PHP

Aditya Lohia

Mayank Sharma

What is PHP?

- PHP stands for **P**HP: **H**ypertext **P**reprocessor
- PHP is a server-side scripting language, like ASP
- PHP scripts are executed on the server
- PHP supports many databases (MySQL, Oracle, PostgreSQL etc.)
- PHP is an open source software
- PHP is free to download and use

- A PHP script always starts with **<?php** and ends with **?>**. A PHP script can be placed anywhere in the document.

- On servers with shorthand-support, you can start a PHP script with <? and end with ?>.

- A PHP file must have a .php extension.

- A PHP file normally contains HTML tags, and some PHP scripting code.

- Each code line in PHP must end with a semicolon.

Example

```
<html>
<body>
        <?php
                echo "Hello World";

                //This is a comment

                /*
                This is
                a comment
                block
                */
        ?>
</body>
</html>
// -> single line comment

/*  */ multi line comment
```

PHP Variables

- PHP variables are used to hold values or expressions.

Rules for PHP variable names:

- Variables in PHP starts with a $ sign, followed by the name of the variable

- The variable name must begin with a letter or the underscore character

- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)

- A variable name should not contain spaces

- Variable names are case sensitive (y and Y are two different variables)

Creating (Declaring) PHP Variables

```php
<?php
$txt="Hello World!";
$x=16;
?>
```

- PHP is a Loosely Typed Language

Arithmetic Operators

Operator	Name	Description	Example	Result
x + y	Addition	Sum of x and y	2 + 2	4
x - y	Subtraction	Difference of x and y	5 - 2	3
x * y	Multiplication	Product of x and y	5 * 2	10
x / y	Division	Quotient of x and y	15 / 5	3
x % y	Modulus	Remainder of x divided by y	5 % 2 10 % 8 10 % 2	1 2 0
a . b	Concatenation	Concatenate two strings	"Hi" . "Ha"	HiHa

Assignment Operators

Assignment	Same as...	Description
x = y	x = y	The left operand gets set to the value of the expression on the right
x += y	x = x + y	Addition
x -= y	x = x - y	Subtraction
x *= y	x = x * y	Multiplication
x /= y	x = x / y	Division
x %= y	x = x % y	Modulus
a .= b	a = a . b	Concatenate two strings

Incrementing/Decrementing Operators

Operator	Name	Description
++ x	Pre-increment	Increments x by one, then returns x
x ++	Post-increment	Returns x, then increments x by one
-- x	Pre-decrement	Decrements x by one, then returns x
x --	Post-decrement	Returns x, then decrements x by one

Comparison Operators

Operator	Name	Description	Example
x == y	Equal	True if x is equal to y	5==8 returns false
x === y	Identical	True if x is equal to y, and they are of same type	5==="5" returns false
x != y	Not equal	True if x is not equal to y	5!=8 returns true
x <> y	Not equal	True if x is not equal to y	5<>8 returns true
x > y	Greater than	True if x is greater than y	5>8 returns false
x < y	Less than	True if x is less than y	5<8 returns true
x >= y	Greater than or equal to	True if x is greater than or equal to y	5>=8 returns false
x <= y	Less than or equal to	True if x is less than or equal to y	5<=8 returns true

Logical Operators

Operator	Name	Description	Example
x && y	And	True if both x and y are true	x=6 y=3 (x < 10 && y > 1) returns true
x || y	Or	True if either or both x and y are true	x=6 y=3 (x==5 || y==5) returns false
! x	Not	True if x is not true	x=6 y=3 !(x==y) returns true

Conditional Statements

- The **if** Statement

if (*condition*)

{*code to be executed if condition is true;*}

- The **if...else** Statement

if (*condition*){ *code to be executed if condition is true;* }
else { *code to be executed if condition is false;*   }

- The **if...elseif....else** Statement

if (*condition*) { *code to be executed if condition is true;* }
elseif (*condition*) { *code to be executed if condition is true;* }
else { *code to be executed if condition is false;* }

The PHP Switch Statement

```php
switch (n)
{
    case label1:
        code to be executed if n=label1;
        break;
    case label2:
        code to be executed if n=label2;
        break;
    default:
        code to be executed if n is different from both
        label1 and label2;
}//end of switch
```

PHP Arrays

- An array is a special variable, which can store multiple values in one single variable.

- **Numeric array** - An array with a numeric index

- **Associative array** - An array where each ID key is associated with a value

- **Multidimensional array** - An array containing one or more arrays

The while and do while Loop

```
while (condition)
  {
        code to be executed;
  }


do
  {
        code to be executed;
  }
while (condition);
```

The for and foreach Loop

```
for (initialization; condition; increment)
  {
        code to be executed;
  }


foreach ($array as $value)
  {
        code to be executed;
  }
```

- function *functionName*([argument_list])
  {

      code to be executed;

  }

- The predefined $_GET variable is used to collect values in a form with method="get"

- When using method="get" in HTML forms, all variable names and values are displayed in the URL.

- The predefined $_POST variable is used to collect values from a form sent with method="post".

- Information sent from a form with the POST method is invisible to others and has no limits on the amount of information to send.

- MySQL is a database.

- The data in MySQL is stored in database objects called tables.

- A table is a collection of related data entries and it consists of columns and rows.

- Queries

A query is a question or a request.

Example :-SELECT LastName FROM Persons

- Before you can access data in a database, you must create a connection to the database.

- In PHP, this is done with the mysql_connect() function.

Syntax

mysql_connect(servername,username,password);

Select Data From a Database Table

```php
SELECT column_name(s) FROM table_name

<?php
$con = mysql_connect("localhost","peter","abc123");
if (!$con)
  {
  die('Could not connect: ' . mysql_error());
  }
mysql_select_db("my_db", $con);
$result = mysql_query("SELECT * FROM Persons");
while($row = mysql_fetch_array($result))
  {
  echo $row['FirstName'] . " " . $row['LastName'];
  echo "<br />";
  }
mysql_close($con);
?>
```

Output
Peter Griffin
Glenn John

Thank You