

## Lesson 7 Action Script Working with Forms, Validation, PHP - R. Berdan

### Objectives:

- 1) Work with forms to validate a text field
- 2) Output html formatted text to a text field e.g. message.htmltext
- 3) Create a simple form and send the data by e-mail
- 4) Create a form and send the data to a text file in order to create a simple database using a server site scripting language PHP
- 5) Note PHP requires access to a PHP enabled server

In order to collect data for surveys, shopping carts, job applications etc – we often use forms. Forms built using HTML are relatively easy to create, however HTML is severely limited in how it can format the appearance of the forms. Flash allows an almost unlimited control over the appearance of forms.

To begin – we will use the tutorial from lesson 13 to create a script that validates – that is checks to see if a form box has been correctly filled out before permitting it to be sent. In this exercise we will not complete the chapter but simply carry out the first validation step, check if the name field has been entered and that the data entered was text not a number.

### Start on Page 402 in text book - Open validate1 fla

Select Frame 1 and enter

```
stop ();  
errors = new Array();
```

errors is an array (variable) which will hold one or more error messages that will be displayed as a result of the validation process.

Then add the following

```
stop();  
errors = new Array();  
  
function clearForm()  
{  
    name.text = "";  
    email.text= "";  
    state.text = "";  
    zip.text = "";  
    errorLog.text = "";  
    errors.length = 0;  
}
```

When called this function resets the value of the scene elements to their initial values. It also removes any errors within the error array. Basically it resets the form.

Now select the clear button and add the following action to call the clearForm() function.

```
on (release)  
{  
clearForm()  
}
```

save the file as validate2 fla.

### **Validating strings – text**

Strings have the property they must first be “text or type” and they have a length – number of characters that make up the string.

In frame 1 of validate2 fla file add the following script

```
function validateName()  
{  
    if (name.text.length < 2 || isNaN(name.text) == false)  
    {  
        errors.push("Please enter a valid name");  
    }  
}
```

if name.text.length < 2 returns a value of false and therefore returns an error the value could have been set to <1 which means no characters have been entered.

isNaN(name.text)

isNaN(“text”) returns a value of true - in other words it is true it is not a number.

isNaN(465) returns a value of false – it is false because 465 is a number

The whole (conditional test) must evaluate to either true or false the || or symbol means if data is not entered or the value typed in is a number – evaluate the expression to true – and register an error

isNaN(name.text) == false “If name is a number”

At the end of the current script add the following code

```

function validateForm()
{
    errorLog.text = "";
    errors.length = 0;
    validateName();

    if (errors.length > 0)
    {
        errorLog.htmlText = "<b>These errors were found:</b><br>";
        var i=-1;
        while (++i < errors.length)
        {
            errorLog.htmlText += errors[i] + newline;
        }
    }
    else
    {
        gotoAndStop("Confirm", 1)
    }
}

```

The first two lines clear the errorLog text field plus any errors

Then we call the function validateName() which checks if text has been entered and it is not a number that has been entered into the text field

If an error is detected in the errors array – display text in the adjacent window called errorLog, note errorLog.htmlText - unlike using errorLog.text, this new property allows you to add <html> tags to modify the output properties of the text in the window.

The next part of the script uses a while loop – it checks if there is an error and so long as there is one it continues to output an error message, the variable “i” is a counter that tracks the number of errors.

Finally select the Submit button and add the following script and then try Control test movie

```

on (release)
{
    validateForm() // see These errors were found: Please enter a valid name
}

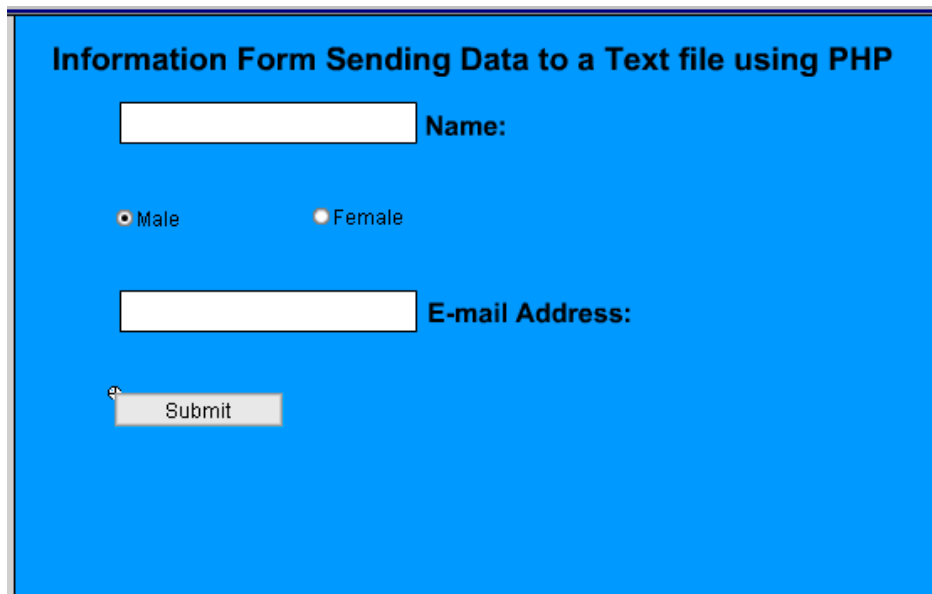
```

## Using PHP with Flash to send data to an e-mail address and a text file

by Juan Rivera and Rob Berdan (Science & Art)

You will build a simple form in Flash with a submit button and have the data sent to a new browser window where you show the data input and thank the user. The data will be appended to a “data.txt” file sitting on the server. Each time the form is submitted you can also send the data to a specified e-mail address. You can FTP and read the file or have the entire file output to a browser window. In effect the data.txt file serves as a simple database. This is not a good method to store secure data or personal data but can be used for surveys and other non-sensitive types of information. The tutorial requires a server that will run PHP (Hypertext Preprocessor) which is a “simple” server side scripting language you embed into HTML documents.

1. Start Flash MX – we will first build a simple form that looks like the picture below.



The image shows a screenshot of a web form titled "Information Form Sending Data to a Text file using PHP". The form is set against a blue background. It includes a text input field for "Name:", two radio buttons for "Male" and "Female", a text input field for "E-mail Address:", and a "Submit" button.

Simple form – the background color and design are not important.

2. Insert a text box using the properties box. Flash MX permits you to create 3 different styles of text (static text, dynamic text and input text). Select the Text Tool in the toolbar and to create text boxes make sure you select input text from the properties box. Create an “Input Text box” and place static text beside it “Name”. Below it add another text box and type “E-mail Address:” beside it. In the properties box you will need to add variable names to each text box. Select the text box and in the properties box Var field type in: tName for the Name textbox and tMail for the E-mail Address. You are basically assigning names to each text box. In order to make it easier to see the text boxes make sure you also select “Show Border around text” in the properties box and click on the Character button and select “Embedd Font outlines for: All Characters”. This is so every character a user enters is accounted for.
3. To create the radio buttons, use the Flash UI Components box and drag two radio buttons onto the movie. If the Components box is not visible select Window>Components, locate the Radio buttons and drag two of them onto the movie like that shown above.
4. Click on the left radio button and open the properties inspector if it is not already open. In the list box within the properties box Enter the following values for the first radio button:

Label: Male  
Initial State: true

Group Name: gSex  
Data: Male  
Label Placement: right

For the second radio button provide the following input into the properties box

Label: Female  
Initial State: False  
Group Name: gSex  
Data: Female  
Label Placement: right

5. Now drag a push button onto the screen and in the properties box give it the label "Submit".

Right Click on the submit button and select "Actions", Enter the following code in the Actions box (use the expert Mode to enter the script).

```
on(release) {  
  tName = _root.tName;  
  tSex = _root.gSex.getValue();  
  tMail = _root.tMail;  
  getURL (" flash.php", "_blank", "POST")  
}
```

```
/* this script initializes the variables for the text boxes and  
radio buttons and will submit the data to a file called flash.php  
which will process the data sending it to a specified e-mail  
address, textfile, database etc., blank displays the text in a new  
document window (recall target="blank" from HTML coding) */
```

6. Now save your flash movie into a folder call it form fla, then publish it so you have a form.swf file along with an form.html file.
7. Now open windows notepad and create an empty file (nothing in it) save this file and call it "data.txt" into the same folder as your flash .swf file and .html file.

8. Now you will create he PHP file inside an HTML document using Notepad.

```
<html>  
<head>
```

```

<title>test</title>
</head>
<body>

<?php

// display text in a browser window
print "We received the following information - thank you <br>\n";
print "Name:\t $tName<br> \n";
print "Sex:\t $tSex<br> \n";
print "E-Mail:\t $tMail<br> \n";

// Send data entered to the e-mail address shown below
$to = "rberdan@scienceandart.org";
$subject = "PHP Simple Database";
$body = "Name:\n $tName \n Sex: $tSex \n E-Mail: $tMail \n";
mail($to, $subject, $body);

// create a file called data.txt, open it, write to it and append new data
$file="data.txt";
$file = fopen("data.txt", "a");
// a+ file pointer is placed at the end of file if it does not exist it
//will attempt to create it
fputs($file, "Name:\t $tName\t Sex:\t $tSex\t E-Mail:\t $tMail \n<br>");
fclose($file);

// outputs all the data in the data.txt to the browser window
//$file="data.txt";
//$body = readfile($file);
// to see data in database output to window remove the comment symbols
//on last to lines $file and $body

?>

</body>
</html>

```

When you are finished typing in the code save this file into your folder along with the .swf and data.txt files - call this file flash.php.

The PHP section of the code starts with <?php and ends with ?>  
 \$VariableNames are preceded with \$ signs in PHP

“// “ sign for single line comment, some of the lines have been commented out e.g. \$file="data.txt"  
 these sections are optional since this statement outputs all the data to the browser window. /\* ... \*/ is a multiline comment in PHP, javascript, C and Java

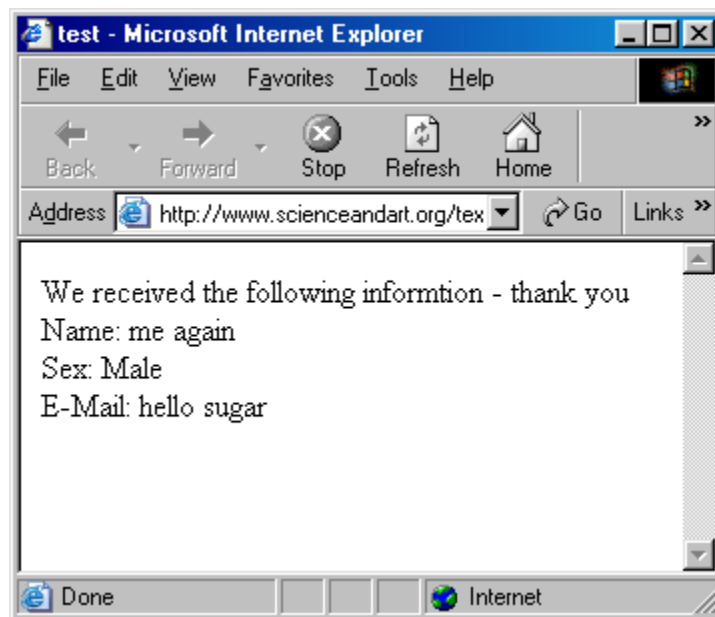
**IMPORTANT:** PHP is case sensitive and every statement must end in a semicolon “;” or the file will not work!

When you are finished copy the folder to your web server using an FTP program like WS\_FTP – type in the URL [www.yourcompany.com/form.html](http://www.yourcompany.com/form.html) enter data into the form and select the submit button. If you did everything correctly the data will appear in a new blank window and if you check your e-mail

you will see the data there as well. After you submit several forms you can FTP the data.txt file to your computer and open it in note pad. It won't look pretty but all the data should be there looking something like that below:

```
Name: test Sex: Male E-Mail: test
Name: Rob Berdan Sex: Male E-Mail: Rob Berdan
Name: Fred Flinstone Sex: Male E-Mail: fred@bedrock.com
Name: Donna Sex: Male E-Mail: donna@hotmail.com
Name: Fred Flintstone Sex: Male E-Mail: fred@bedrock.com
Name: Fred Flintstone Sex: Male E-Mail: fred@bedrock.com
Name: Hello Again Sex: Male E-Mail: bunny@rabbits.com
Name: One more time Sex: Male E-Mail: jimmy@gohoo.com
Name: One more time Sex: Male E-Mail: jimmy@gohoo.com
Name: OK Sex: Male E-Mail: one more time
```

I copied the text from notepad here and it looks better above then it does in note pad. The data could be imported into a spread sheet or into a database. The process can be automated but we need to know some mySQL and/or work with a specific database.



This is what the text looks like in the blank browser window after submitting the form.