

NateMail 3.0

PHP form processing script by Nate Baldwin, www.mindpalette.com

NateMail 3.0 is a PHP form processing script designed to make working with web forms simple. Version 3.0 brings many great features such as...

More Form Validation Options
More Text Formatting Options
Better International Text Encoding

More Configuration Options
More Redirect Page Options
Easier Confirmation Page Styling
...and More!

Contents

Getting Started	2	Recipient Identification	7
Server Requirements	2	Sorting Field Order	7
Hidden From Fields	2	Redirect Pages	8
Basic Set-Up Instructions	3	Fine-Tuning the Script	9
Test Basic Installation	4	Time / Date Format	9
Optional Features	5	Formatting Field Names	10
Email Subject	5	Server Time Offset	10
Required Fields	5	Email Line Spacing	10
Email Formatted Fields	5	Confirmation Page Styling	11
Number Fields	5	Support Options	11
Letter Fields	5		
Lowercase Text	6		
Uppercase Text	6		
Required Matching Fields	6		
Reply To and From Address	6		
Link Back to Site	6		
Excluding Fields	7		

Getting Started

Your Server

Before spending much time setting this script up, I'd recommend making sure that your server not only supports PHP, but has PHP configured properly to be able to send out emails. In the "Extras" folder included in this download, you will find a "Test PHP Email" utility that you can use to make sure your servers have what they need to use this script.

If your server fails the test, it may just be that your web host requires a different extension for PHP files (such as .phtml or .php3), or they may just need to activate PHP for your account. If things do not seem to be working correctly with the server test, contact your web host and find out if there are any special requirements for using PHP on the server, or for letting PHP send out email messages.

Hidden Form Fields

These instructions are for setting up NateMail 3.0 with a standard HTML or text editor. If you are using GoLive or Dreamweaver, check the download for a PDF manual made specifically for your environment.

Many of the features of NateMail use "hidden" form fields. This will allow your form to pass the instructions to the script without having the visitor see these fields on your form. If you are not familiar with HTML hidden form fields, the illustrations below will show you the correct syntax.

Hidden Form Field Code:

```
<input type="hidden" name="hiddenName" value="hiddenValue">
```

The next page will tell you just the basics in setting up the script, and should have you up and running in no time. The following pages will explain all the optional features to fine-tune the script for your site.

Basic Set-up Instructions

One of the problems with many other form processing scripts is that they are vulnerable to spam attacks that hi-jack your script and use it to send out spam email, making it look like YOU sent them. The way we get around that problem is to have a list of recipient email addresses inside the script that NateMail is allowed to send messages to.

This list is hidden from email address harvesting scripts, and even if hi-jacked, NateMail could only send messages to addresses in this pre-defined list, so it would make it pretty worthless to a spammer. That being said, our first task will be to build the list of "recipient" addresses that NateMail is allowed to send mail to.

Open the NateMail.php file into your simple text or HTML editor. You'll find the recipients list starting on line 13. Inside the double quote marks in each line, replace the existing text with an email address. If you don't need that many addresses, just leave the rest as they are. If you need more, add as many extra lines as you need by copying and pasting, but make sure each \$recipients line has a unique key number inside the [brackets].

```
12 // LIST OF EMAIL RECIPIENTS...
13 $recipients[0] = "address@mindpalette.com";
14 $recipients[1] = "address2@mindpalette.com";
15 $recipients[2] = "address@processform.com";
16 $recipients[3] = "enter your email address here";
17 $recipients[4] = "enter your email address here";
18 $recipients[5] = "enter your email address here";
19 $recipients[6] = "enter your email address here";
20 $recipients[7] = "enter your email address here";
21 $recipients[8] = "enter your email address here";
22 $recipients[9] = "enter your email address here";
```

Now you can save and close the NateMail.php file, and we're ready to set up your HTML to work with your script. This tutorial assumes you already have a form page created, or at least know how to build a HTML form.

To use the script, the form will have to know where to submit the form contents to. Set the form action to be the path on your server to get to the NateMail.php file. Also, make sure the form method is set to post. When you're finished, if your script is uploaded to the same directory as your HTML form, the form tag should look something like...

```
<form action="NateMail.php" method="post">
```

Now we just have to tell the form which email address to send the results to. To do this, add a hidden field to your form and set the Name to "recipient", and set the value to be the index number of the address you want the form to send to. The index number is what was inside the [brackets] in the \$recipients code we edited earlier in the script file. If you want one form to send to more than one email address, just add as many index numbers as you need, separated by commas.

```
<input type="hidden" name="recipient" value="0,1,4">
```

Test Basic Installation

At this point, the script should work with all its default settings. I'd recommend giving it a test run before spending much more time fine-tuning the optional settings. You will probably need to upload the form and the .php script file to your server in order to test, since PHP scripts are a "server-side" technology and need to be run on a web server.

Fill in your test form and submit it. If the script handles the form correctly (you should see the default "Thank You" confirmation screen) and you receive the resulting email at the address you entered for the recipient, then we're ready to continue with the optional settings. If not, we have a bit of trouble-shooting to do. For help, see the FAQ.pdf file for support information.

Fine-Tune your Settings

There are 2 ways to modify the script and use the built-in features. For all the primary features, you will use hidden form fields to tell the script what to do. There are quite a few hidden form field options, and these settings can be different for every form that uses the same script, so one NateMail.php script may be used with several different forms.

Aside from hidden form fields, other settings can be changed inside the script itself. These settings will affect all forms that are submitted to the script. If you decide to change any of these settings inside the script, remember that you are changing programming code, and the "syntax" is very important, so forgetting a quotation mark or semi-colon could easily cause the entire script to stop working. I'd highly recommend making a back-up of the NateMail.php file before making any internal changes.

Most of the rest of this tutorial will cover how to use both of these methods to get the script to work best for your needs and to match the look of your site.

Basic Hidden Field Settings

subject

```
<input type="hidden" name="subject" value="Web Form Submission">
```

If you'd like to change the subject of the email that the script sends to your recipient email, then create a new hidden field named "subject" and set the value to whatever you'd like the new email subject to be.

required

```
<input type="hidden" name="required" value="name, email, phone">
```

To make any of your form fields required (the visitor can't submit the form until the required fields have a value entered), then add a hidden field named "required" and for the value, list the names of the fields that must be filled in before submitting. To require more than one field, add as many field names as you need into the value, separated by commas.

email_only

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

To require than any of your form fields be a valid email format, add a new hidden field named "email_only" and set the value to the list of form fields that must be in email format, separated by commas. This function will only determine if the address entered is formatted as an acceptable email address, and will NOT determine if the email address actually exists.

numbers_only

```
<input type="hidden" name="numbers_only" value="item_number">
```

To require that a form field's value has only numbers entered (for quantities, item numbers, etc.), make a new hidden form field named "numbers_only" and set the value to be the list of form fields whose values can only be numbers, separated by commas. Decimal points or periods are also allowed, and numbers submitted with a comma, such as 2,500, will be considered valid, but the commas will be stripped automatically by the script.

letters_only

```
<input type="hidden" name="letters_only" value="item_name">
```

To require that a form fields value only be allowed to contain normal upper or lower case letters (a-z, A-Z), then add a hidden field named "numbers_only" and set the value to be the list of fields that can only be letters, separated by commas.

lowercase

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

To have the script convert any field values to all lower case (such as an email field), add a new hidden field named "lowercase" and set the value to be the list of field names to convert to lower case, separated by commas.

uppercase

```
<input type="hidden" name="uppercase" value="item_number, name">
```

This one works just like "lowercase" except the field is named "uppercase" and converts the field values to all caps.

force_match

```
<input type="hidden" name="force_match" value="password1, password2; email1, email2, email3">
```

To require that 2 fields have the same value entered before allowing the form to be processed (such as a Password and Confirm Password field), add a new hidden field named "force_match" and set the value to be the fields that must have the same value, separated by commas. If you have more than one set of fields that need to match (like 2 email fields that must match, and 2 password fields that must match), separate the groups of fields by semicolons.

sender_email

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

To have the "reply to" and "from" address in the email be an email address entered by the person filling out the form, add a new field named "sender_email", and for the value, enter the name of the other form field that will have the email address of the person completing the form (usually a text field).

sender_name

```
<input type="hidden" name="sender_name" value="your_name">
```

Similar to the "sender_email" function, this option will have the sender name show up as the "from" and "reply to" in the email message. This function can only be used with the "sender_email" function, and will cause the "from" address to look something like... Firstname Lastname <emailaddress> ...in your email application. Set the name of the field to "sender_name" and the value to the name of the other form field that has the name of the sender.

link_url

```
<input type="hidden" name="link_url" value="http://www.mindpalette.com">
```

After the form has been processed, this option will let you add a link to the form results page to get back to your home page (or any other page) after the form has been submitted. Name the field "link_url" and set the value to be the URL of the page your link returns to. If your form was in a popup window, you can enter "close" for the value to have the script generate a JavaScript "close window" link. Enter "back" for the value if you want the script to generate a JavaScript link to go to the previous page.

link_text

```
<input type="hidden" name="link_text" value="MindPalette Home">
```

This option will determine the text for the link with the "link_url" option above. If not used, the script will default to "back to home". If the value of link_url is "close", this will default to "close window", and if the link_url value is "back", it will default to "back to form"

Basic Hidden Field Settings, continued

exclude

```
<input type="hidden" name="exclude" value="submitButtonName, submitx">
```

This option will allow you to exclude certain form fields from the resulting email and the confirmation or redirect page (such as excluding your submit button data and any unnecessary information that the form is trying to pass). Add a new hidden field named "exclude" and set the value to be the list of form field names to exclude, separated by commas.

exclude_display

```
<input type="hidden" name="exclude_display" value="password, confirm_password">
```

Similar to the "exclude" field, this option will exclude any field values from the confirmation or redirect page, but will still send the information in the email message. Add a new hidden field named "exclude_display" and for the value, enter the list of form field names to exclude from the confirmation or redirect page, separated by commas.

exclude_email

```
<input type="hidden" name="exclude_email" value="password, confirm_password">
```

Similar to the "exclude" field, this option will exclude any field values from email message, but will still send the information to the confirmation or redirect page. Add a new hidden field named "exclude_email" and for the value, enter the list of form field names to exclude from the email message, separated by commas.

recipient_name

```
<input type="hidden" name="recipient_name" value="MindPalette">
```

In the default confirmation page, there is a message telling the visitor what email address the form results were sent to. If you'd like to not show the email address to avoid getting unwanted messages, and would rather have the script say the name of the person or company that the email was sent to instead of the email address, then add a new form field named "recipient_name" and for the value, enter the name you'd like to have it use.

sort

```
<input type="hidden" name="sort" value="name, phone, email, comments">
```

This option will let you control the sort order of the form results in the resulting email and the confirmation or redirect page. Add a new hidden field named "sort" and set the value to be the names of your form orders in the order you'd like them to appear. Only the fields found in the sort value will be sent in the email and shown in the confirmation or redirect page.

Custom Redirect Pages

If you'd like to have the script redirect to one of your other pages after the form has been successfully processed instead of the default Confirmation / Thank You page, you'll want to use one of the built-in redirect options.

redirect

```
<input type="hidden" name="redirect" value="formresults.html">
```

the script to use your redirect page. Name the field "redirect" and for the value, enter the URL address of the page you'd like to redirect to. For best result, use a relative URL instead of an absolute URL. It's also best to put the redirect page into the same folder as the NateMail.php script page. If that's the case, you can just enter the name of the page, as shown in the example above.

If you don't need the form results to display in your redirect page, then we're done. Usually, though, you'll want to tell the visitor what information was sent with the form as a confirmation. The challenge is getting the form information passed to your own page so it can be displayed. There are 2 methods included in the script.

redirect_type

```
<input type="hidden" name="redirect_type" value="include">
```

To tell the script which type of redirect to use, add another hidden field named "redirect_type", and the value will either be set to "include" or "query". If no value is entered, or if the "redirect_type" field is not used, the script will default to the "include" method. Both methods are explained in detail below.

"include" redirects

The preferred redirect method is "include" (the default). This method actually imports your redirect page into the script page, so it's important that you put your redirect page in the same folder as the script. It's also important to make sure your redirect page has a .php extension (or whatever other extension your server requires to recognize a file as a PHP page). Build your page however you'd like it to look, then switch to source view in GoLive, and go to the area you'd like the form results to print to. Sometimes it's easier to make a page break
 tag in layout mode, then select it and switch to source mode, if you have trouble finding your place in source. Then, insert this PHP code...

```
<?php @redirectHTML("fieldNameCSS", "fieldValueCSS"); ?>
```

Make sure you type it in exactly as shown. Then, if you'd like to use CSS classes to control the way the form field names and values look, make your CSS classes, and replace the "fieldNameCSS" and "fieldValueCSS" with the CSS class name you'd like it to use. Make sure to use the quotation marks as shown. Alternately, you could just name your CSS classes .fieldNameCSS and .fieldValueCSS and not have to change the PHP code.

Custom Redirect Pages, continued

“query” redirects

If, for some reason, the “include” method will not work for you (for instance, if the redirect page can not be in the same folder as NateMail.php), then you’ll want to use the “query” method. This redirect type uses a URL query string and JavaScript to print the form results into the redirect page.

This used to be the only method available, but I wasn’t happy with the limitations. Some of the possible problems are, if the visitor has JavaScript disabled (it doesn’t happen often, but it does happen), then they won’t be able to see the form results. Also, the form results are printed into the browser address bar, which might seem a bit sloppy. Another disadvantage is that some browsers have a limit on how much data can be passed in a URL query string, so if there are any very long text fields, some of the form data may be cut off in the confirmation page.

To use the “query” method, you’ll need to make sure you use a “redirect_type” field with the value set to “query”. Then, you’ll need to include a simple JavaScript in your redirect page where you’d like the form results to appear. The specific instructions for installing and using the JavaScript are included with the script in the “Extras” folder that was downloaded with NateMail and this PDF file...

Script Configuration Options

Aside from the features we’ve already covered, there are additional configuration options that you can edit as needed inside the script file. These changes will affect all forms that are submitted to the script file and not just individual forms.

\$dateFormat

Around line 28, you’ll see an option for \$dateFormat. The available values are either 1, 2 or 3 (not inside quotation marks). The default options is 1. These numbers correspond with a time/date pattern that will be used by the script.

```
28 | $dateFormat = 1;    // see options listed below (1, 2 or 3)...
```

The patterns that each number uses are as follows:

- 1: January 4, 2004 @ 6:00 pm
- 2: 2004-01-04 @ 18:00
- 3: 4 January 2004, 18:00

When the form has been processed, the current data and time will be written in one of the above patterns into the email message and the default confirmation page.

Script Configuration Options, continued

Lines 37 - 42 (depending on how many recipients you've added) are settings for an assortment of other options. These are explained in detail below.

```
$replaceUnderscores = true; // enter true or false - changes underscores to spaces
$initialCaps = true; // enter true or false - capitalizes the first letter of each word
$serverTimeOffset = 0; // number of hours to add or subtract from server time
$doubleSpaceEmail = true; // true to double space between fields in email message
unset($referrerCheck);
$referrerCheck = true; // make sure form is on same server as script (recommended)
```

\$replaceUnderscores

When working with HTML forms, it's often a good idea to make sure the names of all your form fields contain only letters, numbers and underscores (no spaces or special characters). This is particularly important if you're also using JavaScript validation. While good coding, it doesn't necessarily look good in the email and confirmation page. If

`$replaceUnderscores` is set to true (no quotes), then the script will replace the underscores with spaces in your field names so they are easier to read. For instance, in the email and confirmation page, the field name "first_name" would be changed to "first name". To turn this option off, simply change it to false.

\$initialCaps

This option also exists to make your email and confirmation page easier to read. If set to true (no quotes), it will capitalize the first letter of each word in your form field names. For instance, if used with `$replaceUnderscores`, "first_name" would become "First Name". To turn this option off, change the value to false.

\$serverTimeOffset

This value exists in case you and your server are not in the same time zone. This number (no quotation marks) will adjust the time/date that's printed in the email and confirmation page by however many hours you enter. Negative numbers are also allowed. If your server says 5:00pm and you need 7:00pm, you'd enter 2 for the value. If your server says 5:00pm and you need it to say 2:00pm, you'd enter -3 for the value.

\$doubleSpaceEmail

This option determines whether the results in your email message are single spaced or double spaced. By default it's set for double spacing, so change this value to false (no quotes) if you'd like it single spaced.

Confirmation Page Styling

The built-in confirmation page is highly customizable. The text that appears on this page and the error messages the visitor would see if there was a problem with what they entered is all customizable. These settings are, by default, in lines 78-132 of the script. As with any other script changes, I'd recommend making a back-up of your NateMail.php file just in case. Most values are between quotation marks, so if you use any quotation marks in any of your messages, make sure they're escaped with a backslash, otherwise you'll get a script error. For instance...

```
$fakeVariable = "The word \"hamster\" is surrounded by escaped quotation marks.";
```

If you'd like to change the background color of the page, or any of the font styles of the text, the CSS settings for the confirmation page are in lines 44-76. You do not need to "escape" quotation marks in these lines, but will need to escape any apostrophes or single quotes with a backslash. Otherwise, if you're familiar with CSS, these changes should be easy to make. Even if you're not, most of the language is self-explanatory, so you might just make a back-up and give it a shot. If you aren't comfortable with editing these settings, you might want to look into using a "redirect" page (see page 8).

```
// Styling for Thank You / Confirmation page (in CSS)...  
  
// Page background color and styling for <body> <div> <td> <p>...  
$pageStyle = 'background-color: white';  
  
// Main page text...  
$MPinfo = 'color: black; font-size: 13px; font-family: Verdana, Arial, Helvetica, sans-serif';  
  
// Form field names and values...  
$MPfieldNames = 'color: black; font-weight: bold; font-size: 12px; font-family: Verdana, Arial, Helvetica, sans-serif';  
$MPfieldValues = 'color: #00327d; font-style: normal; font-size: 12px; font-family: Verdana, Arial, Helvetica, sans-serif';  
  
// Headline (Thank You) text...  
$MPheadline = 'color: black; font-weight: bold; font-size: 24px; font-family: Verdana, Arial, Helvetica, sans-serif';
```

Congratulations!

At this point, you're finished with all my ramblings and should pretty much have an idea of how things work. If you run into any problems, there are plenty of ways to get support. Don't forget about the "Test PHP Email" utility in the Extras folder that came with this download. Also, I'd encourage you to check over the Troubleshooting.pdf file that was also included with your download.

If your question has still not been answered, feel free to visit our support forum and post your questions online. I personally monitor that forum along with some other very knowledgeable people, and you're likely to get an answer sooner than sending a direct email...

www.mindpalette.com/forum

THE END :-)