

MSCI 346 – Spring 2019

LAB 2: Introduction to Web Development, HTML and PHP

Introduction to Web Development

We need to perform activities such as:

- Connecting to the database server
- Querying the database
- Displaying results on a web form
- Querying the database using user-input as a parameter

1. HTML to create web forms
2. Server Side Scripting (PHP)

1.1 HTML - Hypertext Markup Language

- HTML is a set of “markup” symbols (ie. code) used to render content for the web.
- The markup tells the web browser how to display the web page’s text, images, sound, and video files to the user.
- The individual markup codes are referred to as elements, but many people also refer to them as tags.

1.2 HTML Documents

- HTML documents are text documents
 - HTML file extensions are .html or .htm
- You can create HTML documents using
 - Notepad in Windows, TextEdit in Mac OSX, gedit in Linux
 - Other popular editors include NVU (<http://www.nvu.com>), Notepad++ (<http://www.notepad-plus-plus.org> / zip package)
- Advanced editors include Google Web Designer or Adobe Dreamweaver

1.3 HTML Example

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Website Title</title>
    <link rel="stylesheet" type="text/css" href="mystyle.css">
  </head>
  <body>
    <h1>Level 1 Heading</h1>
    <p>Paragraph content, "Hello world!"</p>
  </body>
</html>
```

1.4 HTML Form

- HTML forms are used to pass data to a server.
- An HTML form can contain input elements such as text fields, checkboxes, radio-buttons, options, submit buttons, and more.
- A typical form structure.

```
<form>
```

```
    <input></input>
```

```
    <input></input>
```

```
    <input></input>
```

```
</form>
```

1.4 HTML Form (Cont'd)

- `<input type="the_type" name="unique_name"></input>`
 - `type="text"` creates a plaintext input field
 - `type="password"` create an obscured password field
 - `name=""` sets variable names for data when form is submitted
- Example

```
<form>
  <label for="firstname">First Name:</label>
  <input type="text" name="firstname"></input><br>
  <label for="lastname">Last Name:</label>
  <input type="text" name="lastname"></input><br>
  <label for="password">Password:</label>
  <input type="password" name="pass"></input>
</form>
```



First Name: Reid

Last Name: Miller

Password: ●●●●●●●●●●

1.4 HTML Form (Cont'd)

- `<input type="the_type" value="button_value" name="unique_name"></input>`
 - type="checkbox" input can take multiple checked values
 - type="radio" input can take one value in a set
 - value="" sets the real value when a user makes a selection
 - name="" sets variable name for data when form is submitted
- Example

```
...
<form>
  <label for="gender">Gender:</label><br>
  <input type="radio" name="gender" value="male">Man</input><br>
  <input type="radio" name="gender" value="female">Woman</input><br>
  <label for="Interests">Interests:</label><br>
  <input type="checkbox" name="interests" value="hockey">Hockey</input><br>
  <input type="checkbox" name="interests" value="lacrosse">Lacrosse</input><br>
  <input type="checkbox" name="interests" value="football">American
Football</input><br>
</form>
```

Gender:

☒ Man

☐ Woman

Interests:

☐ Hockey

☒ Lacrosse

☒ American Football

1.4 HTML Form (Cont'd)

- `<input type="submit"></input>`
 - Defines a submit button.
 - Sends form data to a server when clicked.
 - The data is sent to the page specified in the form's action attribute.
- Example

```
<form action="submit.html" method="get">  
  <label for="firstname">First Name:</label>  
  <input type="text" name="firstname"></input><br>  
  <label for="lastname">Last Name:</label>  
  <input type="text" name="lastname"></input><br>  
  <input type="submit" value="Submit the Form"></input>  
</form>
```

1.4 HTML Form (Cont'd)

- `<input type="submit"></input>`
 - Defines a submit button.
 - Sends form data to a server when clicked.
 - The data is sent to the page specified in the form's action attribute.
- Example

```
<form action="submit.html" method="get">  
  <label for="firstname">First Name:</label>  
  <input type="text" name="firstName"></input><br>  
  <label for="lastname">Last Name:</label>  
  <input type="text" name="lastName"></input><br>  
  <input type="submit" value="Submit the Form"></input>  
</form>
```

Form data is usually handled by a server side language (e.g. PHP)

1.5 Embed Picture to HTML files

- HTML images are defined with the tag.
- The source file (src), alternative text (alt), and size (width and height) are provided as attributes.
 - src = "file name/file location"
 - alt = "alternative info. to be provided if the image cannot be displayed"
 - Style = define properties of the picture

```
7. <br>
```

1.6 Redirect to Another Webpage

- Create a link to redirect to a different webpage.
- HTML links are defined with the <a> tag.
 - href = “name/destination of the webpage that will be redirected to”

23. `How to add pictures`

[How to add pictures](#)

1.7 HTML References

- <http://www.w3schools.com/html/default.asp>
 - Optional CSS: <http://www.w3schools.com/css/default.asp>

Exercise 1: Deploy the hello_world.html

- Create file “hello_world.html” on your local machine
 - html examples in Sections 1.3 to 1.6
- Transfer the file to your public_html directory
- Rename the file to index.html and hit refresh
- Open a web browser and go to the link :
mansci-db.uwaterloo.ca/~your_username

2. Introduction to PHP

- PHP stands for Hypertext Preprocessor
- It is a server-side programming language
 - Designed primarily for web development
- HTML is interpreted by the web browser while PHP is interpreted by the web server
- PHP can be mixed with HTML to create dynamic content on websites
- PHP can authenticate to and interact with MySQL databases using many available Application Programming Interfaces (APIs) like the mysql extension, mysqli, or PDO_MySQL
- Documentation: <http://php.net/manual/en/index.php>

2.1 PHP syntax: tags and variables

- Opening tag: <?php
- Closing tag: ?>
- Variable names start with the '\$' sign
- Echo prints on the screen/web browser after the .php file is hosted
- '.' concatenates strings

```
<?php
    $name = 'Only PHP';
    echo 'Learning ' . $name;
?>
```


2.2 PHP syntax: HTML embedded in PHP

- HTML tags can be embedded within PHP code
- Here, the paragraph tag <p> is embedded within the PHP code
- HTML has to be written within single quotes if written in the PHP code as the web servers cannot interpret them. The HTML tags, along with other strings, are sent to the web browsers as strings. Here, they are interpreted by the web browser and rendered in their form.

```
<?php
$name = 'HTML embedded in PHP';
echo '<p>Learning ' . $name . '</p>';
?>
```

2.3 PHP syntax: PHP embedded in HTML

- PHP can also be embedded into HTML
- The title of the page is “PHP Test” as indicated by HTML’s <title> tag
 - HTML tags are interpreted by the web browser
 - PHP tags are interpreted by the web server
- Thus, the page needs to be hosted (put in the public_html directory of a web server and accessed through a web browser) to interpret and execute the code in the PHP tags

```
<html>
<head>
  <title>PHP Test</title>
</head>
<body>
<p>
  <?php
    $name = 'PHP embedded in HTML';
    echo 'Learning '. $name;
  ?>
</p>
</body>
</html>
```

2.4 PHP syntax: PHP embedded in HTML

- PHP code can be used to dynamically change value of an input text box or any other input HTML object
- The web page displays a label with “Learning” and a text box of width 300 pixels (interpreted by the web browser)
- The value of the text box “PHP embedded in HTML way 2” is obtained using the PHP code (interpreted by the web server)
- Note: Different php snippets (<?php ... ?>) on a .php file share variables and memory. \$name’s value is obtained from the previous php snippet

```
<p>  
  <label for="learn">Learning</label>  
  <input type="text" name="learn" style="width: 300px;" value="<?php echo $name . ' way 2';?>">  
</p>
```

2.5 Create and deploy PHP files

- There are many free PHP editors that can be found online
 - For this course, it is enough to use Notepad++
 - Save the files with the extension .php
- The web server mansci-db.uwaterloo.ca will host our web form. Transfer the file you want to deploy to the public_html directory of this server. This needs to be done to make our file accessible through the internet
- Transfer your files.
 - Winscp: Connect to mansci-db.uwaterloo.ca on port 22 (using your Quest username and password) and drag the file into the public_html directory there.
- Rename the file as index.php if it is the homepage
- Now this web form can be accessed through a web browser

Exercise 2: Deploy the php_html.php

- Create file “php_html.php” on your local machine
 - PHP code in Sections 2.1 to 2.4
- Transfer the file to your public_html directory
- Rename the file to index.php and hit refresh
 - .html has priority over .php ->please rename index_html created in exercise-1
- Open a web browser and go to the link :
mansci-db.uwaterloo.ca/~your_username
- PHP alone, PHP embedded in HTML and HTML embedded in PHP have been interpreted and are displayed
 - PHP by the web server mansci-db.uwaterloo.ca and HTML by the local machine’s web browser

Exercise 3:

- `<input type="submit"></input>`
 - Defines a submit button.
 - Sends form data to a server when clicked.
 - The data is sent to the page specified in the form's action attribute.
- Example

```
<form action="submit.html" method="get">  
  <label for="firstname">First Name:</label>  
  <input type="text" name="firstName"></input><br>  
  <label for="lastname">Last Name:</label>  
  <input type="text" name="lastName"></input><br>  
  <input type="submit" value="Submit the Form"></input>  
</form>
```

Form data is usually handled by a server side language (e.g. PHP)

Exercise 3 (Cont'd)

- `<input type="submit"></input>`
 - Defines a submit button.
 - Sends form data to a server when clicked.
 - The data is sent to the page specified in the form's action attribute.
- Example

```
<form action="reactionPage.php" method="get">  
  <label for="firstname">First Name:</label>  
  <input type="text" name="firstName"></input><br>  
  <label for="lastname">Last Name:</label>  
  <input type="text" name="lastName"></input><br>  
  <input type="submit" value="Submit the Form"></input>  
</form>
```

Form data is usually handled by a server side language (e.g. PHP)

Exercise 3

- Download reactionPage.php file from Learn
- Transfer the file to your public_html directory
- Create form in your index.php file (previous slide)
 - reactionPage.php in the form action
- Open a web browser and go to the link :
mansci-db.uwaterloo.ca/~your_username