



PHP: Hypertext Preprocessor

Course : CS491B Software Design Lab
Professor: Chengyu Sun
Student : Ya-Chi Liu
Date : 11/12/2004

Introduction to PHP

- HTML-embedded scripting language
- Syntax : is borrowed from C, Java and Perl with PHP-specific features thrown in.
- Goal : to allow web developers to write dynamically generated pages quickly.
- Website : <http://www.php.net>

Overview of PHP

- **Getting Start:**
 1. What is PHP?
 2. What can PHP do?
- **Installation and Configuration**
Example: php.ini
- **Language Reference**
 1. Basic Syntax
 2. Data Types
 3. Constants and Expression
 4. Assignment and Comparison
 5. Array Operators
 6. Control Structures
 7. Functions
 8. Classes and Objects (PHP 5)
- **A Login Example: Login, Database, Session**
- **Reference**

What is PHP?

PHP is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.

Example: hi.php

```
<html>
<head>
  <title>Example</title>
</head>
<body>

  <?php
    echo "Hi, I'm a PHP script!";
  ?>

</body>
</html>
```

Try it : <http://www.monkeyfingers.org/~yliu6/hi.php>

What can PHP do?

- **Main Areas:**
Server-side scripting, Command line scripting, Client-side GUI applications
- **O/S:**
Linux, many Unix variants (including HP-UX, Solaris and OpenBSD), Microsoft Windows, Mac OS X, RISC OS..etc.
- **Web servers:**
Apache, IIS, Personal Web Server, Netscape and iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd...etc.
- **Databases:**
Adabas D, InterBase, PostgreSQL, dBase, FrontBase, SQLite, Empress, mSQL, Solid, FilePro(read-only), Direct MS-SQL, Sybase, Hyperwave, MySQL, Velocis, IBM DB2, ODBC, Unix dbm, InformixOracle (OCI7 and OCI8), Ingres, Ovrinos
- **Protocols:**
LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (on Windows) etc.

Installation and Configuration

- **Windows Installer :**
<http://us2.php.net/downloads.php>
- **php.ini :**
The configuration file is read when PHP starts up.
- **Example: php.ini**

Example: php.ini

```
; Setting parser options
engine = On                      ; PHP engine should be "on" or "off"
short_open_tag = On             ; recognize the standard <?php...?> tag
output_buffering = 2048
expose_php = On                 ; auto check directories for function libraries or classes etc
auto_prepend_file = /home/web/includes/header.php ; specify files that PHP auto appends to the PHP document
auto_append_file = /home/web/includes/footer.php

; Handling errors
error_reporting = E_ALL         ; PHP errors: parsing errors, warnings ka non-fatal errors), and fatal errors.
display_errors = Off            ; turn off the display errors
log_errors = On                 ; write these errors to a custom log file
error_log = "error.log"

; Activating extensions
extension_dir = "C:\Program Files\Internet Tools\Apache\bin\php\extensions" ; the name of the directory for extensions
extension=php_domxml.dll          ; pass the file name of corresponding DLL

; Setting extension-specific variables
SMTP = myserver.localnet.com      ; PHP's mail() function
sendmail_from = me@localhost.com ; address to display in e-mail messages
sendmail_path = /usr/sbin/sendmail ; sending e-mail messages through PHP's mail() function

java.class.path = /php_java.jar   ; set the directories to look in for Java classes and libraries
java.home = c:\jdk
java.library.path = c:\jdk\bin\ ; set the directories to look in for Java classes and libraries
java.library.path = .

session.save_path = c:\windows\temp ; specifies the temporary directory for session information
session.cookie_lifetime = 1800      ; control how long a session cookie remains valid, in seconds
```

Example: php.ini

```
; Security settings : security of your PHP installation
safe_mode = on
safe_mode_include_dir = /usr/local/lib/php/safe-include ; run exec() by placing the program binaries in a special directory
safe_mode_exec_dir = /usr/local/lib/php/safe-bin

open_basedir = /home/web/ ; restrict file operations
max_execution_time = 90 ; seconds that PHP will wait for a script to finish executing

; Configuring file uploads and form variables
file_uploads = On
upload_max_filesize = 2M
register_globals = Off ; <form> variables could only be accessed through the special $_GET and $_POST arrays
post_max_size = 8M ; max data that PHP will accept in a single form submission with the POST method
max_input_time = 90 ; sets a time limit in seconds for receiving input data through POST, GET, and PUT

; Tweaking performance
memory_limit = 8M ; limits on memory usage, be higher than the value of post_max_size.
register_argc_argv = false ; disable the $argc and $argv variables from command line
register_long_arrays = false ; disable the $HTTP_GET_VARS and $HTTP_POST_VARS arrays

; The ini_set() function overrides those settings
ini_set('max_execution_time', 600); ; increases the maximum execution time

// more code
?>
```

Basic Syntax

Escaping from HTML

```
1. <?php echo("<?php echo(\".....\");>"); ?>
2. <? echo ("....."); ?>
3. <script language="php">
  echo (".....");
</script>
4. <% echo ("....."); %>
5. <?php if ($expression) {
  ?>
  <strong>This is true.</strong>
  ?>
  } else {
  ?>
  <strong>This is false.</strong>
  ?>
}
```

Comments

```
<?php
// This is a one-line c++ style comment
echo "This is a test";
/* long .....
..... comment */
?>
```

Data Types

- Four scalar types: string, boolean, integer, float (like 'double')
- Two compound types: array, object
- And finally two special types: resource, NULL

Example: gettype()

```
<?php
$bool = TRUE; // a boolean
$str = "foo"; // a string
$int = 12; // an integer

echo gettype($bool); // prints out "boolean"
echo gettype($str); // prints out "string"

if (is_int($int)) { // If this is an integer, increment it by four
  $int += 4;
}
echo $int;

if (is_string($bool)) { // Is $bool a string ?
  echo "String: $bool";
} else {
  echo "this is not a string!";
}
?>
Try it : http://www.monkeyfingers.org/~yliu6/types.php
```

Constants and Expression

Defining Constants:

```
<?php
define("CONSTANT", "Hello world.");
echo CONSTANT;
echo Constant;
?>
Try it : http://www.monkeyfingers.org/~yliu6/constants.php
```

Expression

```
<?php
function double($i)
{
  return $i*2;
}
$b = $a = 5;
$c = $a++; // post-increment
$d = ++$b; // pre-increment

$f = double($d++);
$g = double(++$e);
$h = $g += 10;
?>
Try it : http://www.monkeyfingers.org/~yliu6/expression.php
```

Assignment and Comparison

Assignment Operators

```
<?php
$a = 3;
$a = 5;
$b = "Hello ";
$b .= "There!";
?>
Try it : http://www.monkeyfingers.org/~yliu6/operator.php
```

Comparison Operators

```
<?php
var_dump(0 == "a"); // 0 == 0 -> true
var_dump("1" == "01"); // 1 == 1 -> true

switch("a") {
  case 0:
    echo "0";
    break;
  case "a": // never reached because "a" is already matched with 0
    echo "a";
    break;
}
?>
Try it : http://www.monkeyfingers.org/~yliu6/comparison.php
```

Array Operators

■ Array Operators

```
<?php
$a = array("a" => "1", "b" => "2");
$b = array("a" => "3", "b" => "4", "c" => "5");

$c = $a + $b; // Union of $a and $b
var_dump($c);

$c = $b + $a; // Union of $b and $a
var_dump($c);

$a = array("cherie", "liu");
$b = array(1 => "liu", "0" => "cherie");

var_dump($a == $b); // bool(true)
var_dump($a === $b); // bool(false)
?>
```

Try it: <http://www.monkeyfingers.org/~yliu6/array.php>

Control Structures

■ If, elseif, while, do-while, for, foreach, break, continue, switch, return, require, include, require_once, include_once.

■ Example:

```
<?php
$arr = array(1, 2, 3, 4);
foreach ($arr as $value)
{
    $value = $value * 2;
} // $arr is now array(2, 4, 6, 8)
?>
```

■ Example:

```
<?php
require 'prepend.php'; // results in a fatal error
include 'vars.php'; // produces a warning
require_once('a.php');
include_once('a.php');
?>
```

Functions

```
<?php
function foo()
{
    echo "In foo()<br />\n";
}

function bar($arg = "")
{
    echo "In bar(); argument was '$arg'.<br />\n";
}

// This is a wrapper function around echo
function echoit($string)
{
    echo $string;
}

$func(); // This calls foo()
$func('test'); // This calls bar()
$func('test'); // This calls echoit()
?>
```

Try it: <http://www.monkeyfingers.org/~yliu6/function.php>

Classes and Objects (PHP 5)

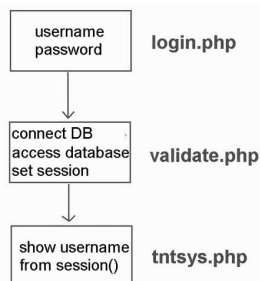
```
<?
class Foo
{
    public $a = 'cherie';
    function aFunc() { print 'yo'; }
}
$foo = new Foo;
?>

<?
$element = 'a';
print $foo->$element;
?>

<?
function getVarName() { return 'a'; }
print $foo->{getVarName()};
print $foo->{aFunc()}; // Prints "Inside `aMemberFunc`"
?>
```

Try it: <http://www.monkeyfingers.org/~yliu6/class.php>

A Login Example



Try it: <http://www.monkeyfingers.org/~yliu6/login.php>

A Login Example : login.php

```
<FORM NAME="myForm" ACTION="validate.php" target="" METHOD="POST">
<TABLE NAME="myTable" BORDER=0>
  <TR>
    <TD><P ID=sub_title>Username</P></TD>
    <TD><input type="text" name=myUsername size=15></TD>
  </TR>
  <TR>
    <TD><P ID=sub_title>Password</P></TD>
    <TD><input type="password" name=myPassword size=15></TD>
  </TR>
</TABLE>
<INPUT TYPE=SUBMIT NAME="mySubmit" VALUE="Submit">
<INPUT TYPE=RESET NAME="myReset" VALUE="Clear">
</FORM>
```

A Login Example : validate.php

```
<?php
require_once ('config.php');
$myUsername = trim($_POST[myUsername]);
$myPassword = trim($_POST[myPassword]);
$query = "SELECT * FROM User WHERE username='$myUsername' AND password = '$myPassword'";
$db = mysql_connect($dbHost, $dbUser, $dbPass)
or die ("DB Error - Could not connect : ". mysql_error());
mysql_select_db($dbName)
or die ("DB Error - Could not select database");
$result = mysql_query($query)
or die ("DB Error - Query failed : ". mysql_error());
while ($row = mysql_fetch_array($result)) {
    if ($row[username]==$myUsername && $row[password]==$myPassword) {
        $myUserType = $row[userType];
        $myCo = $row[co];
        session_start();
        $_SESSION[myUsername] = $myUsername;
        $_SESSION[myPassword] = $myPassword;
        $_SESSION[myUserType] = $myUserType;
        $_SESSION[myCo] = $myCo;
        header("Location:tntsys.php");
        exit();
    }
}
?>
<HTML><BODY>
<?php echo "Invalid username or password! <br/>"; ?>
<BODY></HTML>
```

```
<?php
$dbHost = 'localhost';
$dbUser = 'yilu6';
$dbPass = 'jaysserver';
$dbName = 'cherie';
?>
```

A Login Example : tntsys.php

```
<?php
session_start();
$myUsername = $_SESSION[myUsername];
$myPassword = $_SESSION[myPassword];
$myUserType = $_SESSION[myUserType];
?>
<html>
<body>
<p color=blue>Hello, <?php echo $myUsername; ?> !<p>
</body>
</html>
```

Summary

- **Getting Start:**
 1. What is PHP? HTML-embedded scripting language
 2. What can PHP do? Server-side scripting, Command line scripting, Client-side GUI applications
- **Installation and Configuration:** php.ini
- **Language Reference**
 1. Basic Syntax Ex: <?php echo(".....\n"); ?>
 2. Data Types Ex: string, boolean, integer, float, array, object...etc
 3. Constants and Expression
 - Ex: define("CONSTANT", "Hello world."); \$b = \$a = 5; \$c = \$a++;
 4. Assignment and Comparison
 - Ex: \$b = "Hello " . "There!"; (0 == "a"); ("1" == "01");
 5. Array Operators Ex: \$a = array("a" => "1", "b" => "2");
 6. Control Structures Ex: if, else, foreach, require, include, require_once, include_once
 7. Functions Ex: function foo() { echo "yo"; }
 8. Classes and Objects (PHP 5)
 - Ex: \$foo = new Foo
- **A login example: Database, Session**

Reference

- **PHP website:**
 - <http://www.php.net/>
 - <http://www.zend.com/zend/art/intro.php>
 - <http://conf.php.net/>
- **ACM:** <http://acm.calstatela.edu/>

