

COMP 4021  
Internet Computing

PHP

David Rossiter

# PHP

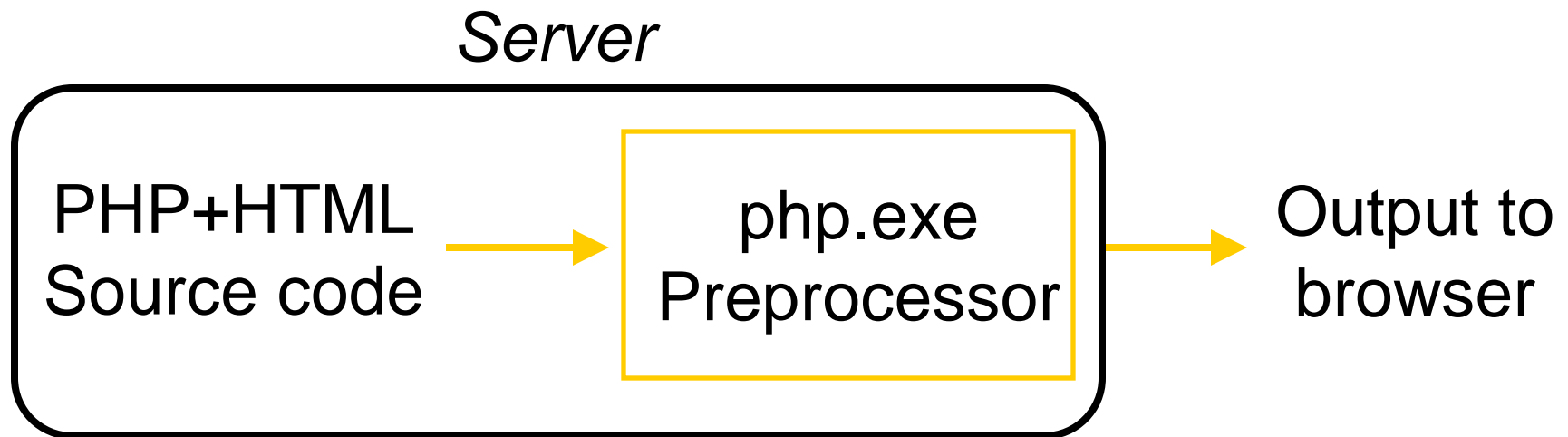
---

'Hypertext' basically means  
'has links' i.e. HTML

- ❑ PHP = Hypertext Preprocessor
- ❑ 'Preprocessor' means that a file gets processed on the server before it gets sent out to the browser
- ❑ So PHP is a server-side scripting language
- ❑ You usually embed PHP commands *inside* HTML
- ❑ Also, you could have a file which contains only PHP

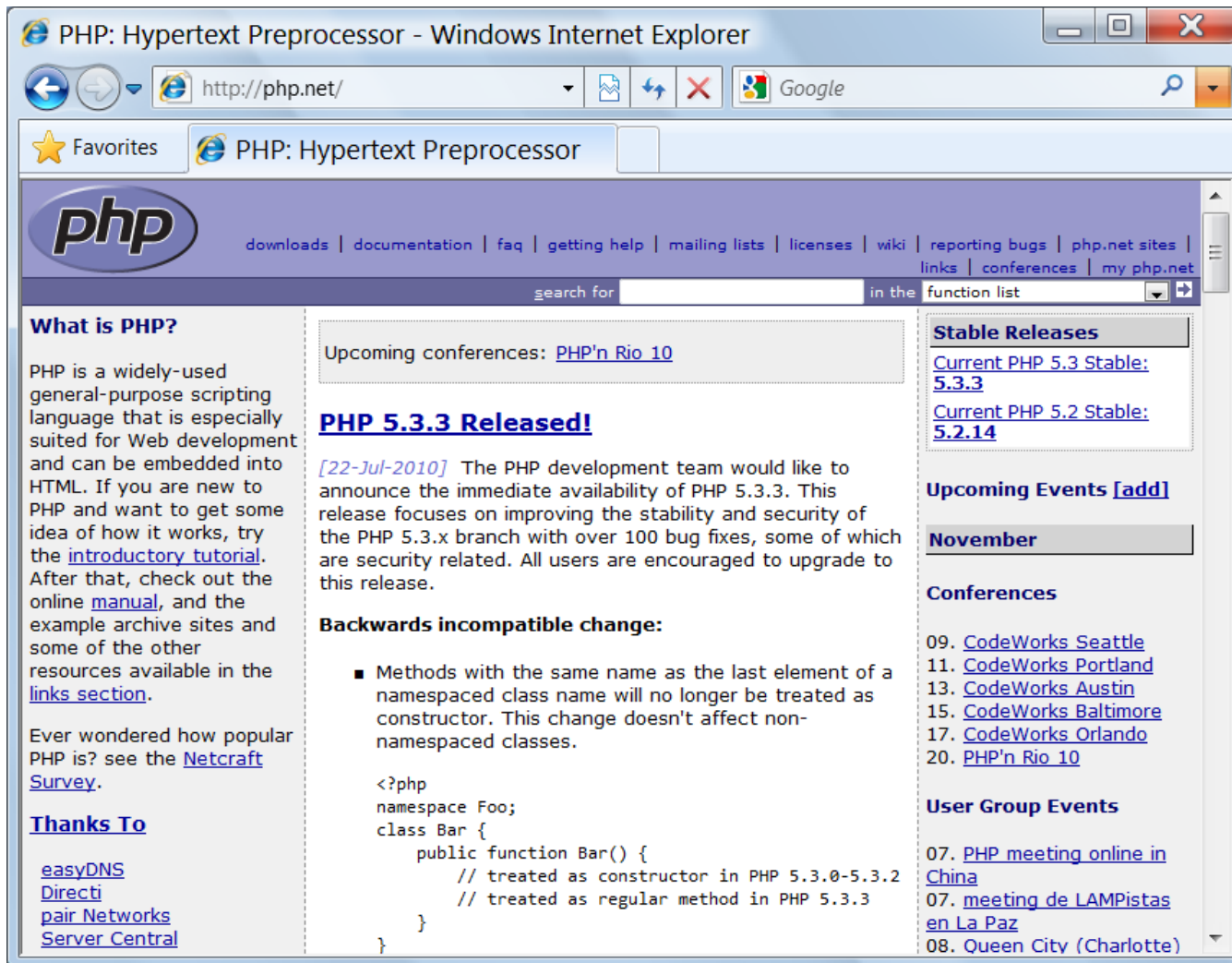
# PHP Operation

---



- ▣ Output could be HTML, JavaScript, XML, . . .

# A Web Site Built on PHP *http://www.php.net*

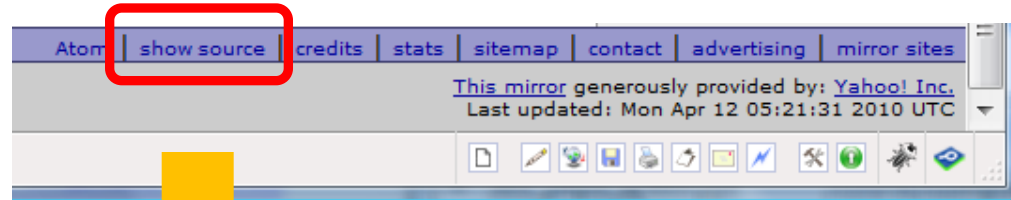


This web site lets you see the PHP source code for any page, which may be useful for learning various PHP techniques

# A Web Site Built on PHP

- Visual result

- At the bottom of the page:



- PHP source



```
<?php // vim: et
/*
    If you're reading this, it isn't because you've found a security hole.
    this is an open source website. read and learn!
*/

/* -----
*/

// Get the modification date of this PHP file
$timestamps = array(@getlastmod());

/* The date of prepend.inc represents the age of ALL
   included files. Please touch it if you modify any
   other include file (and the modification affects
   the display of the index page). The cost of stat'ing
   them all is prohibitive. Also note the file path,
   we aren't using the include path here.
*/
$timestamps[] = @filemtime("include/prepend.inc");

// Calendar, conference teasers & latest releases box are the only "dynamic"
// features on this page
$timestamps[] = @filemtime("include/prgen-events.inc");
$timestamps[] = @filemtime("include/prgen-conf.inc");
$timestamps[] = @filemtime("include/prgen-news.inc");
$timestamps[] = @filemtime("include/version.inc");

// The latest of these modification dates is our real Last-Modified date
$timestamp = max($timestamps);
```

# Configuring Apache For PHP

---

- ❑ To install PHP in an Apache Web Server, you have to add a few lines in the configuration file *httpd.conf*

```
ScriptAlias /php/ "C:/php/"
```

```
AddType application/x-httpd-php .php
```

```
AddType application/x-httpd-php-source .phps
```

```
Action application/x-httpd-php "/php/php.exe"
```

- ❑ After Apache is configured, every file with an extension of *php* or *phps* (=PHP source code) is processed by *C:/php/php.exe* before the output is sent to the browser

# Simple PHP Example

---

```
<html>
  <head>
    <title>PHP Test</title>
  </head>
  <body>
    <?php echo "Hello World";?>
  </body>
</html>
```

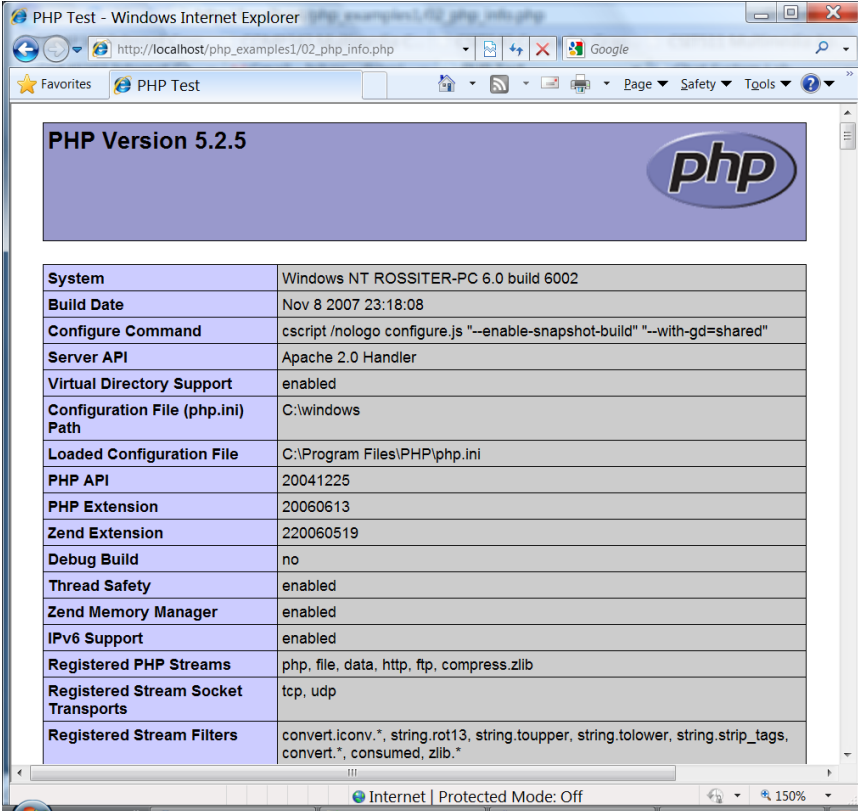



Demo – 01\_starting\_example.php

# PHP General Information

- The instruction `phpinfo()` simply displays some useful information about PHP

`<?php echo phpinfo() ?>`



<b>PHP Version 5.2.5</b>	
	
System	Windows NT ROSSITER-PC 6.0 build 6002
Build Date	Nov 8 2007 23:18:08
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--with-gd=shared"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\windows
Loaded Configuration File	C:\Program Files\PHP\php.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	enabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	php, file, data, http, ftp, compress.zlib
Registered Stream Socket Transports	tcp, udp
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, zlib.*

Demo – 02\_php\_info.php



# PHP Types

---

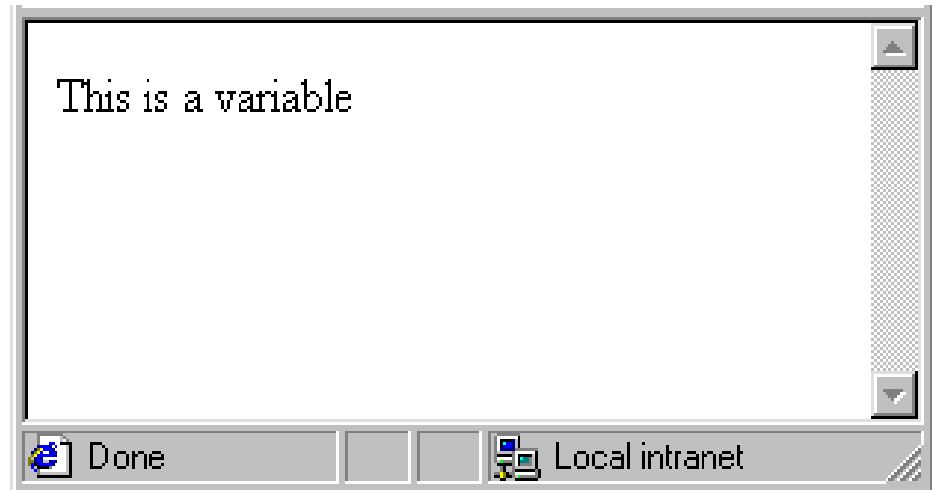
- ❑ PHP is a large language, supporting all the usual things you would expect to find:
  - arrays
  - floating-point numbers
  - integers
  - objects
  - strings

# PHP Variables

---

- Simple example:

```
<?php  
$va= "This is a variable";  
echo $va;  
?>
```



Demo – 03\_variable.php

- A PHP variable always begins with a dollar-sign

# PHP Multi-Dimension Arrays

---

- ▣ Like many languages, PHP supports single dimension arrays as well as multi-dimension arrays. Examples:

`$a[1] = $f;`                # one dimensional

`$a[1][0] = $f;`            # two dimensional

`$a["foo"][2] = $f;`    # two dimensional

# you can mix numeric and associative indices

`$a["foo"][4]["bar"][0] = $f;` # four dimensional

# If Statement

---

**strstr(str1, str2)** is a function for searching a string for another string

```
<?php if( strstr($HTTP_USER_AGENT,"MSIE")) {?>
    <center>
        <b>You are using Internet Explorer</b>
    </center>
<?php } else {?>
    <center>
        <b>You are not using Internet Explorer</b>
    </center>
<?php } ?>
```

# If Statement – Checking the browser

---



Demo – 04\_check\_browser.php

# for Statement

---

- We want to print 1, 2, 3, ..., 9

```
<?php
    for ($i=1;$i<10;$i++)
        print $i;
?>
```



Demo – 05\_for\_loop.php

# Functions

---

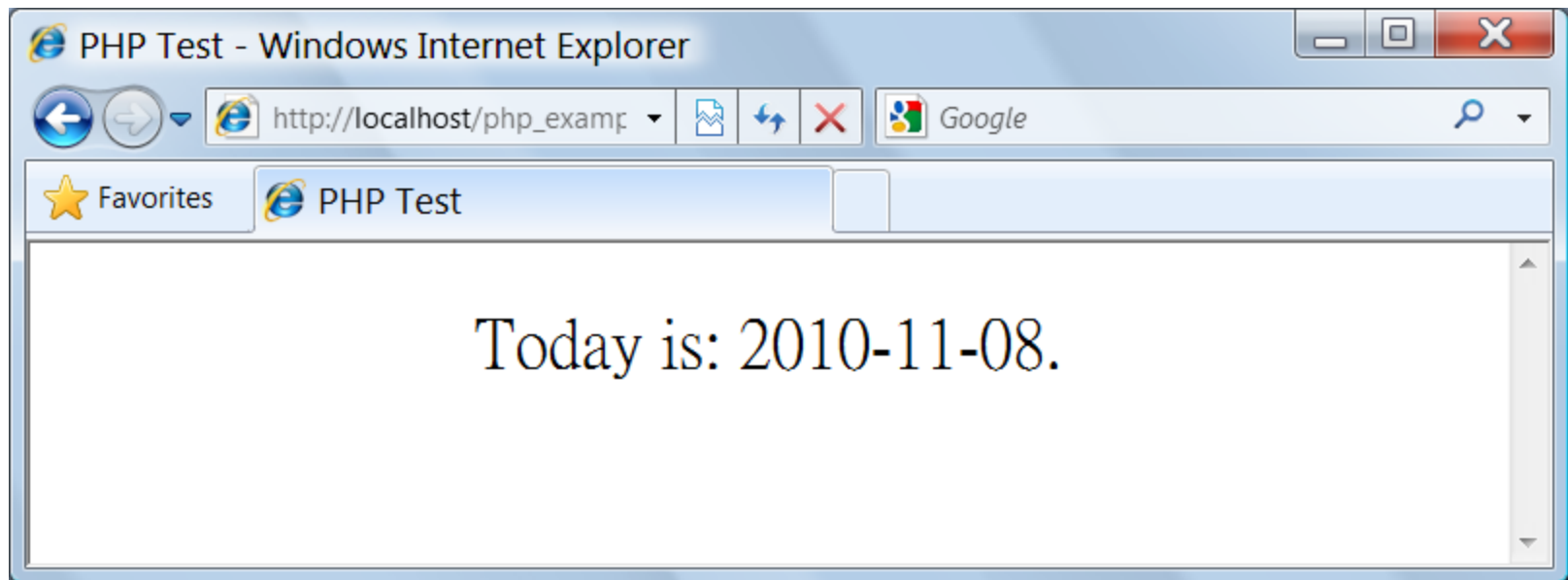
## □ Example:

```
<?php
    function print_date(){
        $today = date("Y-m-d");
        print "<center>Today is:
            $today.</center>";
    }
    print_date();
?>
```

**date(str)** returns a string containing a date formatted according to the parameters

# Functions

---



Demo – 06\_function.php



# Functions – Returning a Value

---

- ▣ General syntax for defining a function:

```
function name($arg_1, $arg_2, ..., $arg_n)
{
    //put your code here
    return value; //return a value
}
```

- ▣ The `return` statement will return a value, like C

# PHP Comments

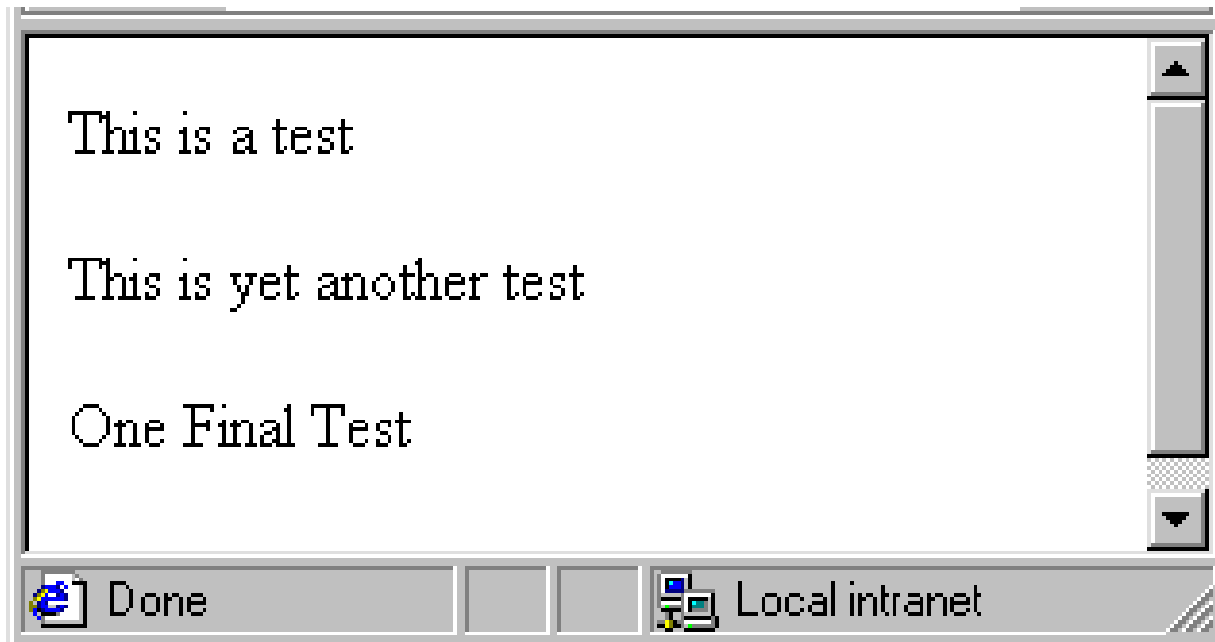
---

## □ Example:

```
<?php echo "<p>This is a test</p>";  
    // This is a one-line c++ style comment  
    /* This is a multi line comment  
    which includes these words */  
    echo "<p>This is yet another test</p>";  
    echo "<p>One Final Test</p>";  
    # This is the last comment  
?>
```

# Comments - Example

---



Demo – 07\_comments.php

# Handling Forms

---

- ❑ Handling HTML forms is very easy with PHP
- ❑ Example on the next few slides

# Handling Forms – HTML code

---

```
<form action="09_action.php" method="post">  
  Your name: <input type="text" name="name">  
  You age: <input type="text" name="age">  
  <input type="submit">  
</form>
```

# Handling Forms – PHP code

---

- The form data gets passed to the following PHP:

```
<html>
```

```
<head><title>PHP Test</title></head>
```

```
<body>
```

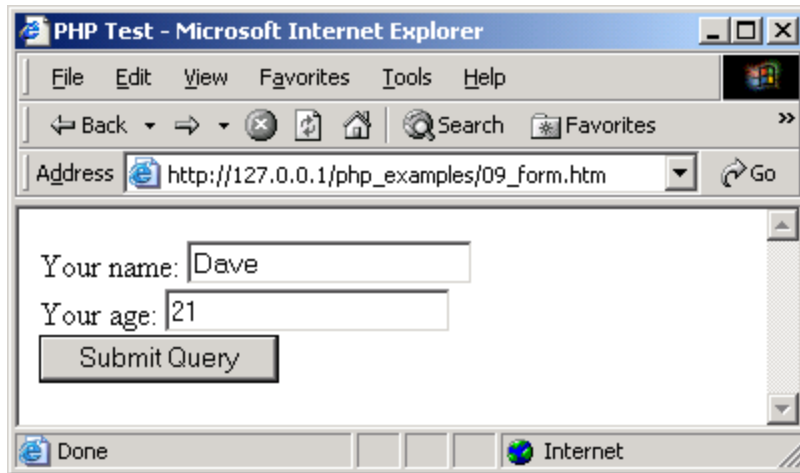
```
  Hi <?php echo $_POST['name'];?>. You are  
    <?php echo $_POST['age'];?> years old.
```

```
</body>
```

```
</html>
```

# An Illustration of the Process

Open the form, input your details,  
submit the data



Demo – 09\_form.htm

PHP program response



Demo – 09\_action.php

# Handle File Upload – HTML code

---

## □ Example form:

```
<form enctype="multipart/form-data"  
      action="10_file_upload.php" method="post">
```

```
<input type="hidden"  
      name="MAX_FILE_SIZE" value="640000">
```

Send this file:

```
<input name="userfile" TYPE="file">  
<input type="submit" VALUE="Send File">  
</form>
```



# Handle File Upload – PHP code

---

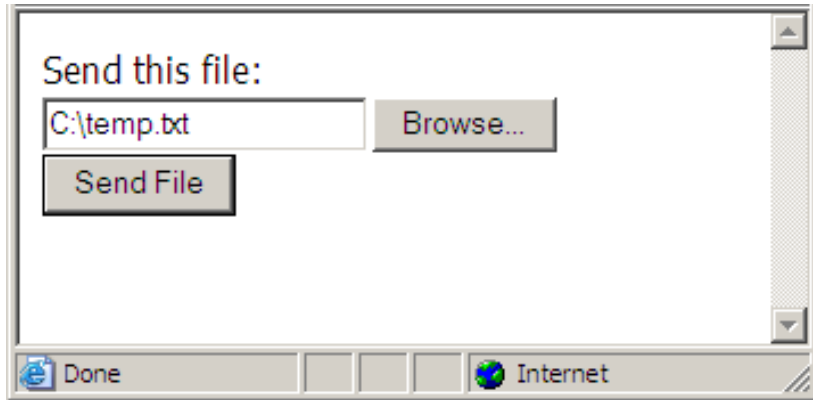
- ❑ The form data is passed to this PHP:

```
<?php
    move_uploaded_file(
        $_FILES['userfile']['tmp_name'],
        $_FILES['userfile']['name']);

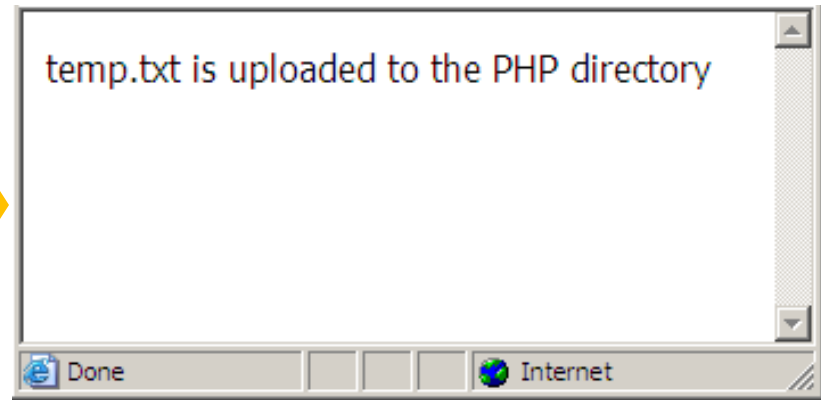
    echo $_FILES['userfile']['name'],
        " is uploaded to the PHP directory";
?>
```

# An Illustration of the Process

---



Demo – 10\_file.html



Demo – 10\_file\_upload.php

# Take Home Message

---

- ❑ PHP is the most popular system for dynamic website development
  - Easy to program
  - Probably not as efficient as JSP and servlets but most websites do not need high efficiency