

PHP (Hypertext Preprocessor)

- PHP code should start with **<?php** and end with **?>**
 - Eg:


```
<?php
.....
?>
```
- To output the text, we have two functions: **echo** and **print**
 - **Eg:** echo "Hello World";
- Each line should be separated with **;** (semicolon)
- **Comments in PHP:**
 - // single line comment
 - /* block
 - comment
 - */
- **Variables:**
 - variables should start with **\$**
 - no need to tell php about the data type of the variable
 - no need to declare a variable before being set
 - variable names are case sensitive
 - Syntax: \$variableName=value;
 - Eg:


```
<?php
$name = "My Name";
$regNo = 1220;

echo "Name=" ■ $name ■ "Number=" ■ $regNo;

?>
```

Note: ■ (Dot) is used to concatenate the strings

➤ String Functions

Function Name	Description	Example	Output
strlen	find the length of a string	echo strlen("Hello World");	11
strpos	search for a string or character within a string	echo strpos("Hello World", "World");	6
ltrim rtrim	Removes whitespaces from the left/right side of a string	echo ltrim(" Hello"); echo rtrim("Hello ");	Hello Hello
trim	Removes spaces from both sides	echo trim(" Hello World ");	Hello World
print	slower than echo	<?php \$name = "Ravi"; \$number = 12; print "Name:\$color"; print " "; print 'No:\$number'; ?>	Name:Ravi No:12
str_replace	replaces some characters with some other characters(2 nd argurement) in a string.	<?php echo str_replace("World", "Ravi", "Hello World"); ?>	Hello Ravi
str_split	splits a string into an array of strings(each string length is by default 1)	1. \$arr=str_split("Ravi Sankar",3); 2. \$arr=str_split("ABC");	1. Rav i S ank ar 2. A B C

strcasecmp()	compares two strings.(case-insensitive)	1. strcmp("ravi","Ravi"); 2. strcmp("A","B");	1. 0 2. <0
strcmp()	compares two strings.(case-sensitive)	1. strcmp("ravi","Ravi"); 2. strcmp("A","B");	1. >0 2. <0
strtok()	splits a string into smaller strings depends on some criteria	<?php \$string = "Ravi, you are good boy"; \$token = strtok(\$string, " ,"); while (\$token != false){ echo "\$token "; \$token = strtok(" ,"); }//while ?>	Ravi you are good boy
strtolower() strtoupper()	Converts a string to lowercase/uppercase.	echo strtolower("Hello"); echo strtoupper("Hello");	Hello HELLO
substr()	returns a part of a string Syntax: substr(string,start,length)	1. substr("Hello World!",6); 2. substr("Hello World",6,5);	1.World! 2. World

➤ Operators

• Arithmetic Operators

+	-	*	/	%	++ and --
---	---	---	---	---	-----------

• Assignment Operators

Operator	Example	Is equal to
=	x=y	x=y
+=	x+=y	x=x+y
-=	x-=y	x=x-y
=	x=y	x=x*y
/=	x/=y	x=x/y
%=	x%=y	x=x%y

• Comparison Operators

==	!=	>	<	>=	<=
----	----	---	---	----	----

• Logical Operators

&&		!
----	--	---

➤ Conditional Statements

▪ if - else if - else

Eg:

```
<?php
    $d=date("D");
    if ($d=="Fri"){
        echo "Have a nice weekend!";
    }else if($d=="Sun"){
```

```
        echo "Have a nice Sunday!";
    }else{
        echo "Have a nice day!";
    }//else
?>
```

- **switch Statement**

Eg:

```
<?php
$x=2;
switch ($x){
    case 1:
        echo "Number 1";
        break;
    case 2:
        echo "Number 2";
        break;
    case 3:
        echo "Number 3";
        break;
    default:
        echo "No number between 1 and 3";
}//switch
?>
```

Output: Number 2

- **for loop**

Eg:

```
<?php
for ($i=1; $i<=3; $i++){
    echo "The number is " . $i . "<br />"
}//for
?>
```

Output:

The number is 1
The number is 2
The number is 3

- **foreach loop**

- is used to loop through arrays.

Eg:

```
<?php
$arr=array("one", "two", "three");
foreach ($arr as $value){
    echo "Value: " . $value . "<br />";
}//foreach
?>
```

Output:

Value:one
Value:two
Value:three

- **while loop**

Eg:

```
<?php
```

```
$i=1;
while($i<=3){
    echo "The number is " . $i . "<br />";
    $i++;
} //while
```

?>

Output:

The number is 1
The number is 2
The number is 3

▪ **do while loop**

Eg:

```
<?php
    $i=0;
    do{
        $i++;
        echo "The number is " . $i . "<br />";
    }while ($i<=3);
?>
```

Output:

The number is 1
The number is 2
The number is 3
The number is 4

➤ **Arrays**

-There are three types of arrays in PHP

1. Numeric array - stores each element with a numeric ID key.

Eg1:

```
$empNames = array("Ravi","Sankar","Kumar");
```

Eg2:

```
$empNames[0] = "Ravi";
$empNames[1] = "Sankari";
$empNames[2] = "Kumar";
```

```
echo empNames[0] . empNames[1] . empNames[2]
```

2. Associative array - An array where each ID key is associated with a value (key-value pairs)

Eg1:

```
$regNumbers = array("Ravi"=>1220, "Sankar"=>1221, "Kumar"=>1222);
```

Eg2:

```
$regNumbers ['Ravi'] = "1220";
$regNumbers ['Sankar'] = "1221i";
$regNumbers ['Kumar'] = "1222r";
```

```
echo "Ravi Reg Number: " . $regNumbers['Ravi']
```

3. Multidimensional array - In a multidimensional array, each element in the main array can also be an array. And each element in the sub-array can be an array, and so on.

Eg:

```
$myCompany = array
(
    "Group1"=>array
    (
        "Ravi",
        "Sankar"
```

```
    ),  
    "Group2"=>array  
    (  
        "Kumar",  
        "John"  
    )  
);  
  
echo myCompany['Group1'][0] . " is in Group1"
```

➤ Functions

Eg1:

```
<?php  
function writeData($myName){  
    echo "***** <br/>";  
    echo $myName . "<br />";  
    echo "***** <br/>";  
}  
//function writeData  
  
echo "My name is ";  
writeData("Ravi");  
  
echo "Your name is ";  
writeData("Sankar");  
?>
```

Eg2:

```
<?php  
function myAddition($first,$second){  
    $total = $first + $second;  
    return $total;  
}  
  
echo "sum of 10 and 20 is = " . myAddition(10,20);  
?>
```

➤ Date() in PHP

Syntax: `date(format,timestamp)` where

format: specifies how to format the date/time

d - The day of the month (01-31)

D - Day name (Mon, Tue ...)

l - A full textual representation of a day(Lower case L)

m - The current month, as a number (01-12)

M - Short month name(Jan)

F - Full month name (January,...)

Y - The current year in four digits

y - two digit representation of year (08)

a - Lowercase am or pm

A - Uppercase AM or PM

h - 12-hour format of an hour (01 to 12)

H - 24-hour format of an hour (00 to 23)

i - Minutes with leading zeros (00 to 59)

s - Seconds, with leading zeros (00 to 59)

T - Timezone setting of the PHP machine (Examples: EST, MDT)

timestamp: is the no.of seconds since 1st,Jan,1970 at 00:00:00 GMT. Default is the current date and time.

Eg: `echo date("Y/m/d");`

Output: 2008/02/21

Adding a timestamp to the existing date.

Syntax: `mktime(hour,minute,second,month,day,year,is_dst)`

Eg:

```
<?php
    $tomorrow = mktime(0,0,0,date("m"),date("d")+1,date("Y"));
    echo "Tomorrow is ".date("Y/m/d", $tomorrow);
?>
```

Output: Tomorrow is 2008/02/22

➤ **PHP in Web applications**

We can send the form data either in get or post method form. Data sent by GET method will be visible in address bar and it has limitation in size of data. POST method doesn't have any limitations.

We can receive the form data either through `$_GET["variableName"]` or `$_POST["variableName"]`

Note: We can use `$_REQUEST["variableName"]` to get the result from form data sent with both the GET and POST methods and also COOKIE method.

Eg1: index.php

```
<html>
  <body>
    <form action="FirstWeb.php" method="post">
      Name: <input type="text" name="txtName">
      RegNo: <input type="text" name="txtRegNo">
      <input type="submit">
    </form>
  </body>
</html>
```

FirstWeb.php

```
<html>
  <body>
    Entered Name: <?php echo $_POST["txtName"]; ?>. <br />
    Entered Reg No: <?php echo $_POST["txtRegNo"]; ?>
  </body>
</html>
```

➤ **mail function in PHP**

Syntax: `mail(to,subject,message,headers,parameters)`

Parameter	Description
to	Required. Specifies the receiver / receivers of the email
subject	Required. Specifies the subject of the email. Note: This parameter cannot contain any newline characters
message	Required. Defines the message to be sent. Each line should be separated with a LF (\n). Lines should not exceed 70 characters. Windows note: If a full stop is found on the beginning of a line in the message, it might be removed. To solve this problem, replace the full stop with a double dot: <pre><?php \$txt = str_replace("\n.", "\n..", \$txt); ?></pre>
headers	Optional. Specifies additional headers, like From, Cc, and Bcc. The additional headers should be separated with a CRLF (\r\n). Note: When sending an email, it must contain a From header. This can be set with this parameter or in the php.ini file.
parameters	Optional. Specifies an additional parameter to the sendmail program (the one

defined in the sendmail_path configuration setting). (i.e. this can be used to set the envelope sender address when using sendmail with the -f sendmail option)

Eg1: Simple Mail

```
<?php
$txtContent = This is first line of text\n This is second line of text";

// Use wordwrap() if lines are longer than 70 characters
$txt = wordwrap($txt,70);
// Send email
mail("somebody@example.com","My subject",$txtContent);
?>
```

Eg2: email with extra headers:

```
<?php
    $to = "somebody@example.com";
    $subject = "My subject";
    $txt = "Hello world!";
    $headers = "From: webmaster@example.com" . "\r\n" .
               "CC: somebodyelse@example.com";

    mail($to,$subject,$txt,$headers);
?>
```

➤ **PHP with Database (MySQL)**

Assume you have a table called EMP having two fields.

Eg:

```
CREATE TABLE EMP(seqno INTEGER AUTO_INCREMENT PRIMARY KEY,eno INT(4),ename
VARCHAR(20));
```

```
INSERT INTO EMP(eno,ename) VALUES(1220,'Ravi');
INSERT INTO EMP(eno,ename) VALUES(1221,'Sankar');
```

```
<html>
  <body>
    <table border="1">
      <tr>
        <td>Eno</td>
        <td>EName</td>
      </tr>
    </table>
  </body>
</html>

<?php
$host="1.2.3.4";//database location; it can be localhost also depends on ur configuration
$user="scott";
$pass="tiger";
$db_name="MySID"; //sid; for oracle, you can get this name from tnsnames.ora

//database connection
$con=mysql_connect($host,$user,$pass);
if(!$con){
    printf("error while connecting to database");
    die('Could not connect: ' . mysql_error());
}
mysql_select_db($db_name,$con);

$sqlstr=mysql_query("SELECT * FROM emp",$con);
while($row=mysql_fetch_array($sqlstr,$con)){
    $eno=$row['eno'];
    $ename=$row['ename'];
    echo "<tr>";
    echo "<td>$eno</td>";
```

```
        echo "<td>$ename</td>";  
        echo "</tr>";  
    }//while(  
  
    mysql_close($con);  
?>  
  
</body>  
</html>
```