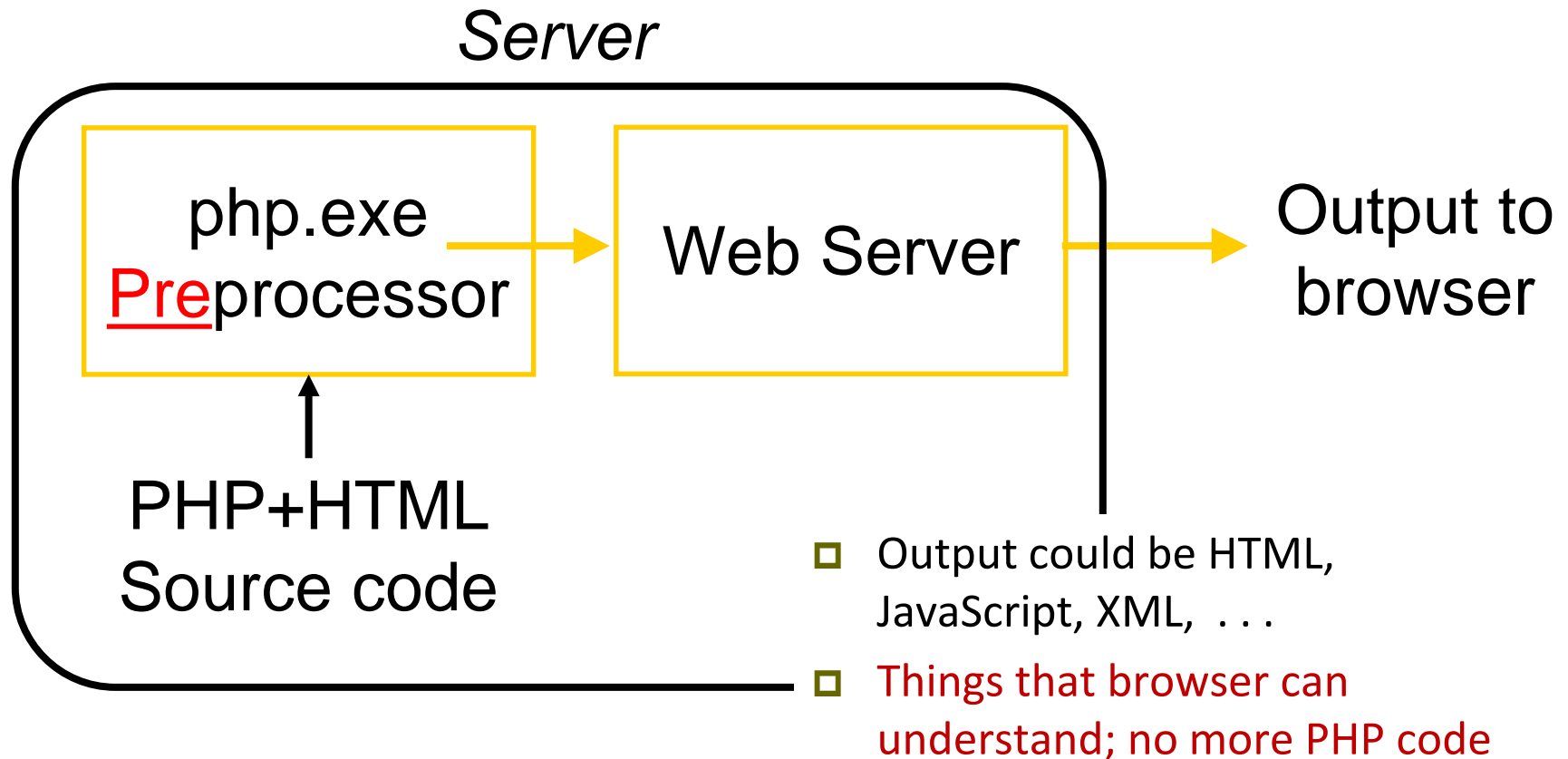# COMP 4021
Internet Computing

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

Server

php.exe
<u>Pre</u>processor

Web Server

Output to browser

PHP+HTML
Source code

- Output could be HTML, JavaScript, XML, . . .
- Things that browser can understand; no more PHP code

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/"** Files under /php/ are scripts/programs
  URL: … **/php/**abc is in **C:\php**abc
  AddType application/x-httpd-php **.php**
  AddType application/x-httpd-php-source **.phps** PHP code in files like abc.phps will NOT be executed by PHP
  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>
```

Hello World

Done   Local intranet
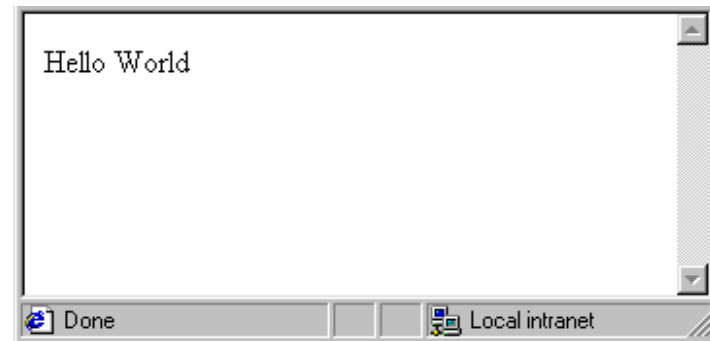
View Page
Source cannot
show any PHP
code:

```
1  <html>
2  <head>
3  <title>PHP Test</title >
4  </head>
5  <body>
6  <!-- A very simple example pri
7  // -->
8  Hello World</body>
9  </html>
```

Run Demo:
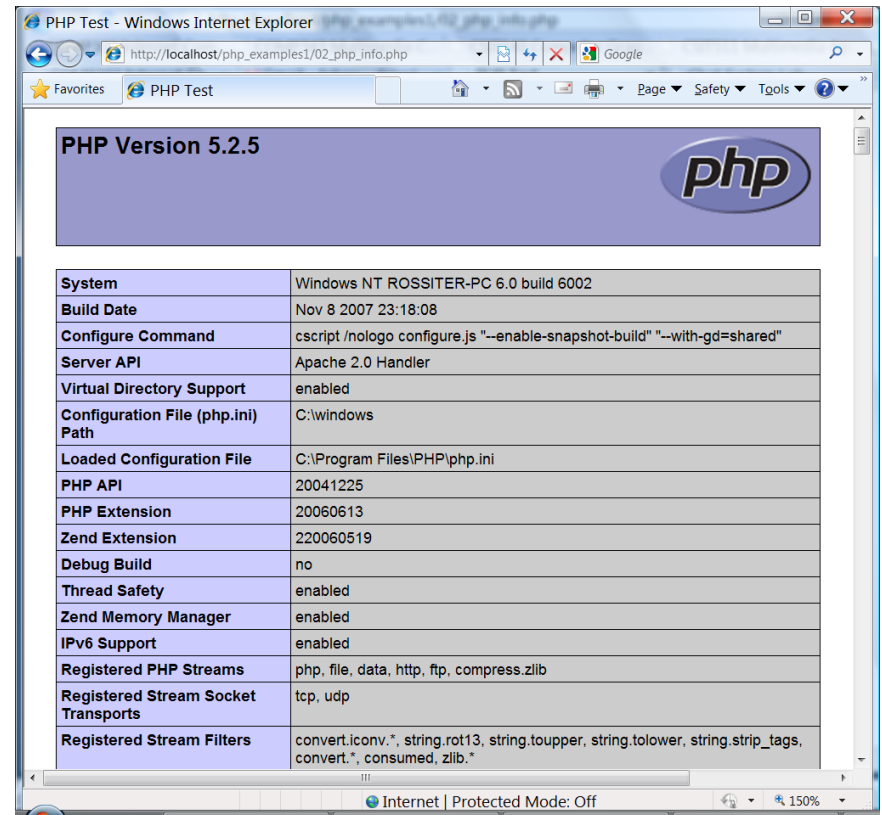01_starting_example.php

Run demo to see the source:
01_starting_example.phps

Command and Response Flow

01_starting_
example.php

PHP
Program

(3) Run the PHP script

PHP

(2) It is PHP!

Web
Server

(1) GET
01_starting_example.php

Browser

(4)

<html> <head>

<title>PHP Test</title>

</head>

<body>

<?php echo "Hello World";?>

</body> </html>

(5)

(6)

<html> <head>

<title>PHP Test</title>

</head>

<body>

Hello World

</body> </html>
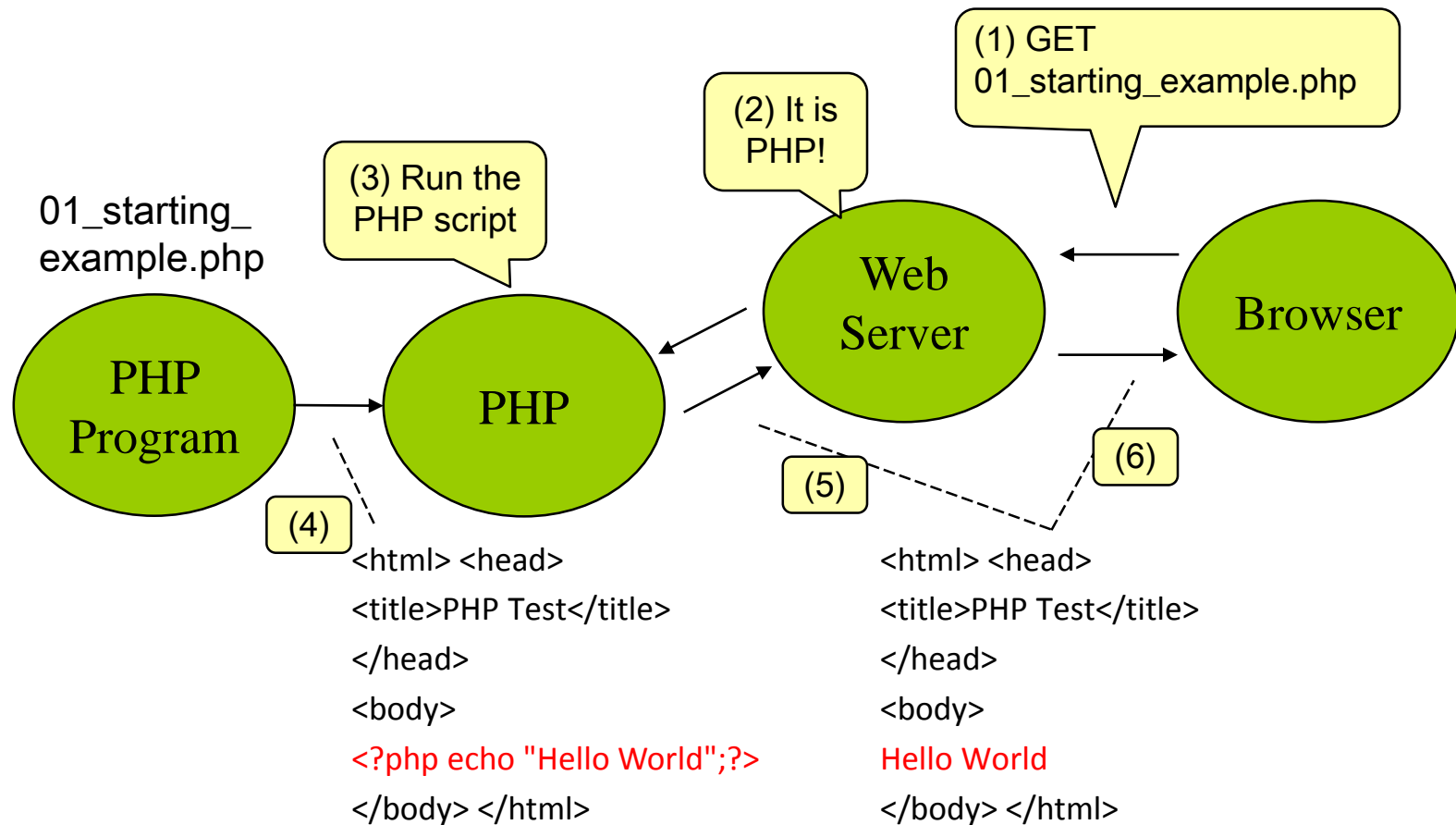
PHP General Information

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

<?php echo phpinfo() ?>
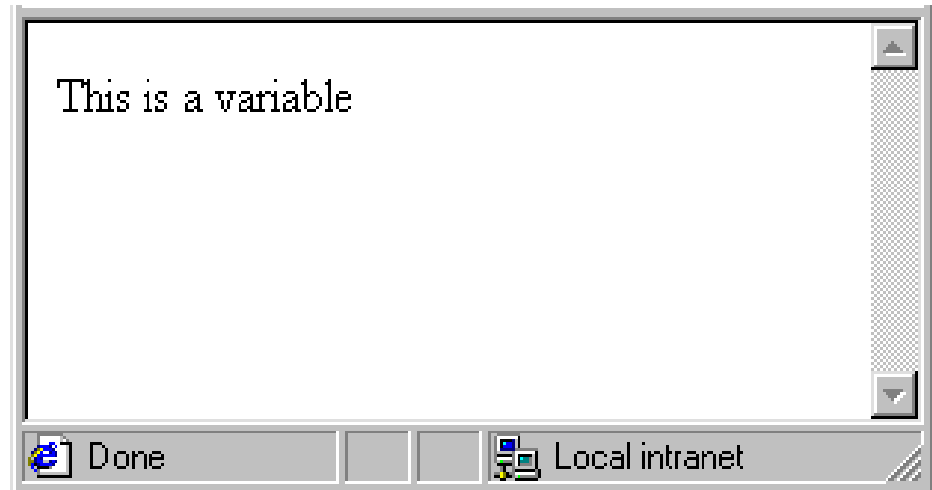

Run Demo – 02_php_info.php

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

This is a variable

Done · · Local intranet

Run 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
<?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

You are using Internet Explorer

Done   Local intranet

Demo – 04_check_browser.php

`for` Statement

□ We want to print 1, 2, 3, …, 9

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

123456789

Done   Local intranet

Demo – 05_for_loop.php
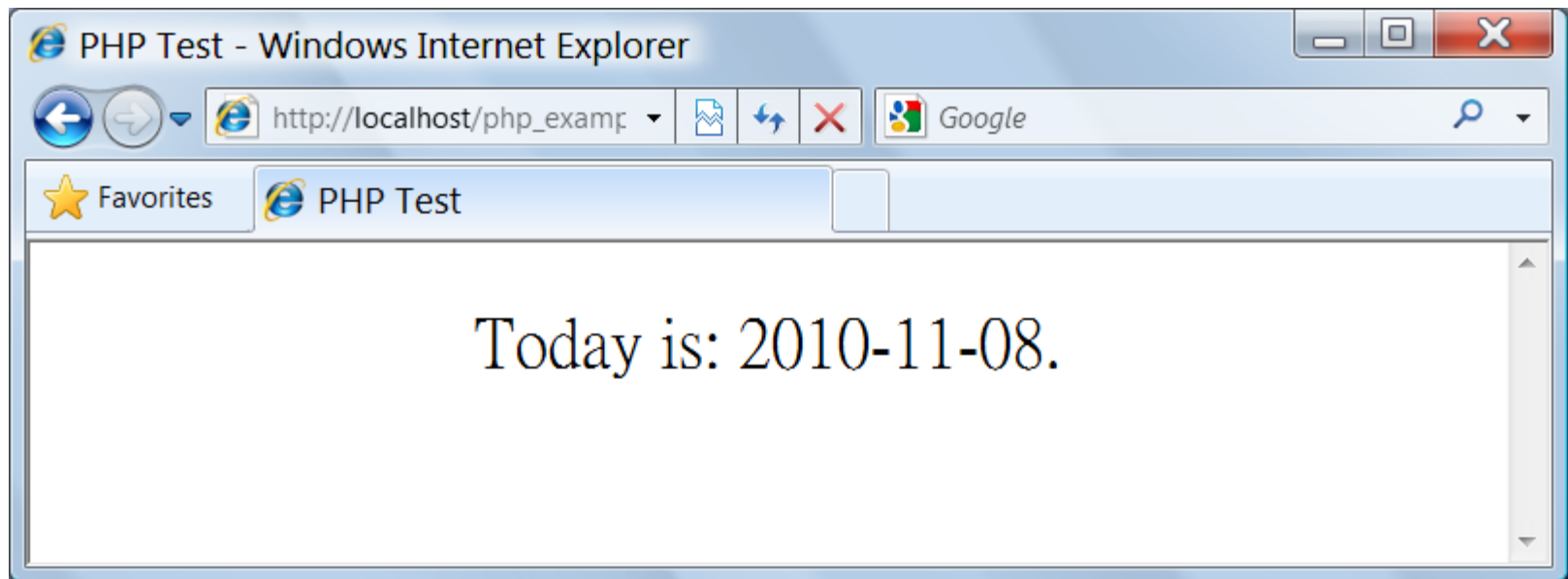
Functions

□ Example:

```php
<?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
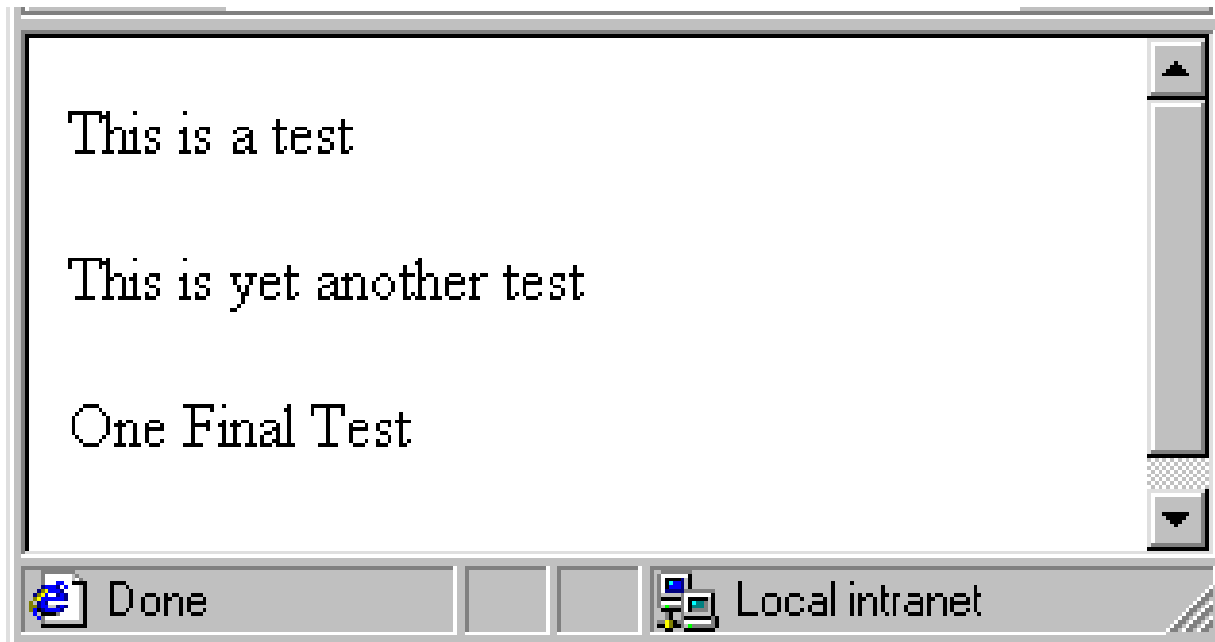}

- 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

```html
<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:
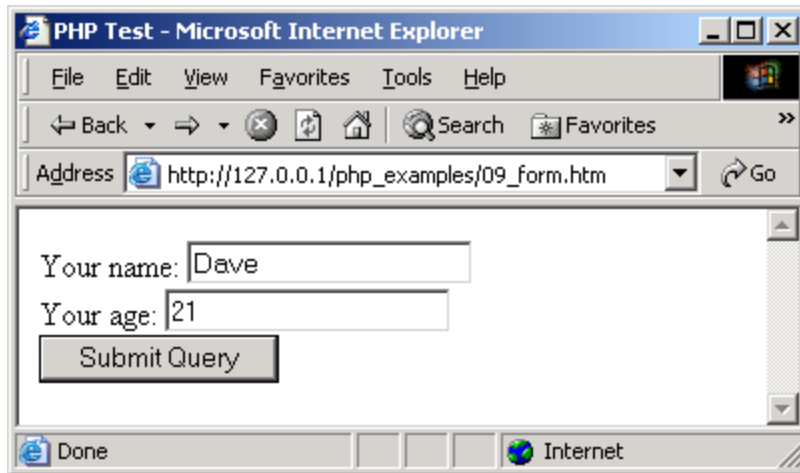
```
<html>
<head><title>PHP Test</title></head>
<body>
  Hi <?php echo $_POST['name'];?>. You are
     <?php echo $_POST['age'];?> years old.
</body>
</html>
```

Run Demo

An Illustration of the Process

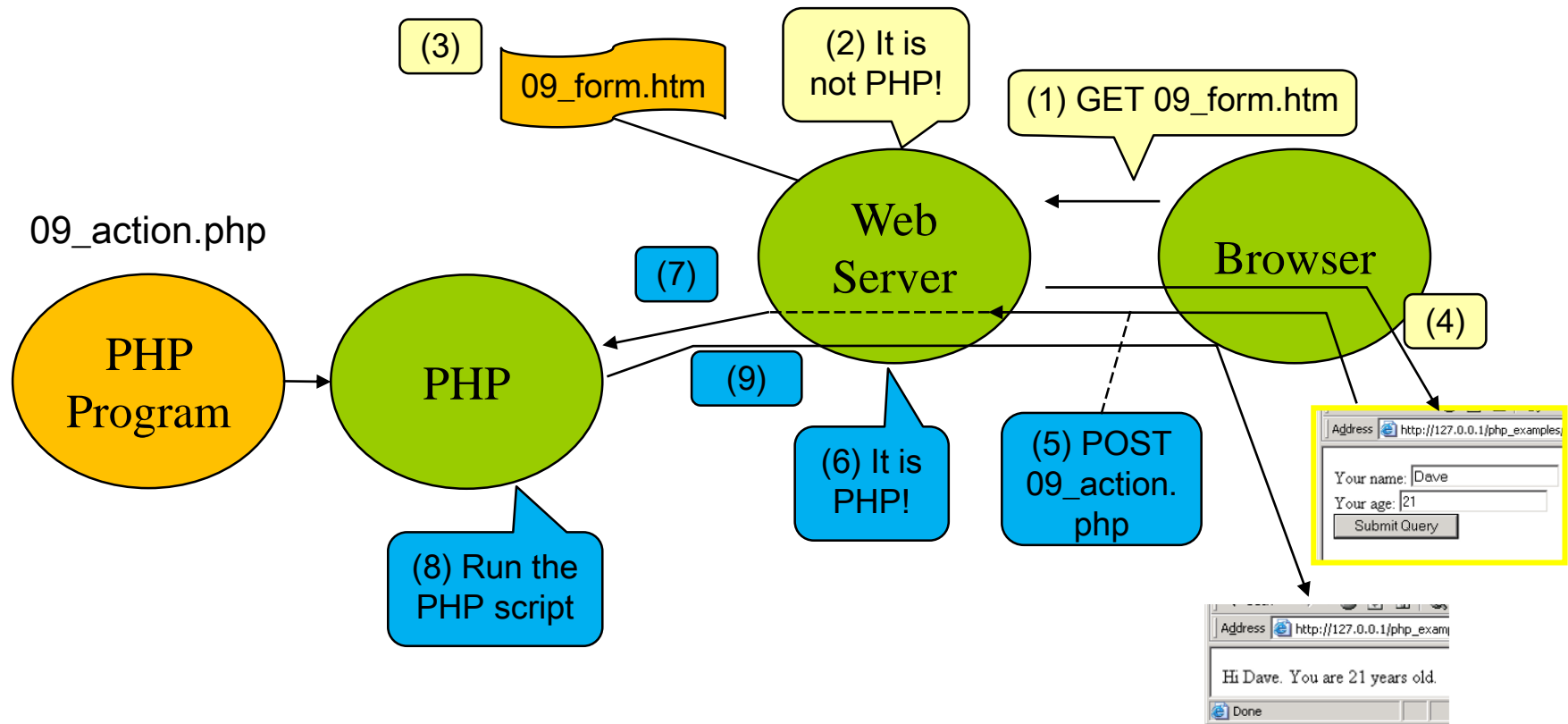Open the form, input your details, submit the data

PHP program response

Demo – 09_form.htm

Demo – 09_action.php

Command and Response Flow

Handle File Upload – HTML code
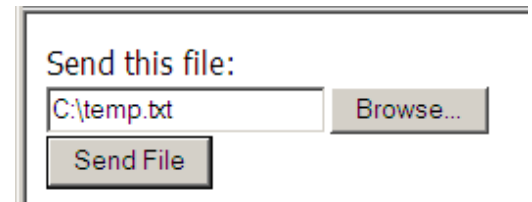
- Example form: Encoding method for form data

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

Hidden form field

<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:

userfile: Field name given in <form>, i.e., userfile

PHP builtin function to move files

Source file: tmp_{filename of upload file)
e.g., tmp_temp.txt

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

    $_FILES['userfile']['name'] );

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

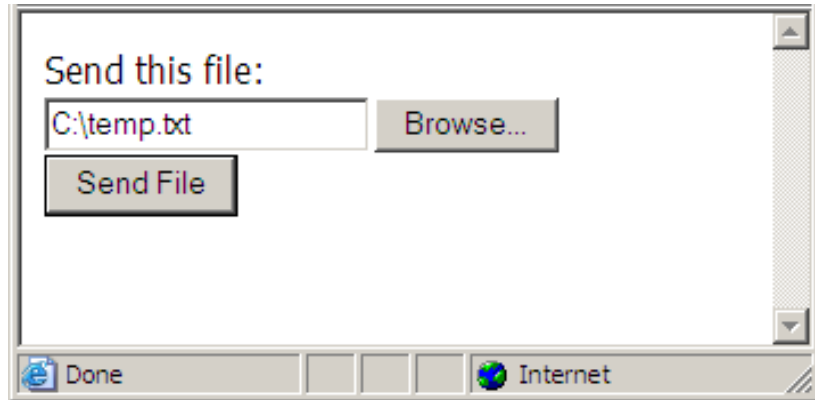Destination file: filename given by user in form

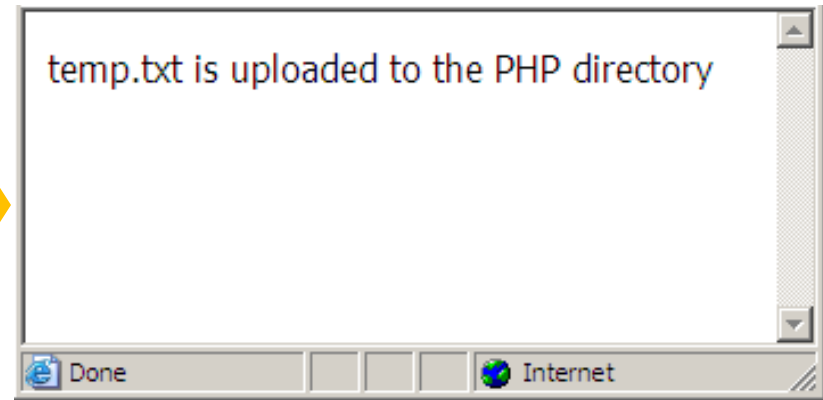Client filename temp.txt
Server filename: tmp_temp.txt
Application filename: temp.txt
* Application could use temp_20181111085910.txt to avoid overwrites

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
- PHP can be used to build large website (e.g., Facebook)