



Internet and Web-Based Systems Development

Lecture 6

Introduction to PHP



Lecture 6

- Dynamic Web Pages
- Server Side Processing
- Introduction to PHP
 - Creating Variables in PHP
 - Working with Numeric Variables
 - Getting Information from a Form
 - Creating a Random Number

Dynamic Web Pages

- As far as we know, a Web server sends back a resource, e.g., index.html, to your PC based on your HTTP request
 - The content of index.html is always the same
- What if we want the content of the page varies, depending on the information included in the HTTP request (submitted in the FORM)?
 - For example, in E-Banking, we make a HTTP request to do a transaction, say depositing \$100 to your account. Your updated balance will be shown in a Webpage, say, balance.php
 - Next time when you deposit \$200 to your account, balance.php will show a different balance even though the same page is accessed



Server Side Processing

- Programming logic executed in the server side (Web Server)
- Scripts are embedded in the XHTML, executed first to produce result (usually XHTML also) which are then sent back to the browser
- Well known technologies
 - CGI
 - ASP, ASP.NET
 - PHP
 - JSP, Servlet



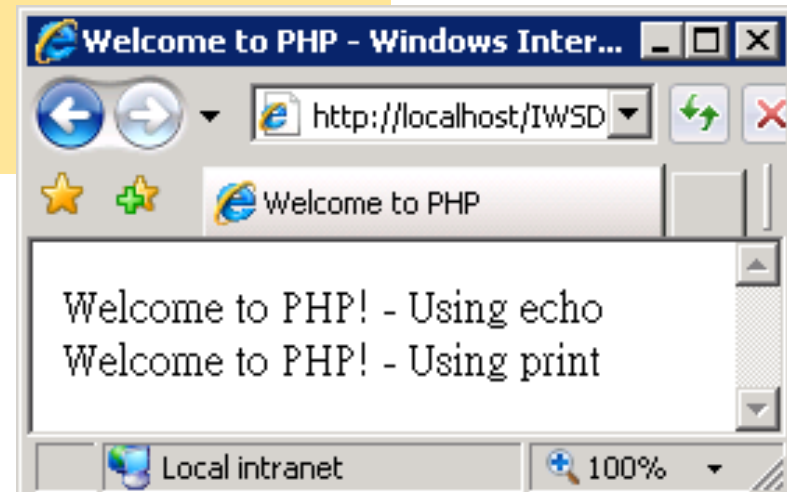
Introduction to PHP

- PHP (Hypertext Pre-processor) is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML
- When the user requests a PHP page, the server first examines the page and executes any script elements before it sends the resulting XHTML page to the user
- PHP pages are having a file extension `.php`
- Current latest version is 5.2.8

Your First PHP Program

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
    Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-  
    transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<title>Welcome to PHP</title>  
</head>  
<body>  
<?php  
    echo "Welcome to PHP! - Using echo <br \>";  
    print "Welcome to PHP! - Using print <br \>";  
?>  
</body>  
</html>
```

welcome.php



Your First PHP Program

- A PHP program looks like a typical XHTML page
`<?php ?>`
 - This tag specifies the existence of PHP code, which is read by the *PHP interpreter* in the Web server
 - Based on the code, different XHTML code may be generated for the Web page
- PHP function **echo** and **print**
 - Write the content on the XHTML file
 - Both have the same effect
- Remember to add a **semicolon (;)** at the end of each statement, just like what you did in C++

User's View on the Source Code

- The browser cannot see the tags in the resulting Web page source

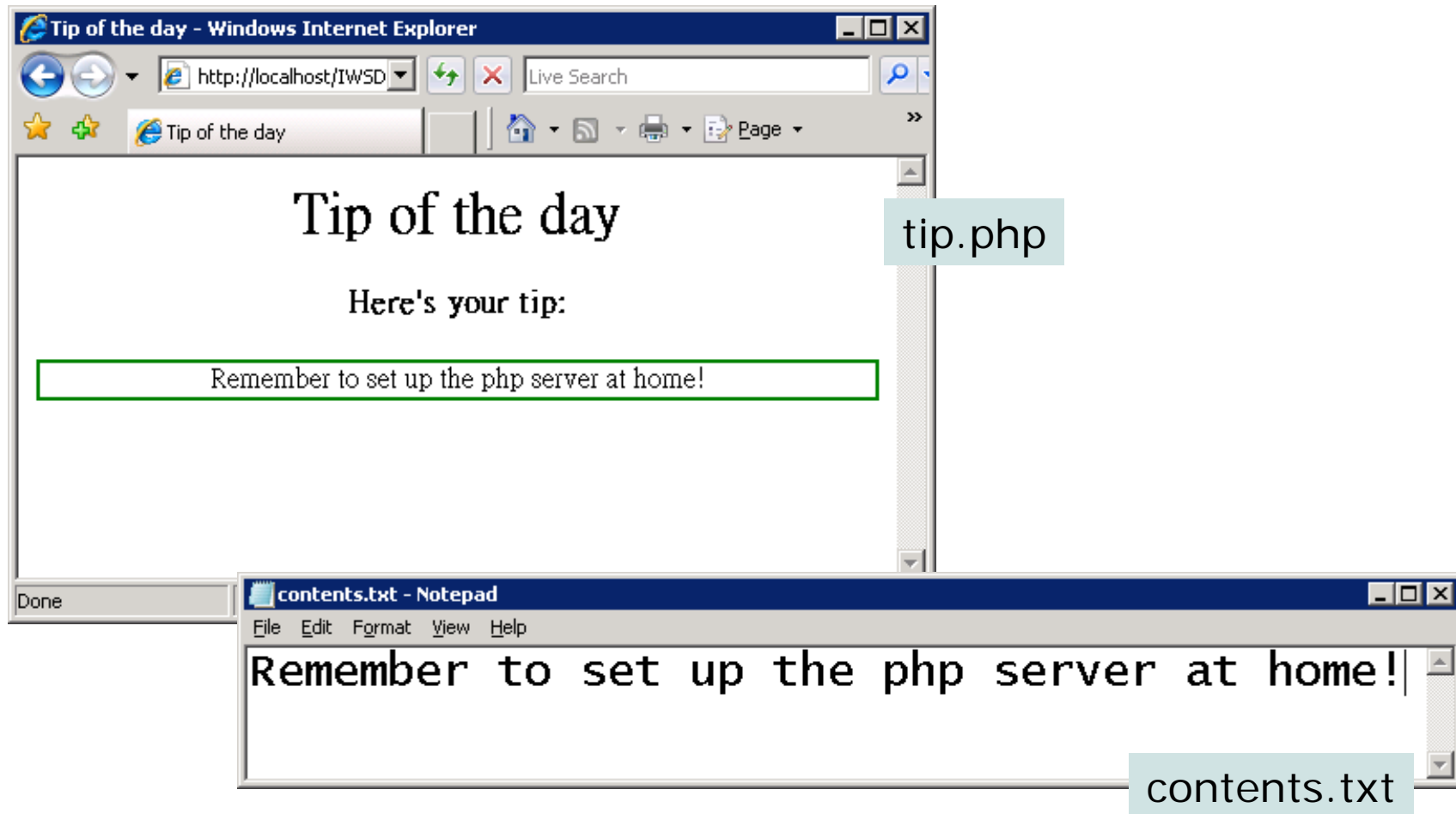
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Welcome to PHP</title>
</head>
<body>
Welcome to PHP! - Using echo <br \>Welcome to PHP! - Using print<br \>
</body>
</html>
```


Another Example

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Tip of the day</title>
</head>
<body style="text-align:center">
<h1>Tip of the day</h1>
<?php
    print "<h3>Here's your tip: </h3>";
?>
<p style="border-color:green; border-style:solid; border-width:2px">
<?php
    readfile("contents.txt");
?>
</p>
</body>
</html>
```

tip.php

Another Example



Another Example

- Function `readfile()`

- Takes an argument – name of a file
- Reads the contents of that file and displays it onto the corresponding position of the page

`readfile("contents.txt")`

- The file *contents.txt* is being stored in the same folder as *tip.php*
 - Or you can use an absolute path, say,
`readfile("c:/contents.txt")`



Creating Variables in PHP

- As what you have learnt in C++ , a variable is a container for holding information in the computer's memory
- Every variable has a name
- Store information into a variable or get information out of a variable

Variable Example

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Variable Example</title>
</head>
<body>
<h1>Demonstrates using a variable</h1>
<?php
```

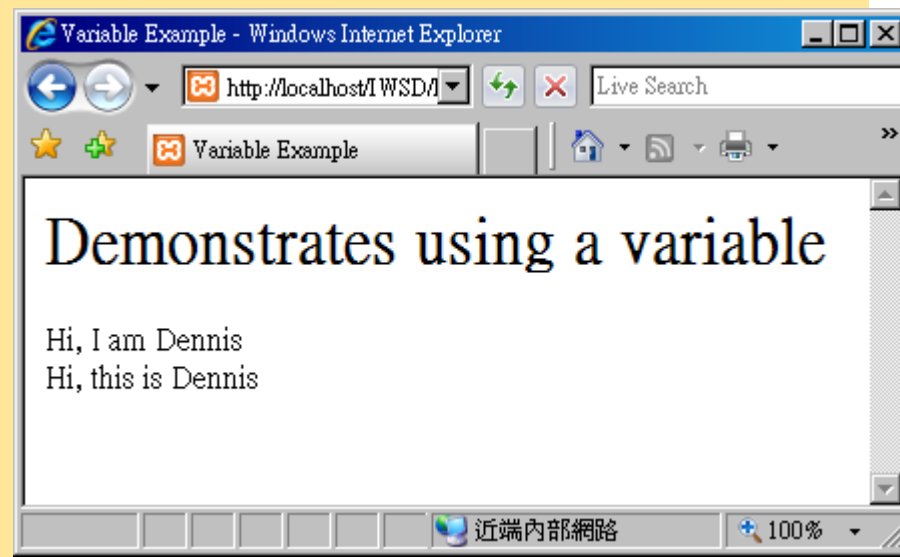
variable.php

```
$userName = "Dennis";
```

```
print "Hi, I am $userName<br>";
print "Hi, this is " . $userName;
```

```
?>
```

```
</body>
</html>
```



Variable Example

- The line `$userName = "Dennis";` means to declare a variable called `$userName` and a value of `Dennis` is assigned to this variable
- In PHP, all variables begin with a **dollar sign** (\$) to distinguish them from other program elements
- Unlike C++, the data type of a variable is not set by the programmer
- PHP decides the data type of variables after interpreting code

Guidelines of Naming Variables

- Make the name descriptive
 - Avoid naming variables like `$a`, `$zzz` that do not reflect the usage of the variables
- Don't use spaces
- Don't use special symbols
 - E.g., `#`, `*`, `/`, and `$`
 - Except underscore `_`
- Case sensitive
 - `$USERNAME` and `$UserName` are not the same variable
 - Use all lowercase except when separating words
 - E.g., `$userName`

Working with Numeric Variables

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-  
transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<title>Numeric values</title>
```

```
</head>
```

```
<body>
```

```
<h3>Demonstrates use of numeric variables</h3>
```

```
<?php
```

```
    $x = 3;
```

```
    $y = 5;
```

```
    print "$x + $y = ";
```

```
    print $x + $y;
```

```
    print "<br><br>";
```

numeric.php (Page 1 of 2)


```
print "$x - $y = ";  
print $x - $y;  
print "<br><br>";
```

```
print "$x * $y = ";  
print $x * $y;  
print "<br><br>";
```

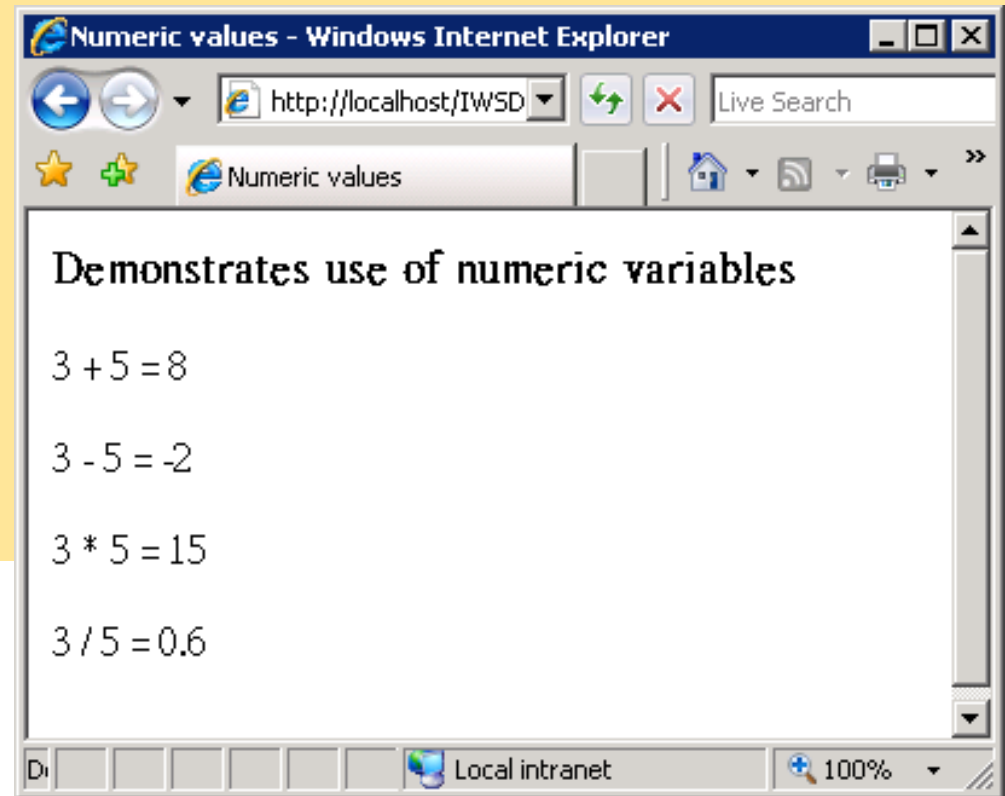
```
print "$x / $y = ";  
print $x / $y;  
print "<br><br>";
```

```
?>
```

```
</body>
```

```
</html>
```

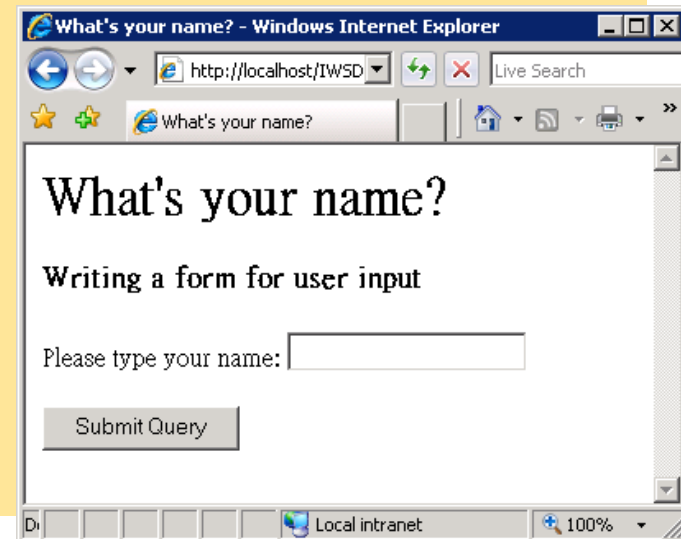
numeric.php (Page 2 of 2)



Getting Information from a Form

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>What's your name?</title>
</head>
<body>
<h1>What's your name?</h1>
<h3>Writing a form for user input</h3>
<form method="get" action="hiUser.php">
    Please type your name:
    <input type="text" name="userName" />
    <br />
    <br />
    <input type="submit" />
</form>
</body>
</html>
```

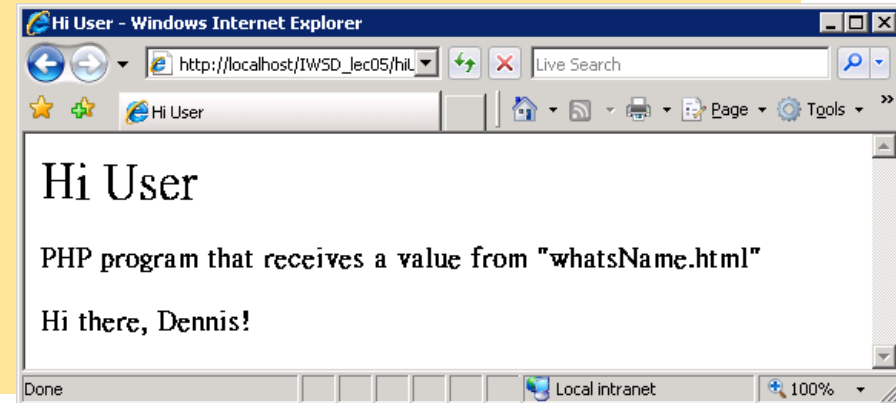
whatsName.html



Getting Information from a Form

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Hi User</title>
</head>
<body>
<h1>Hi User</h1>
<h3>PHP program that receives a value from
    &quot;whatsName.html&quot;; </h3>
<?php
    $user = $_GET["userName"];
    print "<h3>Hi there, $user!</h3>";
?>
</body>
</html>
```

hiUser.php



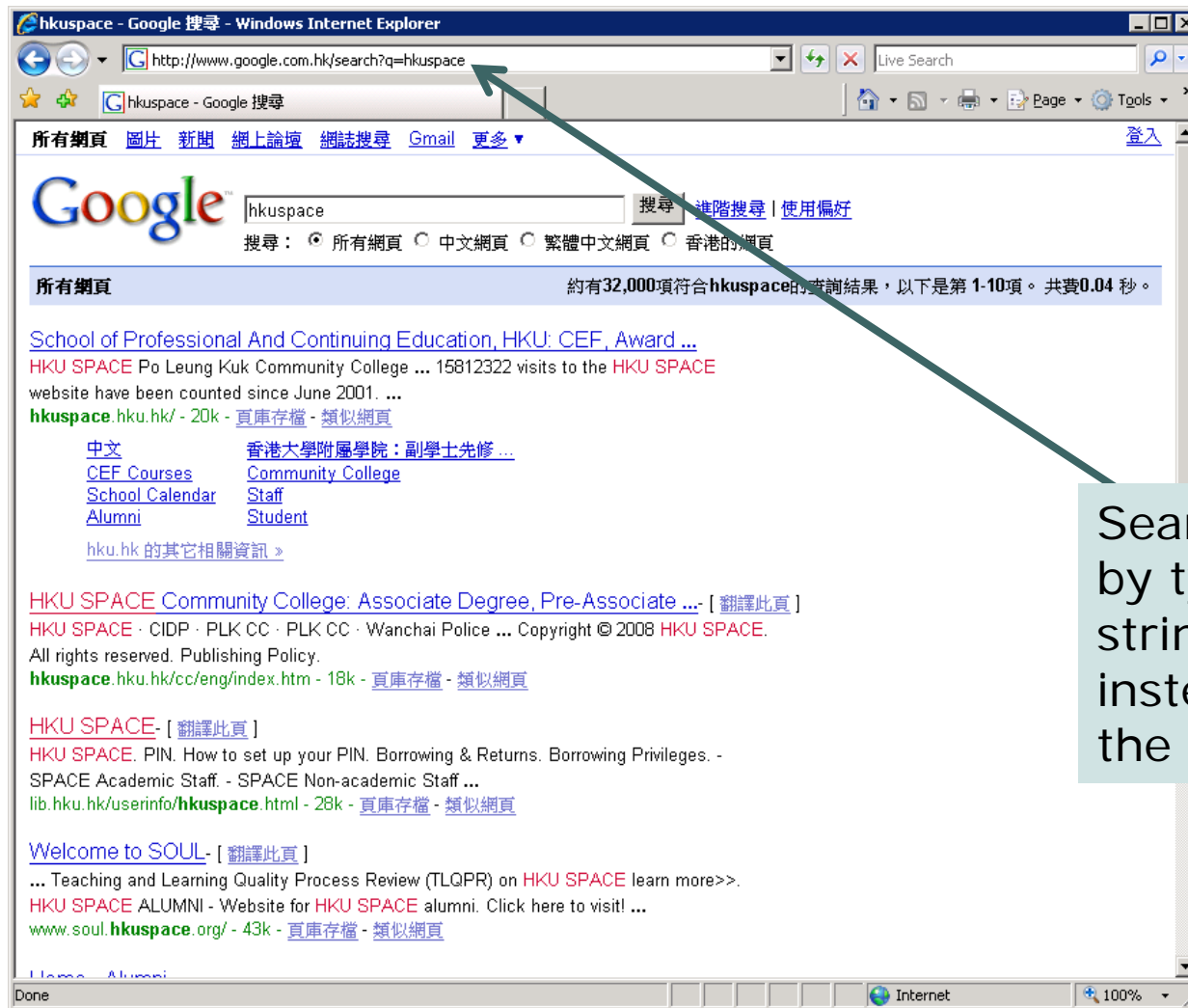
Getting Information from a Form

- A **get** method is used to send the form data to `hiUser.php`
`$_GET["userName"];`
 - The array **\$_GET** contains all values in the form, indexed by the name of each element, using the method **get**
- If you append the name and value pair at the end of the Web address, say,

`http://localhost/IWSD_lec06/hiUser.php?userName=Dennis`

and access **hiUser.php** directly, there will be same result

Using a URL to Embed Form Data




Searching keyword by typing the query string directly, instead of submitting the form

Getting Information from a Form

- The following example use the method **post** to send data to the server

`$_POST["theStyle"];`

- The array **\$_POST** contains all values in the form, indexed by the name of each element, using the method **post**
- Read carefully how data is obtained from different elements of the form




```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Border Maker</title>
</head>
<body style="text-align:center">
  <h1>Border Maker</h1>
  <h3>Demonstrates how to read HTML form elements</h3>
  <form method="post" action="borderMaker.php">
    <h3>Text to modify</h3>
    <textarea name="basicText" rows="10" cols="40">
```

borderMaker.html (Page 1 of 3)

Four score and seven years ago our fathers brought forth on this continent a new nation, conceived in liberty and dedicated to the proposition that all men are created equal. Now we are engaged in a great civil war, testing whether that nation or any nation so conceived and so dedicated can long endure.


```
</textarea>
  <table border="2">
    <tr>
      <td><h3>Border style</h3></td>
```

```
<td colspan="2"><h3>Border Size</h3></td>
</tr>
<tr>
  <td><select name="borderStyle">
    <option value="ridge">ridge</option>
    <option value="groove">groove</option>
    <option value="double">double</option>
    <option value="inset">inset</option>
    <option value="outset">outset</option>
  </select>
</td>
  <td><select size="5" name="borderSize">
    <option value="1">1</option>
    <option value="2">2</option>
    <option value="3">3</option>
    <option value="5">5</option>
    <option value="10">10</option>
  </select>
</td>
  <td style="text-align:left"><input type="radio"
name="sizeType" value="px" />pixels<br />
borderMaker.html (Page 2 of 3)
```

```
<input type="radio" name="sizeType" value="pt" />
points<br />
<input type="radio" name="sizeType" value="cm" />
centimeters<br />
<input type="radio" name="sizeType" value="in" />
inches<br />
</td>
</tr>
</table>
<br />
<br />
<input type="submit" value="Show me" />
</form>
</body>
</html>
```

borderMaker.html (Page 3 of 3)



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Your Output</title>
</head>
<body style="text-align:center">
<h1>Your Output</h1>
<?php
$borderSize = $_POST["borderSize"];
$sizeType = $_POST["sizeType"];
$borderStyle = $_POST["borderStyle"];
$basicText = $_POST["basicText"];

$theStyle = "\"border-width:$borderSize$sizeType; border-
style:$borderStyle; border-color:green\"";

print "<div style=$theStyle>";
print $basicText;
print "</div>";
?>
</body>
</html>
```

borderMaker.php

Getting Information from a Form

The screenshot shows a web browser window titled "Border Maker - Windows Internet Explorer". The address bar shows "http://localhost/IWSD_jec05/borderMak". The page content includes the title "Border Maker", a subtitle "Demonstrates how to read HTML form elements", and a section "Text to modify" containing a text area with the text: "Four score and seven years ago our fathers brought forth on this continent a new nation, conceived in liberty and dedicated to the proposition that all men are created equal. Now we are engaged in a great civil war, testing whether that nation or any nation so conceived and so dedicated can long endure." Below the text area are two form elements: a "Border style" dropdown menu set to "double" and a "Border Size" section with a list of sizes (1, 2, 3, 5, 10) and radio buttons for units (pixels, points, centimeters, inches). The "10" size is selected, and the "points" unit is selected. A "Show me" button is at the bottom.

borderMaker.html

The screenshot shows a web browser window titled "Your Output - Windows Internet Explorer". The address bar shows "http://localhost/IWSD_jec05/bc". The page content includes the title "Your Output" and a text area with the same text as the previous screenshot: "Four score and seven years ago our fathers brought forth on this continent a new nation, conceived in liberty and dedicated to the proposition that all men are created equal. Now we are engaged in a great civil war, testing whether that nation or any nation so conceived and so dedicated can long endure." The text is enclosed in a green rectangular border. A "Done" button is at the bottom left, and a status bar at the bottom right shows "100%".

borderMaker.php

Getting Information from a Form

- Escape character \
- Since we have to print the double quotes onto the XHTML, we have to “escape” the meaning of " in php code
 - We use \"

XHTML tag `<div> ... </div>`

- A division/section in the HTML document
- Can add formatting style to a particular section

Creating a Random Number

- Sometimes, we may need to create a random number, e.g., throwing a dice
- Use the function `rand()`
 - Take two arguments
 - E.g., `rand(1, 6)`
 - Randomly choose an integer between 1 and 6
- From PHP 4.2.0, there is no need to call `srand()` to generate a seed

Creating a Random Number

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Roll a Dice</title>
</head>
<body>
<h1>Roll a Dice</h1>
<h3>Demonstrates rolling a die</h3>
<?php
    $roll = rand(1,6);
    print "You rolled a $roll";
    print "<br \>";
    print "<img src=\"die$roll.jpg\" \>";
?>
<br />
Refresh this page in the browser to roll another die.
</body>
</html>
```

