

```
<?php
/*
formhandler.php
Script by Bernard Pegis (bp@amdgssoftware.com) 3/16/02
```

```
version 1.0
- initial version
```

```
version 1.1
- fixes a bug in the comma delimited file date stamp
- put in click here... link at end before displaying thanks
```

This script is called by an HTML form using the POST command with this file as the action.

Example: `<FORM METHOD="POST" ACTION="formhandler.php">`

FormHandler is a configurable PHP script which is used for receiving data from an HTML form and either displaying, emailing or saving the results. When saving the results, a comma delimited file is created on the server with the last two fields being the date and time, respectively, at which the data was entered. After saving or emailing the results, they can be also displayed on the screen to the user. Displayed and emailed results can be shown in either the comma delimited format or in a more user-friendly format which shows the field name and the data in tabular columns. Descriptions of the options and examples are below. Each option is selected by adding a hidden field to the form with the name of the option. The field's value will be the setting for the option.

```
-----
Option:      todo
Values:      writefile, email, display
Use:  writefile: create a comma delimited file on the server. The filename
        is established in the filename option.
        email: sends the results of the form to the email address in the
        mailto option with the subject in the mailsubject option in the
        format selected by the displayformat option. The originating IP
        address and script, HTML referring page, and the user's browser
        type are also emailed.
        display: neither writes a file nor emails the results. Only shows
        the data from the form on the screen in the format selected by
        the displayformat option
```

Example: to write data to a file, add the following field to your form:
`<input type=hidden name="todo" value="writefile">`

```
-----
Option:      filename
Values:      any valid file name
```

Use: data is written to the file in a comma delimited format. If the filename ends in .php then the data will be more or less obscured from unauthorized viewing. This is accomplished by commenting out the data and causing the PHP parser to ignore the data when the file is requested for viewing. To get the data, you must use the FTP GET command, rather than using HTTP. If the filename ends in anything else, anybody will be able to view the data by simply entering the filename as an HTTP request in a browser. Relative and absolute paths are accepted and used. If no path is included, the file is written in the same directory as the script and form.

Example: `<input type=hidden name="filename" value="xy.txt">`

Option: afterpage

Values: "" or any valid html or php document.

Use: if blank, the script will prompt the user with a generic "thank you" message, otherwise the document indicated will be loaded after the form has been submitted and processed.

Example: `<input type=hidden name="afterpage" value="thanks.html">`

Option: mailto

Values: any valid email address

Use: This is required if todo is set to "email". This is the address to which the form's results will be emailed. The format selected by the displayformat option will be used.

Example: `<input type=hidden name="mailto" value="bp@amdgssoftware.com">`

Option: mailsubject

Values: any string value

Use: This is the subject which will be used in the email message which will be sent if todo is set to "email".

Example: `<input type=hidden name="mailsubject" value="survey results">`

Option: displayformat

Values: comma, tabular

Use: this is the format used when the data is either displayed or emailed. comma will have the data in a comma delimited format and tabular will show the data along with the field names.

Example: `<input type=hidden name="displayformat" value="comma">`

Option: showinfo

Values: true, false

Use: This option is ignored if todo is set to "display". This causes the form's data to be shown to the user after form submission when todo is set to either "email" or "writefile"

Example: <input type=hidden name="showinfo" value="false">

```
-----  
*/  
  
$od = "";  
$x = "";  
$goafter = "";  
$mailto = "";  
$mailsubject = "";  
$displayformat = "";  
$resultsd = "You submitted the following information:<br><center><table border>";  
$resultsm = "";  
$todayd = date("m-d-Y");  
$todayt = date("H:i:s");  
reset($HTTP_POST_VARS);  
while (list ($key, $val) = each ($HTTP_POST_VARS))  
{  
  if ($key == "todo"):  
    $od = $val;  
  elseif ($key == "filename"):  
    $filename = $val;  
  elseif ($key == "mailto"):  
    $mailto = $val;  
  elseif ($key == "mailsubject"):  
    $mailsubject = $val;  
  elseif ($key == "displayformat"):  
    $displayformat = $val;  
  elseif ($key == "afterpage"):  
    $goafter = $val;  
  elseif ($key == "showinfo"):  
    $showinfo = $val;  
  else:  
    $x .= "\" . $val . "\"";  
    $resultsm .= "$key = $val\n";  
    if ($val == ""):  
      $val = "&nbsp;";  
    endif;  
    $resultsd .= "<tr><td>$key</td><td>$val</td></tr>";  
  endif;  
}  
  
$x = substr($x, 0, strlen($x) - 1);  
$resultsd .= "</table>";  
$resultsm .= "\n\n-----\n\n";  
$resultsm .= "Sent by: " . $HTTP_SERVER_VARS['PHP_SELF'] . "\n";
```

```

$results = "Possible prior page: " . $_SERVER['HTTP_REFERER'] .
"\n";
$results = "Browser: " . $_SERVER['HTTP_USER_AGENT'] . "\n";
$results = "From IP: " . $_SERVER['REMOTE_ADDR'] . "\n";

if ($od == "writefile"):
    if (substr($filename, strlen($filename) - 3, 3) == "php"):
        $obfuscate = true;
    else:
        $obfuscate = false;
    endif;

    if (file_exists($filename)):
        $fp = fopen ($filename, "a+"); //this opens the file
    else:
        $fp = fopen ($filename, "w"); //this creates the file
        if ($obfuscate):
            /*
            This file header serves to hide the data, regardless of the
            unix file permissions. While it is bad form to have the initiating
            php tag (i.e. <?php) and not the terminating tag in the file, it is
            too much effort to ensure that the ?> stays at the end, and it works
            perfectly well without it. The way it works is to force the php
            parser to ignore the data by telling it that it is a series of comments
            */
            fwrite($fp, "<center><b>Error. You May Not View This File</b></center><?php
            /*\n");
            fwrite($fp, "-----*\n");
        endif;
    endif;
    $x = ",\'' . $todayd . "\",\'' . $todayt . \"\"";
    if ($obfuscate)
        {$x = "/*" . $x . "*/";}
    fwrite($fp, $x . "\n");
    fclose($fp);
// chmod($filename, 0606);
    if ($showinfo == "true"):
        echo $resultsd;
    endif;
elseif ($od == "email"):
    if ($displayformat == "comma"):
        mail($mailto, $mailsubject, $x);
    else:
        mail($mailto, $mailsubject, $results);
    endif;
    if ($showinfo == "true"):

```

```
    echo $resultsd;
endif;
elseif ($od == "display"):
    echo $resultsd;
endif;

if ($goafter == ""):
    echo "<center><font size = +3>Thank you for submitting your form.</font></center>";
else:
    if (($od == "display") or ($showinfo == true)):
        echo "<center><hr><a href=\"\" . $goafter . \"\">Click here when
finished</a></center>";
    else:
        echo "<html><head><META HTTP-EQUIV=\"Refresh\" CONTENT=\"0;URL=" .
$goafter . "\"></head><body></body></html>";
    endif;
endif;
?>
```