

---

[25] See en.wikipedia.org/wiki/Captcha. Pseudo-captcha is my term for the "common sense question" approach described at en.wikipedia.org/wiki/Captcha#Attempts_at_more_accessible_CAPTCHAs. While not highly secure, they are proving to be "secure enough" for this application.

5I. Msg1AddrList

```
<input name="Msg1AddrList" type="hidden"
value="Bob Smith, Bob.Smith.XXXXX, JSend.com,
Sue Smith, Sue.Smith.XXXXX, JSend.com">
```

NB: you may test this script *after* this test is complete. Do not test the script until *after* you make the 5I change. Do not forget to upload the changes you have made before you test.

This parameter specifies the email addresses of the people to receive the Email Message The email addresses are split into three parts separated by commas.

The first part is the Name Part. The second part is the Local Address Part. The third part is the Domain Name Part.

Consider this email address:
"Frank Jones <Frank.Jones@JSend.com>".

The Name Part is the name of the addressee. It does not directly affect delivery of the email. In this example "Frank Jones" is used as the Name Part.

The Local Name Part is the part of the email address immediately before the "@". In this example "Frank.Jones" is used as the Local Name Part.

The Domain Name Part is the part of the email address immediately after the "@". In this example "JSend.com" is used as the Domain Name Part.

You can have *as many* addresses listed as you want. In the example, I have listed two.

See *MsgxAddrList* on page 134.

Change Needed: You should change this parameter to reflect your own name and email address so the email will be sent to you. If you test before making this change the script will send the mails to *me*. That won't help you test at all!!

Editing Notes: The script performs lots of editing on email addresses. These edits apply both to the email addresses you code directly and those which are entered by the user:

- The email address entered will be tested for format
- The email address entered will be tested for domain name
- The email address and email confirmation address will be compared

NB: *After* you have completed this change step and uploaded the change, you may test the script, even though all the changes are not yet done.

5J. MSG1FIELDNAMEEXCLUDELIST

```
<input name="Msg1FieldNameExcludeList" type="hidden" value=
            "confirm_email_address, test78, email2">
```

This parameter tells the script what fields to exclude when creating the email to send for Msg1. In this case we exclude **confirm_email_address**, **test78** and **email2**.

We exclude **confirm_email_address**, because we used it in *FormEmailFieldList* and so we know the script will force it to be the same as **email_address** and hence we don't need to report it.

We exclude **test78** because we already have an edit that will force it to be "hot".

And we exclude **email2** because we have an edit that will force it to be blank.

For these reasons we don't need to report these three fields so we exclude them. See *MsgxFieldNameExcludeList* on page 141. We will **not** change this parameter.

5K. MSG1SUBJECT

```
<input name="Msg1Subject" type="hidden"
value=" Email from Basic HTML Form Page">
```

This parameter tells the script what content to put in the subject of the email for Msg1. You should definitely change this to a good, relevant descriptive subject. See *MsgxSubject* on page 157.

Change Optional: You *should* change this parameter to reflect your own choice for the email subject. *If you wish*, you may leave it now until you understand more how it works.

5L. MSG1TEXTBOTTOM

```
<input name="Msg1TextBottom" type="hidden"
value="The above data was entered in the form.">
```

This parameter provides the text that will appear at the bottom of the Msg1 email. See *MsgxTextBottom* on page 160.

Change Optional: You *should* change this parameter to reflect your own choice for the bottom text. *If you wish*, you may leave it now until you understand more how it works.

5M. MSG1TEXTTOP

```
<input name="Msg1TextTop" type="hidden"
value="The following data was entered in the form.">
```

This parameter provides the text that will appear at the top of the Msg1 email. See *MsgxTextTop* on page 161.

Change Optional: You *should* change this parameter to reflect your own choice for the top text. *If you wish*, you may leave it now until you understand more how it works.

5N. MSGECHOFIELDNAMEEXCLUDELIST

```
<input name="MsgEchoFieldNameExcludeList" type="hidden" value="*">
```

This field tells the script what input fields to excludes from the Msg Echo email. Using an asterisk ("*") tells it to omit the same fields as were omitted on the Msg1 email. See *MsgxFieldNameExcludeList* on page 141. We will ***not*** change this parameter.

5O. MSGECHOFROMADDR

```
<input name="MsgEchoFromAddr" type="hidden"
value="Confirmation from Simple Form Page 1, Frank.Jones, JSend.com">
```

This parameter specifies the email address to use as the sending address for the echo email. The email address is split into three parts separated by commas.

The first part is the Name Part. The second part is the Local Address Part. The third part is the Domain Name Part.

Consider this email address:
"Confirmation from Simple Form Page 1 <Frank.Jones@JSend.com>".

The Name Part is the name of the addressee. It does not directly affect delivery of the email. In this example "Confirmation from Simple Form Page 1" is used as the Name Part.

The Local Name Part is the part of the email address immediately before the "@". In this example "Frank.Jones" is used as the Local Name Part.

The Domain Name Part is the part of the email address immediately after the "@". In this example "JSend.com" is used as the Domain Name Part.

You may only specify one email address.

See MsgEchoFromAddr on page 132.

For You To do: You *should* change this parameter to reflect your choices and your actual email address so the person who gets the email will know it is from you and be able to reply.

5P. MSGECHOSUBJECT

```
<input name="MsgEchoSubject" type="hidden"
value="Data You Entered in Simple Form Page 1">
```

This parameter tells the script what content to put in the subject of the email for the Echo Message sent to the form user1. Feel free to change this value to whatever you would like. See *MsgxSubject* on page 157.

Change Optional: You *should* change this parameter to reflect your own choice for the subject. *If you wish*, you may leave it now until you understand more how it works.

5Q. MSGECHOTEXTBOTTOM

```
<input name="MsgEchoTextBottom" type="hidden"
value="The above data was entered in the form.">
```

This parameter provides the text that will appear at the bottom of the Echo message email sent back to the form user. See *MsgxTextBottom* on page 160.

Change Optional: You *should* change this parameter to reflect your own choice for the bottom text. *If you wish*, you may leave it now until you understand more how it works.

5R. MSGECHOTEXTTOP

```
<input name="MsgEchoTextTop" type="hidden"
value="The following data was entered in the form.">
```

This parameter provides the text that will appear at the top of the Echo email message email sent back to the form user. See *MsgxTextTop* on page 161.

Change Optional: You *should* change this parameter to reflect your own choice for the top text. *If you wish*, you may leave it now until you understand more how it works.

6. TEST YOUR WORK *USING THE SCRIPT ON MY SERVER*

Once you have made these changes then you should load the changed page to your server and execute that page. The page will try to work using the script on my server. You can use the script on my server to see how well you did. Do not proceed until you get the script working on **my server**. Contact me if you need to:

JHuggins@JamesSHuggins.com

or

JHuggins@EclecticPower.com.

7. DETERMINE WHERE YOU WILL PLACE THE PHP SCRIPT

Once it works using the script on my server you will want to load the script to your server. First, decide on the name and location of the PHP Script.

I strongly suggest (and respectfully request) that you name the script file using the same name as the script file you receive:

huggins-email-form-script-v2.2.5.php

The reasons for this are:

1. Keeping the name the same will let you easily know, at all times, exactly which version you have installed. This is especially helpful when you write me for assistance.

2. Using a fully qualified name for the installed file will enable you to install a new version and move Web Forms on your site to use the new script in a controlled way.

3. Using a fully qualified name will help others visiting your site and "looking under the covers" to find the script you are using. This will promote use of the script.

You may place the script in any location within your website, but make a note of what will be the fully qualified path/name to the script.

That said, I generally recommend **not** placing the script into the cgi folder, because some hosts restrict that folder and it can complicate installation. [26]

For example, if your site is www.Someplace.com and you place the script in a folder named scripts, the fully qualified path/name would be:

```
http://www.Someplace.com/scripts/huggins-email-form-script-v2.2.5.php
```

On the other hand, if your site is www.Someplace.com and you place the script in the root folder, the fully qualified path/name would be:

```
http://www.Someplace.com/huggins-email-form-script-v2.2.5.php
```

8. UPLOAD THE SCRIPT TO YOUR SERVER

However you normally upload such things, upload the script to your server.[27]

The script is the file named huggins-email-form-script-v2.2.5.php5 which you previously downloaded.

✓   NB: the file is distributed as huggins-email-form-script-v2.2.5.php.txt". Please rename and remove the final ".txt".

Be sure to note where you put it.

---

[26] The "cgi" directory gets its name from "Common Gateway Interface". The CGI standard is a way for web pages to interface with computer programs (like the email processing script). Some web servers require that the script be placed in the "cgi" directory. At one time, using the "cgi" directory was important. However, today, the overwhelming majority of web servers do not care and permit the script to be placed anywhere in the site.

If your server does not require you to put the script in the "cgi" directory, I recommend you avoid it. Instead, put it somewhere else. "Permissions" on the "cgi" directory can sometimes interfere with script functions.

Regardless of where the script is placed on the site, it is important to know where it is. The script is executed by directly referencing it using its full path and name. Whatever you do, make a note where you put it.

[27] The most common way of uploading is to use FTP. FrontPage users who "publish" their files, however, may place the script in the web/site structure on their PC and then "publish" the script. Either way works just fine.

9. CHANGE THE FORM STATEMENT

Change the form statement that refers to the script on my server to refer to the script on your server.

Use the correct, fully qualified path/name of the script on your server.

Also, for simplicity, include the "http://" part of the script name.[28]

10. TEST IT OUT

If you did everything right, the script should now work on your site.

Your form should now:

- Require that you type in the email address.
- Require that you also type in the email confirmation address.
- Confirm that the email address and the email confirmation address match each other.
- Confirm that the email address is in the right format.
- Confirm that the email address domain name is legal.
- If everything checks out, it should send an email to you with everything typed in, send a confirmation email to the Form User confirming everything typed in and then go to the web page you specified.

Here are some tests you can try:[29]

- Omit the email address.
- Type the email address, but omit the email confirmation address.
- Type both the email address and the email confirmation address, but make them different.
- Type the email address in the wrong format. For example, put in no "@" signs, or two "@" signs, or nothing before the "@" or nothing after the "@".
- Type the email address with a domain name you are sure is not a valid domain name.

---

[28] It is not always necessary to include the http:// part or include the full path. But doing so will eliminate some possibilities for error I've frequently needed to help people with so I recommend just putting them there. However, if the script and form are on the same server, you don't need the http:// and if you are in the same folder, you don't need the full path. But still, putting them there *doesn't hurt*.

[29] For the best effect, try these one at a time.

11. MAKE YOUR OWN "THANK YOU" PAGE

If you haven't changed it already, the "Thank You" page it goes to is on *my website*. You want it to go to a "Thank You" page on *your* website. If you haven't already made your own "Thank You" page, make one now.

12. SUBSCRIBE TO *HEFS BULLETIN*

I've said it before. Let me say it again. Now, while you are thinking about it, subscribe to the *HEFS Bulletin* to keep up to date about fixes and enhancements. To subscribe to *HEFS Bulletin* go to:

www.JamesSHuggins.com/hefs-bulletin

For more information see *Tracking Fixes and Enhancements* on page 199.

13. LEARN HOW TO DO MORE

Your form now works. But you probably want to learn how to use some of the more advanced features of the script.

The next section, titled *Continuing the Quick Start* on page 48, will take you through some of this process.

CONTINUING THE QUICK START

WHAT IS THIS SECTION?

In the prior section (Quick Start for a Simple Form beginning on page 32) I went through simple steps to get a basic form working. Feedback from users indicates that this form may satisfy the basic needs of more than 80% of my users. However, some users will require additional functionality.

This section introduces some of that additional functionality.

Note, however, that this section does *not* completely explore all the functionality of Huggins' Email Form Script. Rather, it highlights some of the most common additional functionality needed for many forms.

WHERE ARE WE?

At the end of the prior section) we have created a simple form that

1. sends an email with all Form Fields to one address, then
2. goes to a specified "Thank You" page.

This section will introduce more possibilities.

ADDING FUNCTIONALITY TO THE FORM

1. BASE THE REST OF THE WORK ON THE WORK SO FAR

Rather than start from scratch, the remaining directions build on the basic form you have already built. If you have not built it, back up and do that now. Go back to ??? on page 13.

2. ALL THE POSSIBLE PARAMETERS

You have two files in the script package

- "HEFS Form Statement and Parameters V2.2.5 for HTML.txt"
- "HEFS Form Statement and Parameters V2.2.5 for XHTML.txt"

These two files contain complete sets of sample parameters for the HTML form. If you are using HTML use the first. If you are using XHTML use the second.

I provided these so you can easily copy/paste and then change the full range of parameters.

3. DEFINING THE FIELDS

A key form is the parameter called *FormFieldNameLabelPlusList* (page 114). This parameter defines all the fields in our form.

If we have a small form, this is easy. If you have a form with 40 or 50 or even more fields, it can be a bit of a chore. To help with this task, I have provided a special tool.

4. A SPECIAL TOOL TO GIVE YOU THE FORMFIELDNAMELABELPLUSLIST INFORMATION

I have created a special tool to allow the system to extract form field information to help you build parameters. This is particularly helpful if you have a large form.

To use this tool, add this Hidden Field Parameter to the Web Form:[30]

```
<input name="FormOutputFieldInfo" type="hidden" value="yes">
```

Then, load the Web Form page to the server, and go to the Web Form Page. First be sure to check every checkbox. If you don't check a checkbox, the Web Form Page won't send information on that checkbox. Then press the submit button.

The script will output a web page with "preformatted" HTML you can copy/paste for various needs.

Copy this entire page and paste it into a Notepad file. Save that Notepad file someplace where you will be able to find it.

This HTML will give you html for:

- A concatenated field name list,
- A stacked field name list, and
- FormFieldNameLabelPlusList

Once you have done this be sure to change the value to "no" and reload the changed page back onto your server.

---

[30] For more information about this Hidden Field Parameter, see FormOutputFieldInfo on page 121.

5. COPYING THE FORMFIELDNAMELABELPLUSLIST INFORMATION

The HTML for *FormFieldNameLabelPlusList* will show all the Form Fields. Each line will have the Form Field name, the label (the same as the name), a value of "N" (meaning it is not required) and minimum and maximum sizes of zero.

- Copy this HTML and put it into the Web Form
- Turn off the parameter *FormOutputFieldInfo*. You can do this by changing the value to "n" or "no".

6. UPDATE FORMFIELDNAMELABELPLUSLIST

Update each field.[31] One at a time:

- **Choose a Field Label for the field, and type it in.**
  Again, each line will have the Form Field name, the Label (the same as the name), a value of "N" (meaning it is not required) and minimum and maximum sizes of zero. The Label is the text that the script will use to refer to that field in error reports and emails. For example, you might label **email** as "Email Address", **firstname** as "First Name", and so on. To change the Label, just replace the second parameter of each line. The label may contain spaces and special characters. It may not contain a comma.

- **Specify the Required Fields.**
  If the field is required, change the "N" to an "R" (for "Required").

- **Specify the Checkbox Fields.**
  If the field is a "checkbox", change the "N" to a "C" (for "Checkbox"). Forms do not send information for checkboxes that are not checked. The value of "C" tells the system that this is a checkbox, and not to get freaked out if the form doesn't send information.

- **Specify the Minimum Size.**
  If there is a minimum acceptable size for the field, change the "0" to that size. Experience says this will be rarely used.

- **Specify the Maximum Size for text fields.**
  Each text field has a maximum size. It is specified once by the size of the text field in the HTML for the form. Change the "0" for maximum size to the maximum size of the text fields. This should match the field size specified in the HTML.

After you have done this you have specified the basic information about each field on the form. And, each field will have five items specified for it.

Unless you override it, this will automatically set several other actions:

---

[31] To see more about FormFieldNameLabelPlusList, see the full description on page 114.

- The email address entered will be tested for format
- The email address entered will be tested for domain name
- The email address and email confirmation address will be compared
- The email address entered will be used to send an echo/confirmation email if you tell the script to send one.

ADDITIONAL CHANGES

The preceding changes are just a few of those possible for this form. Here are some of the additional things to check out as possibilities:

- **Pull down lists**
  Adding a drop down list can be an excellent protection against spambots using your form/script. See *Field Edit 6 – Drop Down: Ensure a Drop Down Value is Selected (E013)* on page 107.

- **Additional Field Validation**
  Finally, you should check out the other field edits you may want to use. More information about these is in the description of *FormFieldNameEditList* on page 98.

YOU HAVE FINISHED THE QUICK START

YOU ARE DONE! (AT LEAST FOR THE FIRST PART)

You have now finished the Quick Start. At this point you should now have a form page that **works**.

You may wish to make it "prettier".

And you may wish to add even more "functionality".

The rest of this manual is written to add all the "other stuff" to help you make additional changes you may want.

A FEW NOTES ON THIS NEW VERSION

RECENT CHANGES

Version 2.1.x, a prior version, introduced several significant enhancements, including:

- Formatting the Error Page
- Saving Responses in a Server File
- Attempting Compatibility with Windows Servers
- "Excluded" Fields

Even though we are now in version 2.2.x these enhancements are still relevant, particularly for users upgrading from a prior version or users looking for the newest features of the script.

FORMATTING THE ERROR PAGE

Version 2.1.1 substantially increased the ability of the user to format the error page. This includes:

Formatting the Text of the 6 Components of the Error Page

1. The First Heading Line
2. The Second Heading Line
3. The Top Summary Text
4. The Error Messages
5. The Bottom Summary Text
6. The Footer Information

The user, through parameters change directly specify the Font Face, Font Size, Font Color and whether the font should be bold. (This is true except for Error Messages and Footer Information. For those two items, the "boldness" is controlled by the program and cannot be changed.

Alternatively, the user may specify for all 6 items the specific paragraph style (CSS class) to be specified. This requires that the user also specify an error page "template".

For these features see:

- *FormErrorPageHeading1*   page 83
- *FormErrorPageHeading2*   page 86
- *FormErrorPageLineOpening*   page 92
- *FormErrorPageErrorMsg*   page 77

- *FormErrorPageLineClosing*   page 89
- *FormErrorPageFooter*   page 80

Specifying an Error Page Template

The user may specify an error page template. The error page template is constructed as a normal HTML page, and specifies "where" error messages should be written. The script will write the HTML before the errors, write the errors, then write the HTML after the errors.

If you wish to format the 6 components of the error page, you must specify an error page template and must include a reference to styles (embedded or external) in that template.

For this feature see:

- *FormErrorTemplate*   page 95

SAVING THE RESPONSES IN A SERVER FILE

The user may specify a file[32] on the server into which the script will save responses to the form.

The file will be the same directory which the form script is installed unless a path is specified.

The file will be saved as a CSV file[33] and can then be downloaded and used.

If the file does not exist, it is created and the field names are posted as the first two rows.

If the file already exists, it is added to by each subsequent response.

For this feature see:

- *FormCSVFileName*   page 70

ATTEMPTING COMPATIBILITY WITH WINDOWS SERVERS

In an attempt to allow the script to better perform on Windows Servers, I've added a Hidden Field Parameter that will format emails in the non-standard way that Windows Servers require.

---

[32] Specifically, a CSV or Comma Separated Value file. See http://en.wikipedia.org/wiki/Comma-separated_values

[33] http://en.wikipedia.org/wiki/Comma-separated_values

For this feature see:

- *FormPleaseForgiveMyUseOfA   Server* on page *123*

MISCELLANEOUS CHANGES

Exclude Value in FormFieldNameLabelPlusList

This was a change previously made but not documented. It allows a field to be specified as present but to set it's default characteristic as "exclude" so it will not be reported unless specifically requested.

For this feature see:

- *FormFieldNameLabelPlusList*   page 114

SPECIFIC CHANGES FOR THIS LATEST RELEASE: VERSION 2.2.5

This release made one primary change:

- A "recent" version of PHP (5.3.0) introduced a "wrinkle" which causes an error that says: "Deprecated: Function ereg() is deprecated". As I found out about this change I also discovered that many hosts were not yet using the latest PHP (even though this change happened almost a year before – and even now, many hosts have not moved!). For those that have not moved the prior version (2.2.3) works just fine. The new version (2.2.5), tested for almost 2 years, works for both.

SUMMARY

The following changes have been identified but have not yet been completed:

- Updating the email edits to accommodate more unusual emails
- Allowing file uploads

UPDATING THE EMAIL EDITS

I used http://www.addedbytes.com/php/email-address-validation in the construction of my original email edits. I have discovered an update at http://www.addedbytes.com/blog/email-address-validation-v2. My plan is to study and incorporate this update, but I have not done this yet.

UPLOADING A FILE

I am evaluating changes which would allow the upload of a file, along with the form. This would be "sort of" an attachment. Preliminary investigation is still ongoing.

STRUCTURE OF THE SCRIPT

STRUCTURE OF HUGGINS' EMAIL FORM SCRIPT

Huggins' Email Form Script is written in PHP.

It is designed to be invoked from an HTML Web Form through the POST function.

It receives two types of data from the Web Form:

1. The "Visible Fields" containing the data input by the Form User, and
2. "Hidden Fields" containing configuration parameters supplied by the Form Designer. I refer to these as Hidden Field Parameters.

The script is designed so that multiple Web Forms on a site may use the same script without the need to modify the script. All configuration parameters are passed through the Hidden Field Parameters.

This includes:

- Lists of Form Fields to be reported in the email
- Validation rules for the Form Fields
- Spambot protections for the form/script
- Formatting options for the email
- Lists of addresses to be emailed
- From address for the confirmation email
- Name of the error page template file
- Formatting options for the error page
- Name of the optional CSV file to be created
- Destination URL after processing

HIDDEN FIELD PARAMETERS

WHAT IS THIS SECTION

This section provides full documentation for all the available Hidden Field Parameters used by Huggins' Email Form Script.

WHY USE THIS SECTION

This section enables Form Designers to fully customize the processing of the Web Form using Huggins' Email Form Script.

However, before using this section, new Form Designers may wish to complete the exercises presented in the sections beginning on page 13 and *Continuing the Quick Start* beginning on page 48.

Getting the Web Form working at a basic level will introduce the Form Designer to the functional concepts of the script and let the Form Designer more easily debug their use of the script.

Once the Web Form is working at a basic level, the parameter values can be modified and expanded to provide full functionality for the Web Form.

WHAT ARE HIDDEN FIELD PARAMETERS?

Huggins' Email Form Script receives two types of data from the Web Form:

1. The "Visible Fields" containing the data input by the Form User, and
2. "Hidden Fields" containing configuration parameters supplied by the Form Designer. I refer to these as Hidden Field Parameters.

The HTML form specifies these Hidden Field Parameters in this general format:

```
<input name="hidden-field-parameter-name" type="hidden"
value="parameter-value">
```

This section describes all the Hidden Field Parameters for Huggins' Email Form Script.

SUB-PARAMETERS

Some of the Hidden Field Parameters have multiple "sub-parameters". Sub-parameters are separated either by commas or by a vertical bar ("|"). Anywhere you can use a comma, you can use a vertical bar.

The vertical bar can be used to "cluster" groups of sub-parameters.

For example, MsgxAddrList has sub-parameters identifying email addresses. Each email address is identified by three sub-parameters.

Consider that you wished to identify three different emails to receive the output of the form: Dan Foreman [DForeman@JSend.com], Linda Nulty [LNulty@BigBlueBanaBoat.com] and Monty Python [MPython@ILikeMontyPython.com]. These three email addresses could be specified as:

```
<input name="Msg1AddrList" type="hidden" value="
        Dan Foreman, DForeman, JSend.com,
        Linda Nulty, LNulty, BigBlueBananaBoat.com,
        Monty Python, MPython, ILikeMontyPython.com">
```

Alternatively, the Form Designer may wish to use the "vertical bar" as a "comma substitute" for improved readability, like this.

```
<input name="Msg1AddrList" type="hidden" value="
        Dan Foreman, DForeman, JSend.com|
        Linda Nulty, LNulty, BigBlueBananaBoat.com|
        Monty Python, MPython, ILikeMontyPython.com">
```

The vertical bar works exactly like a comma. As far as the script is concerned, the two specifications are identical. Some will prefer commas only. Some will prefer the vertical bar. Both options are available. You choose.

NOTES ON MESSAGE 2

MULTIPLE MESSAGES

Sometimes you may wish to send the information in two different formats to two different sets of people.

For example, you may wish to send the message to people in one format, but also send the message to an automated message script in a second format.

The script has the ability to do this.

Note however, that you should *not* use Message 2 to send the same format message to a second address. If you want to send the same message in the same format to two (or even three or four or five or six or more) addresses, just specify all these addresses in *Msg1AddrList* (page 134).

The second message is referred to as Message 2 and the Form Fields for the second message are prefixed with "Msg2".

✓ **Repeated Caution**: Do not use Message 2 to send a copy of Message 1 to a second person. The field Msg1AddrList can accept 1 or multiple email addresses. If you want to send Message 1 to multiple people, put the multiple addresses into Msg1AddrList.

USE OF MESSAGE 2 (MSG2) PARAMETERS

You need to use Message 2 (Msg2) parameters only if you wish to send a Message 2 email. If you do not intend to send a Message 2 email, you may omit all Message 2 (Msg2) parameters.

COPYING PARAMETERS FROM MESSAGE 1

If you wish for a parameter for Message 2 to be the same as the corresponding parameter for Message 1, use a value of an asterisk ("*"). The parameter processing logic in this program frequently recognizes an asterisk ("*") to mean "the same value as the similar parameter previously specified" and can be thought of as a "back reference". This makes entry and maintenance of parameters much easier.

NOTES ON THE ECHO/CONFIRMATION MESSAGE

TELLING THE FORM USER WHAT THEY DID

Sometimes you may wish to echo (or confirm) the information back to the person who fills out the Web Form. And, you may wish for the email that does this to have a different format.

The script has the ability to do this.

This message is referred to as the Echo/Confirmation Message and the Form Fields for the second message are prefixed with MsgEcho.

USE OF ECHO/CONFIRMATION MESSAGE (MSGECHO) PARAMETERS

You need to use Echo/Confirmation Message (MsgEcho) parameters only if you wish to send an Echo/Confirmation Message email. If you do **not** intend to send an Echo/Confirmation Message email, you may omit all Echo/Confirmation Message (MsgEcho) parameters.

COPYING FROM MESSAGE 2 OR MESSAGE 1

If you wish for a parameter for the Echo/Confirmation Message to be the same as the corresponding parameter for Message 2 (or Message 1 if there is no Message 2), use a value of an asterisk ("*").

HIDDEN FIELD PARAMETER SUMMARY REFERENCE TABLES

These Hidden Field Parameters are listed by group, and in alphabetical order within the group. These tables serves as summaries of the Hidden Field Parameters.

More complete descriptions follow these Summary Reference Tables:

- **Form Level Parameters**
  Parameters which apply to the form generally

- **Message Level Parameters**
  Parameters which are specific to one or more of the email messages

CHANGED AND NEW PARAMETERS

In the tables that follow, the column labeled with an asterisk ("*") shows parameters which were changed in Version 2.0 of this script ("C") and parameters which are new in Version 2.0 of this script("N").This information was primarily intended for those migrating from Version 1.0 many years ago and can be ignored for new/recent users.

MESSAGE LEVEL PARAMETERS

In the table for Message Level Parameters, the columns labeled "1", "2" and "E" indicate whether the parameter is legal for Message 1, Message 2 and the Echo/Confirmation Message, respectively.

Form Level Parameters Summary Table

NB: The column labeled asterisk ("*") shows parameters changed ("C") or new ("N") and the Version of that change.

Hidden Field Parameter & Edits	*	Description	Page
CopyUser *Edits: C003 with FormEchoUser, C004, C022*	C2.0	Indicates (yes/no) whether an Echo/Confirmation Email should be sent. ✓ *The default format for the Echo/Confirmation Email is the same as Msg2 (if specified) or Msg1.* ✓ *This field was replaced in V2.0 by FormEchoUser. In addition, the default value has changed to "yes". Before V2.0, Echo/Confirmation emails were not sent by default. In V2.0 they are sent by default.*	69
FormCSVFileName *Edits: C022*	N2.1	Specifies (optionally) the name of a CSV File to receive the contents of the form entries. If the file does not exist, it is created. If it does exist, information is added.	70
FormEchoUser *Edits: C003 with CopyUser, C004*	N2.0	Indicates (yes/no) whether an Echo/Confirmation Email should be sent. (Default is yes.) ✓ *The default format for the Echo/Confirmation Email is the same as Msg2 (if specified) or Msg1.* ✓ *This new parameter (V2.0) replaces CopyUser. The default value of "yes" is a change. Before V2.0, Echo/Confirmation emails were not sent by default. In V2.0 they are sent by default.*	72
FormEmailField *Edits: C001, C003 with FormEmailFieldList, C022* *Field pointed to is edited with: E005, E006, E007, E008, E009*	C2.0	The Field Name of the Form Field on the Web Form into which the Form User can enter his/her email address. ✓ *This field was replaced in V2.0 by FormEmailFieldList.*	73
FormEmailFieldList *Edits: C001, C003 with FormEmailField, C022* *Field pointed to is tested with: E005, E006, E007, E008, E009* *Confirmation field is edited with E011*	N2.0	The Field Name of the Form Field on the Web Form into which the Form User can enter his/her email address. This parameter also specifies (optionally) the name of the email confirmation field. ✓ *This new parameter (V2.0) replaces FormEmailField.*	75
FormErrorPageErrorMsg *C022, C026, C027, C028, C029, C030, C031, C032, C033, C034, C035, C036*	N2.1	Allows specification of the font face, font size and color of the error messages on the error page. Alternatively, allows a CSS paragraph class to be specified for formatting the error messages.	77
FormErrorPageFooter *C022, C026, C027, C028, C029, C030, C031, C032, C033, C034, C035, C036*	N2.1	Allows specification of the font face, font size and color of the footer on the error page. Alternatively, allows a CSS paragraph class to be specified for formatting the footer.	80

Form Level Parameters Summary Table

NB: The column labeled asterisk ("*") shows parameters changed ("C") or new ("N") and the Version of that change.

Hidden Field Parameter & Edits	*	Description	Page
FormErrorPageHeading1 *C022, C026, C027, C028, C029, C030, C031, C032, C033, C034, C035*	N2.0 C2.1	The text to be displayed as the primary heading on the error page. Also allows specification of the font face, font size, color and "boldness" of the primary heading. Alternatively, allows a CSS paragraph class to be specified for formatting the primary heading.	83
FormErrorPageHeading2 *C022, C026, C027, C028, C029, C030, C031, C032, C033, C034, C035*	N2.0 C2.1	The text to be displayed as the secondary heading on the error page. Also allows specification of the font face, font size, color and "boldness" of the secondary heading. Alternatively, allows a CSS paragraph class to be specified for formatting the secondary heading.	86
FormErrorPageLineClosing *C022, C026, C027, C028, C029, C030, C031, C032, C033, C034, C035*	N2.0 C2.1	The text to be displayed after all errors are reported on the error page. Also allows specification of the font face, font size, color and "boldness" of the closing line. Alternatively, allows a CSS paragraph class to be specified for formatting the closing line.	89
FormErrorPageLineOpening *C022, C026, C027, C028, C029, C030, C031, C032, C033, C034, C035*	N2.0 C2.1	The text to be displayed immediately before reporting the errors on the error page. Also allows specification of the font face, font size, color and "boldness" of the opening line. Alternatively, allows a CSS paragraph class to be specified for formatting the opening line.	92
FormErrorPageTemplate *C022, C024, C025*	N2.1	The file containing the HTML template for the error page.	95
FormErrorPageTitle *C022*	N2.0	The HTML Title for the error page. This title appears in the Window Title Bar.	96
FormErrorPageTrackingInfo *Edits: C004*	N2.0	Indicates (yes/no) whether Tracking Information should be displayed on any Error Page. (Default is yes.)	97

Form Level Parameters Summary Table

NB: The column labeled asterisk ("*") shows parameters changed ("C") or new ("N") and the Version of that change.

Hidden Field Parameter & Edits	*	Description	Page
FormFieldNameEditList *Edits: C001, C006, C012, C013, C014, C022* *For drop down edit: E013* *For compare edit: E012* *For captcha edit: E001* *For attraction edit: E014* *For equal edit: E015* *For not equal edit: E016* *For integer edit: E017, E018, E019* *For number edit: E020, E021, E022, E023* *For text edit: E024* *For email edit: E011*	N2.0	A list of field names, the edits to be performed on them and the related values for these edits. ✓ *This is one of the more complex parameters introduced in V2.0. Please read the documentation carefully.*	97
FormFieldNameLabelPlusList *Edits: C002, C005, C016, C017, C022* *For fields pointed to: E002, E003, E004*	N2.0 C2.1	A master list of all field names, their labels, their required status and their minimum and maximum sizes. ✓ *Version 2.0 allowed a status of "e" for "excluded", but this was not documented until Version 2.1.*	114
FormNameField *Edits: C001, C003 with FormNameFieldList, C022*	C2.0	The Field Name of the Form Field on the Web Form into which the Form User can enter his/her name. ✓ *This field was replaced in V2.0 by FormNameFieldList.*	118
FormNameFieldList *Edits: C001. C003 with FormNameField, C022*	N2.0	The list of field names which will be put together to construct a full name. For example, firstname and lastname. ✓ *This new parameter (V2.0) replaces FormNameField.*	119
FormNextURL *Edits: C003 with NextURL, C022*	N2.0	The *fully qualified* URL of the page to be displayed after the Web Form is processed. ✓ *This new parameter (V2.0) replaces NextURL.*	120
FormOutputFieldInfo *Edits: C004*	N2.0	Indicates (yes/no) whether to scan the form and output the field information in HTML format to facilitate form setup.	121
FormPleaseForgiveMyUseOfAWindowsServer *Edits: C004*	N2.1	Indicates (yes/no) whether emails should be degraded for a Windows Server. If yes, the emails are sent using email address only and ignoring any affiliated name.	123
FormTestEmailDomain *Edits: C004*	N2.0	Indicates (yes/no) whether to test the domain name of the email address to ensure it is a real domain name configured for email. (Default is yes.)	125

Form Level Parameters Summary Table

NB: The column labeled asterisk ("*") shows parameters changed ("C") or new ("N") and the Version of that change.

Hidden Field Parameter & Edits	*	Description	Page
FormTestEmailFormat *Edits: C004*	N2.0	Indicates (yes/no) whether to test the format of the data entered in the email address field to ensure it is a valid format. (Default is yes.)	126
FormTestFieldMaxLengths *Edits: C004*	N2.0	Indicates (yes/no) whether to test the fields against their specified maximum lengths. (Default is yes.)	127
FormTestFieldMinLengths *Edits: C004*	N2.0	Indicates (yes/no) whether to test the fields against their specified minimum lengths. (Default is yes.)	128
FormTestInjectionExploits *Edits: C004*	N2.0	Indicates (yes/no) whether to test the Form Fields for possible Injection Exploits. (Default is yes.) ✓ *Hidden Form Fields are also tested for Injection exploits. See C018. The testing of Hidden Form Fields cannot be turned off.*	129
NextURL *Edits: C003 with FormNextURL, C022*	C2.0	The *fully qualified* URL of the page to be displayed after the Web Form is processed. ✓ *This field was replaced in V2.0 by FormNextURL.*	130

Message Level Parameters

NB: The column labeled asterisk ("*") shows parameters changed ("C") or new ("N") and the version of that change or new.
NB: The columns labeled with 1, 2, E indicate to which message(s) (1, 2 or Echo) the parameter applies.
NB: The column labeled "B", indicates which variables can use an asterisk for "Back Reference".

Hidden Field Parameter & Edits	*	Description	1	2	E	B	Page
MsgEchoFromAddr *Edits: C007, C008, C009, C010 C011, C015, C022*		The "from" address to be used when sending the Echo/Confirmation Message to the person who filled out the Web Form. This value defaults to the first address of Msg1AddrList.			X		132
MsgxLabelSubjectAndAddressesDropDown *Edits: C007, C008, C009, C010, C011, C016, C021, C022*	N2.0	A drop down parameter allowing selection of email addresses and the subject that goes with them. ✓ *Note that MsgxAddrList and MsgxSubject are not permitted if this parameter is used.*	X	X			154
MsgxAddrList *Edits: C007, C008, C009, C010, C011, C016, C021, C022*		The email addresses of the people to receive the email. ✓ *Note that Msg1AddrList or Msg1LabelSubjectAndAddressesDropDown are the only Hidden Parameter required for the script to function.* ✓ *Note that MsgxLabelSubjectAndAddressesDropDown is not permitted if either MsgxAddrList or MsgxSubject is used.*	X	X			134
MsgxDefaultFromAddr *Edits: C007, C008, C009, C010 C011, C015, C022 (Msg1 only)*		The email address to be shown as the "from address" for Msg1 and Msg2, if no email address is available from the Form User.	X	X		*	137
MsgxFieldLabelValueSeparator *C022 (Msg1 only)*	N2.0	Specifies the text to use in an email to separate the Field Label and the Field Value. Special values are also used to indicate different email format.	X	X	X	*	139
MsgxFieldNameExcludeList *Edits: C022 (Msg1 only)*	N2.0	A list of fields to exclude from a message. This is useful when one message should have the same fields as a prior message, except for some.	X	X	X	*	141
MsgxFieldNameLabelList *Edits: C002, C016 (this edit temporarily suspended pending problem resolution with Format specifications), C022 (Msg1 only)*		The Field Names and Field Reporting Labels of the Form Fields to be shown in the email. ✓ *Note that if omitted and needed, the list for the previous message is used. For example, if an Echo/Confirmation is being sent but MsgEchoFieldNameLabelList is not specified, the list from Msg2 or Msg1 will be used.*	X	X	X	*	142

Message Level Parameters

NB: The column labeled asterisk ("*") shows parameters changed ("C") or new ("N") and the version of that change or new.

NB: The columns labeled with 1, 2, E indicate to which message(s) (1, 2 or Echo) the parameter applies.

NB: The column labeled "B", indicates which variables can use an asterisk for "Back Reference".

Hidden Field Parameter & Edits	*	Description	1	2	E	B	Page
MsgxFieldNameValueSubstitutionList *Edits: C002, C016, C022 (Msg1 only)*		Specifies data values and substitute values for the reporting in the email.	X	X	X	*	146
MsgxFieldNameValueSubstitutionList2 *Edits: C002, C016, C022 (Msg1 only)*		Specifies data values and substitute values for the reporting in the email.	X	X	X	*	151
MsgxForceDefaultFromAddr *Edits: C004*		Identifies whether to force the use of MsgxDefaultFromAddr. ✓ *The default value is "no".*	X	X			153
MsgxSubject *Edits: C021, C022 (Msg1 only)*		The email subject. ✓ *Note that MsgxLabelSubjectAndAddressesDropDown is not permitted if either MsgxAddrList or MsgxSubject is used.*	X	X	X	*	154
MsgxSubjectField *Edits: C001, C022 (Msg1 only)*	N2.0	*The Field Name of a Form Field into which the Form User can specify the subject for the email.*	X	X	X	*	159
MsgxTextBottom *C022 (Msg1 only)*		Brief text to be printed at the bottom of the email.	X	X	X		160
MsgxTextTop *C022 (Msg1 only)*		Brief text to be printed at the top of the email.	X	X	X		161

FORM LEVEL PARAMETERS – DETAIL DESCRIPTIONS

Form Level Parameters are those parameters which apply to the form generally.

COPYUSER (DEPRECATED)

Description

This parameter was deprecated in Version 2.0.[34]

It was replaced by *FormEchoUser* (page 70). This replacement changes the name of the parameter for naming consistency.

If used, this parameter has exactly the same rules as *FormEchoUser*.

Old (pre-version 2.0) forms do not need to change this parameter. The new script recognizes and uses the old parameter for backward compatibility.

New implementations should not use this field and should use *FormEchoUser* instead.

Values

See *FormEchoUser* (page 70).

Default

The default value is yes. This is a version change. See below.

Examples

See *FormEchoUser* (page 70).

Version Change

Note this change. The previous default value for CopyUser was "no". The new default value both for CopyUser and for FormEchoUser is "yes". This may change behavior of a form.

Edits

- C003: Both *CopyUser* and *FormEchoUser* may not be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 166)
- C022: This field may not have a value of asterisk ("*"). (page 173)

---

[34] A deprecated item is one that has been outdated by newer constructs. Deprecated items are defined in this document but are clearly marked as deprecated. Deprecated items still work, but may become obsolete at any time in future versions of this software. The script may still support deprecated items for backward compatibility. However, users should begin to use the substitutes. For a list, see the section *Deprecated and Obsolete Items* on page 198.

FORMCSVFILENAME

Description

This parameter specifies whether or not to write the form contents to a CSV File[35] on the server.

If the parameter is empty, no writing occurs.

If the parameter is not empty, the form contents are written to the file name.

If the file does not exist, it is created.

If the file already exists, it is added to.

NB: The parameter is ***not*** downshifted. If you specify capital letters, the script will use those letters. Be absolutely sure you specify this parameter using the correct case.

Values

Specify the "simple" name of a file.

Do not specify "http", the domain name, the disk or the path. The file will be in the same directory/folder as the script.

Be sure that the directory has "sufficient" write properties (possibly up to the max "777, allowing the owner, group and world to read, write and execute the files in that directory). [36]

Failure to confirm the write properties may cause this functionality to fail without any error message.

---

[35] CSV stands for "Comma Separated Values". This is a simple file structure that is supported by common spreadsheet software, like Excel. For more information see en.wikipedia.org/wiki/Comma-separated_values and en.wikipedia.org/wiki/CSV_application_support.

[36] "777" is *Owner*: Read, Write, Execute; *Group*: Read, Write, Execute; *Other*: Read, Write, Execute; this allows everyone to do everything; more restrictive permissions may work; For example; depending on the environment, if your program qualifies as "owner", then reducing the capabilities of "group" and "other" to "5" resulting in "755" as a possibility; A "757" is also a possibility and is more restrictive than a full open of "777"; In general, use a '755' if that will work, a "757" if the "755" doesn't work and then a "777" if necessary; this file has more information for background www.tldp.org/LDP/intro-linux/html/sect_03_04.html

Examples

For a file name of "form-values.csv" use:

```
<input name="FormCSVFileName" type="hidden"
value="form-values.csv">
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)

FORMECHOUSER

Description

This parameter specifies whether or not to send an Echo/Confirmation Email to the Form User.

The Echo/Confirmation Email is an email sent to the Form User to report the information the Form User typed into the Web Form.

This new parameter (V2.0) renames/replaces *CopyUser*.

Values

Legal values are "y", "yes", "n" and "no". Case does not matter.

Set the value to "yes" to send an Echo/Confirmation Email.

Set the value to "no" to *not* send an Echo/Confirmation Email.

Default

The default value is "yes".

Examples

A value of "no" is specified as:

```
<input name="FormEchoUser" type="hidden"
value="no">
```

Version Change

Note this change. The previous default value for CopyUser was "no". As of Version 2.0.0, the new default value both for CopyUser and for FormEchoUser is "yes". This may change behavior of a form.

Edits

- C003: Both *CopyUser* and *FormEchoUser* may not be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 166)
- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

FORMEMAILFIELD (DEPRECATED)

Description

This parameter was deprecated in Version 2.0.[37]

It was replaced by *FormEmailFieldList* (see page 97). This replacement is to change the name of the parameter for naming consistency and also to add the ability to specify a email confirmation field.

If used, this parameter has exactly the same rules as *FormEmailFieldList*, except that no email confirmation field may be specified.

Old (pre-version 2.0) forms do not need to change this parameter. The new script recognizes and uses the deprecated parameter for backward compatibility.

However, new implementations should not use this parameter and should instead use *FormEmailFieldList*.

Values

See *FormEmailFieldList* on page 97.

Default

See *FormEmailFieldList* on page 97.

Examples

See *FormEmailFieldList* on page 97.

Edits

- C003: Both *FormEmailField* and *FormEmailFieldList* may not be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 86)
- C022: This field may not have a value of asterisk ("*"). (page 173)
- E005: Check email address for number of characters and "@" signs (page 178)
- E006: Check email address for format of local address part (page 178)

---

[37] A deprecated item is one that has been outdated by newer constructs. Deprecated items are defined in this document but are clearly marked as deprecated. Deprecated items still work, but may become obsolete at any time in future versions of this software. The script may still support deprecated items for backward compatibility. However, users should begin to use the substitutes. For a list, see the section *Deprecated and Obsolete Items* on page 198.

- E007: Check email address for number of parts in domain name (page 178)
- E008: Check email address for format of domain name (page 179)
- E009: Check email address for whether domain name can receive email (page 179)

FORMEMAILFIELDLIST

Description

This parameter specifies the Field Name of the Form Field on the Web Form into which the Form User can enter his/her email address.

It also specifies, optionally, the Field Name of the Form Field on the Web Form which serves to confirm the email field (i.e., by having the Form User enter it again.)

This field is optional. The Web Form does not have to have an email field. However, I recommend that all Web Forms do have such an email field *at least as an optional field* so that Form Users, if they wish, can specify an email address and receive a copy of the data they submit.

Unless you override it, this will automatically set several other actions:

- The email address entered will be tested for format
- The email address entered will be tested for domain name
- The email address and email confirmation address will be compared
- The email address entered will receive an echo/confirmation email

This new parameter (V2.0) renames/replaces *FormEmailField*.

See also *FormFieldNameEditList, Field Edit 7 – Text: Compare Two Fields*, on page 108.

Values

The value must be a Field Name of a Form Field on the Web Form.

Default

There is no default.

Examples

If the field containing the email address of the person is **email**, you would specify this as:

```
<input name="FormEmailFieldList" type="hidden" value="email">
```

If the field containing the email address of the person is **email**, and if you have an email confirmation field named **emailconfirmation** you would specify this as:

```
<input name="FormEmailFieldList" type="hidden" value="email,
emailconfirmation">
```

Edits

- C001: Referenced fields must exist on the form; the special fields ip, browser and referer are not allowed (page 166)
- C003: FormEmailField and Form Field Email Field may not both be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 86)
- C022: This field may not have a value of asterisk ("*"). (page 173)
- E005: Check email address for number of characters and "@" signs (page 178)
- E006: Check email address for format of local address part (page 178)
- E007: Check email address for number of parts in domain name (page 178)
- E008: Check email address for format of domain name (page 179)
- E009: Check email address for whether domain name can receive email (page 179)
- E011: Compare email address entered against the confirmation email address entered (page 180)

FORMERRORPAGEERRORMSG

Description

This parameter specifies the format of error messages on the Error Page.

This parameter allows the Form Designer to customize the formatting of these error messages to simulate the look/feel of his website.

See *The Error Page* on page 163.

Similar Parameters

There are six Error Page Format parameters. The specifications for these six parameters have been designed to be almost identical.

These are: (1) *FormErrorPageErrorMsg*, (2) *FormErrorPageFooter*, (3) *FormErrorPageHeading1*, (4) *FormErrorPageHeading2*, (5) *FormErrorPageLineClosing* and (6) *FormErrorPageLineOpening*.

The differences are:

1. *FormErrorPageErrorMsg* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other Error Page Format parameters.

2. *FormErrorPageFooter* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other similar Error Page Format parameters.

Sub-parameters

The parameter contains a string of sub-parameters, separated by commas, in one of two formats:

```
Format 1: does not specify a CSS paragraph class
up to 5 values

"error-msg-text-empty, error-msg-font-faces, error-msg-font-size, error-msg-
font-color, error-msg-font-bold-empty"

(NB: not all 5 values are required; empty values may be truncated)


Format 2: specifies a CSS paragraph class
1 real value and 5 empty values

"error-msg-text-empty, , , , , error-msg-paragraph-class"

(NB: to specify error-msg-paragraph-class, the four "middle" sub-parms must
be omitted)
```

error-msg-text-empty: this value must be omitted. It is an error to specify a value for this sub-parameter. In similar Error Page Format parameters, the sub-parameter in position one contains text to be used on the page. However, for error messages, the script provides the text and this parameter may not provide any text.

error-msg-font-faces: the font faces to be used; multiple faces are separated by asterisks. The default for error-msg-font-faces is "Georgia, serif".

error-msg-font-size: a number between 1 and 12 indicating the display font size. The default for error-msg-font-size is "2".

error-msg-font-color: the color of the font. Colors must be specified as a pound sign ("#") followed by a 6 character hex number (e.g., "#000000" through "#ffffff")"or must be a standard accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". The default for error-msg-font-color is "red".

error-msg-font-bold-empty: this value must be omitted. It is an error to specify a value for this sub-parameter. In similar Form Error Page parameters, the sub-parameter in position five specifies whether or not to bold the text. However, for error messages, the script controls bolding and this parameter is not permitted.

error-msg-paragraph-class: a CSS class the script will apply to the error messages paragraphs. This sub-parameter may only be provided in Format 2. In addition, if *any* Error Page Format parameter uses Format 2, then *all* Error Page format parameters must use Format 2. This also requires that the *FormErrorPageTemplate* hidden parameter be specified (see page 95).

Defaults

error-msg-font-faces:	"Georgia, serif"
error-msg-font-size:	"2"
error-msg-font-color:	"red"

Examples

If you wish to have this field use the Verdana face, with a size of 6, and a color of blue, you would specify this as:

```
<input name="FormErrorPageErrorMsg" type="hidden"
value=", Verdana, 6, blue">
```

If you wish to have this field use the CSS paragraph class of "error-msg" you would specify this as:

```
<input name="FormErrorPageErrorMsg" type="hidden"
value=", , , , , error-msg"
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)
- C026: Number in a sub-parameter contains an illegal character, (page 174)
- C027: Number in a sub-parameter contains too many periods. (page 174)
- C028: Number in a sub-parameter has too low a value. (page 174)
- C029: Number in a sub-parameter has too high a value. (page 174)
- C030: Illegal color specification in a sub-parameter – not a number or color value. (page 175)
- C031: Illegal color specification in a sub-parameter – wrong length number. (page 175)
- C032: Illegal color specification in a sub-parameter – invalid hex number. (page 175)
- C033: Illegal bold flag – not yes or no. (page 175)
- C034: Expected CSS Class Value not found. (page 176)
- C035: Unexpected "old style" error page format found. (page 176)
- C036: Sub-parameter isn't blank (null). (page 176)

FORMERRORPAGEFOOTER

Description

This parameter specifies the format of the footer information on the Error Page.

This parameter allows the Form Designer to customize the formatting of this footer information to simulate the look/feel of his website.

See *The Error Page* on page 163.

Similar Parameters

There are six Error Page Format parameters. The specifications for these six parameters have been designed to be almost identical.

These are: (1) *FormErrorPageErrorMsg*, (2) *FormErrorPageFooter*, (3) *FormErrorPageHeading1*, (4) *FormErrorPageHeading2*, (5) *FormErrorPageLineClosing* and (6) *FormErrorPageLineOpening*.

The differences are:

1. *FormErrorPageErrorMsg* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other Error Page Format parameters.

2. *FormErrorPageFooter* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other similar Error Page Format parameters.

Sub-parameters

The parameter contains a string of sub-parameters, separated by commas, in one of two formats:

```
Format 1: does not specify a CSS paragraph class
up to 5 values

"footer-text-empty, footer-font-faces, footer-font-size,
footer-font-color, footer-font-bold-empty"

(NB: not all 5 values are required; empty values may be truncated)


Format 2: specifies a CSS paragraph class
1 real value and 5 empty values

" footer-text-empty, , , , , footer-paragraph-class"

(NB: to specify footer-paragraph-class, the four "middle" values must be
omitted)
```

footer-text-empty: this value must be omitted. It is an error to specify a value for this sub-parameter. In similar Error Page Format parameters, the sub-parameter in position one contains text to be used on the page. However, for the footer, the script provides the text and this parameter may not provide any text.

footer-font-faces: the font faces to be used; multiple faces are separated by asterisks. The default for footer-font-faces is "Tahoma, Arial, Helvetica, sans-serif".

footer-font-size: a number between 1 and 12 indicating the display font size. The default for footer-font-size is "1".

footer-font-color: the color of the font. Colors must be specified as a pound sign ("#") followed by a 6 character hex number (e.g., "#000000" through "#ffffff")"or must be a standard accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". The default for footer-font-color is "#000000" (black).

footer-font-bold-empty: this value must be omitted. It is an error to specify a value for this sub-parameter. In similar Form Error Page parameters, the sub-parameter in position five specifies whether or not to bold the text. However, for error messages, the script controls bolding and this parameter is not permitted.

footer-paragraph-class: a CSS class the script will apply to the error messages paragraphs. This sub-parameter may only be provided in Format 2. In addition, if *any* Error Page Format parameter uses Format 2, then *all* Error Page format parameters must use Format 2. This also requires that the *FormErrorPageTemplate* hidden parameter be specified (see page 95).

Defaults

footer-msg-font-faces:	"Tahoma, Arial, Helvetica, sans-serif"
footer-msg-font-size:	"1"
footer-msg-font-color:	"#000000" (black)

Examples

If you wish to have this field use the Verdana face, with a size of 6, and a color of blue, you would specify this as:

```
<input name="FormErrorPageFooter" type="hidden"
value=", Verdana, 6, blue">
```

If you wish to have this field use the CSS paragraph class of "error-page-footer" you would specify this as:

```
<input name="FormErrorPageFooter" type="hidden"
value=", , , , , error-page-footer"
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)
- C026: Number in a sub-parameter contains an illegal character, (page 174)
- C027: Number in a sub-parameter contains too many periods. (page 174)
- C028: Number in a sub-parameter has too low a value. (page 174)
- C029: Number in a sub-parameter has too high a value. (page 174)
- C030: Illegal color specification in a sub-parameter – not a number or color value. (page 175)
- C031: Illegal color specification in a sub-parameter – wrong length number. (page 175)
- C032: Illegal color specification in a sub-parameter – invalid hex number. (page 175)
- C033: Illegal bold flag – not yes or no. (page 175)
- C034: Expected CSS Class Value not found. (page 176)
- C035: Unexpected "old style" error page format found. (page 176)
- C036: Sub-parameter isn't blank (null). (page 176)

FORMERRORPAGEHEADING1

Description

This parameter specifies the format of the first heading line on the Error Page.

This parameter allows the Form Designer to customize the formatting of heading line to simulate the look/feel of his website.

See *The Error Page* on page 163.

Similar Parameters

There are six Error Page Format parameters. The specifications for these six parameters have been designed to be almost identical.

These are: (1) *FormErrorPageErrorMsg,* (2) *FormErrorPageFooter,*
(3) *FormErrorPageHeading1*, (4) *FormErrorPageHeading2,*
(5) *FormErrorPageLineClosing* and (6) *FormErrorPageLineOpening.*

The differences are:

1. *FormErrorPageErrorMsg* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other Error Page Format parameters.

2. *FormErrorPageFooter* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other similar Error Page Format parameters.

Sub-parameters

The parameter contains a string of sub-parameters, separated by commas, in one of two formats:

```
Format 1: does not specify a CSS paragraph class
up to 5 values

"heading1-text, heading1-font-faces, heading1-font-size,
heading1-font-color, heading1-font-bold-flag"

(NB: not all 5 values are required; empty values may be truncated)


Format 2: specifies a CSS paragraph class
1 real value and 5 empty values

"heading1-text, , , , , heading1-paragraph-class"

(NB: to specify heading1-paragraph-class, the four "middle" values must be
omitted)
```

heading1-text: the value to be displayed as heading1. The default for heading1-text is "HUGGINS' EMAIL FORM SCRIPT" (all caps).

heading1-font-faces: the font faces to be used; multiple faces are separated by asterisks. The default for heading1-font-faces is "Tahoma, Arial, Helvetica, sans-serif".

heading1-font-size: a number between 1 and 12 indicating the display font size. The default for heading1-font-size is "5".

heading1-font-color: the color of the font. Colors must be specified as a pound sign ("#") followed by a 6 character hex number (e.g., "#000000" through "#ffffff")"or must be a standard accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". The default for heading1-font-color is "#000000" (black).

heading1-font-bold-flag: whether to bold this line. The default for heading1-font-bold-flag is "yes".

heading1-paragraph-class: a CSS class the script will apply to heading1. This sub-parameter may only be provided in Format 2. In addition, if *any* Error Page Format parameter uses Format 2, then *all* Error Page format parameters must use Format 2. This also requires that the *FormErrorPageTemplate* hidden parameter be specified (see page 95).

Defaults

heading1-text:	"HUGGINS' EMAIL FORM SCRIPT" (all caps)
heading1-font-faces:	"Tahoma, Arial, Helvetica, sans-serif"
heading1-font-size:	"5"
heading1-font-color:	"#000000" (black)
heading1-font-bold-flag:	"yes"

Examples

If you wish to have this line have a text of "Script Error Page", use the Verdana face, with a size of 6, and a color of blue, but not bold, you would specify this as:

```
<input name="FormErrorPageHeading1" type="hidden"
value="Script Error Page, Verdana, 6, blue, no">
```

If you wish to have this field use the CSS paragraph class of "error-page-heading1" you would specify this as:

```
<input name="FormErrorPageHeading1" type="hidden"
value=", , , , , error-page-heading1"
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)
- C026: Number in a sub-parameter contains an illegal character, (page 174)
- C027: Number in a sub-parameter contains too many periods. (page 174)
- C028: Number in a sub-parameter has too low a value. (page 174)
- C029: Number in a sub-parameter has too high a value. (page 174)
- C030: Illegal color specification in a sub-parameter – not a number or color value. (page 175)
- C031: Illegal color specification in a sub-parameter – wrong length number. (page 175)
- C032: Illegal color specification in a sub-parameter – invalid hex number. (page 175)
- C033: Illegal bold flag – not yes or no. (page 175)
- C034: Expected CSS Class Value not found. (page 176)
- C035: Unexpected "old style" error page format found. (page 176)

FORMERRORPAGEHEADING2

Description

This parameter specifies the format of the first heading line on the Error Page.

This parameter allows the Form Designer to customize the formatting of heading line to simulate the look/feel of his website.

See *The Error Page* on page 163.

Similar Parameters

There are six Error Page Format parameters. The specifications for these six parameters have been designed to be almost identical.

These are: (1) *FormErrorPageErrorMsg*, (2) *FormErrorPageFooter*, (3) *FormErrorPageHeading1*, (4) *FormErrorPageHeading2*, (5) *FormErrorPageLineClosing* and (6) *FormErrorPageLineOpening*.

The differences are:

1. *FormErrorPageErrorMsg* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other Error Page Format parameters.

2. *FormErrorPageFooter* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other similar Error Page Format parameters.

Sub-parameters

The parameter contains a string of sub-parameters, separated by commas, in one of two formats:

```
Format 1: does not specify a CSS paragraph class
up to 5 values

"heading2-text, heading2-font-faces, heading2-font-size,
heading2-font-color, heading2-font-bold-flag"

(NB: not all 5 values are required; empty values may be truncated)


Format 2: specifies a CSS paragraph class
1 real value and 5 empty values

"heading2-text, , , , , heading2-paragraph-class"

(NB: to specify heading2-paragraph-class, the four "middle" values must be
omitted)
```

heading2-text: the value to be displayed as heading2. The default for heading2-text is "Form Processing Error Page".

heading2-font-faces: the font faces to be used; multiple faces are separated by asterisks. The default for heading2-font-faces is "Tahoma, Arial, Helvetica, sans-serif".

heading2-font-size: a number between 1 and 12 indicating the display font size. The default for heading2-font-size is "4".

heading2-font-color: the color of the font. Colors must be specified as a pound sign ("#") followed by a 6 character hex number (e.g., "#000000" through "#ffffff")"or must be a standard accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". The default for heading2-font-color is "#000000" (black).

heading2-font-bold-flag: whether to bold this line. The default for heading2-font-bold-flag is "yes".

heading2-paragraph-class: a CSS class the script will apply to heading2. This sub-parameter may only be provided in Format 2. In addition, if *any* Error Page Format parameter uses Format 2, then *all* Error Page format parameters must use Format 2. This also requires that the *FormErrorPageTemplate* hidden parameter be specified (see page 95).

Defaults

heading2-text: "Form Processing Error Page"

heading2-font-faces: "Tahoma, Arial, Helvetica, sans-serif"

heading2-font-size: "4"

heading2-font-color: "#000000" (black)

heading2-font-bold-flag: "yes"

Examples

If you wish to have this line have a text of "Error Page", use the Verdana face, with a size of 6, and a color of blue, but not bold, you would specify this as:

```
<input name="FormErrorPageHeading2" type="hidden"
value="Error Page, Verdana, 6, blue, no">
```

If you wish to have this field use the CSS paragraph class of "error-page-heading2" you would specify this as:

```
<input name="FormErrorPageHeading2" type="hidden"
value=", , , , , error-page-heading2"
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)
- C026: Number in a sub-parameter contains an illegal character, (page 174)
- C027: Number in a sub-parameter contains too many periods. (page 174)
- C028: Number in a sub-parameter has too low a value. (page 174)
- C029: Number in a sub-parameter has too high a value. (page 174)
- C030: Illegal color specification in a sub-parameter – not a number or color value. (page 175)
- C031: Illegal color specification in a sub-parameter – wrong length number. (page 175)
- C032: Illegal color specification in a sub-parameter – invalid hex number. (page 175)
- C033: Illegal bold flag – not yes or no. (page 175)
- C034: Expected CSS Class Value not found. (page 176)
- C035: Unexpected "old style" error page format found. (page 176)

FORMERRORPAGELINECLOSING

Description

This parameter specifies the format of the closing line on the Error Page.

This parameter allows the Form Designer to customize the formatting of heading line to simulate the look/feel of his website.

See *The Error Page* on page 163.

Similar Parameters

There are six Error Page Format parameters. The specifications for these six parameters have been designed to be almost identical.

These are: (1) *FormErrorPageErrorMsg*, (2) *FormErrorPageFooter*, (3) *FormErrorPageHeading1*, (4) *FormErrorPageHeading2*, (5) *FormErrorPageLineClosing* and (6) *FormErrorPageLineOpening*.

The differences are:

1. *FormErrorPageErrorMsg* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other Error Page Format parameters.

2. *FormErrorPageFooter* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other similar Error Page Format parameters.

Sub-parameters

The parameter contains a string of sub-parameters, separated by commas, in one of two formats:

```
Format 1: does not specify a CSS paragraph class
up to 5 values

"line-closing-text, line-closing-font-faces, line-closing-font-size,
line-closing-font-color, line-closing-font-bold-flag"

(NB: not all 5 values are required; empty values may be truncated)


Format 2: specifies a CSS paragraph class
1 real value and 5 empty values

"line-closing-text, , , , , line-closing-paragraph-class"

(NB: to specify line-closing-paragraph-class, the four "middle" values must
be omitted)
```

line-closing-text: the value to be displayed as line-closing. The default for line-closing-text is "Please press the "Back" button, correct the errors and resubmit."

line-closing-font-faces: the font faces to be used; multiple faces are separated by asterisks. The default for line-closing-font-faces is "Tahoma, Arial, Helvetica, sans-serif".

line-closing-font-size: a number between 1 and 12 indicating the display font size. The default for line-closing-font-size is "2".

line-closing-font-color: the color of the font. Colors must be specified as a pound sign ("#") followed by a 6 character hex number (e.g., "#000000" through "#ffffff")"or must be a standard accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". The default for line-closing-font-color is "#000000" (black).

line-closing-font-bold-flag: whether to bold this line. The default for line-closing-font-bold-flag is "yes".

line-closing-paragraph-class: a CSS class the script will apply to line-closing. This sub-parameter may only be provided in Format 2. In addition, if *any* Error Page Format parameter uses Format 2, then *all* Error Page format parameters must use Format 2. This also requires that the *FormErrorPageTemplate* hidden parameter be specified (see page 95).

Defaults

line-closing-text:　"Please press the "Back" button, correct the errors and resubmit."

line-closing-font-faces:　"Tahoma, Arial, Helvetica, sans-serif"

line-closing-font-size:　"2"

line-closing-font-color:　"#000000" (black)

line-closing-font-bold-flag:　"yes"

Examples

If you wish to have this line have a text of "Go back and fix", use the Verdana face, with a size of 6, and a color of blue, but not bold, you would specify this as:

```
<input name="FormErrorPageLineClosing" type="hidden"
value="Go back and fix, Verdana, 6, blue, no">
```

If you wish to have this field use the CSS paragraph class of "error-page-line-closing" you would specify this as:

```
<input name="FormErrorPageLineClosing" type="hidden"
value=", , , , , error-page-line-closing"
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)
- C026: Number in a sub-parameter contains an illegal character, (page 174)
- C027: Number in a sub-parameter contains too many periods. (page 174)
- C028: Number in a sub-parameter has too low a value. (page 174)
- C029: Number in a sub-parameter has too high a value. (page 174)
- C030: Illegal color specification in a sub-parameter – not a number or color value. (page 175)
- C031: Illegal color specification in a sub-parameter – wrong length number. (page 175)
- C032: Illegal color specification in a sub-parameter – invalid hex number. (page 175)
- C033: Illegal bold flag – not yes or no. (page 175)
- C034: Expected CSS Class Value not found. (page 176)
- C035: Unexpected "old style" error page format found. (page 176)

FORMERRORPAGELINEOPENING

Description

This parameter specifies the format of the opening line on the Error Page.

This parameter allows the Form Designer to customize the formatting of heading line to simulate the look/feel of his website.

See *The Error Page* on page 163.

Similar Parameters

There are six Error Page Format parameters. The specifications for these six parameters have been designed to be almost identical.

These are: (1) *FormErrorPageErrorMsg*, (2) *FormErrorPageFooter*, (3) *FormErrorPageHeading1*, (4) *FormErrorPageHeading2*, (5) *FormErrorPageLineClosing* and (6) *FormErrorPageLineOpening*.

The differences are:

1. *FormErrorPageErrorMsg* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other Error Page Format parameters.

2. *FormErrorPageFooter* may not specify the first sub-parameter nor the fifth sub-parameter. This is where the "text" and "bold flag" are positioned in the other similar Error Page Format parameters.

Sub-parameters

The parameter contains a string of sub-parameters, separated by commas, in one of two formats:

```
Format 1: does not specify a CSS paragraph class
up to 5 values

"line-opening-text, line-opening-font-faces, line-opening-font-size,
line-opening-font-color, line-opening-font-bold-flag"

(NB: not all 5 values are required; empty values may be truncated)


Format 2: specifies a CSS paragraph class
1 real value and 5 empty values

"line-opening-text, , , , , line-opening-paragraph-class"

(NB: to specify line-opening-paragraph-class, the four "middle" values must
be omitted)
```

line-opening-text: the value to be displayed as line-opening. The default for line-opening-text is "The form you submitted had the following errors:".

line-opening-font-faces: the font faces to be used; multiple faces are separated by asterisks. The default for line-opening-font-faces is "Tahoma, Arial, Helvetica, sans-serif".

line-opening-font-size: a number between 1 and 12 indicating the display font size. The default for line-opening-font-size is "2".

line-opening-font-color: the color of the font. Colors must be specified as a pound sign ("#") followed by a 6 character hex number (e.g., "#000000" through "#ffffff")"or must be a standard accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". The default for line-opening-font-color is "#000000" (black).

line-opening-font-bold-flag: whether to bold this line. The default for line-opening-font-bold-flag is "yes".

line-opening-paragraph-class: a CSS class the script will apply to line-opening. This sub-parameter may only be provided in Format 2. In addition, if *any* Error Page Format parameter uses Format 2, then *all* Error Page format parameters must use Format 2. This also requires that the *FormErrorPageTemplate* hidden parameter be specified (see page 95).

Defaults

line-opening-text: "Please press the "Back" button, correct the errors and resubmit."

line-opening-font-faces: "Tahoma, Arial, Helvetica, sans-serif"

line-opening-font-size: "2"

line-opening-font-color: "#000000" (black)

line-opening-font-bold-flag: "yes"

Examples

If you wish to have this line have a text of "Go back and fix", use the Verdana face, with a size of 6, and a color of blue, but not bold, you would specify this as:

```
<input name="FormErrorPageLineOpening" type="hidden"
value="Go back and fix, Verdana, 6, blue, no">
```

If you wish to have this field use the CSS paragraph class of "error-page-line-opening" you would specify this as:

```
<input name="FormErrorPageLineOpening" type="hidden"
value=", , , , , error-page-line-opening"
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)
- C026: Number in a sub-parameter contains an illegal character, (page 174)
- C027: Number in a sub-parameter contains too many periods. (page 174)
- C028: Number in a sub-parameter has too low a value. (page 174)
- C029: Number in a sub-parameter has too high a value. (page 174)
- C030: Illegal color specification in a sub-parameter – not a number or color value. (page 175)
- C031: Illegal color specification in a sub-parameter – wrong length number. (page 175)
- C032: Illegal color specification in a sub-parameter – invalid hex number. (page 175)
- C033: Illegal bold flag – not yes or no. (page 175)
- C034: Expected CSS Class Value not found. (page 176)
- C035: Unexpected "old style" error page format found. (page 176)

FORMERRORPAGETEMPLATE

Description

This parameter specifies the file containing an HTML template for the error page.

This parameter allows the Form Designer to customize the formatting of the page title simulate the look/feel of his website.

See *The Error Page* on page 163.

Values

This value should be the "short name"[38] of the file containing the template. The file should be stored in the same folder with the script

Default

None. If the value is blank (null) a template file is not used.

Special Consideration

The template file must have the characters "[[INSERT HEFS ERROR MESSAGES HERE]]" in the file. The string must be in all capitals. This string will not be displayed. Instead, errors will be inserted there in the page.

Examples

Remember to specify the short name and to place this file in the same folder as the script:

```
<input name="FormErrorPageTemplate" type="hidden"
value="error-page-template.htm">
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)
- C024: Can't open Error Page template file. (page 174)
- C025: Specified file is incorrectly formatted. (page 174)

---

[38] The "short name" or "simple name" of a file does not include the drive designation nor the path. For example: if the full name is "C:directory1\directory2\filename.txt" then the "short name" is simply "filename.txt".

FORMERRORPAGETITLE

Description

This parameter specifies the text which appears in the title bar of the Error Page window.

This parameter allows the Form Designer to customize the formatting of the page title simulate the look/feel of his website.

See *The Error Page* on page 163.

Values

Any text value (except asterisk) is permissible. No other edits are performed on this Hidden Field Parameter.

Default

If not specified, the default value is "Error Page – Huggins' Email Form Script".

Examples

For example, if you wish to have this field say "Error Page – Dan's Dynamite Website", you would specify this as:

```
<input name="FormErrorPageTitle" type="hidden"
value="Error Page – Dan's Dynamite Website">
```

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173)

FormErrorPageTrackingInfo

Description

This parameter specifies whether to display tracking information on the Error Page.

The tracking information includes:

- Date/Time the script was executed[39]
- IP information
- Referer information[40]
- PHP version
- Script version
- The short and long names of the script being executed
- Browser information

See *The Error Page* on page 163.

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to display the tracking information.

Set the value to "no" to not display the tracking information.

Default

The default value is "yes".

Examples

To turn off the tracking information, specified this as:

```
<input name="FormErrorPageTrackingInfo" type="hidden" value="no">
```

Edits

- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

---

[39] Showing the date/time requires PHP Version 4.1.0. If you run this script with a prior version of PHP this will show an erroneous date, probably Thu, 01 Jan 1970.

[40] For information on why "referer" is "misspelled", see Special Fields on page 145.

FORMFIELDNAMEEDITLIST

Description

This parameter specifies a list individual field edits for the script to apply to Form Fields.

Each edit for each field has a specific format.

The edits are placed in the list one after another, as many as necessary.

Because this parameter is one of the more complex parameters for Huggins' Email Form Script, I will illustrate the various options using examples.

Values

See the descriptions of each particular edit for values.

Default

There is no default.

Examples

See the descriptions of each particular edit for values.

Edits

- C001: Referenced fields must exist on the form; the special fields ip, browser and referer are not allowed (page 166)
- C006: Edit specification must be one of the legal values (page 98)
- C012: Check for terminator of sub-parameters (page 170)
- C013: Check for terminator of individual edit specifications (page 170)
- C014: Check Sub-Parameters within a Text Edit (page 171)
- C022: This field may not have a value of asterisk ("*"). (page 173)
- E001: Check for Pseudo-Captcha Error
- E011: Compare email address entered against the confirmation email address entered (page 180)
- E012: Check value entered against confirmation value entered (page 180)
- E013: Confirm a value was selected in a drop down list (page 180)
- E014: Confirm an "attraction field" was left empty (page 180)
- E015: Check a field against a list of permitted values (page 180)
- E016: Check a field against a list of prohibited values (page 180)
- E017: Check a field that should be an integer; look for illegal characters (page 181)

- E018: Check a field with an integer against the minimum value (page 181)
- E019: Check a field with an integer against the maximum value (page 181)
- E020: Check a field that should be a number; look for illegal character (page 181)
- E021: Check a field that should be a number; look for too many periods (page 181)
- E022: Check a field with a number against the minimum value (page 182)
- E023: Check a field with a number against the maximum value (page 182)
- E024: Check a text field for illegal characters (page 182)
- E029: Check number of selected checkboxes against a minimum (page 182)
- E027: Check number of selected checkboxes against a maximum (page 182)

Multiple Field Edits

The individual edits are specified one after the other in the list.

Each individual edit terminates with a tilde ("~").

For example, the code below will perform edits on **field1**, **field2**, **field3**, **email**, **accountnumber**, **banana** and e-**mail**.

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, letters, numbers, other: #-, ~,
       field2, integer, 0, 100, ~,
       field3, drop down, select your choice, ~,
       accountnumber, compare, accountnumberconfirmation, ~,
       banana, captcha, blue, ~,
       e-mail, attraction, ~">
```

In each of the subsections that follows, the example shows the format as though there were only one edit. Please remember that usually, there will be multiple edits, "stacked" within this one edit field.

Types of Field Edits

The nine (9) possible field edits are:

Field Edit 1 – Text: Edit Text for Specific Characters (E024)

If you have a general text field, you may wish to limit the specific characters permitted in that field.

The general format is:

```
Name-of-Field, text, options, ~,
```

Each edit set ends with a "tilde" ("~").

Options can be:

1. "letters"

To require that the field only have letters, use the sub-parameter "letters". If you do this and the Form User enters a number or an embedded space or a special character, the script will generate an error. For example:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, letters, ~, … >
```

With this parameter, "abcde" would be legal, but "12345", "abc5e", "abc e" and "abc$e" would all be illegal.

2. "numbers"

To require that the field only have numbers (digits or numerals)), use the sub-parameter "numbers". If you do this and the Form User enters a letter or an embedded space or a special character, the script will generate an error. For example:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, numbers, ~, … >
```

With this parameter, "12345" would be legal, but "abcde", "123a5", "123 5" and "123$5" would all be illegal.

3. "spaces"

To permit such a field to have embedded spaces, use the sub-parameter "spaces". If you do this and the Form User enters an embedded space, the script will permit it.

For example, to allow letters and spaces:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, letters, spaces, ~, … >
```

With this parameter, "abcde" and "abc e" would both be legal, but "12345", "abc5e" and "abc$e" would all be illegal.

As another example, to allow numbers and spaces:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, numbers, spaces, ~, … >
```

With this parameter, "12345" and "123 5" would both be legal, but "abcde", "123a5" and "123$5" would all be illegal.

4. **"other:"**
   To permit such a field to have specific characters, use the sub-parameter "other:", followed by a list of acceptable characters.

For example, to allow the characters #, $, %, & and *, numbers and spaces:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, other: #$%&*, ~, … >
```

More typical would be to allow letters *and also* the characters #, $, %, & and *:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, letters, other: #$%&*, ~, … >
```

Or, to allow numbers, letters, embedded spaces *and also* the characters # and -:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, letters, numbers, other: #-, ~, … >
```

Field Edit 2 – Integer: Edit the Input for an Integer (E017, E018, E019)

This edit ensures that the Form User input an integer value (a whole number).

The general format is:

```
Name-of-Field, integer, low-value, high-value, ~,
```

Each edit set ends with a "tilde" ("~").

The script will ensure that an integer is entered. Then, it will check the integer against the low-value and high-value to ensure it is within limits.

For example, to require an integer between 0 and 10:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, integer, 0, 10, ~, … >
```

To require an integer between 0 and 100:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, integer, 0, 100, ~, … >
```

To require an integer between -100 and 100:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, integer, -100, 100, ~, … >
```

To require an integer between 4,000 and 8,000:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, integer, 4000, 8000, ~, … >
```

Note that in each of the above examples, entering a number that is not an integer (e.g., 9.4 or 2.6) will generate an error

Field Edit 3 – Number: Edit the Input for a Number (E020, E021, E022, E023)

This edit ensures that the Form User input an numeric value (a number).

Unlike integer edits, a number edit permits fractional numbers (e.g., 9.4 and 2.6).

The general format is:

```
Name-of-Field, number, low-value, high-value, ~,
```

Each edit set ends with a "tilde" ("~").

The script will ensure that a number is entered. Then, it will check the number against the low-value and high-value to ensure it is within limits.

For example, to require a number between 0 and 10:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, number, 0, 10, ~, … >
```

To require a number between 0 and 100:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, number, 0, 100, ~, … >
```

To require a number between -100 and 100:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, number, -100, 100, ~, … >
```

To require a number between 4,000 and 8,000:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, number, 4000, 8000, ~, … >
```

Field Edit 4 – Equal: Require One of a List of Legal Values (E015)

This edit ensures that input is one of a set of legal value.

This edit is *not* frequently used with "words or phrases" because when this is required, a better approach is to design the form with either a pull-down list or a set of radio buttons.

When it is used, it is most frequently used with single character inputs.

The general format is:

```
Name-of-Field, equal, value1, value2, value3, …, ~,
```

Each edit set ends with a "tilde" ("~").

The word "equals" also works for "equal".

For example, to require the Form User to type "apple", "pear" or "banana":

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, equal, apple, pear, banana, ~, … >
```

As another example, to require the Form User to type "frog", "snake", spider" or "ant":

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, equal, frog, snake, spider, ant, ~, … >
```

A more typical example would require the Form User to type a single letter, "a", "h", "m" or "q", for example:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, equal, a, h, m, q, ~, … >
```

Field Edit 5 – Not Equal: Prohibit a List of Illegal Values (E016)

This edit ensures that input is not one of a set of prohibited values.

This edit is *not* frequently used with "words or phrases" because of the difficulty of specifying all prohibited values.

When it is used, it is most frequently used with single character inputs.

The general format is:

```
Name-of-Field, not equal, value1, value2, value3, …, ~,
```

Each edit set ends with a "tilde" ("~").

The word "notequal" also works for "not equal".

For example, to require the Form User to *not* type "apple", "pear" or "banana":

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, not equal, apple, pear, banana, ~, … >
```

As another example, to require the Form User to *not* type "frog", "snake", spider" or "ant":

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, not equal, frog, snake, spider, ant, ~, … >
```

A more typical example would require the Form User to *not* type the letter, "a", "h", "m" or "q", for example:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, not equal, a, h, m, q, ~, … >


NB: This example does not prohibit the letters h, h, m or q as part of a
word. Rather it prohibits these letters as the entire entry.
```

Field Edit 6 – Drop Down: Ensure a Drop Down Value is Selected (E013)

Frequently, when Form Designers create a drop down list of values, they set the first value as an "instruction" (e. g, "Select your choice" or "Why are you contacting us?"). The Form Designer wants to ensure that the Form User actually chooses a value. They want to prohibit the default/first value.

This function can be done with *Field Edit 5 – Not Equal: Prohibit a List of Illegal Values* above. However, Huggins' Email Form Script implements a special option as well. The difference between using *Field Edit 5 – Not Equal: Prohibit a List of Illegal Values* to prohibit the first value and using this edit is the error message generated.

This edit, *Field Edit 6 – Drop Down: Ensure a Drop Down Value is Selected (E013),* generates an error that clearly references a drop down list.

The general format is:

```
Name-of-Field, drop down, prohibited-value, ~,
```

Each edit set ends with a "tilde" ("~").

For example, to prohibit a first value that says "Select your choice":

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, drop down, select your choice, ~, … >
```

Field Edit 7 – Text: Compare Two Fields (E012)

This field edit compares two fields for equality.

The most typical use of this *type* of comparison is to compare the email field with an email confirmation field. The script provides this function automatically as part of the parameter *FormFieldEditField*.

This field edit may be used to perform the same type of function on other fields.

The general format to compare Field1 with Field2, is:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, compare, field2, ~, … >
```

Each individual edit terminates with a tilde ("~").

For example, if you want the script to compare the field **accountnumber** with the field **accountnumberconfirmation**, and also to compare the field **phonenumber** with the field **phonenumberconfirmation**, you would specify this as:

```
<input name="FormFieldNameEditList" type="hidden"
value="accountnumber, compare, accountnumberconfirmation, ~,
value="phonenumber, compare, phonenumberconfirmation, ~, … >
```

Field Edit 8 – Email: Validate/Compare One or Two Email Addresses (E011)

This edit is similar to the performed against *FormEmailFieldList*:

- It tests the format of email addresses, like *FormEmailFieldList* (page 75);
- It provides for comparison of two email address fields for confirmation; and
- The error message is specific to the concept of a mismatched email (E011: Email Address Does Not Match Confirmation, page 180)

In other words, this edit is functionally like the *FormEmailFieldList*, but the field it specifies is not used for anything else.

This functionality is useful when you wish to ask a Form User for a second email, like an business or alternate contact email.

The general format to validate **EmailField1** without comparison to a second field is:

```
<input name="FormFieldNameEditList" type="hidden"
value="EmailField1, email, ~, … >
```

The general format to validate **EmailField1** and also compare it with **EmailField2** is:

```
<input name="FormFieldNameEditList" type="hidden"
value="EmailField1, email, EmailField2, ~, … >
```

Each individual edit terminates with a tilde ("~").

For example, if you want the script to validate field **businessemail,** you would specify this as:

```
<input name="FormFieldNameEditList" type="hidden"
value="businessemail, edit, ~, … >
```

Also, if you want the script to validate **businessemail** and also to compare it with field **businessemailconfirmation**, you would specify this as:

```
<input name="FormFieldNameEditList" type="hidden"
value="businessemail, edit, businessemailconfirmation, ~, … >
```

Field Edit 9 – Pseudo-Captcha: Test for Pseudo-Captcha Error (E001)

This field edit allows for testing a field against a required value.

It is similar to the option for "equals", except that the error message is specific to the concept of a CAPTCHA test to prevent spambots.

CAPTCHA stands for "Completely Automated Public Turing test to tell Computers and Humans Apart". This is not true CAPTCHA. True CAPTCHA are those hard to read, distorted, letters you have to type into a form. But this is something similar.

What it will do is allow you to ask a simple question on the form (e.g., "What color is the sky?", or "Summer isn't cold. What is it?" or "What is the opposite of black?", receive the answer and check for the answer. In these three examples, you would expect the user to type "blue", "hot" and "white", respectively. It is not sophisticated. But it is effective at stopping simple spambots.

I recommend that you not name the field using the word "captcha" or "spam". Instead, choose a different name that will not easily alert a spambot to the test.

Recommendation: do not name this Form Field using the word "captcha" or "spam".

For example, suppose your form asked this question: "What color is the sky?". The field designated to receive the answer might be named "banana" to confuse the spambot. The label (for error messages) might be "Color of the sky". And, the expected value would be "blue". You would specify this as:

```
<input name="FormFieldNameEditList" type="hidden"
value="banana, captcha, blue, ~, … >
```

(Remember that each individual edit terminates with a tilde ("~").)

While it is unusual to do so, note that you can have as many pseudo-captcha field tests as you would like.

Field Edit 10 – Attraction: A Spambot Attraction Test (E014)

This field edit allows for testing a field to be sure *nothing* is entered into it.

The purpose of this test is to trap spambots which fill out fields they shouldn't.

What you can do is to place a field on a form with a message like this:

Please leave this field empty so I will know you are not a spambot.

Another option is to put the message on the form that says:

For office use only. Leave this field empty.

Then, this test will check to be sure they really do.

To further try to entice the spambot, I may give the field an attractive name, like **e-mail**.

For example, suppose your form had the field named **e-mail** that you wanted to use as a spambot trap. You would specify this as:

```
<input name="FormFieldNameEditList" type="hidden"
value="e-mail, attraction, ~, … >
```

(Remember that each individual edit terminates with a tilde ("~").)

One additional trick with this is to use CSS to locate this field "off the visible screen" (e.g., <div style="position:absolute; top: -9000px; left:-9000px;">). If you do this a person will not see this field (and the instructions for the field, but a spambot will.

Field Edit 11 – CheckboxGroup: Checkbox Group Test (E029, E027)

This field edit allows for testing a group of checkboxes to ensure (a) at least some number are checked, (b) at most some number are checked, or (c) both.

The purpose of this test is to ensure that a Form User checks "the right number" of checkboxes.

For example, you may offer a group of checkboxes (e.g., a person's areas of interest, or items they want you to send them) and you may wish to ensure they check at least one. Or at least two. Or you might want to ensure they don't check more than three. Whichever combination of these min/max requirements this edit handles it.

For example, suppose your form had checkbox fields named c1, c2, c3 … c20. These 20 checkboxes are "areas of interest". And suppose you want to require that they check at least 3 but not more than 6. You would specify this as:

```
<input name="FormFieldNameEditList" type="hidden"
value="Areas of Interest, checkboxgroup, 3, 6, c1, c2, c3, c4, c5, c6, c7,
c8,
c9, c10, c11, c12, c13, c14, c15, c16, c17, c18, c19, c20, ~, … >
```

(Remember that each individual edit terminates with a tilde ("~").)

The first sub-parameter ("Areas of Interest") is the label that will be used if an error needs to be reported. It is not a "real" field name. Rather it is a "pseudo field name" that is *only* used for error reporting.

The second sub-parameter ("checkboxgroup") triggers the edit.

The third and fourth sub-parameters are the minimum and maximum number of checkboxes that must be checked.

Following that, list all the checkbox field names that form part of this group.

The value of zero for either the min or max tells the script not to check that. In other words, a 0 min means no min and a 0 max means no max.

Combining Multiple Field Edits

The parameter *FormFieldNameEditList* supports an "unlimited" list of fields and edits.

For example, suppose you want to:

- Allow numbers, letters, embedded spaces *and also* the characters "#" and "-" for **field1**

- Require **field2** to be an integer between 0 and 100 for **field2**

- Force a drop down selection for **field3**

- Compare **email** with **emailconfirmation**

- Compare **accountnumber** with **accountnumberconfirmation**

- Have a pseudo-Captcha field of **banana** requiring a response of "blue", and

- Have a spambot trap test for a field named **e-mail**.

In this case you would code:

```
<input name="FormFieldNameEditList" type="hidden"
value="field1, text, letters, numbers, other: #-, ~,
       field2, integer, 0, 100, ~,
       field3, drop down, select your choice, ~,
       accountnumber, compare, accountnumberconfirmation, ~,
       banana, captcha, blue, ~,
       e-mail, attraction, ~">
```

The script steps through the list performing each edit on each field in sequence.

FORMFIELDNAMELABELPLUSLIST

Description

This parameter serves as a type of "master field list".

This parameter is not required and the form can work without this parameter.

If you omit this parameter the script will automatically scan the form information and identify all the fields.[41]

However, because of the power of this parameter, I ***strongly recommend*** that you carefully consider using this field.

This list identifies:

- The fields in the form
- Their preferred label for messages and error reporting
- Whether they are required
- Their minimum size (if entered)
- Their maximum size

It may also specify email formatting, including blank lines and lines with just a "label".

The format are:

Format 1: Specifying a field
field-name, field-label, field-tag, min-size, max-size

Format 2: Specifying a label-line
labelline, line-label

Format 3: Specifying a blank line[42]
blankline
or
newline

Values

field-name must be a field on the form.

---

[41] The script will ***not*** automatically identify use of MsgxLabelSubjectAndAddressesDropDown. Reporting of this field requires specification in FormFieldNameLabelPlusList.

[42] In Version 2.0.1 the script *erroneously* required that "blankline", "newline" and "labelline" be specified as all caps. This error was detected during the revisions and testing for Version 2.1.0.

field-label may be any text. The special character "`" can be used to add leading spaces to the field-label.

field-tag

"y" or "yes" or "r" or "required" to indicate required

"n" or "no" to indicate not required

"c" or "checkbox" to indicate the field is a checkbox[43]

"e" or "exclude" to indicate that the field should not be reported by default[44]

min-size and max-size must be integers.

line-label is the text to be output by itself, typically as a group separator in the email

Default

There is no default

Examples

Suppose the form had the following fields:

- **email**
  A required field, with a min size of 0 characters, a max size of 60 characters, and a label of "Email Address"

- **firstname**
  An optional field, with no minimum size, a max size of 30 characters, and a label of "First Name"

- **middlename**
  An optional field, with no minimum size, a max size of 30 characters, and a label of "Middle Name"

- **lastname**
  An optional field, with no minimum size, a max size of 30 characters, and a label of "Last Name"

- **message**
  A required field, with a min size of 10 characters, a max size of 400 characters, and a label of "Message"

You would specify this as:

---

[43] Some of the error checking for the script (see errors C001 and C002, page 166) check to see if the referenced field was actually sent to the script. Checkboxes are a special case and require special handling because they sent to the script *only if they are checked*.

[44] An "e" or "exclude" field will not be reported by default. It will only be reported if explicitly requested.

```
<input name="FormFieldNameLabelPlusList" type="hidden"
value="email, Email Address, r, 0, 60,
       firstname, First Name, n, 0, 30,
       middlename, Middle Name, n, 0, 30,
       lastname, Last Name, n, 0, 30,
       message, Message, r, 10, 400">
```

Suppose you wanted to put a blank line in the email after it listed the email address and before the message. You would specify this as:

```
<input name="FormFieldNameLabelPlusList" type="hidden"
value="email, Email Address, r, 0, 60,
       blankline,
       firstname, First Name, n, 0, 30,
       middlename, Middle Name, n, 0, 30,
       lastname, Last Name, n, 0, 30,
       blankline,
       message, Message, r, 10, 400">
```

Now suppose you wanted to put a "label line" in the email that says "Name" and you want to put this before it lists the **firstname**, **middlename** and **lastname**. You would specify this as:

```
<input name="FormFieldNameLabelPlusList" type="hidden"
value="email, Email Address, r, 0, 60,
       blankline,
       labelline, Name,
       blankline,
       firstname, First Name, n, 0, 30,
       middlename, Middle Name, n, 0, 30,
       lastname, Last Name, n, 0, 30,
       blankline,
       message, Message, r, 10, 400">
```

Finally, suppose that you want to "indent" the labels for **firstname**, **middlename** and **lastname**. Suppose you want to "indent" them 5 spaces each. You cannot just put 5 spaces in front of the field-label. Consider, for example, if you say:

```
       firstname,      First Name, n, 0, 30,
       middlename,      Middle Name, n, 0, 30,
       lastname,      Last Name, n, 0, 30,
```

The script won't indent them. This is because the script strips off leading spaces.

To do this you need to find and use the "`" character. On most keyboards this is to the left of the "1", on the same key as "~". This character allows you to add leading spaces to indent the labels. You would say:

```
<input name="FormFieldNameLabelPlusList" type="hidden"
value="email, Email Address, r, 0, 60,
      blankline,
      labelline, Name,
      blankline,
      firstname, `     First Name, n, 0, 30,
      middlename, `      Middle Name, n, 0, 30,
      lastname, `      Last Name, n, 0, 30,
      blankline,
      message, Message, r, 10, 400">
```

This will cause "First Name", "Middle Name" and "Last Name" to be indented 5 spaces (the number of spaces after the "`" and before the text.

Edits

- C002: Referenced fields must exist on the form; the special fields ip, browser and referer are allowed (page 166)
- C005: The field-tag sub-parameter must have a value of "y", "yes", "n", "no", "r", "required", "c" or "checkbox"). (page 167)
- C017: Confirm that min and max sizes contain legal integers (page 171)
- C022: This field may not have a value of asterisk ("*"). (page 173)
- E002: Check if required fields are missing (page 177)
- E003: Check to see if too much data was entered (page 177)
- E004: Check to see if enough data was entered (page 177)

Notes

1. Requiring a field and specifying a minimum size are not the same. The minimum size applies *only if a field is entered.* A field that is not required with a minimum size of 10 characters would require the minimum size *only if data is entered.* If nothing is entered, the check is not performed. It would be, in that case, permissible to not enter anything at all.

2. The field label is used for emails and error messages. Note that this "master" field label can be overridden, if desired, for any email being sent.

3. To indicate a field is required, the legal values are "y", "yes", "n", "no", "r" and "required". Case does not matter.

4. The primary use for maximum size checking is to stop spambots. When you set up a form, the form will naturally limit the number of characters that can be typed into a field. However, when spambots attempt to hijack a form/script, they will pass more characters in a field than the form was designed for. Double checking the max size will stop spambots which do this.

5. A value of zero for minimum or maximum size turns off checking for minimum or maximum size.

FORMNAMEFIELD (DEPRECATED)

Description

This parameter was deprecated in Version 2.0.[45]

It was replaced by *FormNameFieldList* (see 113). This replacement is to change the name of the field for naming consistency and also to add additional functionality to the parameter.

If used, this parameter has different rules and format from *FormNameFieldList*. Specifically, this parameter could only specify one field while the new *FormNameFieldList* can specify multiple fields.

Old (pre-version 2.0) forms do not need to change this parameter. The new script recognizes the deprecated field for backward compatibility.

However, new implementations should not use this field and should instead use *FormNameFieldList*.

Values

See FormNameFieldList on page 39.

Default

See FormNameFieldList on page 39.

Examples

See FormNameFieldList on page 39.

Edits

- C001: Referenced fields must exist on the form; the special fields ip, browser and referer are not allowed (page 166)
- C003: Both *FormNameField* and *FormNameFieldList* may not be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 86)
- C022: This field may not have a value of asterisk ("*"). (page 173)

---

[45] A deprecated item is one that has been outdated by newer constructs. Deprecated items are defined in this document but are clearly marked as deprecated. Deprecated items still work, but may become obsolete at any time in future versions of this software. The script may still support deprecated items for backward compatibility. However, users should begin to use the substitutes. For a list, see the section *Deprecated and Obsolete Items* on page 198.

FORMNAMEFIELDLIST

Description

This parameter specifies a list of field names which will be put together to construct a full name for the receiving name of the echo/confirmation email and also as the sending name for email Message 1 and Message 2.

This new parameter (V2.0) replaces *FormNameField*. See the description below for more info.

Previously, the script allowed only one field. Version 2.0 allows a list of fields (e.g., **firstname**, **lastname**) which will be put together to construct the full name. This full name will be used as the sending name for Message 1 and Message 2 and as the receiving name for the echo/confirmation email.

Values

The values must be a Field Name of a Form Field on the Web Form.

Default

There is no default.

Examples

For example, if the form had a field **firstname** and a field **lastname**, you would specify this as:

```
<input name="FormNameFieldList" type="hidden" value="firstname, lastname">
```

Edits

- C002: Referenced fields must exist on the form; the special fields ip, browser and referer are allowed (page 166)
- C003: Both *FormNameField* and *FormNameFieldList* may not be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 86)
- C022: This field may not have a value of asterisk ("*"). (page 173)

FORMNEXTURL

Description

This parameter specifies the *fully qualified* URL of the page to be displayed after the Web Form is processed.

This new parameter (V2.0) renames/replaces *NextURL*.

Values

This value should be a fully qualified, legal URL. It should probably be the URL of a "thank you page" on the site. *No editing is performed to ensure the URL is valid.*

Default

The default is http://www.JamesSHuggins.com/h/web1/huggins-email-form-script-thank-you.htm

Examples

If you wanted the Form Page to go to Google when it is finished processing, you would set the value as:

```
<input name="FormNextURL" type="hidden" value="http://www.Google.com">
```

Edits

- C003: Both *NextURL* and *FormNextURL* may not be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 86)
- C022: This field may not have a value of asterisk ("*"). (page 173)

FORMOUTPUTFIELDINFO

Description

This parameter specifies whether to output a page showing the field information from the form.

This field information is useful for creating the *FormFieldNameLabelPlusList* parameter.

In order to specify the *FormFieldNameLabelPlusList* parameter, you need to create HTML for the form showing all the fields. This parameter makes this easy.

By specifying this parameter with a value of "yes", the script will output a page with the HTML that can be easily copied and modified to specify *FormFieldNameLabelPlusList*.

This parameter is never used for production. It halts processing of the form. However, it can be specified during setup to assist parameter specification.

Note that the script omits fields beginning with:

- copyuser
- formecho
- formemail
- formerror
- formfield
- formname
- formnexturl
- formoutput
- formtest
- msg1
- msg2
- msgecho
- nexturl

Note that the script does *not* omit the field named "submit".

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to display the page.

Set the value to "no" to not display the page.

Default

The default value is "no".

Examples

To indicate that the page should be output, you would specify this as:

```
<input name="FormOutputFieldInfo" type="hidden" value="yes">
```

Edits

- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

Checkbox Fields

The script *only* reports the fields which are sent to the script. Note that checkbox fields are sent *only if they are checked.*

If your form has *any* checkbox fields, check *all of them* before clicking submit.

FORMPLEASEFORGIVEMYUSEOFAWINDOWSSERVER

Description

This parameter specifies whether to alter script performance to accommodate limitations of Windows Servers running PHP. For example, many Windows Servers do not allow an email address to have *both* a name and an email address. Also, checking email validity on non-Windows servers involves checking the MX records of a domain name, and this function tends to fail on Windows Servers. The parameter changes these actions to accommodate the peculiarities and functional limitations of Windows Servers.

Recommendations

For these and other similar reasons, I recommend that you do not use Windows servers *unless you need to.* It is perfectly fine to use them if you need to, but I have some users who, when asked why they use Windows servers, did not have a good reason. Instead, they had chosen to use Windows servers because they did not know/understand "Linux" and/or "Apache". In short, do not choose Windows servers unless you need to. But if you need to, this script will accommodate those limitations.

Effects

Turning this parameter on ("y" or "yes") causes the following effects.

1. Email addresses ignore the "name" part and only use the email address part.

2. Edits of email address validity do not check MX records on the destination domain name.

The result is that script functionality is reduced in order to accommodate the limitations of Windows Servers.

Values

Legal values are "y", "yes", "n" and "no".

Default

The default value is "no".

Examples

To indicate that the email format *should* be downgraded, you would specify this as:

```
<input name=" FormPleaseForgiveMyUseOfAWindowsServer" type="hidden"
value="yes">
```

Edits

- ▪ C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

FORMTESTEMAILDOMAIN

Description

This parameter specifies whether to test the domain name of the email address to be sure the domain name exists and is configured for email. The email address is the data entered in the field specified by *FormEmailFieldList*.

The test checks the *domain name* of the email address entered to be sure it can be found and that it has MX records. This test does *not* check to ensure the email address is a real working address.

> **NB: This test does not work on a Windows Server because Windows does not support the required PHP function.**

This Hidden Parameter relates to error E009.

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to perform the test.

Set the value to "no" to not perform the test.

Default

The default value is "yes".

Examples

To indicate that the test should *not* be performed, you would specify this as:

```
<input name="FormTestEmailDomain" type="hidden" value="no">
```

Edits

- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

FORMTESTEMAILFORMAT

Description

This parameter specifies whether to test the email address for "format". The email address is the data entered in the field specified by *FormEmailFieldList*.

The test checks the *format* of the email address entered to be sure it complies with the rules for construction of an email address. This test does *not* check to ensure the email address is a real working address.

This Hidden Parameter relates to errors E005, E006, E007 and E008.

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to perform the test.

Set the value to "no" to not perform the test.

Default

The default value is "yes".

Examples

To indicate that the test should *not* be performed, you would specify this as:

```
<input name="FormTestEmailFormat" type="hidden" value="no">
```

Edits

- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

FORMTESTFIELDMAXLENGTHS

Description

This parameter specifies whether to test the maximum lengths for fields specified in *FormFieldNameLabelPlusList*.

Testing for maximum lengths requires that *FormFieldNameLabelPlusList* be specified.

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to perform the test.

Set the value to "no" to not perform the test

Default

The default value is "yes".

Examples

To indicate that the test should *not* be performed, you would specify this as:

```
<input name="FormTestFieldMaxLengths" type="hidden" value="no">
```

Edits

- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

FORMTESTFIELDMINLENGTHS

Description

This parameter specifies whether to test the minimum lengths for fields specified in *FormFieldNameLabelPlusList*.

Testing for minimum lengths requires that *FormFieldNameLabelPlusList* be specified.

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to perform the test.

Set the value to "no" to not perform the test.

Default

The default value is "yes".

Examples

To indicate that the test should *not* be performed, you would specify this as:

```
<input name="FormTestFieldMinLengths" type="hidden" value="no">
```

Edits

- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

FORMTESTINJECTIONEXPLOITS

Description

This parameter specifies whether to test the input fields for "injection exploits".

Injection Exploits are attempts by spambots to place values like "cc:" and "bcc:" in the fields to hijack the form/script and use it for spam.

Testing for injection exploits helps prevent spambots from using your form/script for spam.

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to perform the test.

Set the value to "no" to not perform the test.

Default

The default value is "yes".

Examples

To indicate that the test should *not* be performed, you would specify this as:

```
<input name="FormTestInjectionExploits" type="hidden" value="no">
```

Edits

- C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)

NextURL (DEPRECATED)

Description

This field was deprecated in Version 2.0.[46]

It was replaced by *FormNextURL* (see 120). This replacement is only to change the name of the field for naming consistency.

If used, this field has exactly the same rules as *FormNextURL*.

Old (pre-version 2.0) forms do not need to change this field. The new script recognizes and uses this deprecated field.

However, new implementations should not use this field and should instead use *FormNextURL*.

Values

See *FormNextURL* on page 120.

Default

See *FormNextURL* on page 120.

Examples

See *FormNextURL* on page 120.

Edits

- C003: Both *NextURL* and *FormNextURL* may not be specified. They are redundant and mutually exclusive Hidden Field Parameters. (page 86)
- C022: This field may not have a value of asterisk ("*"). (page 173)

---

[46] A deprecated item is one that has been outdated by newer constructs. Deprecated items are defined in this document but are clearly marked as deprecated. Deprecated items still work, but may become obsolete at any time in future versions of this software. The script may still support deprecated items for backward compatibility. However, users should begin to use the substitutes. For a list, see the section *Deprecated and Obsolete Items* on page 198.

MESSAGE LEVEL PARAMETERS – DETAIL DESCRIPTIONS

These Hidden Field Parameters apply to the email messages sent by the script.

BACK REFERENCE: DUPLICATING A "PRIOR VALUE"

Sometimes it is desired that a "prior value" be used. For example, The most common example is when you want the Echo/Confirmation Email to be the same format as either Message 1 or Message 2.[47]

When possible, the system uses a value of asterisk ("*") to indicate the use of a "prior value". The script will look "backwards" to find the most appropriate value to use.

For example, for *MsgxFieldNameLabelList*:

- For the Echo/Confirmation Message, the script looks first for the Message 2 parameter (*Msg2FieldNameLabelList*), or, if that is not specified, then the Message 1 parameter (*Msg1FieldNameLabelList*), or if that is not specified, then the Form parameter (*FormFieldNameLabelPlusList*), or if that is not specified, all the fields that the script finds in the form.

- For Message 2, the script looks for the Message 1 parameter (*Msg1FieldNameLabelList*), or if that is not specified, then the Form parameter (*FormFieldNameLabelPlusList*), or if that is not specified, all the fields that the script finds in the form.

- For Message 1, the script looks for the Form parameter (*FormFieldNameLabelPlusList*), or if that is not specified, all the fields that the script finds in the form.

- If the parameter is omitted, the script reports all fields using Format 1.

- If you want to override this, you must specify this parameter.

The availability of Back Reference capability is defined for each variable.

---

[47] It is unlikely you would ever want Message 2 to have the same format as Message 1. The reason is that if Message 2 were to have the same format as Message 1, you would just add an additional address to Message 1 and would not even use Message 2.

MSGECHOFROMADDR

Specific Fields

This Hidden Field Parameter applies only to *MsgEchoFromAddr*. The fields *Msg1FromAddr* and *Msg2FromAddr* do not exist.

Description

This parameter specifies the sending address for the Echo/Confirmation Message. The Form User will see this address as the sender of the email message.

If you do not specify this parameter, the script will automatically use the first address in *Msg1AddrList* as the default.

However, if you want to use a *different* email address as the "from address", use this parameter.

Values

The email address is split into three parts separated by commas.

The first part is the Name Part. The second part is the Local Address Part. The third part is the Domain Name Part.

```
<input name="Msg1AddrList" type="hidden"
value="name-part, local-address-part, domain-name-part">
```

The Name Part is the name of the addressee. It does not directly affect delivery of the email.

The Local Name Part is the part of the email address immediately before the "@".

The Domain Name Part is the part of the email address immediately after the "@".

Note that commas are not permitted in the Name Part of an email address. Unlike some other Form Fields, the use of an asterisk ("*") will not insert a comma, but will, instead, be removed.

Default

It is not necessary to supply this value, if you want the Echo/Confirmation "from address" to be the same as the first Message 1 address .

Examples

> For example, if you wanted to use "Confirming Your Message
> <DForeman@JSend.com>" and you would do that this way:

```
<input name="MsgEchoFromAddr" type="hidden"
value="Confirming Your Message, DForeman, JSend.com">
```

Back Reference: Duplicating a "Prior Value"

> There is no Back Reference capability for this variable.

Edits

- C007: Check email address for number of characters and "@" signs (page 168)
- C008: Check email address for format of local address part (page 168)
- C009: Check email address for number of parts in domain name (page 169)
- C010: Check email address for format of domain name (page 169)
- C011: Check email address for whether domain name can receive email (page 170)
- C015: Check for a specific number of sub-parameters (page 171)
- C022: This field may not have a value of asterisk ("*"). (page 173)

MSGXADDRLIST

Specific Fields

This Hidden Field Parameter applies to both *Msg1AddrList* and *Msg2AddrLlist*.

Description

This parameter specifies the email addresses of the people to receive the Email Message (Message 1 or Message 2).

(This parameter does not apply to the Echo/Confirmation Message because that email message is sent to the name/email address specified on the Web Form.)

This parameter can also contain a list of multiple addresses.

To expand the list to multiple addresses, just add additional addresses to the string.

> **This is the only Hidden Parameter that is required for the script to function.**

Values

The email addresses are split into three parts separated by commas.

The first part is the Name Part. The second part is the Local Address Part. The third part is the Domain Name Part.

```
<input name="Msg1AddrList" type="hidden"
value="name-part, local-address-part, domain-name-part">
```

The Name Part is the name of the addressee. It does not directly affect delivery of the email.

The Local Name Part is the part of the email address immediately before the "@".

The Domain Name Part is the part of the email address immediately after the "@".

Note that commas are not permitted in the Name Part of an email address. Unlike some other Form Fields, the use of an asterisk ("*") will not insert a comma, but will, instead, be removed.

Default

There is no default.

Examples

If the full email address is

"Dan Foreman <DForeman@JSend.com>",

the address would be entered as

```
<input name="Msg1AddrList" type="hidden" value="Dan Foreman, DForeman,
JSend.com">
```

Suppose you had these three addresses:

- Dan Foreman <DForeman@JSend.com>
- Linda Nulty <LNulty@BigBlueBananaBoat.com>
- Monty Python <MPython@ILikeMontyPython.com

You would code for these three addresses like this:

```
<input name="Msg1AddrList" type="hidden" value="Dan Foreman, DForeman,
JSend.com, Linda Nulty, LNulty, BigBlueBananaBoat.com, Monty Python, MPython,
ILikeMontyPython.com">
```

Restrictions

If *MsgxLabelSubjectAndAddressesDropDown* is specified, the corresponding *MsgxAddrList* and *MsgxSubject* may *not* be specified.

Back Reference: Duplicating a "Prior Value"

There is no Back Reference capability for this variable. It is not possible to make Msg2 be the same as Msg1. An asterisk ("*") is an illegal value. An empty value means it will not be used.

Edits

- C007: Check email address for number of characters and "@" signs (page 168)
- C008: Check email address for format of local address part (page 168)
- C009: Check email address for number of parts in domain name (page 169)
- C010: Check email address for format of domain name (page 169)
- C011: Check email address for whether domain name can receive email (page 170)
- C016: Confirm number of sub-parameters is an even multiple (page 171)

- C021: Check to ensure that neither *MsgxAddrList* nor *MsgxSubject* is specified if *MsgxLabelSubjectAndAddressesDropDown* is specified (page 173)

- C022: This field may not have a value of asterisk ("*"). (page 173)

MSGXDEFAULTFROMADDR

Specific Fields

Msg1: Yes	Msg2: Yes	MsgEcho: No

This Hidden Field Parameter applies to both *Msg1DefaultFromAddr* and *Msg2DefaultFromAddr*. There is no field named *MsgEchoDefaultFromAddr*.

Description

This parameter specifies what email address should be shown as the "from address" for Message 1 or Message 2 if no email address is available from the Form User.

Set the value of Msg1DefaultFromAddr to be one email address—the email address to be shown as the "from address" for Message 1.

MsgxDefaultFromAddr is used in a variety of circumstances:

- **The Web Form does not have an email address field.**
  Normally the form's email address field would be used as the "sender" address for messages sent by the script. But if the Web Form does not have an email address field, some other sender address must be used. This field provides that address.

- **The Web Form has a field, but the Form User left it blank.**
  Since the Form User left it blank, it cannot be used as the "sender" address. Instead, the script uses the specified Msg1DefaultFromAddr.

- **The Form Designer doesn't want to use the Form User email address.**
  The Web Form specifies an email address field for use by Message 1 (or Message 2) or to receive an Echo/Confirmation Email, but, for some reason, Message 1 should not use that field and the Web Form sets Msg1ForceDefaultFromAddr to "yes". In this case the script uses the specified Msg1DefaultFromAddr even if the Form User specifies an email address.

Values

The email address is split into three parts separated by commas.

The first part is the Name Part. The second part is the Local Address Part. The third part is the Domain Name Part.

```
<input name="Msg1AddrList" type="hidden"
value="name-part, local-address-part, domain-name-part">
```

The Name Part is the name of the sender. It does not directly affect delivery of the email.

The Local Name Part is the part of the email address immediately before the "@".

The Domain Name Part is the part of the email address immediately after the "@".

Note that commas are not permitted in the Name Part of an email address. Unlike some other Form Fields, the use of an asterisk ("*") will not insert a comma, but will, instead, be removed.

Default

If you do not specify this parameter, and *if it is needed*, the script will automatically use the first address in *Msg1AddrList* as the default.

Examples

If the full email address is

"Dan Foreman <DForeman@JSend.com>",

the address would be entered as

```
<input name="Msg1DefaultFromAddr" type="hidden"
value="Dan Foreman, DForeman, JSend.com">
```

Back Reference: Duplicating a "Prior Value"

For *Msg2DefaultFromAddr*, a value of asterisk ("*") results in the value of *Msg1DefaultFromAddr* being used. Back Reference is not available for Message 1. If you do not specify this parameter, and *if it is needed,* the script will automatically use the first address in *Msg1AddrList*.

Edits

- ▪ C007: Check email address for number of characters and "@" signs (page 168)
- ▪ C008: Check email address for format of local address part (page 168)
- ▪ C009: Check email address for number of parts in domain name (page 169)
- ▪ C010: Check email address for format of domain name (page 169)
- ▪ C011: Check email address for whether domain name can receive email (page 170)
- ▪ C015: Check for a specific number of sub-parameters (page 171)
- ▪ C022: This field may not have a value of asterisk ("*"). (page 173) NB: for Msg1 only

MSGXFIELDLABELVALUESEPARATOR

Specific Fields

This Hidden Field Parameter applies to all three:
Msg1FieldLabelValueSeparator, Msg2FieldLabelValueSeparator, and
MsgEchoFieldLabelValueSeparator.

Description

This parameter specifies the text to use in an email to separate the Field Label and the Field Value. Special values are also used to indicate different email formats.

Special Values

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1).

Other special values are:

- "crlf" means to put the label on one line, the value on the next, and have a blank line after the value
- "crlf+1" means the same as "crlf", but the value is indented on the 2nd line by one space
- "crlf+2" through "crlf+10" mean the same as "crlf+1", but the number of indented spaces changes from 2 through 10

Default

The default is ": ". That is, the email will list the Field Label, a colon, a space, and the Field Value.

Examples

For example, to separate the Field Label from the Field Value in the email with three hyphens, you would specify this as:

```
<input name="Msg1FieldLabelValueSeparator" type="hidden" value=" --- ">
```

To separate the Field Label from the Field Value in the email with three ">", you would specify this as:

```
<input name="Msg1FieldLabelValueSeparator" type="hidden" value=" >>> ">
```

To put the Field Label on one line, and the Field Value on the next line, indented by 8 spaces, you would specify this as:

```
<input name="Msg1FieldLabelValueSeparator" type="hidden" value="crlf+8">
```

Back Reference: Duplicating a "Prior Value"

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value uses the default.

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGXFIELDNAMEEXCLUDELIST

Specific Fields

This Hidden Field Parameter applies to all three: *Msg1FieldNameExcludeList, Msg2FieldNameExcludelist* and *MsgEchoFieldNameExcludeList*.

Description

This parameter specifies a list of fields to exclude from a message. This is useful when one message should have the same fields as a prior message, except for some fields.

The parameter specifies the fields to exclude by name.

Values

The Field Names specified must be fields which are on the form.

Default

The default is to not exclude any fields.

Examples

Three of the most common fields to exclude (frequently from the echo/confirmation message) are **ip**, **browser** and **referer**. This is specified as:

```
<input name="MsgEchoFieldNameExcludeList" type="hidden"
value="ip, browser, referer">
```

Back Reference: Duplicating a "Prior Value"

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value does not exclude any fields.

Edits

- C002: Referenced fields must exist on the form; the special fields ip, browser and referer are allowed (page 166)
- C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGXFIELDNAMELABELLIST

Specific Fields

This Hidden Field Parameter applies to all three: *Msg1FieldNameLabelList, Msg2FieldNameLabelList* and *MsgEchoFieldNameLabelList.*

Description

This parameter specifies the Field Names of the Form Fields to be reported in Message 1, Message 2 or the Echo/Confirmation Message.

If you wish to report a subset of the fields, you may list them here, or you may specify which ones to exclude by coding the *MsgxFieldNameExcludeList* (page 49).

Values

There are three primary formats available for the email messages:

- **Format 1**
- Places the Field Label and the Field Value on a line or pair of lines.
- Places each label/value on a new line or on a pair of lines (depending on the value of *MsgxFieldLabelValueSeparator*).
- Can specify the value of the label for each field.
- **Format 2**
- The output of Format 1 and Format 2 is the same. The difference is that with Format 2 you cannot specify the value of the label for each field.
- Places the Field Label and the Field Value on a line or pair of lines.
- Places each label/value on a new line or on a pair of lines (depending on the value of *MsgxFieldLabelValueSeparator*).
- **Format 3**
- Places groups of Field Labels and the Field Value on a line.

Default

The default is to report all fields using Format 1.

Example of Format 1 & 2 (basic)

```
Name: John Jones
Email: John@Wherever.com
Address: 12345 Some Street
City: Houston
State: Texas
Zip: 77058
```

Example of Format 1 & 2 (alternate[48])

```
Name
John Jones

Email
John@Wherever.com

Address
12345 Some St.

City
Houston

State
Texas

Zip
77058
```

Example of Format 3

```
Name: John Jones     Email: John@Wherever.com
Address: 12345 Some St.    City: Houston     State: Texas     Zip: 77058
```

Specifying Format 1

For Format 1, you specify "Format 1"[49] followed by pairs of parameters, separated by commas. The first value is the Field Name, the second is the Field Label to be used.

For example, to have the email show a Form Field **name** with a Form Label of "User Name" and then a Form Field **email** with a Form Label of "Email Address", enter

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="Format 1, name, User Name, email, Email Address">
```

To specify a Field Reporting Label with an embedded comma, use a tilde ("~") instead of a comma.

---

[48] The alternate format is created by the special values of *MsgxFieldLabelValueSeparator*. See page 139.

[49] A value of "Format1" (with no space) is also legal. Capitalization does not matter. In addition, for compatibility with Version 1, if the first item in the list is not "Format 1" or "Format 2" or "Format 3", the list is presumed to be in Format 1. This is because Version 1 did not have different formats available.

For example, to have the email show a Form Field **name** with a Form Label of "Name, User" and then a Form Field **email** with a Form Label of "Address, Email", enter

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="Format 1, name, Name~ User, email, Address~ Email">
```

If you want to use the Field Label specified in *FormFieldNameLabelPlusList*, code an asterisk ("*") as the Field Label.

For example, to have the email show the Form Fields **name** and **email**, using the Field Labels specified in *FormFieldNameLabelPlusList*, you would specify this as:

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="Format 1, name, *, email, *">
```

Specifying Format 2

For Format 2, you specify "Format 2"[50] followed by a list of Field Names. Unlike Format 1, you cannot specify the Field Label to be used. Format 2 uses the master Field Label as specified in *FormFieldNameLabelPlusList*.

For example, to have the email show a Form Field **name** and then a Form Field **email** enter

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="Format 2, name, User Name, email, Email Address">
```

Specifying Format 3

For Format 3, you specify "Format 3"[51] followed by a list of Field Names.

This list of Field Names is separated by "newline" and/or "cond-newline".

(Like Format 2 you cannot specify the Field Label to be used. Format 3 uses the master Field Label as specified in *FormFieldNameLabelPlusList*.)

The script places each Field Label/Value pair on the same line of the email, until it encounters either "newline" or "cond-newline".

The parameter "newline" forces a new line in the email.

---

[50] A value of "Format2" (with no space) is also legal. Capitalization does not matter.

[51] A value of "Format3" (with no space) is also legal. Capitalization does not matter.

For example, to have the email show a the Form Field **name** and then a Form Field **email** on the same line, followed by the **firstname** and then **lastname** on the next line, enter

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="Format 3, name, email, newline, firstname, lastname">
```

The parameter "cond-newline" forces a new line *only if* some values have been reported. Cond-newline is useful if you are specifying that fields should be omitted in certain conditions. Using "cond-newline" eliminates the blank lines that result when values are omitted.[52]

Special Fields

In addition to the Form Fields you have created for the Web Form, you may use any of three special Form Fields in the list. These special Form Fields reference data supplied by the Web Form process:

- **ip**
  The IP Address of the Form User's computer when the Web Form is processed.
- **browser**
  The browser specification string sent from the Form User's Internet browser to identify the browser
- **referer[53] or referrer**
  The URL of the Form Page, sent by the Form User's Internet browser when the Web Form is processed.

Note that all three of these Field Names are lower case.

These Form Fields may be used just as any other Form Field.

For example:

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 1, ip, IP Address, browser, Browser, referer, Referring Page">
```

---

[52] Omitting values is specified by MsgxFieldNameValueSubstitutionList on page 149.

[53] Note that the spelling of this field is either "referer" or "referrer". The reason for this is that "referer" is a misspelling that is now standard. The Free On-line Dictionary of Computing (www.foldoc.org) says that "referer" is 'A misspelling of "referrer" which somehow made it into the HTTP standard. A given web page's referer (sic) is the URL of whatever web page contains the link that the user followed to the current page. Most browsers pass this information as part of a request.' For this reason, The Huggins' Email Form Script script accepts either the standard (but incorrect) spelling or the non-standard (but correct) spelling.

Here is an example showing a name field, an email address field and all three of the special Form Fields

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 1, name, User Name, email, Email Address, ip,
IP Address, browser, Browser, referer, Referring Page">
```

Blank Lines in Format 1

You may wish to separate Form Fields with "blank lines" in the email.

To add a blank line, just insert a new "Dummy Field Name" with a Field Reporting Label of "blank" or "newline".[54]

The Dummy Field Name can be any text that is not a Field Name. Note that each Dummy Field Name can be used only once.

The following example uses a Dummy Field Name of "b1" to specify a blank line between the email address and the IP address:

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 1, name, User Name, email, Email Address,
b1, blank, ip, IP Address, browser, Browser, referer, Referring Page">
```

The following example uses a Dummy Field Name of "b1" to specify a blank line between the email address and the IP address, and a Dummy Field Name of "b2" to specify a blank line between the browser value and the referring page:

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 1, name, User Name, email, Email Address,
b1, blank, ip, IP Address, browser, Browser,
b2, blank, referer, Referring Page">
```

Blank Lines in Format 2

To add a blank line, just insert a new "Dummy Field Name" of either "blank" or "newline".[55]

The following example uses a Dummy Field Name of "blank" to specify a blank line between the email address and the IP address:

---

[54] The value of "blank" provides compatibility with Version 1. The value of "newline" is consistent with the specification in Format 2 and Format 3.

[55] The value of "blank" provides compatibility with Version 1. The value of "newline" is consistent with the specification in Format 2 and Format 3.

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 2, name, email, blank, ip, browser, referer">
```

The following example uses a Dummy Field Name of "blank" to specify a blank line between the email address and the IP address, and a Dummy Field Name of "blank" to specify a blank line between the browser value and the referring page:

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 2, name, email, blank, ip, browser, blank, referer">
```

Blank Lines in Format 3

New lines are specified in Format 3 by the keyword "newline". To add a blank line, just insert two "newline".[56]

The following example puts name and email on the first line and also specifies a blank line between that line and the next line:

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 3, name, email, newline, newline, ip, browser, referer">
```

The following example uses a Dummy Field Name of "blank" to specify a blank line between the email address and the IP address, and a Dummy Field Name of "blank" to specify a blank line between the browser value and the referring page:

```
<input name="Msg1FieldNameLabelList" type="hidden"
value="format 3, name, email, blank, ip, browser, blank, referer">
```

Back Reference: Duplicating a "Prior Value"

Sometimes it is desired that the "prior value" be used. The most common example is when you want the Echo/Confirmation Email to be the same format as either Message 1 or Message 2.[57]

The value of asterisk ("*")indicates to use the "prior value". The script will look "backwards" to find the most appropriate value to use:

---

[56] "blank" provides compatibility with Version 1. "newline" is consistent with the specification in Format 2 and Format 3.

[57] It is unlikely you would ever want Message 2 to have the same format as Message 1. The reason is that if Message 2 were to have the same format as Message 1, you would just add an additional address to Message 1 and would not even use Message 2.

- For the Echo/Confirmation Message, the script looks first for the Message 2 parameter (*Msg2FieldNameLabelList*), or, if that is not specified, then the Message 1 parameter (*Msg1FieldNameLabelList*).

- For Message 2, the script looks for the Message 1 parameter (*Msg1FieldNameLabelList*).

- For Message 1, Back Reference is not permitted.

- If the parameter is omitted, the script reports using the specification in *FormFieldNameLabelPlusList*, or if that parameter is not specified, reports all fields using Format 1.

Edits

- C002: Referenced fields must exist on the form; the special fields ip, browser and referer are allowed (page 166)

- C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGXFIELDNAMEVALUESUBSTITUTIONLIST

Specific Fields

This Hidden Field Parameter applies to all three parameters
Msg1FieldNameValueSubstitutionList,
Msg2FieldNameValueSubstitutionList
and *MsgEchoFieldNameValueSubstitutionList.*

See also *MsgxFieldNameValueSubstitutionList2* on page 151 for an alternate
format to specify this information.

Description

This parameter specifies values to substitute for values entered into the Web
Form for Message 1, Message 2 or the Echo/Confirmation Message.

Values

To specify the substitution, enter (a) field-name, (b) field-value and
(c) substitution-value, separated by commas, repeating as many times as
necessary.

field-name must be a field on the form.

- For Radio Buttons, use the Group Name as the Field Name and the
  Value assigned to that particular button as the Field Value, being
  sure to match case.

field-value may be any anticipated value for the field

- Use "ON" to represent a checkbox being checked.[58]
- Use "OFF" to represent a checkbox *not* being checked.
- Use "OFF" to represent any string field being "empty" or "null".
- Use a value of "OMIT" to cause a field to not be reported.
- Use a value of "OFF" and a value of "OMIT" to cause an empty or null
  field to not be reported.
- To specify either a Field Value or a Substitution Value with a comma,
  use an asterisk instead of a comma.

substitution-value may be any desired value to substitute

Default

There is no default.

---

[58] Note that *case* is important for the values of "ON" and "OFF". The value of "OMIT" is not case
sensitive.

Checkbox Example

Suppose you have a checkbox named **C1**. If that box is checked, you want to report a value of "Yes, contact me.". If that box is *not* checked, you want to report a value of "No, do not contact me.". You would specify values for this checkbox this way:

```
<input name="Msg1FieldNameValueSubstitutionList" type="hidden"
value="C1, ON, Yes~ contact me,
C1, OFF, No~ do not contact me.">

Note the use of the tilde, to indicate an embedded comma.
```

Radio Button Example

Suppose you have a series of Radio Buttons with a Group Name of **R1**, and values of **V1**, **V2** and **V3**. You want **V1** to mean "Black". You want **V2** to mean "Blue". And you want **V3** to mean "Polka Dot". You would specify values for these buttons this way:

```
<input name="Msg1FieldNameValueSubstitutionList" type="hidden"
value="R1, V1, Black,
R1, V2, Blue,
R1, V3, Polka Dot">
```

Omit Field Example

Suppose that you want to report the name, *unless* it is empty or null. You could use this:

```
<input name="Msg1FieldNameValueSubstitutionList" type="hidden"
value="individualname, OFF, OMIT">
```

Back Reference: Duplicating a "Prior Value"

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value does not exclude any fields.

Edits

- C002: Referenced fields must exist on the form; the special fields ip, browser and referer are allowed (page 166)
- C016: Confirm number of sub-parameters is an even multiple (page 171)
- C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGXFIELDNAMEVALUESUBSTITUTIONLIST2

| Msg1: Yes | Msg2: Yes | MsgEcho: Yes |

Specific Fields

This Hidden Field Parameter applies to all three: *Msg1FieldNameValueSubstitutionList2, Msg2FieldNameValueSubstitutionList2* and *MsgEchoFieldNameValueSubstitutionList2*.

See also *MsgxFieldNameValueSubstitutionList* on page 149 for an alternate format to specify this information.

Description

This parameter specifies values to substitute for values entered into the Web Form for Message 1, Message 2 or the Echo/Confirmation Message. This parameter has the same function as MsgxFieldNameValueSubstitutionList, but specifies the sub-parameters in a different format that may be easier in some cases.

Values

To specify the substitution, enter (a) field-name, followed then by pairs of (b) field-value and (c) substitution-value, separated by commas, followed by a tilde ("~"),repeating as many times as necessary.

field-name must be a field on the form.

- For Radio Buttons, use the Group Name as the Field Name and the Value assigned to that particular button as the Field Value, being sure to match case.

field-value may be any anticipated value for the field

- Use "ON" to represent a checkbox being checked.[59]
- Use "OFF" to represent a checkbox *not* being checked.
- Use "OFF" to represent any string field being "empty" or "null".
- Use a value of "OMIT" to cause a field to not be reported.
- Use a value of "OFF" and a value of "OMIT" to cause an empty or null field to not be reported.
- To specify either a Field Value or a Substitution Value with a comma, use an asterisk instead of a comma.

substitution-value may be any desired value to substitute

---

[59] Note that *case* is important for the values of "ON" and "OFF". The value of "OMIT" is not case sensitive.

Default

> There is no default.

Checkbox Example

> Suppose you have a checkbox named **C1**. If that box is checked, you want to report a value of "Yes, contact me.". If that box is *not* checked, you want to report a value of "No, do not contact me.". You would specify values for this checkbox this way:

```
<input name="Msg1FieldNameValueSubstitutionList" type="hidden"
value="C1, ON, Yes~ contact me, OFF, No~ do not contact me., ~">

Note the use of the tilde, to indicate an embedded comma.
Note also the use of a single tilde to end the sub-parameter string.
```

Radio Button Example

> Suppose you have a series of Radio Buttons with a Group Name of **R1**, and values of **V1**, **V2** and **V3**. You want **V1** to mean "Black". You want **V2** to mean "Blue". And you want **V3** to mean "Polka Dot". You would specify values for these buttons this way:

```
<input name="Msg1FieldNameValueSubstitutionList" type="hidden"
value="R1, V1, Black, V2, Blue, V3, Polka Dot, ~">
```

Omit Field Example

> Suppose that you want to report the name, *unless* it is empty or null. You could use this:

```
<input name="Msg1FieldNameValueSubstitutionList" type="hidden"
value="individualname, OFF, OMIT, ~">
```

Back Reference: Duplicating a "Prior Value"

> A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value does not exclude any fields.

Edits

> - C002: Referenced fields must exist on the form; the special fields ip, browser and referer are allowed (page 166)
> - C016: Confirm number of sub-parameters is an even multiple (page 171)
> - C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGxFORCEDEFAULTFROMADDR

Specific Fields

Msg1: Yes   Msg2: Yes   MsgEcho: No

This Hidden Field Parameter applies to both *Msg1ForceDefaultFromAddr* and *Msg2ForceDefaultFromAddr*. There is no field named *MsgEchoForceDefaultFromAddr*.

Description

This parameter specifies whether to force the use of Msg1DefaultFromAddr, even if the Form User specifies an email address on the form.

Values

Legal values are "y", "yes", "n" and "no".

Set the value to "yes" to force use of the Default From Address.

Set the value to "no" to use the Default From Address *only if necessary* – if the Form User did not specify an email address or if the form itself does not provide an email address.

Default

The default value is "no".

Examples

If you want to use *Msg1DefaultFromAddr* to be the sender of Message 1 *even if* the Web Form has an email field and *even if* the Form User specifies an email address, set this value to "yes".

```
<input name="Msg1ForceDefaultFromAddr" type="hidden" value="yes">
```

Back Reference: Duplicating a "Prior Value"

There is no Back Reference capability for this variable. It is not possible to make Msg2 be the same as Msg1. An asterisk ("*") is an illegal value. An empty value is equivalent to "no".

Edits

- ▪ C004: This Yes/No parameter must have a value of "y", "yes", "n" or "no". (page 167)
- ▪ C022: This field may not have a value of asterisk ("*"). (page 173)

MsgxLabelSubjectAndAddressesDropDown

Specific Fields

This Hidden Field Parameter applies to both *Msg1LabelSubjectAndAddressesDropDown* and *Msg2LabelSubjectAndAddressesDropDown*.

NB: This is not truly a hidden field parameter because it is not hidden but is visible in the form.

Description

This parameter specifies the email addresses of the people to receive the Email Message as well as the corresponding subject (for Message 1 or Message 2) and also the "value" to be used in emails.

(This parameter does not apply to the Echo/Confirmation Message because that email message is sent to the name/email address specified on the Web Form.)

This parameter is intended to be specified as a drop down selection list.

Important Note

This parameter is different to all the "regular" parameters because it actually defines a real field on the form. That is, this particular parameter is ***not*** "hidden". In addition, the first sub-parameter of this particular parameter can be reported in each email.

Values

There are 5 values:

```
Substitution Value, Subject Value, Addressee Name Part, Addressee Local
Address Part, Addressee Domain Name Part …
```

The Substitution Value is the value that will be sent in the email for this field, if the field is referenced in the email. That is, when the email reports the value of *MsgXLabelSubjectAndAddressesDropDown*, it will report this first sub-value.

The Subject Value is the subject that will be used in the email to the specified addressees. Note that any email back to the submitter will use

The Addressee Information (Name Part, Local Address Part and Domain Name Part) is used to construct the addresses to send the emails. These three sub-values may repeat as often as needed for as many addressees as you wish.

The first part is the Name Part. The second part is the Local Address Part. The third part is the Domain Name Part.

The Name Part is the name of the addressee. It does not directly affect delivery of the email.

The Local Name Part is the part of the email address immediately before the "@".

The Domain Name Part is the part of the email address immediately after the "@".

Note that commas are not permitted in the Name Part of an email address. Unlike some other Form Fields, the use of an asterisk ("*") will not insert a comma, but will, instead, be removed.

Default

There is no default.

Examples

This code shows an example drop down with three options.

Option 1 shows as "Select Addressee" and does not have a specified value. (Do not specify a value for such an item.) If this is chosen, the system responds with error E025. The trigger for this error is selecting an option that has only one sub-parameter. If you do not want to edit for this, you can omit this option. **However, for maximum spam protection, I strongly recommend including such an option.**

Option 2 shows as "Customer Service" and has a value of "Customer Service, Customer Service Inquiry from the Web, Frank Jones, Frank, JSend.com, Linda Jones, Linda, JSend.com". This gives a subject of "Customer Service Inquiry from the Web" and two addressees (Frank Jones <Frank@JSend.com> and Linda Jones <Linda@JSend.com>). The value of "Customer Service" can be used as desired by reporting *MsgxLabelSubjectAndAddressesDropDown*.

Option 3 shows as "Technical Support" and has a value of "Technical Support, Technical Support Inquiry from the Web, Bob Smith, Bob, JSend.com, Sue Smith, Sue, JSend.com". This gives a subject of "Technical Support Inquiry from the Web" and two addresses (Bob Smith <Bob@JSend.com> and Sue Smith <Sue@JSend.com>). The value of "Technical Support" can be used as desired by reporting *MsgxLabelSubjectAndAddressesDropDown*.

Note that while their *appears* to be some relationship between the wording of parameter 1 and 2, this is not in any way necessary.

```
<select size="1" name="Msg1LabelSubjectAndAddressesDropDown">

<option selected>Select Addressee</option>

<option value="Customer Service, Customer Service Inquiry from the Web, Frank
Jones, Frank, JSend.com, Linda Jones, Linda, JSend.com">Customer
Service</option>

<option value="Technical Support, Technical Support Inquiry from the Web, Bob
Smith, Bob, JSend.com, Sue Smith, Sue, JSend.com">Technical Support</option>

</select>
```

Restrictions

If *MsgxLabelSubjectAndAddressesDropDown* is specified, the corresponding *MsgxAddrList* and *MsgxSubject* may *not* be specified.

Back Reference: Duplicating a "Prior Value"

There is no Back Reference capability for this variable. It is not possible to make Msg2 be the same as Msg1. An asterisk ("*") is an illegal value. An empty value means it will not be used.

Special Concerns

MsgxLabelSubjectAndAddressesDropDown is not reported automatically. It is only reported if specifically specified in *FormFieldNameLabelPlusList*.

Edits

- C008: Check email address for format of local address part (page 168)
- C009: Check email address for number of parts in domain name (page 169)
- C010: Check email address for format of domain name (page 169)
- C011: Check email address for whether domain name can receive email (page 170)
- C020: Check number of sub-parameters (a multiple of 3, plus 1) (page 172)
- C021: Check to ensure that neither *MsgxAddrList* nor *MsgxSubject* is specified if *MsgxLabelSubjectAndAddressesDropDown* is specified (page 173)
- C022: This field may not have a value of asterisk ("*"). (page 173)

E025: Check to ensure something was selected (page 182) [Actually checks to see if the parameter chosen only had one sub-parameter.]

M<small>SGX</small>S<small>UBJECT</small>

Specific Fields

This Hidden Field Parameter applies to all three: *Msg1Subject, Msg2Subject* and *MsgEchoSubject.*

Description

The email subject for Message 1, Message 2 or the Echo/Confirmation Message.

Values

Any text value is permissible. No edits are performed on this Hidden Field Parameter.

Default

The default value is:

- For Message 1 and 2 – "Message from Web Form".
- For the Echo Message – "Confirming Your Message on the Web Form".

Examples

For example,

```
<input name="MsgEchoSubject" type="hidden"
value="You Filled Out the Email Form on www.Someplace.com">
```

Restrictions

If *MsgxLabelSubjectAndAddressesDropDown* is specified, the corresponding *MsgxAddrList* and *MsgxSubject* may *not* be specified.

Back Reference: Duplicating a "Prior Value"

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value does not exclude any fields.

Edits

- C021: Check to ensure that neither *MsgxAddrList* nor *MsgxSubject* is specified if *MsgxLabelSubjectAndAddressesDropDown* is specified (page 173)
- C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGXSUBJECTFIELD

Specific Fields

This Hidden Field Parameter applies to both *Msg1SubjectField* and *Msg2SubjectField*. There is no field named *MsgEchoSubjectField*.

Description

The field into which the Form User may type the email subject for Message 1 or Message 2.

Values

The specified Field Name must exist on the form.

Default

No default.

Examples

To specify **subject1** as the Field Name:

```
<input name="Msg1SubjectField" type="hidden" value="subject1">
```

Notes

This parameter may be used *even if MsgxSubject or MsgxLabelSubjectAndAddressesDropDown* is used

If the Form User does specify a value in the specified field, it will *override* any value specified in *MsgxSubject or MsgxLabelSubjectAndAddressesDropDown*.

Back Reference: Duplicating a "Prior Value"

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value does not exclude any fields.

Edits

- ▪ C001: Referenced fields must exist on the form; the special fields ip, browser and referer are not allowed (page 166)
- ▪ C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGXTEXTBOTTOM

Specific Fields

This Hidden Field Parameter applies to all three: *Msg1TextBottom, Msg2TextBottom* and *MsgEchoTextBottom*.

Description

Brief text to be printed at the bottom of Message 1, Message 2, or the Echo/Confirmation Message.

Values

Any text value is permissible. No edits are performed on this Hidden Field Parameter. The text should be less than 200 characters in length.

Default

The default value is:

- For Message 1 and 2 – "[End Message]".
- For the Echo Message – "The data above was sent for you.".

Examples

For example,

```
<input name="MsgEchoTextBottom" type="hidden" value="[end of message]">
```

Back Reference: Duplicating a "Prior Value"

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value does not exclude any fields.

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

MSGXTEXTTOP

Specific Fields

Msg1: Yes   Msg2: Yes   MsgEcho: Yes

This Hidden Field Parameter applies to all three: *Msg1TextTop, Msg2TextTop* and *MsgEchoTextTop*.

Description

This Hidden Field Parameter applies to all three message types: Msg1, Msg2 and MsgEcho.

Brief text to be printed at the top of the email in Message 1, Message 2 or the Echo/Confirmation Message.

Values

Any text value is permissible. No edits are performed on this Hidden Field Parameter. The text should be less than 200 characters in length.

Default

The default value is:

- For Message 1 and 2 – "[Begin Message]".
- For the Echo Message – "You entered the following data:".

Examples

For example,

```
<input name="Msg1TextTop" type="hidden" value="This email is being sent
because a user filled out the Web Form on the website.">
```

Back Reference: Duplicating a "Prior Value"

A value of asterisk ("*") means to use the value of the prior message (for Message 2, use Message 1; for MsgEcho, use Message 2 or 1). Back Reference is not available for Message 1. An empty value does not exclude any fields.

Edits

- C022: This field may not have a value of asterisk ("*"). (page 173) NB: Message 1 only.

THE ERROR MESSAGES

Huggins' Email Form Script displays error messages when it encounters an error.

These error messages have been written to be easily understood and to provide maximum information to both the Form User and the Form Designer.

Two types of error messages are provided:

- **Form User**
  These error messages are for the person completing the form and generally indicate that an incorrect value was entered or that a required answer has not been given. These error numbers begin with "E".

- **Form Designer / Configuration**
  These error messages are for the person setting up the form on the website. These generally indicate that a Hidden Field parameter was set incorrectly. These error numbers begin with "C".

THE ERROR PAGE

ERROR MESSAGE CONVENTIONS

In these error messages, several conventions are followed:

- **Error Message Number**
  All error messages include the error message number at the end of the message.

- Form Designer / Configuration error messages begin with "C".

- Form User error messages begin with "E".

- **$EditFieldLabel1**
  This is where the "label" for a field is placed in the error message. The label of a field is the text, specified by the Form Designer, to display in an email as the name of a field. Labels of fields are usually more descriptive than the name of the field. This is usually bolded in the error message.

- **$EditFieldMaxSize1**
  This is where the maximum field size is placed in the error message.

- **$EditFieldMinSize1**
  This is where the minimum field size is placed in the error message.

- **$EditFieldPos1**
  This is where the "position" for sub-parameter is placed in the error message. Many Hidden Parameters contain a list of sub-parameters. The "position" of a sub-parameter identifies which particular sub-parameter is causing the error. $EditFieldPos1 identifies the position of the field name causing the error.

- **$EditFieldName1**
  This is where the "name" for a field is placed in the error message. The name of a field is the text used in the form to name the field. Names of fields are usually less descriptive than the label of the field. This is usually bolded in the error message.

- **$EditFieldValue1**
  This is where the "value" for a field (the value the Form User entered) is shown. The $EditFieldValue1 is usually shown in parentheses, following the $EditFieldLabel1.

- **$EditParmName1**
  This is where the "name" of a Hidden Parameter is placed in the error message. Hidden Parameters are the values specified by the Form Designer to tell the script how to operate. This is usually bolded in the error message.

- **$EditParmValue1**
  This is where the "value" of a Hidden Parameter is placed in the error message. Hidden Parameters are the values specified by the Form Designer to tell the script how to operate.

- **$EditSubParmName1**

    This is where the "name" of a sub-parameter is placed in the error message. Sub-parameters are individual parameters that, together, comprise a Hidden Parameters. Some Hidden parameters are comprised of sub-parameters or even groups of sub-parameters. This is usually bolded in the error message.

- **$EditSubParmPos1**

    This is where the "position" for sub-parameter is placed in the error message. Many Hidden Parameters contain a list of sub-parameters. The "position" of a sub-parameter identifies which particular sub-parameter is causing the error. $EditSubParmPos1 identifies the position of a sub-parameter (other than a field name) causing the error.

- **Other "$-names"**

    Within the error message, there may be other $-names. These represent other values that will be inserted into the error message. They are explained beneath the error message. They may be bolded or highlighted for emphasis.

CONFIGURATION ERROR MESSAGES

C001: FIELD NOT SENT TO FORM SCRIPT

The control parameter named **$EditParmName1** contains the name of a field (**$EditFieldName1**) that was ***not sent to the Form Script***. The field is named in sub-parameter #$EditFieldPos1. You must enter the name of a field that is actually on the form and will be sent to the form script. (Note that "browser", "ip", "referer" and "referrer" are ***not*** legal in this use.) (C001)

The script checks Hidden Field Parameters that specify the names of fields to ensure that the named field was actually passed to the script. This error is similar to C002, except that for this error, "browser", "ip", "referer" and "referrer" are not permitted.

This error can occur with: FormEmailField, FormEmailFieldList, FormFieldNameEditList and MsgxFieldNameValueSubstitutionList.

C002: FIELD NOT SENT TO FORM SCRIPT

The control parameter named **$EditParmName1** contains the name of a field (**$EditFieldName1**) that was **not sent to the Form Script**. The field is named in sub-parameter #$EditFieldPos1. You must enter the name of a field that is actually on the form and will be sent to the form script. (Note that "browser", "ip", "referer" and "referrer" ***are*** legal in this use.) (C002)

The script checks Hidden Field Parameters that specify the names of fields to ensure that the named field was actually passed to the script. This error is similar to C001, except that for this error, "browser", "ip", "referer" and "referrer" are permitted.

This error can occur with: FormFieldNameLabelPlusList, FormNameFieldList, MsgxFieldNameExcludeList, and MsgxFieldNameLabelList.

C003: REDUNDANT PARAMETERS SPECIFIED

The control parameter named **$EditParmName1** ("$EditParmValue1") and **$EditParmName2** ("$EditParmValue2") are ***both specified as parameters***. These two parameters are redundant. ***Only one*** of these parameters is permitted. You must remove one of these parameters. *Because* **$EditParmName1** *is deprecated, we recommend keeping* **$EditParmName2** *and removing* **$EditParmName1**. (C003)

The script checks redundant parameters to ensure that only one is specified. Redundant parameters occur when a previously use parameter is deprecated, and replaced in functionality with a new parameter. The old, deprecated parameter may be used or the newer parameter may be used – but not both.

This error can occur with: CopyUser & FormEchoUser, FormEmailField & FormEmailFieldList and FormNameField & FormNameFieldList.

C004: PARAMETER MUST BE YES/NO

The control parameter named **$EditParmName1** ("$EditParmValue1") does *not contain "y", "yes", "n" or "no"* (case does not matter). These are the *only* four values permitted for this parameter. You must use one of these values. (You may use any mix of upper-case and lower-case.) (C004)

The script checks yes/no fields to ensure they contain a legal value. This error is similar to C005, but C005 checks sub-parameters within the parameter.

This error can occur with: CopyUser, FormEchoUser, FormTestEmailDomain, FormTestEmailFormat, FormTestFieldMinLengths, FormTestFieldMaxLengths, FormTestInjectionExploits and MsgxForceDefaultFromAddr.

C005: SUB-PARAMETER MUST BE YES/NO, OR MAYBE SOME OTHER VALUE

The control parameter named **$EditParmName1** has an *illegal value in a sub-parameter*. The sub-parameter contains "$EditSubParmValue1\ but the only legal values are $EditSubParm1LegalValues. You must use one of these values. (You may use any mix of upper-case and lower-case.) The sub-parameter is for the field named **$EditFieldName1**. The field name is in sub-parameter #$EditFieldPos1 and the illegal value is in sub-parameter #$EditSubParmPos1. Note that this error is frequently caused either by a missing comma or extra comma coming *before* the sub-parameter with the error. (C005)

The script checks sub-parameters which must be yes/no to ensure they contain a legal value. This error is similar to C004, but C004 checks parameters without sub-parameters.

This error can occur with: FormFieldNameLabelPlusList (required flag). For this field, the legal values are "y", "yes", "n", "no", "r" and "required".

As noted in the error message, this error is frequently caused by a missing comma or an extra comma in the parameter which occurs earlier in the parameter.

Version 2.2.2 substantially improved this error message to better identify the actual error.

C006: UNKNOWN EDIT SPECIFICATION

In the control parameter named **$EditParmName1**, *the edit ("$EditEditValue1") specified for* **$EditFieldName1** *is not recognized*. You must specify a valid edit. The edit specification begins in sub-parameter #$EditFieldPos1 and the erroneous edit value is in sub-parameter #$EditEditPos1. (C006)

The script checks the edits specified to be sure they are a known edit.

This error can occur with: FormFieldNameEditList.

C007: ILLEGAL EMAIL ADDR – WRONG NUMBER OF CHARACTERS OR "@"

The control parameter named **$EditParmName1** contains an *illegal email address* ("$EditSubParmValue1"). The email address is illegal because it has the *wrong number of characters in one section or the wrong number of @ symbols*. You must enter a valid email address into this field. The erroneous email address begins in sub-parameter #$EditSubParmPos1. The two "parts" of the email address are in sub-parameter #$EditSubParmPos2 and sub-parameter #$EditSubParmPos3. (C007)

The script checks the specification of an email address in a Hidden Field Parameter. Email addresses are specified in three "parts": the name part, the part before the "@", and the domain name part after the "@". The two parts referenced in this error are the part before the "@" and the domain name part after the "@".

This error checks to see if the parts contain too many characters or if the "@" is duplicated.

This error can occur with: MsgEchoFromAddr, MsgxAddrList and MsgxDefaultFromAddr.

This edit is part of a group of edits (C007, c008, C009, C010 and C011) which check validity of email addresses configured by the Form Designer. Similar field edits are E005, E006, E007, E008 and E009.

C008: ILLEGAL EMAIL ADDR – INCORRECT LOCAL ADDRESS PART

The control parameter named **$EditParmName1** contains an *illegal email address* ("$EditSubParmValue1"). The email address is illegal because the *local address part before the @-sign ("$EditSubParmValue2") is invalid*. You must enter a valid email address into this field. The erroneous email address begins in sub-parameter #$EditSubParmPos1. The two "parts" of the email address are in sub-parameter #$EditSubParmPos2 and sub-parameter #$EditSubParmPos3. (C008)

The script checks the specification of an email address in a Hidden Field Parameter. Email addresses are specified in three "parts": the name part, the part before the "@", and the domain name part after the "@". The two parts referenced in this error are the part before the "@" and the domain name part after the "@".

This error checks to see if the part before the "@" is invalid (e.g., contains an illegal character like "[".)

This error can occur with: MsgEchoFromAddr, MsgxAddrList and MsgxDefaultFromAddr.

This edit is part of a group of edits (C007, C008, C009, C010 and C011) which check validity of email addresses configured by the Form Designer. Similar field edits are E005, E006, E007, E008 and E009.

C009: ILLEGAL EMAIL ADDR – NOT ENOUGH PARTS IN DOMAIN NAME

The control parameter named **$EditParmName1** contains an ***illegal email address*** ("$EditSubParmValue1"). The email address is illegal because ***there are not enough parts in the domain name ("$EditSubParmValue2")***. You must enter a valid email address into this field. The erroneous email address begins in sub-parameter #$EditSubParmPos1. The two "parts" of the email address are in sub-parameter #$EditSubParmPos2 and sub-parameter #$EditSubParmPos3. (C009)

The script checks the specification of an email address in a Hidden Field Parameter. Email addresses are specified in three "parts": the name part, the part before the "@", and the domain name part after the "@". The two parts referenced in this error are the part before the "@" and the domain name part after the "@".

This error checks to see if the domain name contains at least two parts around a separating period (e.g., "yahoo" would fail, but "yahoo.com" would pass).

This error can occur with: MsgEchoFromAddr, MsgxAddrList and MsgxDefaultFromAddr.

This edit is part of a group of edits (C007, c008, C009, C010 and C011) which check validity of email addresses configured by the Form Designer. Similar field edits are E005, E006, E007, E008 and E009.

C010: ILLEGAL EMAIL ADDR – DOMAIN NAME BUILT WRONG

The control parameter named **$EditParmName1** contains an ***illegal email address*** ("$EditSubParmValue1"). The email address is illegal because ***the domain name ("$EditSubParmValue2") contains illegal characters or is built wrong***. You must enter a valid email address into this field. The erroneous email address begins in sub-parameter #$EditSubParmPos1. The two "parts" of the email address are in sub-parameter #$EditSubParmPos2 and sub-parameter #$EditSubParmPos3. (C010)

The script checks the specification of an email address in a Hidden Field Parameter. Email addresses are specified in three "parts": the name part, the part before the "@", and the domain name part after the "@". The two parts referenced in this error are the part before the "@" and the domain name part after the "@".

This error checks to see if the domain name contains illegal characters or is built wrong.

This error can occur with: MsgEchoFromAddr, MsgxAddrList and MsgxDefaultFromAddr.

This edit is part of a group of edits (C007, c008, C009, C010 and C011) which check validity of email addresses configured by the Form Designer. Similar field edits are E005, E006, E007, E008 and E009.

C011: ILLEGAL EMAIL ADDR – DOMAIN NAME DOESN'T EXIST OR CAN'T RECEIVE EMAIL

The control parameter named **$EditParmName1** contains an **illegal email address** ("$EditSubParmValue1"). The email address is illegal because **either the domain name ("$EditSubParmValue2") does not exist, or it is not set up to receive email**. You must enter a valid email address into this field. The erroneous email address begins in sub-parameter #$EditSubParmPos1. The two "parts" of the email address are in sub-parameter #$EditSubParmPos2 and sub-parameter #$EditSubParmPos3. If you are using a Windows server, you may need to specify the FormPleaseForgiveMyUseOfAWindowsServer parameter. (C011)

The script checks the specification of an email address in a Hidden Field Parameter. Email addresses are specified in three "parts": the name part, the part before the "@", and the domain name part after the "@". The two parts referenced in this error are the part before the "@" and the domain name part after the "@".

This error checks to see if the domain name exists and if it is configured to receive mail (i.e., has MX records configured for the domain name). This edit does **not** check to confirm that the email address is an actual email address that can receive mail.

This edit can be disabled by turning on the FormPleaseForgiveMyUseOfAWindowsServer parameter. See page 123.

This error can occur with: MsgEchoFromAddr, MsgxAddrList and MsgxDefaultFromAddr.

This edit is part of a group of edits (C007, c008, C009, C010 and C011) which check validity of email addresses configured by the Form Designer. Similar field edits are E005, E006, E007, E008 and E009.

C012: MISSING TERMINATOR

The control parameter named **$EditParmName1** was **not properly terminated**. It should have terminated with a tilde ("~"). You must correct this parameter to terminate with a tilde ("~"). On many keyboards the tilde ("~") is on the key left of the "1" at the top of the keyboard. (C012)

Some Hidden Field Parameters which contain a list of sub-parameters terminate that list with a "~". This error identifies a parameter that requires a terminating "~" but does not have one.

This error can occur with: FormFieldNameEditList.

C013: INDIVIDUAL EDIT SPECIFICATION NOT PROPERLY TERMINATED

In the control parameter named **$EditParmName1**, **one of the edits was not properly terminated**. The edit probably has **an incorrect number of sub-parameters**. You must correct this edit. The edit begins with sub-parameter #$EditFieldPos1 for the field **$EditFieldLabel1**. It is a "$EditEditValue1" edit. It should have terminated with a "~" in sub-parameter #$EditSubParmPos4. Instead that sub-parameter contained "$EditSubParmValue4". (C013)

FormFieldNameEditList contains a list of sub-parameters which specify individual edits. Each individual edit terminates with a "~". This error identifies an individual edit that does not properly terminate with a "~".

This error can occur with: FormFieldNameEditList.

C014: ILLEGAL TEXT EDIT SUB-PARAMETER

In the control parameter named **$EditParmName1**, for the "text" edit, ***one of the sub-parameters is not recognized***. You must correct this edit to specify a valid sub-parameter. The edit begins with sub-parameter #$EditFieldPos1 for the field $EditFieldLabel1. The illegal sub-parameter is at position #$EditSubParmPos1 and has a value of "$EditSubParmValue1". (C014)

FormFieldNameEditList contains a list of sub-parameters which specify individual edits. One type of exit is a "text" edit. One of the sub-parameters within the text edit is illegal and not recognized.

This error can occur with: FormFieldNameEditList.

C015: WRONG NUMBER OF SUB-PARAMETERS (ABSOLUTE)

The control parameter named **$EditParmName1** has the ***wrong number of sub-parameters ($EditSubParmValue1)***. The number of sub-parameters *must be $EditSubParmValue2*. You must correct this edit to specify $EditSubParmValue2 sub-parameters. (C015)

Several Hidden Field Parameters require a specific number of sub-parameters. This edit checks to ensure the right number of sub-parameters is present.

This error can occur with: MsgEchoFromAddr and MsgxDefaultFromAddr.

C016: WRONG NUMBER OF SUB-PARAMETERS (MULTIPLE)

The control parameter named **$EditParmName1** has the ***wrong number of sub-parameters ($EditSubParmValue1)***. The ***number of sub-parameters must be a multiple of $EditSubParmValue2***. You must correct this edit to specify a correct number of sub-parameters. (C016)

Several Hidden Field Parameters require specific numbers of sub-parameters. These parameters typically require that the number of sub-parameters be a multiple of 2, or 3, or some other number. This edit checks to ensure the right number of sub-parameters is present.

This error can occur with: MsgxAddrList, MsgxFieldNameLabelList and MsgxFieldNameValueSubstitutionList.

C017: SUB-PARAMETER CONTAINS AN ILLEGAL INTEGER

$EditParmName1 contains an ***illegal integer*** as a sub-parameter. The ***sub-parameter ("$EditSubParmValue1")should be an integer but contains an illegal character*** ($EditCharFormatted1) in *position* #$EditCharPos1. You must correct this sub-parameter to contain a legal integer. This sub-parameter is the **$EditSubParmName1** for the specification of the field named **$EditFieldName1**. You ***must*** enter a legal integer for this field. (C017)

This edit identifies a sub-parameter which must be an integer, but contains illegal characters.

This error can occur with: FormFieldNameLabelPlusList.

C018: INJECTION EXPLOIT

The control parameter named **$EditFieldParm1** contains a ***possible injection exploit***. When a spambot places *content-type*, *bcc:*, *cc:*, *document.cookie*, *onclick*, or *onload* in a field this is called an "injection exploit". *Placing these values in a field is one way spambots hijack the form/script to send spam.* If you are not a spambot and have placed one of these codes in any field, you must find another way to send this information. Type your information without using one of these codes. (C018)

This edit checks fields to see if "injection exploits" have been sent in any Hidden Field Parameters. Injection exploits are special values (like "cc:") which can let a spambot hijack a form/script.

This error relates to Hidden Field Parameters. Currently, it is performed on all Hidden Field Parameters. There is no option to turn this edit off. For the similar edit on Form Fields, see error E010.

C019: TEMPORARY PARM MUST BE REPLACED

The control parameter named **$EditParmName1** was used temporarily used during development of Pre-Releases of Version 2.0 while it was in test but ***is now invalid***. Instead, the parameter became named **$EditParmName2**. You must replace your use of **$EditParmName1** with **$EditParmName2** instead. In addition you should check the manual to see if any format changes are required. (C019)

This edit checks fields to see if people have used Pre-Releases of Version 2.0 while it was in test and used parameter names that were changed. This helps people who did beta testing to find and make these changes.

This error can occur with: FormFieldEmailField, FormFieldNameFieldList, and FormFieldNameValueSubstitutionList

C020: WRONG NUMBER OF SUB-PARAMETERS (MULTIPLE PLUS SOME)

The control parameter named **$EditParmName1** has the ***wrong number of sub-parameters ($EditSubParmValue1)***. The number of sub-parameters must be $EditSubParmValue3 more than a multiple of $EditSubParmValue2. You must correct this edit to specify a correct number of sub-parameters. (C020)

Several Hidden Field Parameters require a specific number of sub-parameters. This edit checks to ensure the right number of sub-parameters is present.

This error can occur with: Msg1LabelSubjectAndAddressesDropDown and Msg2LabelSubjectAndAddressesDropDown.

See also C015 (page 171) and C016 (page 171).

C021: REDUNDANT PARAMETERS SPECIFIED

The control parameters named **$EditParmName1** ("$EditParmValue1") and **$EditParmName2** ("$EditParmValue2") are ***both specified as parameters***. These two parameters are redundant. Only one of these parameters is permitted. You must remove one of these parameters. (C021)

This error occurs when MsgxLabelSubjectAndAddressesDropDown is specified and either MsgxAddrList or MsgxSubject is also specified.

Also, pre-release 333 of Version 2.0 added MsgxFieldNameValueSubstitutionList2. The restriction is that you can either use the "list" format or the "list2" format, but not both.

This error can occur with: MsgxFieldNameValueSubstitutionList and MsgXFieldNameValueSubstitutionList2.

C022: ASTERISK ("*") IS ILLEGAL IN THIS FIELD

The control parameter named **$EditParmName1** ("$EditParmValue1") ***is not permitted to have a value of asterisk***. There is no Back Reference or prior value for an asterisk to reference. You must remove the asterisk. (C022)

This error can occur with: CopyUser, FormCSVFileName, FormEmailField, FormEmailFieldList, FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening, FormErrorPageTemplate, FormErrorPageTitle, FormFieldNameEditList, FormFieldNameLabelPlusList, FormNameField, FormNameFieldList, FormNextURL, NextURL, MsgEchoFromAddr, Msg1LabelSubjectAndAddressesDropDown, Msg2LabelSubjectAndAddressesDropDown, Msg1AddrList, Msg2AddrList, Msg1DefaultFromAddr, Msg1FieldLabelValueSeparator, Msg1FieldNameExcludeList, Msg1FieldNameLabelList, Msg1FieldNameValueSubstitutionList, Msg1FieldNameValueSubstitutionList2, Msg1Subject, Msg1SubjectField, Msg1TextBottom, Msg1TextTop

C023: FORMFIELDNAMEVALUESUBSTITUTIONLIST WAS ELIMINATED AS A PARAMETER

The control parameter named **FormFieldNameValueSubstitutionList** ("$EditParmValue1") ***was eliminated as a parameter***. You should probably assign the same values to **Msg1FieldNameValueSubstitutionList**. You may also wish to consider using an asterisk ("*") for **Msg2FieldNameValueSubstitutionList** and **MsgEchoFieldNameValueSubstitutionList** to pick up the value of **Msg1FieldNameValueSubstitutionList**. This change was introduced in V2.0-PR-329. (C023)

Pre-release 329 of Version 2.0 eliminated the use of FormFieldNameValueSubstitutionList. This error traps this use and guides the transition.

This error can occur with: FormFieldNameValueSubstitutionList.

C024: CANNOT OPEN ERROR PAGE TEMPLATE FILE

The control parameter named **$EditParmName1** ("$EditParmValue1") was **unable to open the specified file**. You must specify a correct file. (C024);

This error can occur with: FormErrorPageTemplate.

C025: SPECIFIED FILE IS INCORRECTLY FORMATTED

The control parameter named **$EditParmName1** ("$EditParmValue1") specifies a file that **does not contain "$TemplateErrorPlacementString\"**. You must specify a correct file. (C025)

This error can occur with: FormErrorPageTemplate.

C026: NUMBER IN A SUB-PARAMETER CONTAINS AN ILLEGAL CHARACTER

The control parameter named **$EditParmName1** has an **illegal number in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)**. This value should be a number but has an illegal character ($EditCharFormatted1) in position #$EditCharPos1. You must enter a legal number for this field. (C026)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C027: NUMBER IN SUB-PARAMETER CONTAINS TOO MANY PERIODS

The control parameter named **$EditParmName1** has an **illegal number in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)**. This value should be a number but has too many periods. Only one period is allowed as a decimal point. You must enter a legal number for this field. (C027)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C028: NUMBER IN SUB-PARAMETER IS TOO LOW

The control parameter named **$EditParmName1** has an **illegal number in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)**. The value is too low. The lowest possible value is $EditLowValue. You must enter a legal number for this field. (C028)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C029: NUMBER IN SUB-PARAMETER IS TOO HIGH

The control parameter named **$EditParmName1** has an **illegal number in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)**. The value is too high. The highest possible value is $EditHighValue. You must enter a legal number for this field. (C029)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C030: ILLEGAL COLOR SPEC IN A SUB-PARAMETER – ILLEGAL COLOR VALUE

The control parameter named **$EditParmName1** has an ***illegal color value in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)***. This value does not start with a "#" and doesn't match a color value. You must enter a legal color for this field. Colors must be pound sign ("#") followed by a 6 character hex number or must be an accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". (C030)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C031: ILLEGAL COLOR SPEC IN A SUB-PARAMETER – WRONG LENGTH NUMBER

The control parameter named **$EditParmName1** has an ***illegal color value in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)***. This value starts with a "#" but is the wrong length. You must enter a legal color for this field. Colors must be pound sign ("#") followed by a 6 character hex number or must be an accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". (C031)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C032: ILLEGAL COLOR SPEC IN A SUB-PARAMETER – INVALID HEX NUMBER

The control parameter named **$EditParmName1** has an ***illegal color value in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)***. This value starts with a "#" but is an invalid hex number. You must enter a legal color for this field. Colors must be pound sign ("#") followed by a 6 character hex number or must be an accepted color value. The accepted color values are "black", "silver", "gray", "white", "maroon", "red", "purple", "fuchsia", "green", "lime", "olive", "yellow", "navy", "blue", "teal" and "aqua". (C032)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C033: ILLEGAL BOLD FLAG

The control parameter named **$EditParmName1>** has an **illegal bold flag in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)**. The bold flag must be "y", "yes", "n", "no" or empty. You must enter a legal value for this field. (C033)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C034: EXPECTED CSS CLASS VALUE NOT FOUND

The control parameter named **$EditParmName1** *does not specify a CSS Class value*. The control parameters are specifying CSS Class values for some of the Error Page formatting. When this is done all Error Page formatting must be designated with CSS Class values. Further, CSS Class values are required for all six (6) Error Page Format items, including: (1) Heading 1, (2) Heading 2, (3) the error page opening line, (4) the error messages themselves, (5) the error page closing line and (6) the footer. If you wish to use CSS Class values, you must enter a CSS Class value for this field. (C034)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C035: UNEXPECTED "OLD STYLE" ERROR PAGE FORMAT FOUND

The control parameter named **$EditParmName1** *specifies a CSS Class while the parameter named $EditParmName2 specifies "old style" formatting (e.g., face, size, color and "boldness")*. You must use one or the other for for all six (6) Error Page Format items, including: (1) Heading 1, (2) Heading 2, (3) the error page opening line, (4) the error messages themselves, (5) the error page closing line and (6) the footer. (C035)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter, FormErrorPageHeading1, FormErrorPageHeading2, FormErrorPageLineClosing, FormErrorPageLineOpening

C036: SUB-PARAMETER ISN'T BLANK (NULL)

The control parameter named **$EditParmName1** has a *non-blank (non-null) value in sub-parameter #$EditSubParmPos1 ($EditSubParmValue1)*. This sub-parameter must be blank (null). You must reconfigure the parameter to present this sub-parameter as empty. (C036)

This error can occur with: FormErrorPageErrorMsg, FormErrorPageFooter

FORM USER ERROR MESSAGES

E001: PSEUDO-CAPTCHA ERROR

The field named **$EditFieldLabel1** ("$EditFieldValue1") is an anti-spam field, but ***does not contain the "correct answer"***. You must enter the "correct answer" for the system to process your form. *This question is important because it helps ensure that spambots do not use this form/script to process spam.*

This edit relates to the pseudo-captcha edit that can be used to help prevent spambots from using the form/script.

This edit is processed for the Form Field specified in the "pseudo-captcha" edit, in the FormFieldNameEditList (page 109).

E002: MISSING REQUIRED FIELD

The field named **$EditFieldLabel1** is a ***required field*** but you ***did not enter a value***. You must enter a value in this field. (E002)

This edit checks fields specified as required in FormFieldNameLabelPlusList.

E003: TOO MUCH DATA ENTERED

The field named **$EditFieldLabel1** ("$EditFieldValue1") has too much information in it. It is "too long". You ***entered too much information --- too many characters.*** *In theory, if the webmaster configured the form correctly, you should not be able to get this error. This error should occur when spambots try to stuff values into fields to hijack the form/script.* If you are not a spambot, please type less. Do not enter more than $EditFieldMaxSize1 characters. Also, please contact the webmaster to correct the form. (E003)

This edit checks fields against the maximum length specified in FormFieldNameLabelPlusList. This length should be the same length specified for the field in the form.

Textboxes for the form should always specify "maxlength". This will cause the browser to restrict entry to this max number of characters. If maxlength is specified, and if maxlength matches the max size specification in the parameters, this error should not occur in normal use. This error should occur only when spam bots over-stuff a field as they hijack the form/script. This error occurs in normal use when no maxlength is specified for the textbox or when the maxlength specified does not match the maximum length specified in FormFieldNameLabelPlusList.

E004: NOT ENOUGH DATA ENTERED

The field named **$EditFieldLabel1** ("$EditFieldValue1") does not have enough information in it. It is "too short". You ***did not enter enough information --- not enough characters.*** You must enter at least $EditFieldMinSize1 characters into this field. (E004)

This edit checks fields against the maximum length specified in FormFieldNameLabelPlusList.

E005: ILLEGAL EMAIL ADDR – WRONG NUMBER OF CHARACTERS OR "@"

The field named **$EditFieldLabel1** ("$EditFieldValue1") is not a valid email address either because it has the ***wrong number of characters in one section or the wrong number of @ symbols***. You must enter a valid email address into this field. (E005)

The script checks the specification of an email address entered by a Form User. Email addresses are entered by Form Users in two "parts": the part before the "@", and the domain name part after the "@".

This error checks to see if the parts contain too many characters or if the "@" is duplicated.

This edit is part of a group of edits (E005, E006, E007, E008 and E009) which check validity of email addresses entered by the Form User. Similar configuration edits are C007, C008, C009, C010 and C011.

Fields checked include those pointed to by FormEmailField and FormEmailFieldList.

E006: ILLEGAL EMAIL ADDR – INCORRECT LOCAL ADDRESS PART

The field named **$EditFieldLabel1** ("$EditFieldValue1") contains an ***illegal email address***. The email address is illegal because the ***local address part before the @-sign ("$EditFieldValue2") is invalid***. You must enter a valid email address into this field. (E006)

The script checks the specification of an email address entered by a Form User. Email addresses are entered by Form Users in two "parts": the part before the "@", and the domain name part after the "@".

This error checks to see if the part before the "@" is invalid (e.g., contains an illegal character like "[".)

This edit is part of a group of edits (E005, E006, E007, E008 and E009) which check validity of email addresses entered by the Form User. Similar configuration edits are C007, C008, C009, C010 and C011.

Fields checked include those pointed to by FormEmailField and FormEmailFieldList.

E007: ILLEGAL EMAIL ADDR – NOT ENOUGH PARTS IN DOMAIN NAME

The field named **$EditFieldLabel1** ("$EditFieldValue1") contains an ***illegal email address*** . The email address is illegal because there are ***not enough "parts" in the domain name ("$EditFieldValue2")***. You must enter a valid email address into this field. (E007)

The script checks the specification of an email address entered by a Form User. Email addresses are entered by Form Users in two "parts": the part before the "@", and the domain name part after the "@".

This error checks to see if the domain name contains at least two parts around a separating period (e.g., "yahoo" would fail, but "yahoo.com" would pass).

This edit is part of a group of edits (E005, E006, E007, E008 and E009) which check validity of email addresses entered by the Form User. Similar configuration edits are C007, C008, C009, C010 and C011.

Fields checked include those pointed to by FormEmailField and FormEmailFieldList.

E008: ILLEGAL EMAIL ADDR – DOMAIN NAME BUILT WRONG

The field named **$EditFieldLabel1** ("$EditFieldValue1") contains an ***illegal email address***. The email address is illegal because the ***domain name ("$EditFieldValue2") contains illegal characters or is built wrong***. You must enter a valid email address into this field. (E008)

The script checks the specification of an email address entered by a Form User. Email addresses are entered by Form Users in two "parts": the part before the "@", and the domain name part after the "@".

This error checks to see if the domain name contains illegal characters or is built wrong.

This edit is part of a group of edits (E005, E006, E007, E008 and E009) which check validity of email addresses entered by the Form User. Similar configuration edits are C007, C008, C009, C010 and C011.

Fields checked include those pointed to by FormEmailField and FormEmailFieldList.

E009: ILLEGAL EMAIL ADDR – DOMAIN NAME DOESN'T EXIST OR CAN'T RECEIVE EMAIL

The field named **$EditFieldLabel1** ("$EditFieldValue1") contains an ***illegal email address***. The email address is illegal because ***either the domain name ("$EditFieldValue2") does not exist, or it is not set up to receive email***. You must enter a valid email address into this field. (E009)

The script checks the specification of an email address entered by a Form User. Email addresses are entered by Form Users in two "parts": the part before the "@", and the domain name part after the "@".

This error checks to see if the domain name exists and if it is configured to receive mail (i.e., has MX records configured for the domain name). This edit does ***not*** check to confirm that the email address is an actual email address that can receive mail.

This edit is part of a group of edits (E005, E006, E007, E008 and E009) which check validity of email addresses entered by the Form User. Similar configuration edits are C007, C008, C009, C010 and C011.

Fields checked include those pointed to by FormEmailField and FormEmailFieldList.

E010: INJECTION EXPLOIT

The field named **$EditFieldLabel1** ("$EditFieldValue1") contains ***illegal content***. Such illegal content may be an ***injection exploit***. When a spambot places "*content-type*", "*bcc:*", "*cc: *", "*document.cookie*", "*onclick*", or "*onload*" in a field this is called an "injection exploit". *Placing these values in a field is one way spambots hijack the form/script to send spam.* If you have placed one of these codes in any field, you must find another way to send this information. Type your information without using one of these codes. (E010)

This edit checks fields to see if "injection exploits" have been sent in any fields. Injection exploits are special values (like "cc:") which can let a spambot hijack a form/script.

This error relates to Form Fields. For the similar edit on Hidden Field Parameters see error C018.

This edit can be toggled using FormTestInjectionExploits.

E011: EMAIL ADDRESS DOES NOT MATCH CONFIRMATION

The field named **$EditFieldLabel1** ("$EditFieldValue1") contains an email address that ***does not match the confirmation email address*** you entered in **$EditFieldLabel2** ("$EditFieldValue2"). You must enter matching values into these two fields. (E011)

This edit relates to the pair of values entered into the two fields specified by FormEmailFieldList. It also relates to the pair of values entered into the two fields specified by an "email" edit in FormFieldNameEditList.

E012: FIELD VALUE DOES NOT MATCH CONFIRMATION

The field named **$EditFieldLabel1** ("$EditFieldValue1") contains a value that **does not match the confirmation value** you entered in **$EditFieldLabel2** ("$EditFieldValue2"). You must enter matching values into these two fields. (E012)

This edit relates to the pair of values entered into the two fields specified by a "compare" edit.

E013: NO VALUE SELECTED IN DROP DOWN LIST

For the drop down list named **$EditFieldLabel1**, ***no value was selected***. (The value still shows as "$EditFieldValue1".) You must select a value in this drop down list. *This question is important because it helps ensure that spambots do not use this form/script to process spam.* (E013)

This edit relates to the value selected in the drop down list by a "drop down" edit.

E014: FIELD NOT LEFT EMPTY

The field named **$EditFieldLabel1** ("$EditFieldValue1") ***is not empty***. You must leave this field empty. *This question is important because it helps ensure that spambots do not use this form/script to process spam.* (E014)

This edit relates to the value entered in a field specified by an "attraction" edit.

E015: FIELD DOES NOT MATCH PERMITTED VALUES

The field named **$EditFieldLabel1** ("$EditFieldValue1") ***does not contain any of the permitted values***. You must correct your entry to match a permitted value. (E015)

This edit relates to the value entered in a field specified by an "equal" edit.

E016: FIELD MATCHES A PROHIBITED VALUE

The field named **$EditFieldLabel1** ("$EditFieldValue1") ***matched one of the prohibited values***. You must correct your entry to not match a prohibited value. (E016)

This edit relates to the value entered in a field specified by an "not equal" edit.

E017: INTEGER CONTAINS ILLEGAL CHARACTER

The field named **$EditFieldLabel1** ("$EditFieldValue1") should contain an integer but ***contains an illegal character ($EditCharFormatted1) in character position #$EditCharPos1***. You must enter a legal integer for this field. (E017)";

This edit relates to the value entered in a field specified by an "integer" edit.

$EditCharFormatted1 is either the character, surrounded by quotation marks, or is the phrase "a space", depending on what is found.

E018: INTEGER HAS TOO LOW A VALUE

The field named **$EditFieldLabel1** ("$EditFieldValue1") has ***too low a value*** ($EditTestValue). You must enter an integer and the lowest possible value is $EditLowValue. (E018)

This edit relates to the value entered in a field specified by an "integer" edit.

E019: INTEGER HAS TOO HIGH A VALUE

The field named **$EditFieldLabel1** ("$EditFieldValue1") has ***too high a value*** ($EditTestValue). You must enter an integer and the highest possible value is $EditHighValue. (E019)

This edit relates to the value entered in a field specified by an "integer" edit.

E020: NUMBER CONTAINS ILLEGAL CHARACTER

The field named **$EditFieldLabel1** ("$EditFieldValue1") should contain a number but ***contains an illegal character ($EditCharFormatted1) in position #$EditCharPos1***. You must enter a legal number for this field. (E020)";

This edit relates to the value entered in a field specified by a "number" edit.

$EditCharFormatted1 is either the character, surrounded by quotation marks, or is the phrase "a space", depending on what is found.

E021: TOO MANY PERIODS IN A NUMBER

The field named **$EditFieldLabel1** ("$EditFieldValue1") should contain a number but ***contains too many periods.*** Only one period is allowed as a decimal point. You must enter a legal number for this field. (E021)";

This edit relates to the value entered in a field specified by a "number" edit.

E022: Number Has Too Low a Value

The field named **$EditFieldLabel1** ("$EditFieldValue1") has ***too low a value*** ($EditTestValue). You must enter a number and the lowest possible value is $EditLowValue. (E022)

This edit relates to the value entered in a field specified by a "number" edit.

E023: Number Has Too High a Value

The field named **$EditFieldLabel1** ("$EditFieldValue1") has ***too high a value*** ($EditTestValue). You must enter a number and the highest possible value is $EditHighValue. (E023)

This edit relates to the value entered in a field specified by a "number" edit.

E024: Text Field Has an Illegal Character

$EditFieldLabel1 ("$EditFieldValue1") contains an ***illegal character ($EditCharFormatted1) in position #$EditCharPos1***. You must ***enter a legal text value*** for this field. (E024)

This edit relates to the value entered in a field specified by a "text" edit.

E025: No Value Selected in Drop Down List for Addressees

There is a drop down list that asks you where this email should go. ***You did not select a value***. (The value still shows as "$EditParmValue1".) You must select a value in this drop down list. *This question is important because it helps ensure that spambots do not use this form/script to process spam.* (E025)

This edit relates to the value selected in the field MsgxLabelSubjectAndAddressesDropDown.

See also E013 on page 180.

E026: Too Few Checkboxes Checked

The checkbox group known as **$EditSubParmLabel1** ***did not have enough boxes checked***. You checked $EditSubParmValue1 boxes. You must check at least $EditSubParmValue2 boxes. (E026)

This edit relates to FormFieldNameEditList, and specifically to the checkboxgroup edit.

E027: Too Many Checkboxes Checked

The checkbox group known as **$EditSubParmLabel1** ***had too many boxes checked***. You checked $EditSubParmValue1 boxes. You must not check more than $EditSubParmValue2 boxes. (E027)

This edit relates to FormFieldNameEditList, and specifically to the checkboxgroup edit.

WEB SPIDERS AND SPAMBOTS

WEB SPIDERS AND SPAMBOTS

Web spiders and spambots are two threats to email forms across the web. Understanding web spiders and spambots is important for webmasters and Form Designers.

Huggins' Email Form Script was designed to protect the form and script both from web spiders and from spambots.

WEB SPIDERS

Web spiders (also called web crawlers) follow links around the web, looking at web pages for email addresses. These spiders not only find the obvious email addresses, they also have been programmed to identify many of the common attempts to hide email addresses. The issue for Web Forms is that ***many*** of the available scripts do ***not*** attempt to hide the email addresses in ***any*** way. Instead, they simply place the email address plainly in a "hidden field" in the form. Web spiders find such email addresses in a heartbeat.

Huggins' Email Form Script has addressed this issue from the very start by separating the email into three parts. For example, the email address "John Doe <JDoe@aol.com>" is separated into these three parts:

- The Name: "John Doe"
- The part of the email address before the "@": "JDoe"
- The domain name: "aol.com"

At the present time, my testing has not shown any web spider which looks for this technique and reassembles the email address.

For more information this, see en.wikipedia.org/wiki/Web_spider.

SPAMBOTS

Another type of web spider (web crawler) looks for unprotected forms/scripts it can use to send spam directly.[60] One way it does this is by injecting HTML and header fields like "CC:" and "BCC" to get the script to send the spam for it.

For more information this, see http://en.wikipedia.org/wiki/Spambot.

---

[60] While the term "spambot" is also frequently used to mean web spiders that harvest email addresses, I am using it here to specifically refer to the spiders that abuse forms and scripts.

Version 2.0 of Huggins' Email Form Script implemented new techniques specifically designed to stop these spambots that have proven to be very effective.

SPAMBOT PREVENTION: CAPTCHA-LIKE VALIDATION TEST

CAPTCHA stands for "Completely Automated Public Turing test to tell Computers and Humans Apart".[61] This change does not implement true CAPTCHA. Instead, it allows the form to ask a simple question (e.g., "What color is the sky?", or "Summer isn't cold. What is it?" or "What is the 'opposite' of black?"), and check for the "right" answer. It is not sophisticated. But it does stop a significant number of spambots.[62]

SPAMBOT PREVENTION: BLANK REQUIRED SPAM TRAP

This enhancement allows the Form Designer to put a field on a form that a person will not fill in, but a spambot will. One technique is to create a blank field named "email" or "email2" (the name acts as "bait") and then to label the field "For Office Use Only" so that a person will ***not*** place data in it. A spambot, on the other hand, will likely place data in that field, causing the script to fail the edit.[63]

SPAMBOT PREVENTION: FIELD MAXIMUM SIZE CHECKS

When a web designer lays out the form, s/he will specify the maximum size of each field. The HTML will prevent the user from exceeding that size. But when a spambot attempts to hijack the form, it can send more data. This check looks for fields which exceed their maximum size, probably indicating a spambot.[64]

SPAMBOT PREVENTION: INJECTION EXPLOIT CHECKS

This feature automatically removes email code which has been "injected" into the form to hijack the form for spambot use. For example, it removes "BCC:" and "CC:" which spambots use to send spam email to other people.[65]

---

[61] The term CAPTCHA is a trademark of Carnegie Mellon University. For more on CAPTCHA, see www.captcha.net and en.wikipedia.org/wiki/Captcha.

[62] See *FormFieldNameEditList* on page 98 and *Field Edit 9 – Pseudo-Captcha: Test for Pseudo-Captcha Error (E001)* on page110.

[63] See *FormFieldNameEditList* on page 98 and Field Edit 10 – Attraction: A Spambot Attraction Test (E014) on page 111.

[64] See *FormFieldNameLabelPlusList* on page 98.

[65] See *FormTestInjectionExploits* on page 129.

SPAMBOT PREVENTION: REMOVE HTML MARKUP AND CODE

The script also automatically removes additional HTML markup and code that spambots might use to hijack the form/script.

FREQUENTLY ASKED QUESTIONS: THE FAQ

1. **What is a FAQ?**

   The Online Hacker Jargon File, version 4.4.7, 29.Dec.2003 (www.Jargon.org) defines FAQ (www.catb.org/~esr/jargon/html/F/FAQ.html) as a Frequently Asked Question. The *American Heritage Dictionary of the English Language* notes that it is "A list of frequently asked questions and their answers about a given subject."

2. **How often is "frequently". I mean, just how frequently are these questions asked?**

   Not all that frequently actually. But hey, give me a break and go with the flow.

3. **How much does this program cost?**

   It is free.

   Huggins' Email Form Script is made available through the Creative Commons GNU GPL License (2.0) (CreativeCommons.org/licenses/GPL/2.0). This is a free license and permits use on commercial websites. It restricts incorporation into some commercial products.

   This documentation is made available through the Creative Commons Attribution-No Derivative Works 3.0 Unported Licenses (3.0) (CreativeCommons.org/licenses/by-nd/3.0). This is a free license but restricts development of derivative works.

   This manual (ebook) explains the license in more detail in the section named *The License* beginning on page 200.

4. **Are there any conditions for use?**

   The script may be used for personal or commercial use without restriction. In return for the free use of Huggins' Email Form Script, I do request that Form Designers include a linked acknowledgement on their Form Page. This manual (ebook) explains the licenses in more detail in the section named The License on page 200.

5. **Does Huggins' Email Form Script work everywhere?**

   No. Testing has shown that some hosts have configured their hosting to create problems. To assist you in testing this, I have created PHP Test Scripts.
   I document these in the section titled Checking Your Host on page 23. Also see the section titled *Host Issues* that begins on page 189. I am also willing to help anyone with a problem host to find a better host.

6. How fast can Huggins' Email Form Script be implemented?

While Huggins' Email Form Script provides a plethora of options, it can also be implemented for simple use with only one or two parameters. While most Form Designers will want to spend more time to take advantage of the various features, Huggins' Email Form Script can be implemented in minutes if desired.

7. Do I have to use all the Hidden Field Parameters all the time?

No. The only truly required Hidden Field Parameter is *Msg1AddrList*.

FormNextURL is technically not required, but a Form Designer won't want the form without it so I am betting it will also be used 99.999% of the time.

The rest? Use 'em if you need 'em.

8. Are there any other fields I should especially consider?

Yes. Look at *FormFieldNameLabelPlusList*. This particular Hidden Field Parameter is so powerful that if you have more than just a few fields on your form you should definitely consider using it.

9. I need to send the message to three different people. I can use Message 1 and Message 2. But how do I get Message 3?

You don't do it that way. If you want to send the same message to three different people, you put all three addresses into one address list, like *Msg1AddrList*.

Do **not** use Message 2 to send the same message to a different person. **Only** use Message 2 to send a **different** message to different people. The message might be different because:

- It reports a different set of Form Fields.
- It reports the Form Fields using different Field Reporting Labels.
- It reports the Form Fields using different Substitution Values.
- A combination of these.

10. How often do people need to use Message 2?

I am betting not all that often. At this time, no one has reported a use for Message 2. I have never used if on any sites I manage. I added it there years ago "just in case" and it is there if you need it. But so far, as far as I knowno one has/

11. I'm getting a 404 error when I press submit on my form. What is causing that?

This error is caused by incorrectly specifying the fully qualified path to the script. There are four common errors:

(1) You may have left off your domain name.

(2) You may have left off intermediate folders/directories.

(3) You may have omitted the "http://" part of the path name.

(4) You may have changed the name of the script on your server (e.g., changed "php5" to "php") and not changed it in the form statement.

Double check all those possibilities.

12.   What are web spiders?

Web spiders are programs that automatically surf the web. Google, for example, uses spiders to look at pages, and the pages they point to, and the pages those pages point to, and so on and so on. Some spiders go page to page "harvesting" emails. That is, they look for emails on pages across the net and collect those emails so they can be used for spam. Huggins' Email Form Script has features specifically designed to defeat email harvesting by web spiders.

13.   What are spambots?

Spambots are programs that use other people's forms and form scripts to send spam. These programs can send spam to the people who normally get the form messages. And, they can also hijack the form script by embedding special codes to use the form and script to send spam to other people. Huggins' Email Form Script has a number of features specifically designed to defeat the spambots.

14.   How do I keep track of the fixes and enhancements of Huggins' Email Form Script?

Subscribe to the *HEFS Bulletin* to keep up to date about fixes and enhancements. You can subscribe by going to www.JamesSHuggins.com/hefs-bulletin. For more information see *Tracking Fixes and Enhancements* on page 199.

15.   Why aren't there any more questions?

I'm just getting started. I'm waiting for the users to ask me some. (Do you have a question? Please ask it.)

OVERVIEW

Through the testing done by volunteer beta testers, I have identified some of the problems you may encounter with Huggins' Email Form Script.

This section attempts to document some of those problems and assist with their resolution.

GENERAL LIMITATIONS ON PHP

Although PHP is more and more common, support for PHP is far from universal. And even hosts which support PHP may not support all of PHP.

For example, some web hosts

- Have not implemented PHP,
- Have not implemented it completely,
- Have disabled some functions of PHP,
- Require that PHP scripts be placed in specific directories, or
- Impose arbitrary limits on the use of PHP.

SPECIFIC EXAMPLES

This subsection presents some specific examples of these problems[66] for:

- EasyCGI.com
- BlueRazor.com
- M6.net
- Gate.com
- GoDaddy.com
- FastHosts.co.uk

---

[66] I would *love* to have more specific examples. If you encounter specific example, please document them in detail and send them to me at JHuggins@JamesSHuggins.com or JHuggins@EclecticPower.com.

EASYCGI.COM

Despite its name, this web host (www.EasyCGI.com) does not appear to support the ability of an invoked script to use the PHP mail function. All requests for assistance from the host tech support returned a standard answer: that their test PHP script worked. Unfortunately, their test PHP script did not include Form Posting or email sending functionality and their tech support seemed unwilling to address this issue.

BLUERAZOR.COM

- This web host (BlueRazor.com) had a number of limitations[67]
- They disabled some PHP functions on Linux accounts
- They disabled other PHP functions on Windows accounts
- They limited emails to 1000 per month
- They had some contradictory information in their help files. For example, their article on limiting PHP functions on Linux accounts said they had **disabled** the "mail" function. But the article on the 1000 email per month limit said they had **enabled** the "mail" function.

Again, things have changed since these issues were encountered, but these issues are some examples of the type of limitation users must look for.

M6.NET

This host specializes in Windows Server hosting.

I registered the domain name WindowsPHPTest.com and paid for an account on M6.net specifically to test Windows Server hosting. Their help desk was responsive and made great attempts. I have good praise for their attitude and timeliness. However, despite their valiant efforts, they were unable to create an environment in which my test scripts were able to send emails. And they were unable to explain why. [68]

One of the possible reasons for this is that, generally, individuals with skills in Windows Servers do not have skills using PHP.

---

[67] Some of these have changed since this was initially explored with BlueRazor.com. For example, recent checking indicates that the Linux problems may have been fixed. However, they have completely removed PHP support for Windows Servers and suggest that you move to a Linux server if you need PHP. The description of these issues is not intended to suggest that BlueRazor.com is not suitable as a host for your particular requirements, but only to describe my individual experiences.

[68] The description of these issues is not intended to suggest that M6.net is not suitable as a host for your particular requirements, but only to describe my individual experiences.

GATE.COM

One of the users of my script, Clo Knibbe[69] encountered this problem.[70] He reported that on the support page on their host[71] it says:

> Due to anti-spam prevention measures, Gate.com mail forms will only send to Gate.com mailboxes on your domain. This disables the ability to exploit a script. The scripts will not send to an external address.

In addition all emails sent through the script on his site are sent with a common "from address" despite the script's attempt to use a specific, custom "from address".

GODADDY.COM

In her testimonial to me,[72] Joyce Cimbalista wrote "I just completed implementing your script on a site, and will definitely be using it on others (except for the one on GoDaddy, which I found disallows the php mail function)."

I have not yet been able to find confirmation on the GoDaddy.com site, but it is still one more example of hosting issues[73].

FASTHOSTS.CO.UK

A user attempting to use the script on FastHosts.co.uk wrote me in May 2009 having received this error from the host.

---

[69] He also wrote a nice testimonial about this manual. See Testimonials on page 204.

[70] This was in May of 2008.

[71] http://kb.gate.com/display/2/index.asp?c=&cpc=&cid=&cat=&catURL=&r=0.971081

[72] See Testimonials on page 204.

[73] As of Jun.2009 I have not been able to confirm this problem, and, in fact, I know of some people successfully using this script on GoDaddy.

```
* The email is not generated as 'from' a Fasthosts  registered email address.
Because the email is sent from our SMTP server, you will need to ensure that
it is sent from a Fasthosts mailbox so that it can authenticate through the
server. This will also need to be included in  the 'ini_set' line of your
script (eg - ini_set('sendmail_from', "info@bala-backpackers.co.uk");)

* The '-f' parameter has not been included. When creating the  email in the
script, you will have a line similar to :

$sent  = mail($email_to, $email_subject, $message, $headers);

An extra  parameter must be added to the end of this line, defining the fro
address  as below :

$sent = mail($email_to, $email_subject, $message,  $headers, '-
f'$email_from);

I understand that your scripts  will not read exactly as the above lines, but
these lines define the  minimum required information needed to send an email
through our system.  More information can be found at:

http://www.fasthosts.co.uk/knowledge-base/?article_id=1077
```

In other words, they are saying that my script will **not** work on their host.

In response I wrote this host attempting to get a reply and clarification to confirm this. They first declined to clarify and advised me directly that they will **not** help me clarify because I am not a customer.

They wrote "This is most definitely a support enquiry - for specifics into what our scripting supports, or issues with scripting on our servers please email support@fasthosts.co.uk with your account credentials." This was a tad difficult since I am not one of their customers. However, after a bit of correspondence over a couple days, they did eventually phone me from UK to my home/office here in the colonies (USofA) to clarify.

Based on this telephone response, it seems that unlike many commercial web hosts (see below for examples), they are not equipped to offer the services required by my script as a part of their standard basic package offerings. However, if someone requires it, for an additional charge, they can offer an option of additional services (e.g., an additional server) at an additional price to accommodate my script.

For these reasons, it appears to me (my opinion) that using my script on this host is not an optimum option for most users and that use of alternate hosting services (either in addition to **or** instead of this host; see below) will provide the services my script requires at significant cost savings.

If this an issue for you, do not hesitate to contact me:

JHuggins@JamesSHuggins.com

or

JHuggins@EclecticPower.com.

THE .PHP5 ISSUE

The script you download from my site has a file type of ".php5". Many servers cannot handle this file type. If your server indicates that the script file cannot be posted to, change the name to have a file type of ".php". You will need to change the name of the script you store and also change the name of the script to match, in the html execution files you create.

TESTING YOUR HOST

In the effort to assist some beta testers with installation and use of the Huggins' Email Form Script, I have identified some problems and developed pages and scripts to help test the host.

For more information on these, see the section Checking Your Host on page 23.

WHAT DO PEOPLE DO?

There are a few things that well meaning hosts do to make life difficult.

For *many* people these are showstoppers ... for example, these well meaning hosts may send a confirming copy to the user. Some hosts also prohibit the email to the site owner from "coming from" the user. This prohibits just using a simple "reply" to communicate back.

But there is a simple, though brute force workaround. Some people using my script have purchased an account on a more permissive server ... just to host and use the script!! They keep their site on the old host but invoke the script on the new host. For a surprising number of people this brute force work around actually *works*!!

Others have moved their whole site and written a rather brutal email to the host.

A USE OF A USOFA HOST

Here is one plan you might consider if you are in the USofA:

BlueHost.com
http://www.BlueHost.com/?k_id=8237010
$3.99/month - $47.88/year
Includes hosting for up to 20 domains for 20 different websites![74]

NB: I use 1and1 for all my own hosting and I am also an "affiliate" so if you use this link to buy your hosting, I make a small percentage.[75] It does not increase your cost.

If you encounter this kind of problem, please let me know. I'd love to help however I can.

EUROPEAN HOSTS

I asked my users to tell me what European hosts they use so that I can offer some suggestions. I offer a few of those responses here, but note that I have no personal experience with them.

hostpapa.co.uk from Phil L. (Phil@Justice4Kate.com)

I can recommend hostpapa.co.uk. I've used them for a couple of years and found them reliable, inexpensive and when there has been the odd problem they have sorted them quick with a really good free phone helpline although I understand they are based in Canada. Your script works great with them.

www.blackfoot.co.uk from Neil Lemon (NeilLemon@googlemail.com]

I am a UK user of your wonderful script and I too originally encountered a problem with Fasthosts all of about 18 months ago. Since then I have been using http://www.blackfoot.co.uk and have never had an issue with your script working perfectly. They do not offer telephone support and their hosting is pretty basic (all shared servers) but when I have had an issue my emails have been responded to and more importantly sorted within minutes. They are pretty cheap and the servers are based here in the UK which is good for me as the companies I make websites for are also based over here. If your user is looking for a cheap solution and doesn't need their hand held with telephone support I can't think of a better company.

and again from Leah Taylor (info@lanternwebdesign.co.uk)

I use Blackfoot (www.blackfoot.co.uk) who are very good and have excellent support. I have also implemented your form with no problems on their servers.

---

[74] I last confirmed this pricing on 05.May.2009.

[75] I am also a user of NameCheap.com for Domain Name Registration and use them exclusively for all my domain name registrations. I strongly recommend them: www.NameCheap.com and here are some coupons for discounts: www.NameCheapCoupon.com

www.hostpoint.ch from W. Louis (WLouis@CalTechNix.com.mx)

> I use www.hostpoint.ch. My experience is excellent service.

www.1and1.co.uk from David Crowe (David.CROWE@coe.int)

> I am very satisfied with my hosting company, 1 and 1.
>
> For UK based users, they are at www.1and1.co.uk. They trade also as 1and1.fr , 1und1.de (and BlueHost.com internationally). Physical storage is in Karlsruhe, Germany.

www.heartinternet.co.uk from Rob Knight (Rob@Net-Results.co.uk)

> I use Heart Internet: www.heartinternet.co.uk. We know your script works there, 'cos we've got it running. Just needs a PHP5 set in the htaccess file.

FRONTPAGE PROBLEMS

OVERVIEW

I was originally motivated to write this script to serve the needs of my friends and colleagues in the FrontPage community. As a result, I found out more about FrontPage issues than I did about say, Dreamweaver issues. It isn't a personal snub. It is just the community I am in.

The replacement of FrontPage with Expression Web has rendered most of this chapter obsolete. But I leave it for anyone still using FrontPage

If you have your own issues, let me know. I'll work on them all.

FRONTPAGE CHANGING THE CODE

There is a known bug in FP[76].

If you have two forms on one page, FP will work to make the field names unique, particularly the field names of hidden fields (which are all the parameters to the script).

It does this by adding a numeral to the end of the field name which effectively renders them wrong.

This bug presents itself even if you unselect "make ID unique on paste" in Tools / Page Options / General

But there is a workaround.

Wrap the second form in code that tells FP to leave it alone:

```
<!--webbot bot="HTMLMarkup" startspan -->
Page Form
<!--webbot bot="HTMLMarkup" endspan -->
```

You can also make that wrap smaller to limit its effects

For example:

```
<!--webbot bot="HTMLMarkup" startspan -->
Just put the hidden fields here
<!--webbot bot="HTMLMarkup" endspan -->
```

[76] The "death" of FrontPage and its replacement with Expression Web has drastically reduced the need for this information. But I leave it because there are still a few FrontPage users.

This should not effect you unless you have two forms on the same page and are using FrontPage.

DEPRECATED AND OBSOLETE ITEMS

WHAT ARE DEPRECATED AND OBSOLETE ITEMS?

DEPRECATED ITEMS

A deprecated item is one that was outdated by newer constructs. Deprecated items are defined in this document but are clearly marked as deprecated. Deprecated items still work, but may become **_obsolete_** at any time, in future versions of this software. The script still supports deprecated items for backward compatibility. However, users should begin to use the substitutes.

I anticipate that the current Deprecated Items listed below, will be completely eliminated in the next major release, and may occur "very soon".

OBSOLETE ITEMS

Obsolete items are similar to deprecated items, but no long work. These items must be changed. The script checks for these items and guides the "fix". For example, see C023 (page 173)

THE LIST OF DEPRECATED ITEMS

1. *CopyUser*   Replaced by *FormEchoUser*
2. *FormEmailField*   Replaced by *FormEmailFieldList*
3. *FormNameField*   Replaced by *FormNameFieldList*
4. *NextURL*   Replaced by *FormNextURL*

THE LIST OF OBSOLETE ITEMS

1. *FormFieldNameValueSubstitutionList*
   Removed as a legal parameter. The functionality was transferred to *Msg1FieldNameValueSubstitutionList*, *Msg2FieldNameValueSubstitutionList* and *MsgEchiFieldNameValueSubstitutionList*.

   In addition, the ability to use an asterisk ("*") for Back Reference was added to *Msg2FieldNameValueSubstitutionList* and *MsgEchiFieldNameValueSubstitutionList*.

TRACKING FIXES AND ENHANCEMENTS

SECOND NOTICE

I've already mentioned all this. But I do it again because it is important.

CHANGES, FIXES AND ENHANCEMENTS

From time to time I make changes to the script and to the documentation of the script. Some of these changes are for problem fixes. Version 1.2, for example, was a response to user notice of a problem working on some servers with some configurations of PHP. Other changes may be enhancements. And still other changes may be just documentation changes, correcting typos and improving the information.

It is critical for you and other Users of Huggins' Email Form Script to know promptly about these changes. For this reason, I created the *HEFS Bulletin* to keep users informed.

HEFS BULLETIN

The *HEFS Bulletin* is a "self-hosted" ezine, powered by Gammadyne Mailer.[77]

I publish it irregularly, as needed, to alert subscribers to changes, fixes and enhancements to Huggins Email Form Script, and related documentation and support files.

SUBSCRIBING TO HEFS BULLETIN

To subscribe to *HEFS Bulletin* go to:

www.JamesSHuggins.com/hefs-bulletin

YOU NEED TO SUBSCRIBE, RIGHT NOW

If you haven't done it already, do it right now, before you do anything else. Subscribe to this publication. It is critical that you subscribe in order to ensure that you receive critical updates. Please. Do this now. Right now. Not 10 minutest from now. Right now. Please.

Go here and subscribe :

www.JamesSHuggins.com/hefs-bulletin

---

[77] For more information see Gammadyne Mailer on page 223.

COPYRIGHT FOR HUGGINS' EMAIL FORM SCRIPT

Huggins' Email Form Script is copyright 2005-2013 by James S. Huggins & The Eclectic Power Company

www.JamesSHuggins.com
EclecticPower.com

ACKNOWLEDGING YOUR USE OF THE SCRIPT

In return for the free use of Huggins' Email Form Script, there are two teeny tiny requirements:

1. Please include a visible linked acknowledgement on the Form Page. This acknowledgement should be substantially like:

```
<a title="Huggins' Email Form Script" target="_blank"
href="http://JSH.us/hefs">This form uses Huggins' Email Form Script</a>
```

NB: the Acknowledgement Link file ("HEFS Acknowledgement Link V2.2.5.txt") includes this HTML in an easy to use, copy/paste format to make this easier.[78]

The fonts, sizes, colors and other similar characteristics should be set to compliment the site. You may modify the text or add text at your discretion provided the meaning is substantially the same.

Feel free to adjust the specifics to compliment your page. The only objective is to help someone else find and use this script.

Special Notice 1: the target="_blank" attribute has been deprecated in standard HTML.I continue to use it because (a) it is still broadly supported in the browsers and (b) I don't approve of the deprecation; BUT if you are a standards stickler, just take that out

---

[78] The naming scheme of V2.2.5 references the version number of the script. The version number is "A.B.C" where "A" is the primary release number, "B" is the secondary release number and "C" is the tertiary release number. Typically, minor bug fixes update the tertiary release number. Small enhancements and changes update the secondary release number. And major changes update the primary release number. All the files in a download package use a similar numbering scheme to show that they are all part of the package. In general, you should always download and use a complete package of files with similar names. This ensures that the files all work together.

Special Notice 2: I've been asked whether the phrase "This form uses Huggins' Email Form Script" could be shortened to "Huggins' Email Form Script". Sure. I'm not a stickler for that level of detail. The goal is to help others find the script so this is clearly another acceptable option.

Placing this linked acknowledgement on the Form Page will help to promote Huggins' Email Form Script. I sincerely appreciate this consideration.

2. Include the following acknowledgement in the HTML code of the Form Page itself:

```
<head>
<!-- Form Credit: This email form uses Huggins' Email Form Script;
   To learn more visit "http://JSH.us/hefs".-->


note: remove this line; the <head> is just to show where to put this so be
careful not to duplicate that
```

NB: the Acknowledgement Notice file ("HEFS Acknowledgement Notice V2.2.5.txt") includes this HTML in an easy to use, copy/paste format to make this easier.

LICENSE

Huggins' Email Form Script is made available through the Creative Commons GNU GPL License (2.0) (CreativeCommons.org/licenses/GPL/2.0). This is a free license and permits use on commercial websites. It restricts incorporation into some commercial products.

This documentation is made available through the Creative Commons Attribution-No Derivative Works 3.0 Unported License (3.0) (CreativeCommons.org/licenses/by-nd/3.0). This is a free license but restricts development of derivative works.

To view copies of these licenses, visit

CreativeCommons.org/licenses/GPL/2.0

CreativeCommons.org/licenses/by-nd/3.0

or, send a letter to

Creative Commons
444 Castro Street, Suite 900
Mountain View, California 94041-2061

LICENSE SUMMARY

THE LICENSE FOR THE SCRIPT

The Creative Commons GNU GPL License (CreativeCommons.org/licenses/GPL/2.0) is a Free Software license (www.gnu.org/philosophy/free-sw.html). Like any Free Software license, it grants to you the four following freedoms:

- The freedom to run the program for any purpose.

- The freedom to study how the program works and adapt it to your needs.

- The freedom to redistribute copies so you can help your neighbor.

- The freedom to improve the program and release your improvements to the public, so that the whole community benefits.

The principal restrictions and conditions are:

- You must conspicuously and appropriately publish on each copy distributed an appropriate copyright notice and disclaimer of warranty and keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of the GNU General Public License along with the Program. Any translation of the GNU General Public License must be accompanied by the GNU General Public License.

- If you modify your copy or copies of the program or any portion of it, or develop a program based upon it, you may distribute the resulting work provided you do so under the GNU General Public License. Any translation of the GNU General Public License must be accompanied by the GNU General Public License.

- If you copy or distribute the program, you must accompany it with the complete corresponding machine-readable source code or with a written offer, valid for at least three years, to furnish the complete corresponding machine-readable source code.

- In addition; you must specify two specific notices, one visible on the page in which you use the script and one within the HTML on the page in which you use the script. See Acknowledging Your Use of the Script on page 200.

THE LICENSE FOR THE DOCUMENTATION

The Creative Commons Attribution-No Derivative Works 3.0 Unported License (3.0) (CreativeCommons.org/licenses/by-nd/3.0) grants to you the right to:

- **Share**
  to copy, distribute and transmit the work

You are free to do these things under the following conditions:

- **Attribution**
  You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
- **No Derivative Works**
  You may not alter, transform, or build upon this work.

For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to

CreativeCommons.org/licenses/by-nd/3.0.

Any of the above conditions can be waived if you get permission from the copyright holder.

Nothing in this license impairs or restricts the author's moral rights.

This section lists some of the testimonials I have received from users of Huggins' Email Form Script. You can also see this information at www.JamesSHuggins.com/h/hefs/huggins-email-form-script-testimonials.htm

---

I have once before written a testimonial thanking James for all the hard work he has done to help many other other people and also in sharing his time and knowledge to enable many people to use his excellent work for free.

Since I last wrote I have had to contact James once again due to me wanting to modify my contact form. It is because of this I feel compelled to write once again thanking James and 'singing his praises'! From my initial contact James once again replied within the hour and he has patiently stuck with me for around 4 days while I rectified my problem.

When I cannot do something I want to know the reason why and rather than James just rectifying the problem he has walked me through each step while giving me guidance, which meant that I needed to study the code and ultimately learn what it all meant. If I got stuck and could not go any further he would then explain what I needed to do.

I cannot thank him enough for his patience, knowledge and help. If you genuinely want a form that works and does what it says then you definitely have to use the Huggins Script. His manual may not be to everyone's taste but it is nowhere near as daunting as it first looks. To get a basic form working is very very easy and with James's support no one could ask for anything more.

Once again, thank you James.

Charles Archer, Director
www.justice4everyone.co.uk
Apr.2012

The Huggins' Email Form Script is an amazingly easy solution for very flexible and customizable PHP contact forms. As soon as my team and I took a look at it, we were sold – except IT'S FREE!! We are not PHP developers and don't intend to be PHP experts any time soon. This extremely well written and flexible script has kept us out of the business of having to do that. What's more, we were very happily surprised to find out that, not only does James fully support his product, but he is dedicated to helping folks make sure that it works for them. We received replies for information within minutes...fantastic customer service!!

Lane Gifford
BoldLinesServCorp.com
May.2012

JHuggins' Email Script works like a charm! It appears to be 'bulky' until you understand that only a few tweaks make it work for you. The Script is easy to implement - just read the instructions. But like all instructions, it's best if the designer of them were right there guiding you, and here is where the support is fantastic. If you need help, you can easily get friendly patient assistance by phone and/or email personally from James.

Because of this I recommend trying his brilliantly written free script; brilliant because of the checks and cross checks it automatically does so that when you get email, it's legitimate and cross checked for malicious code.

Thanks James for all your help.

Bill Sinclair
www.christiankeys.ca
Dec.2010

Hi James,

I want to thank you for your Email Form Script and sharing your expertise with the world. I'm a complete novice where building websites is concerned and am still in the throes of coming to grips with Microsoft expression. It was on the expression forums that I discovered the existence of the Huggins Email Form Script when I needed to set up a secure email form for a website that I'm developing. That's when the fun started.

I must admit I found your documentation a little daunting at first, until I stopped 'skimming' and started to follow the structure and logic of what you were trying to present in your manual. It's comprehensive, it's clear and unambiguous, and it lead me carefully through the building process. It all fell into place nicely and I surprised myself at how quickly I was able to have a working email form set up, tested and running.

Now here's the biggest surprise of all. When I struck a problem that I wasn't able to resolve, on the verge of abandoning the Huggins solution I decided to contact you on the off chance you might respond. I was delighted and grateful to receive a level of support from you that was prompt, effective, and personal. Indeed, as a senior manager in the high end computer industry for many years in Australia, I have rarely come across such enthusiasm and commitment to getting a client's problem resolved in the quickest possible time...and you don't even know me!

Thank you again, James. The Huggins Email Form Script is a great 'product' that's professionally supported and easy to implement. Needless to say I'll be using it again in future projects, and of course I will ensure I don't miss an opportunity to recommend it to anyone with whom the subject of secure web forms is raised.

Vic Kenney, Asia Pacific Technical Support Manager (Retired)
Safety Beach, Melbourne, Australia
Nov.2010

Here's what I told my Dreamweaver Class, and anyone that will listen. I also use the form for our Curling Club's Contact page. And club members love it!

You'll find really good tools to make forms in Dreamweaver. You'll find excellent extensions too. You can even find terrific wizards or form generators on-line, like Accessify's quick form builder.

But, none of them tell you how to implement the form without incurring a lot of spam, how to make the form secure, and make it actually work to deliver to you usable and understandable information. Most importantly it's the connection you need between you and your clients.

Except for Huggins' Free e-mail Form Script!

At first, the contents in the Huggins' Form Mail folder look's intimidating and hard to install, but it isn't at all. Just read the "manual" and follow the instructions. If you have a problem and you contact James, he will respond and help you find the answer you need! Fantastic! Search no further!

Here's two examples of a simple contact form:
http://uofgts.com/dream/contact.html
http://www.evergreencurling.com/about/contact.php

Thanks James!

Garry Stasiuk
uofgts.com
Nov 2010

Hello James,

Like some others, I found your script through Google. To be honest, I was kinda wary of it because of previous failures with properly setting up an email form. After some trial and failure, finally got the hang of it and it worked like a charm ever since. What impressed me so much out of the learning process is your documentation. The amount of information it packed has helped me enormously to understand how the script and its setting work. I most definitely recommend it to others.

Leo Lin
G2 Orbital Complex - 1:6 Custom Action Figure Portal
www.g2orbital.net
Jun 2009

James,

I want to thank you for your continuing work on the form script.

Like many others, I was left somewhat adrift when FrontPage Extensions began disappearing from servers. Having switched to Expression Web, I knew that I could continue on a limited basis on selected servers, but needed to locate a more permanent solution. Of interest, perhaps, is that I found your script listed along with several others on Microsoft's Expression discussion forum.

Naturally, I investigated all of the listed scripts along with several others that I located independently. And none of them come close to the Huggins' Email Form Script:

1. The customizations available to developers are absolutely superior.

2. The documentation with the script is not only complete and comprehensive, but also well written.

3. The personal support that you provide when questions do arise is prompt, specific, and very helpful.

In short, this is an absolutely superior package. I am proud to be a client and happy to provide the reference to others.

Thank you again for your hard work.

Dr. Lynn Eubank
Toulouse School of Graduate Studies
University of North Texas

NB: this testimonial comes from Bob Walto who discovered a problem and communicated it to me.

Thank you so much for your very prompt attention to this and for resolving it so quickly. I was pretty darn sure that it was an error on my part, but it was a very pleasant experience to review this error with you and determine the error was in the script. You have done an excellent job of delivering a very useful script, and your commitment to delivering quality is very much evident in the service you provide.

Bob Walto
www.walto.com
Nov 2008

Hi James,

I now have all three of my web site forms converted to your email script. The other two were easy for me to do after getting the first one up and running. I am a happy camper.

Attached is a "formal" thank you. I sincerely thank you. Your script is exactly what I needed and what thousands of other FrontPage users are looking for no doubt. The fact that I do not write code and could actually get your script up and running on my web site says a lot about your script. It is truly wonderful. Your documentation and online example and form tutorial made it all understandable.

Here are the links for my three forms:

http://www.dixonpubliclibrary.org/illrequest.html
http://www.dixonpubliclibrary.org/illarticlerequest.html and you already saw: http://www.dixonpubliclibrary.org/askalibrarian.html

Our library usually receives requests from 2 of the 3 forms every day. I am the Interlibrary Loan librarian and I am happy to say that I rarely get requests via the article request form since this is not the type of stuff our patrons are usually looking for - but it's there for people that might need it and is used occasionally. But the other two forms get lots of use.

Thanks again.

Betty Beatty
Oct 2008

I appreciate everything you have done to help me to get a form ready.

Vietta Helmle
www.BeagleTech.com
Aug 2008

When I received word from our host that they were phasing out support of FrontPage Extensions I switched to Expression Web. I tried EW's Form Wizard but as rumored I too found it to be lacking, so I began looking for a third party form script.

We have 15 FrontPage-created webs with various types of mail forms, tutorial feedback, surveys, registration forms, and questionnaires.

I tried the Tectite email form but discovered that they "guide" one to a paying subscription by not answering questions on their forum once they have you hooked. Then I tried James' script and what a difference the over all experience has been.

What I appreciate most, beside the fact that it caters to script/coding semi-illiterates like myself, (it's virtually copy-paste) is that I have the experience that James is committed to service. I marvel at his immediate implementations of edit suggestions, it makes me feel like I'm part of his project.

Kerrith H. (Kerry) King
Leadership-Relationship Communication Skills Coach
President, Community Communications
www.comcom121.org
July 2008

By the way, your documentation is outstanding.

Clo Knibbe
www.NortheastFencing.net
May.2008

I was looking for a simple mail script, found your massive script, got impressed and now as a result I use your script for a very tiny little email form. And I find myself reading the huge manual (which i normally avoid at all cost) just for fun.   ;) Good work!

Henri
16th-floor.nl
Apr.2008

James,

Thanks for creating the Email Form Script v 2.0.1.

I was told about it by Ernie Secrest, who hosts my sites.

It is fantastic, and the [manual] is well-written and very helpful.

I just completed implementing your script on a site, and will definitely be using it on others (except for the one on GoDaddy, which I found disallows the php mail function).

I have a link back to your site so others will find out about it, too.

Joyce Cimbalista
Web Site Design
www.SunnyDaySites.com
Mar.2008

Dear James,

I am writing to say how much I appreciate your e-mail form script.

When I decided to set up a site for my husband, I had no idea what HTML code was, thought that CSS and SEO might be acronyms for certain illnesses and fell in to despair when I realized that I would need to put an e-mail form on the contact page. In short, I was a complete and total novice.

I do not know what I would have done without your script. It is simple to use, but when my lack of experience led me to make stupid mistakes you very patiently helped me to understand what I needed to do to install the form.

I have been very happy with the form and very, very grateful for your support. You have truly helped us to set up a site that works. My husband is an artist and, in a few short months, the form has been the bearer of commissions, two exhibitions in London and Portugal and virtually no spam at all.

I would have no hesitation in recommending your script to anyone, no matter what their level of experience or expertise. If I can install and use it, anyone can.

Thank you again for your kindness in providing the script and for your thorough and patient support.

Lorraine Davison
www.pauldavison.fr
Feb.2008

Mr. Huggins...

Things are going very well with [the script] you provided...

So well that the Scoutmaster will not accept any other communications for a Conference that does not come through this form.

Thanks again...

Thomas J Casey
Boy Scout Troop 152
www.troop152.org
Jan.2008

I found you on Google. Thanks a bunch! I have been trying to get someone help me set up a form for 2 weeks. With your [script and manual] I did it in one evening. Thanks,

Philip Matthews
Nov.2007

Mr. Huggins,

I have been looking for a program or script that would handle the forms I put on my web sites, and I have tried quite a few with some limited success, but mostly problems causing wasted time, energy, and money. Your "Huggins' Email Form Script" is simply **the** best script I have found. My only problem is that I did not find it in 1994 or 95 when I first started doing my site.

Of course Microsoft's "Front Page" with their "extensions" would handle **some** of what your script does, but I have tried to move away from that monopoly and do the "coding" myself. The only reason I used "Front Page" was the fact that they could handle a rather simple form. Now that I have found your script, I am completely free to do as I want, and your program puts very little limitations on what I can only dream about incorporating in any of my forms.

Thank you so much for this script. It really is the answer I have been looking for 12 years or so.

Jim P. Bryan
Allendale, SC USA
before Nov 2007

For several reasons, I resolved to move away from requiring the FrontPage Server Extensions on my site. One of the things that had been locking me to the FPSE, however, was the inability to find a reasonably secure and easy-to-use replacement for the email form handler. HEFS proved to be the solution to my needs, and, while I still use FrontPage to manage my site, I have managed to eliminate any need for the FPSE, and can now use reputable hosting services that I had to disregard previously because of the lack of FPSE support. Thanks for this!

Jeff Zeitlin
Editor, *Freelance Traveller*
The Electronic Fan-Supported Traveller® Resource
www.FreelanceTraveller.com
before Nov 2007

Dear James,

I am using your email form script on every one of my sites and a site I manage for a commercial furniture stores. I use the script for both a contact form on all sites and as a family group sheet on my genealogy site. I even use it on one of my sites that does not support php by referring to the script on another of my sites. The script works just as advertised and you have provided great documentation and tech support. It also works well to use it as an include to insert where needed and I can tell exactly which page the form came from.

I appreciate the fact that you continue to work to improve the form and quickly fix any bugs that are found. I do not hesitate to recommend your email form script to anyone who needs to provide forms on their site.

Thanks you for all you do for us.

Pat Geary
www.genealogy-web-creations.com
before Nov 2007

James,

I just wanted to let you know how much I appreciate you and your program. I had been posting in several forums my issue with getting a Web Form mail program to work. You stepped forward and offered yours for ***free*** but better than that you were willing to hand-hold me thru the whole process of getting it installed, set up and working. You took hours to help me, talking with my hosting tech support to try to resolve problems and going the extra mile to help me. You have been there after "normal" business hours to work with me to get my clients a reliable way of processing the Web Forms they need for their websites.

I can not thank you enough…words are not enough. I will sing your praises and point people to you at every opportunity.

My Sincere Thanks,

Debi Carpadus
Founder/ Senior Designer
Bright Hope Designs
4001 Bright Hope Mine Rd, Kelsey, CA 95667
Office: 530-295-3351 Cell: 530-409-6577 Fax: 530-622-4191
www.BrightHopeDesigns.com
before Nov 2007

Read the documentation yesterday and find it very easy to implement. Besides hiding the email address, I especially like the ability to send the email to more than one address. This could allow me to send to an alternate Gmail account where I can automatically archive all the messages.

Thanks!

Troth
www.OriginalGeekster.com
before Nov 2007

IMHO it is a great script and I know it is going to resolve some major issues with our online adoption application.

L. Harral
www.MalinoisRescue.org
before Nov 2007

I never expected a free e-mail program to come with personalized help from the programmer, but that's exactly what I got. Our dog rescue group works with all-volunteer help and James came through for us when we changed servers and needed a new e-mail program to power our adoption application. He even took the time to ask what modifications to the code and documentation would be useful to us. He may be a Windows guru, but he lives with the spirit of Open Source. Thank you so much, James.

Catherine de la Cruz,
Webmistress
Great Pyrenees Rescue of Northern California, Inc
www.gprnc.org
before Nov 2007

The Huggins' Email Form Script has been a lifesaver for me! I use it in a number of areas on my site to send email notifications and collect information. It is so easy to use - the documentation is impeccably written and easy to follow. Kudos, James!

Alicia Staz
www.BeadedRoyalty.com
before Nov 2007

I have been looking for a form in PHP to move from FrontPage Forms, something I've put off for far too long. When I heard about James' form I was really happy. Not only does it work as advertised, knowing James, I knew that any bug, would be instantly looked at, that he would strive to improve it, and not only that, make it do things normally one would have to pay thousands for. I've recommended James' form to many others and use it on all new sites, and will change over each of my sites as I get time. I cannot recommend his form highly enough. I can only hope that James continues with this enterprise for a long time to come. I intend to encourage him all the way.

Tina Clarke
Microsoft MVP – FrontPage
any-expression.com
before Nov 2007

I started to write my own PHP email submittal form but found yours. After finding out about your excellent reputation, I decided to use it instead. It is more than fully featured and well supported. Thanks, James.

Doug DePrenger
Co-owner of Smartlab Software, LLC.
Home: www.SmartlabSoftware.com
Blog: News.SmartlabSoftware.com
before Nov 2007

SPECIAL ACKNOWLEDGEMENTS

I want to acknowledge those who have contributed to this project, some of whom helped just by their internet publication without ever knowing they would assist, and others who intentionally gave their time and effort to make this the best it can be:

This list is also replicated within the program code.

- **Tom von Alten**
  www.FortBoise.org

  For extensive feedback on and editing of the script documentation and also helping with the code.

- **Paul Beach *****
  www.b-tec.com

  For his kind comments reviewing my documentation and helping me to improve it for everyone.

- **Andreas Belivanakis**
  www.MilosIsForLovers.com

  For critical ideas about improving the script.

- **Mike Cherim**
  Green-Beast.com

  For ideas on protecting email forms from spambots. See especially green-beast.com/blog/?p=128 and green-beast.com/blog/?p=220.

- **Dave Child**
  ww.addedbytes.com/php/email-address-validation

- For an algorithm I adapted to perform validation on an email address. But see also http://www.addedbytes.com/blog/email-address-validation-v2 and http://code.google.com/p/php-email-address-validation for more recent info not yet used.

- **R. David Christman**
  www.ChristmanFamily.net

  For encouragement and very patient assistance with Version 2.0.1.

- **Tina Clarke**
  www.AccessFP.com

  For her many observations and insightful suggestions which have helped to improve the script.

- **Jin Cowan**
  www.jincowan.co.nz

  Helped me improve some of my notifications and even suggested the "inside the page code" notification.

- **Vanessa Cox *****
  www.virtual-business-services.com
  For her kind testing of and feedback critique of my new documentation

- **Lorraine Davison**
  www.PaulDavison.fr
  For helping me to identify and debug an issue related to passing arguments by reference.

- **Doug DePrenger**
  www.SmartlabSoftware.com

  Doug provided invaluable assistance helping me to understand a key difference between PHP 4 and PHP 5, to correct a problem that was affecting some users on some servers and who continually provided feedback, ideas and assistance.

- **S. Emerson**
  www.AccreteWebSolutions.ca

  For ideas about general script improvements and general support and assistance.

- **Stephen Fredette**
  www.ProWebsites.net

  For assistance with XHTML.

- **Pat Geary**
  www.Genealogy-Web-Creations.com

  For incredibly valuable and very patient assistance with XHTML and also V 2.1.0.

- **James George**
  http://gsquaredstudios.com

  For writing the excellent article at http://www.sitepoint.com/style-web-forms-css and for responding to my follow up question in absolutely record time.

- **Lyn Harral**
  www.All-About-The-Details.com

  For the dedication to substantial and significant testing and proofreading, and for all the great enhancement ideas she gave me and helped me to implement.

- **Clo Knibbe**
  www.NortheastFencing.net

  For great assistance exploring a FrontPage problem in which FrontPage "corrupts" the form field names and for exposure of yet one more host with crippled PHP.

- **Svetlana Sally Johnston**
  www.Britfeld.com

  For working with me to implement features of V 2.1.0 to let me test them on a "live" site and helping me to see the differences between what I was trying to do and what the users wanted me to do.

- **Al Lawrence**
  www.SunriseDoors.com

  For his unwavering patience as he found a major series of beta bugs, and his supportive assistance in suggesting new improvements.

- **Jonathan Lewis**
  www.LewisPhillips.com

  For helping me improve the simple one page simple forms by adding CSS and editing my XHTML. His additional templates are now a part of the package.

- **Michelle Lyman**
  www.FromArtToZebras.com

  For her critical assistance testing a major pre-release finding the bugs I should have found in the code I should have written better.
- **Joe Marini**
  www.sitepoint.com/article/users-email-address-php

  For an algorithm I adapted to perform validation on an email address.
- **Melvin D. Nava**
  www.Planet-Source-Code.com/vb/scripts/ShowCode.asp?txtCodeId=1316&lngWId=8

  For a tip on using "function_exists" to resolve "checkdnsrr" for Windows Servers.
- **Anne Nevil**

  For her non-stop dedication to testing the script and for forcing me to see things a different way.
- **Ken Rose**

  For alerting me (on 2010.11.13) to the need to turn off E_DEPRECATED for PHP 5.3.0
- **Lance W.**
  www.cla.asn.au

  For assistance with timestamps.
- **Tammy Wolf**
  www.WolfsDesigns.com

  For her intense tests helping me to resolve Windows Servers incompatibilities.
- **Jeff Zeitlin**
  www.FreelanceTraveller.com

  For ideas about and testing of error page formatting.
- **Thomas Zuber**
  www.ZuberPhotographics.com

  For spurring me on to include better support for XHTML Strict in my examples and templates, and for continued support in that effort.
- **All the members of Cricket's classes and forums**
  www.GNC-Web-Creations.com/seo-optimization.htm

  These members enthusiastically used and tested my script, putting up with bugs and failures while I got it right. They are the best test team out there.
- **J. Bailey ("Cricket")**
  www.GNC-Web-Creations.com

  What can I say. She has tirelessly promoted the script. But more than that, she acknowledged me, appreciated me and encouraged me throughout the process, and especially when I needed it most.
- **Mother and Father**

  For the genes that make me say "Someone should fix this!" and the lifetime of acknowledgement, appreciation and love that lets me know I can.

*** Acknowledgements noted with the three asterisks have been recently added to this documentation but not yet added into the program code.

CRICKET'S CLASSES AND DISCUSSION LISTS I CO-MODERATE

I am a co-moderator on these lists and forums.

Cricket's SEO Training Class
JSH.us/cricket-seo or
www.gnc-web-creations.com/seo-optimization.htm

Cricket's Successful Website Marketing Training Class
JSH.us/cricket-web-marketing or
www.gnc-web-creations.com/successful-website-marketing.htm
Focuses on the various marketing issues for online businesses.

Cricket's Small Business Website Development Training Center
JSH.us/cricket-web-training or
www.gnc-web-creations.com/web-training.htm
Includes a variety of different classes, discussion groups and special focus discussion groups. Each one is joined individually as desired.

- **Website Development Training Classroom** (a class)
  This structured online class includes basics of W3C Valid HTML and CSS, along with basic design issues.

- **Tech Training Classroom** (a class)
  Training currently includes RSS feeds, installing WordPress, PC maintenance & security, and developing and using forms.

- **FrontPage Help For SEO** (a class)
  Not intended to replace Microsoft FrontPage groups that already FrontPage, it was specifically created to teach members of the SEO Training class how to implement recommended search engine optimization techniques using FrontPage.

- **Expression Web** (a discussion group)
  Expression Web users working together as a team to learn the most effective ways of using this new Microsoft program to its full potential.

- **Net Business Success** (a discussion group)
  Here you can discuss ideas and strategies to develop a successful online business. These discussions are restricted to a few specially selected small business related topics ***not*** covered in the training classes.

- **Peer Website Reviews** (a discussion group)
  Peer site reviews are conducted on an announcement basis only. We work together to improve the marketing, appearance, and credibility of our websites.

- **Everything Web Design** (a discussion group)
  Working together as a team to help everyone learn to troubleshoot and find solutions for web design issues.

- **Tech Talk Hideaway** (a discussion group)
  Discussing anything involving RAM, hard drives, external hard drives, motherboards, keyboards, processors, sound cards, video cards, broadband, Ethernet, modems, USB ports, power supplies, network adapters, heatsinks & fans, DVD-ROM & DVD-RW drives, routers, networking, hardware installation issues, software installation issues, broadband issues and similar topics.

- **Digital Photography and Photo Editing** (a discussion group)
  Members discuss photography tips, graphics, and image editing for the web. We welcome all skill levels from beginner to advanced.

- **It's All Debatable** (a special focus discussion group)
  Conducted properly, debates about controversial topics can lead to knowledge and important changes in our communities. This group provides an avenue to discuss those topics not open for discussion in many forums because they are considered controversial.

- **The SEO Coffee Shop** (a special focus discussion group)
  This group is a designated area for personal chatting and getting to know each other. Discussing "work" is off limits. The SEO Coffee Shop may add large amounts of meaningless chatter and uncontrollable laughter to your life.

- **The Diet and Fitness Center** (a special focus discussion group)
  Developing a successful online business often means spending a lot of time sitting behind a computer. This can lead to physical fitness and weight issues that we were not expecting. The Diet and Fitness Center allows members to work together as a team to improve their health, nutrition, and physical fitness.

OTHER TECHNICAL LISTS AND FORUMS I CO-MODERATE

I am a moderator or co-moderator (mostly co-) on these lists and forums.

Tiffany Edmonds' FrontPage Discussion List
JSH.us/fplist or tech.groups.yahoo.com/group/fplist
Tiffany is. like me, a Microsoft MVP "alum".

Tiffany Edmonds' Front Page Off-Topic Discussion List
JSH.us/fpofftopic or tech.groups.yahoo.com/group/fpofftopic
Tiffany a Microsoft MVP "alum".

Patricia Geary's Genealogy/FrontPage Discussion List
JSH.us/fpgenealogy or
freepages.genealogy.rootsweb.com/~gearyfamily/frontpage
This list generally addresses genealogy and family history users of FrontPage.

Cheryl Wise's Expression Web Forum
JSH.us/byexpressionforum or Forum.By-Expression.com
Cheryl is a Microsoft MVP.

OTHER TECHNICAL LISTS, FORUMS AND WEBSITES

By-Expression.com
Cheryl Wise's site about the entire Microsoft Expression suite.
Cheryl is a Microsoft MVP.

AccessFP.net
Tina Clarke's extensive website of FrontPage tutorials, tips and resources.
Tina is a Microsoft MVP.

AddOnFP.com
Tina Clarke's site featuring FrontPage add ons.
Tina is a Microsoft MVP.

Any-Expression.com
Tina Clarke's Expression Web site.
Tina is a Microsoft MVP.

AnyFrontPage.com
Offers Tina Clarke's *Any FrontPage Bytes* ezine and a free ebook for subscribing.
Tina is a Microsoft MVP.

www.Expression-Web.net
Tina Clarke's Expression Web blog.
Tina is a Microsoft MVP.

FrontPage-Ebooks.com
Tina Clarke's FrontPage ebooks. She offers a free ebook on DWT (Dynamic Web Templates).
Tina is a Microsoft MVP.

FrontPage-Tips.com
Offers Tina Clarke's *FrontPage Tips* ezine.
Tina is a Microsoft MVP.

Blogs.MSDN.com/devi
Devindra S. Chainani, Group Program Manager, Expression Web.

www.J-Moves.com
Mike Chafin's J-Move add ins for Expression Web.
Mike is, like me, a Microsoft MVP "alum".

www.JimcoSoftware.com
Jim Cheshire's free add-ins for FrontPage.

MSMVPs.com/blogs/frontpage
Tina Clarke's FrontPage news blog on the Microsoft MVPs site.
Tina is a Microsoft MVP.

StartToWeb.com
Cheryl Wise's online courses including FrontPage and Expression Web
Cheryl is a Microsoft MVP.

www.WebsUnlimited.com
Mike Chafin's J-Bot add ins for FrontPage.
Mike is, like me, a Microsoft MVP "alum".

Gammadyne Mailer is a tool for broadcasting emails. Because it is programmable, it can also maintain mailing lists.

I use Gammadyne Mailer to manage double-opt-in lists (lists which require that the subscriber confirm his/her intention to subscribe).

For example, I use Gammadyne Mailer to manage my *HEFS Bulletin* (JSH.us/hefs-bulletin)

I began using the Gammadyne Mailer in late 2002. It is possible to use it very simply, almost out of the box. But it also features a very complete internal programming language to let you do just about anything you want. And, it keeps all the information on my PC in an Access database, where I know the information is secure and under my control.

It is screaming software. I strongly recommend Gammadyne Mailer.

You can read about Gammadyne Mailer on my site here:

www.JamesSHuggins.com/gammadyne

You can see the Gammadyne Mailer site by following my affiliate link here:

www.JamesSHuggins.com/gammadyne2[79]

You can test drive *my* implementation of the Gammadyne Mailer just to see how it works. I created an ezine named *TestZine* to let me test my programming. You can test it too. Subscribe and receive a "sample issue". You can even try out some of the tests I perform, like submitting a subscription request twice in a row, or subscribing when I'm already subscribed or unsubscribing when I'm not subscribed, just to see how it handles these "special conditions". Just go to my *TestZine* page at:

www.JamesSHuggins.com/testzine

---

[79] This is my affiliate link to Gammadyne. If you follow this link and buy Gammadyne, they will send me money. If you object to such affiliate links, you can go directly to: www.gammadyne.com. To learn more about my affiliate link policy, see Affiliate Program Policy on page 224, or go to www.JamesSHuggins.com/affiliate-program-policy.

My site is not, *primarily*, a commercial website. The minimal additional income to offset some of the expenses is nice, but it is not the primary reason for this site. For that reason, I have created my Affiliate Program Policy which I posted on my Disclaimers, Copyrights and Legal page.[80] It says:

Whenever I provide a link to an affiliate program that pays me, I attempt to also provide an alternate link which omits the referral information. These alternate links will not result in payment to me. If you object to affiliate programs, you can use these alternate links to buy and I won't get any money.

The reasons I do this are simple. First, my site is not primarily a commercial site. It is not about making money. Affiliate payments are nice, but if they all went away, I'd still have and would still keep my site. Second, I link to some affiliate sites because they are good, not because I'd get paid. Getting a small dividend is an extra; it is not the motivation. Third, I link to other affiliate sites (those with which I do not have personal buying experience), because my research indicates that they might be of interest to my visitors and that they offer a quality product or service.

For all of these reasons I fully disclose my affiliate relationship and offer a choice. I let the buyer choose to support my site because they like my site, instead of forcing them to support my site because my relationship with the vendor is secret. My personal site is not about advertising and making money. Never has been. Never will be.

Each of these Affiliate Pages, also includes a link to the vendor that will not result in a payment to me.

If you want to visit the vendor but object to my participation in an affiliate program, use that link instead.

In addition, if you experience anything negative, in any way, about any affiliate I have, I want to know about it. Please email me and tell me all the details. I am proud of the affiliates I have and like to monitor them to ensure that they remain appropriate for listing on my website. Because making money is not the primary reason for their listing, I will not hesitate to remove any affiliate program that does not meet my personal standards. I appreciate your help in monitoring their performance.

---

[80] You can see this disclosure on my site at: www.JamesSHuggins.com/affiliate-program-policy

SPECIAL TERMS USED IN THIS MANUAL (EBOOK)

This document uses a few special terms. It also uses common terms in specific ways. This Glossary explains these terms as they are used in this manual (ebook).

BACK REFERENCE

The use of an asterisk ("*") for a Hidden Field Parameter value to indicate that the script should use the value of a prior/similar field. For example, a MSG2 Hidden Field Parameter may use an asterisk to indicate the script should used the value of the corresponding MSG1 Hidden Field Parameter. See *Back Reference: Duplicating a "Prior Value"* on page 133.

CONFIRMATION MESSAGE

An optional message sent by the script back to the Form User. Also called an Echo Message

DEPRECATED

A deprecated item is one that was outdated by newer constructs. Deprecated items are defined in this document but are clearly marked as deprecated. Deprecated items still work, but may become ***obsolete*** at any time in future versions of this software. The script still supports deprecated items for backward compatibility. However, users should begin to use the substitutes. For a list, see the section Deprecated and Obsolete Items on page 198.

ECHO MESSAGE

An optional message sent by the script back to the Form User. Also called a Confirmation Message.

EMAIL FORM

A Web Form designed to capture information from a Form User and send that information to someone via email. The Email Form may also send the Form User a Confirmation Message.

FIELD NAME

The name of a field as specified in a Web Form.

FIELD REPORTING LABEL

The text used to label a particular Field Value in an email being sent.

FIELD VALUE

The data typed into a field in a form.

FORM DESIGNER

The individual responsible for creating the form and implementing the script.

FORM FIELD

A field, other than a Hidden Parameter Field, in which a Form User can input values to the form for processing by the script. This documentation differentiates between Hidden Field Parameter and Form Field. The former is a field used to pass a parameter. The latter is a field in which the Form User types a value.

FORM PAGE

A web page containing a Web Form.

FORM USER

The individual on the web who types information into the Web Form and clicks the "submit button".

HIDDEN FIELD PARAMETER

A field in a Web Form for which the Form Designer specifies a Field Value as a parameter to the script. The field cannot be seen by a Form User and the Form User cannot change the Field Value of a Hidden Field Parameter. This documentation differentiates between Hidden Field Parameter and Form Field. The former is a field used to pass a parameter. The latter is a field in which the Form User types a value.

INJECTION EXPLOIT

Injection Exploits are attempts by spambots to place values like "cc:" and "bcc:" in the fields to hijack the form/script and use it for spam. Testing for injection exploits helps prevent spambots from using your form/script for spam.

MESSAGE 1

The primary email sent by the script.

MESSAGE 2

An optional second message sent by the script.

OBSOLETE

An obsolete item is one which existed in a prior version, but is no longer supported. If you use an obsolete item with the newer versions of Huggins' Email Form Script, results may be unpredictable. For a list, see the section Deprecated and Obsolete Items on page 198.

VISIBLE FIELD

A field in a Web Form into which a Form User can type information. The "opposite" of a Hidden Field

WEB FORM

A standard web interface that can be coded onto a web page and used to send data. A Web Form contains text boxes for a Form User to enter data. Form Users can then submit the Web Form for processing by a Web Form Processing Script.

WEB FORM PROCESSING SCRIPT

A program residing on a server that accepts data from a Web Form and processes that data. The script may send the data via emails (including a possible Confirmation Email to the Form User), may search for information and display results, may update a database, or may trigger other processes.

THE ECLECTIC POWER COMPANY

JAMES S. HUGGINS

James S. Huggins is an eclecticist: a business coach, author, writer, technologist, trainer and professional speaker. Through James' consulting firm, The Eclectic Power Company, he provides a variety of professional services, primarily to sole proprietorships, professional practitioners and small businesses.

James' experience combines more than 30 years in information technology with demonstrated excellence in training, professional speaking, sales, sales support, business development, writing, television, project management, individual coaching, team facilitation and consulting.

James' areas of specialization include:

- Internet Marketing,
- Search Engine Optimization,
- Book and Manual Development,
- White Paper Development,
- Ezine Management,
- Audio Product Development and
- Business Development Coaching

James recently served as the primary writer for Juanell Teague (www.JuanellTeague.com) and as the "third man" in her one-on-one consulting work with her clients. James also co-authored white papers and articles with Juanell, provided overall information technology strategy and implementation and acted as coordinating producer for her conferences and seminars.

James was a Microsoft MVP (2003-2009) for their FrontPage website development and management software. He is a moderator of two FrontPage lists, an Expression Web forum, and SEO Techniques discussion lists. He has presented sessions on Internet Marketing for professional speakers.

In the most recent years, James lived with and assisted his mother who died in July 2011. Since his mother's passing, James has slowly dealt with his disabilities (from two strokes) and has slowly been returning to the technical, writing and consulting world.

THE ECLECTIC POWER COMPANY

The Eclectic Power Company is built on two simple concepts: first, that many people do not have just one talent, but several, and second, that there is power in the synthesis of different, complimentary talents.

Drawing on the multifaceted expertise of the founder, Eclectic Power supplements its strengths with associates and partners, both long term and project based, to fulfill the needs of the client.

CONTACT INFORMATION

James S. Huggins
The Eclectic Power Company
11150 Beamer Road, Suite 109-F3P
Houston, Texas 77089-2331

832.328.8515
fax: 270.596.0140
www.EclecticPower.com
JHuggins@EclecticPower.com