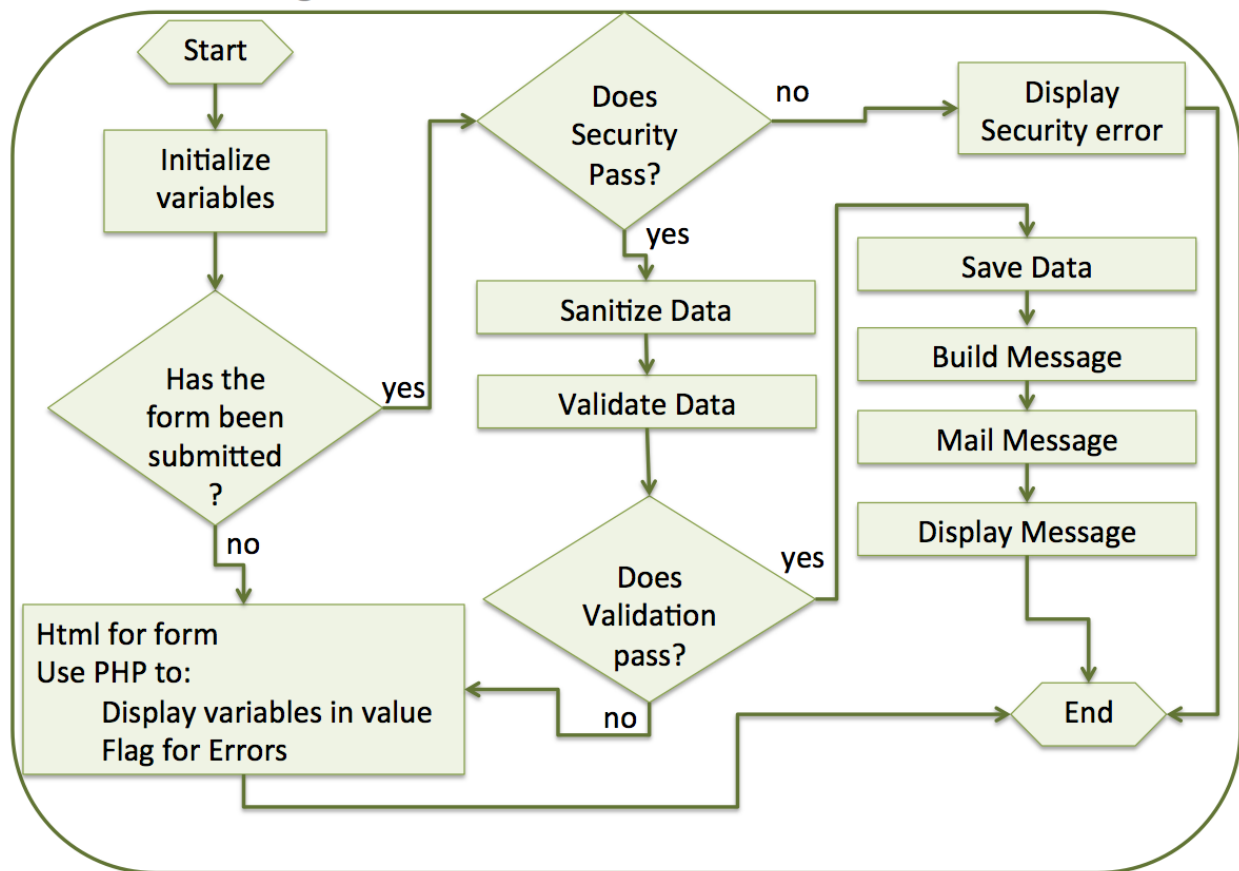


Form Processing Flow Chart



Once our data has been saved lets prepare a message to mail and send to our user. The code can be found in GitHub commit 9: PHP Provide feedback (mail and display form data)

To start I am going to declare a variable called `$mailed` in Section 1. This variable is used as a flag to see if php was able to send the mail. This does not mean the email address was real it just means php sent the mail. If the mail is undeliverable you as the owner of the web site will get an message like this:

Mail Delivery Subsystem

To: robert erickson

Returned mail: see transcript for details

The original message was received at Tue, 20 Mar 16:00:39 -0400
from silk22.uvm.edu []

----- The following addresses had permanent fatal errors -----

[<bob@uvm.edu>](mailto:bob@uvm.edu)

(reason: 550 5.0.0 [<bob@uvm.edu>](mailto:bob@uvm.edu)... User unknown)

Lets start building a message. The message I am creating here is as basic as you can get. There are many other resources you can look at. For starters I suggest glancing at these (reading them as in-depth as you want to):

<https://css-tricks.com/tools-html-email-workflow>

<https://github.com/mailchimp/email-blueprints> look in the read me for a link to HTML Email Reference [<https://templates.mailchimp.com>].

I am going to start with:

```
$message = '<h2>Your information.</h2>';
```

You will notice that I am using html elements to define the content of my message. Most email clients today will interpret the message correctly. You cannot link a style sheet but you can use inline css to style everything. Images can be included with the image element but must have an absolute url. For example:

```

```

Just putting the name of the image will not work.

What I have found is that is useful for your clients is to include in the message what they typed into the form. Similar to when you order online after you check out your order is displayed on the computer screen as well as a copy emailed to you. All the information the user typed in will be in the POST array. So we can loop through the post array and add each piece of information to the message. In this case

```

foreach ( $_POST as $htmlName => $value ) {
    $message .= '<p>';
    $message .= $htmlName;
    $message .= ' = ';
    $message .= htmlentities($value, ENT_QUOTES, "UTF-8");
    $message . '</p>';
}

```

This would produce the email message:

Your information.

txtEmail = Bob@uvm.edu

btnSubmit = Register

I use a three letter prefix on my form objects and this can be removed by using the PHP built in function substr (sub string [<http://php.net/manual/en/function.substr.php>])

```
substr($htmlName, 3)
```

Where we start at the third character skipping characters 0, 1, 2

This works fine for txtEmail (which would be Email) but I also use camel case when I get to two or more word names like txtFirstName and FirstName is not so great. However a regular expression comes in handy to split the word when a capital letter is encountered.

```
preg_split('/(?=[A-Z])/ ', substr($htmlName, 3))
```

What this does is create an array ('First', 'Name') so we can then put the words back together again in a for each loop adding a blank space. That is what this code does where I combine the dropping the first three letters with substr and using the regular expression to put the camel case into an array:

```

$camelCase = preg_split('/(?=[A-Z])/ ', substr($htmlName, 3));
foreach ($camelCase as $oneWord) {
    $message .= $oneWord . ' ';
}

```

So I am adding 'First' plus a blank space to the message. Then I add 'Name' plus a blank space to the message. The end result is we are taking the form element name txtFirstName and converting it First Name so that our email message will look like this:

Your information.

First Name = Bob

Email = bob@uvm.edu

Submit = Register

This is getting more readable by our users. Let me put all the code together so you can see it all at once:

```
//@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
//
// SECTION: 2f Create message
//
// build a message to display on the screen in section 3a and to
mail
// to the person filling out the form (section 2g).
$message = '<h2>Your information.</h2>';
foreach ($_POST as $htmlName => $value) {
    $message .= '<p>';

    // breaks up the form names into words.
    $camelCase = preg_split('/(?=[A-Z])/ ', substr($htmlName, 3));
    foreach ($camelCase as $oneWord) {
        $message .= $oneWord . ' ';
    }

    $message .= ' = '
    $message .= htmlentities($value, ENT_QUOTES, "UTF-8")
    $message .= '</p>';
}
```

This is not bad and is a start. Our end result is the message we will display on the screen and mail to the user. So now that we have the message we need to learn about mailing in php. It is actually just one line of code to have php send an email message [See: <http://php.net/manual/en/function.mail.php>]:

```
mail($to, $subject, $mailMessage, $headers);
```

I have created a function for you to use that includes several security checks and allow you to cc and bcc the mail message. I suggest you DO NOT CC the message to anyone but BCC a copy to yourself so you know when someone fills out the form :) You should read through the function in mail-message.php to get an idea of what needs to be done and you will include mail-message.php in top.php.

```
$mailed = sendMail($to, $cc, $bcc, $from, $subject, $message);
```

The six parameters my functions uses are all self explanatory and before you call the function would should set each value even if we set the value to nothing. For example:

```
$to = $email;
$cc = '';
$bcc = '';
$from = 'WRONG site <customer.service@yoursite.com>';
$subject = 'Changing Earth: ';
```

The from variable should always be a valid email address and for now can be yours. The subject of course should make sense for the site you are building. The code in form.php is pretty simple:

```
//@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
//
// SECTION: 2g Mail to user
//
// Process for mailing a message which contains the forms data
// the message was built in section 2f.
$to = $email; // the person who filled out the form
$cc = '';
$bcc = '';
$from = 'WRONG site <customer.service@yoursite.com>';
// subject of mail should make sense to your form
$subject = 'Changing Earth: ';

$mailed = sendMail($to, $cc, $bcc, $from, $subject, $message);
```

It is very important to note that you will need to include the mail-message.php file in top.php like this:

```
include 'lib/mail-message.php';
```

Of course you need to have a lib folder with the file mail-message.php which can be found in the GitHub repository. Be sure to read all the comments in the file as there are constraints that will cause the message to not be sent.

Summary

Mailing a copy to the user is standard practice today

- POST array
- substr
- preg_split
- function I created: sendMail(\$to, \$cc, \$bcc, \$from, \$subject, \$message)

Self Test Questions

1. If a user types youremail@uvm.edu will the mail be sent? T or F
2. If the variable \$cc is empty will the mail be sent? T or F
3. If the variable \$to is empty will the mail be sent? T or F
4. If the message is "<h2>Thank you</h2>" will the mail be sent? T or F
5. If the message is "<h2> Welcome to the world of PHP and mailing. I hope that you are enjoying your stay</h2>" will the mail be sent? T or F
6. If the message is "<h2 style='color: darkgoldenrod;'>Thank you</h2>" will the mail be sent? T or F
7. Answer the WHY for each question above.

Exercises:

1. Add to the code so that Submit = Register is not part of the message (use an if statement)
2. Add inline css to the email message to create a nicer look
3. **Spammer**
Edit your mail-message function so that it sends ten emails instead of one. (Make sure to only practice this on yourself!)
4. **Time-stamp**
Edit your mail-message function so that it includes the timestamp of when the form was submitted. (Hint: <?php print time(); ?>)
5. **Pre-mail**

Edit your mail-message function so that it sends emails to a predetermined list of 3 people, instead of sending an email to the person who submitted the form.

6. Newsletter & Mailing List

Edit your mail-message function so that it sends the contents of a file to the email that has been submitted through your form. The file should contain your 'Newsletter' for your business or company. Also, append the email to a mailing list csv file that you create.