

Conditional Statements in PHP

Session 10



Objectives

- ◆ *Explain the use of the if statement*
- ◆ *Explain the use of the switch statement*
- ◆ *Explain the use of the ternary (?) operator*

- ◆ A statement
 - ◆ Is a smallest element of any programming language
 - ◆ Consists of commands given by a programmer to a computer
 - ◆ Can be an individual statement or a group of statements within curly braces
 - ◆ Usually ends with a semicolon
- ◆ PHP script consists of a series of statements which are as follows:
 - ◆ An assignment
 - ◆ A function call
 - ◆ A conditional statement
 - ◆ An empty statement that does nothing

- ◆ Control the flow of a program on execution or skip code based on certain criteria
- ◆ Are of two types:
 - ◆ `if` statement
 - ◆ `switch` statement

- ◆ It is a common control structure
- ◆ It contains an expression called as truth expression
- ◆ The truth expression:
 - ◆ Can be a Boolean, variable, constant or an expression
 - ◆ Evaluates to `true`, `false` or `NULL`
 - ◆ If evaluates to `true`, following statements are executed
 - ◆ If evaluates to `false` or `NULL`, statements are not executed

Syntax

```
if(truth expression)
{
    Statements to be executed;
}
```

Where,

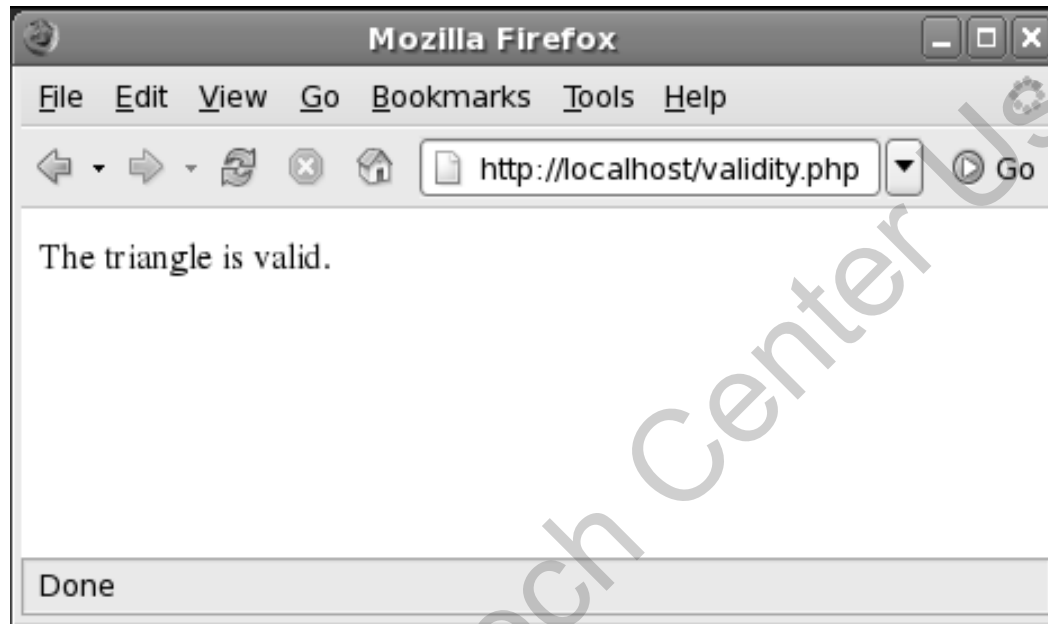
If keyword is followed by truth expression in parenthesis

- ◆ Checking whether a triangle is valid using the `if` statement
 - ◆ Enter the following code in a script named **validity.php**

Snippet

```
<html>
<body>
<?php
$a=60;
$b=60;
$c=60;
if($a+$b+$c == 180)
echo "The triangle is valid.";
?>
</body>
</html>
```

Displays the following output:



If the sum of the degrees $\$a$, $\$b$, and $\$c$ are equal to 180 then the statement following the `if` condition is executed

- ing and displaying the salary of an employee base
and bonus
- bonus.html** – accepts user inputs
- et
- ```
en="salBonus.php" method="GET">

 </td>
type="text" name="sal"></td>

e="submit" value="Submit">
```

## Snippet

```
<html>
<body>
<form action="salBonus.php" method="GET">
 <table>
 <tr>
 <td>Salary </td>
 <td><input type="text" name="sal"></td>
 </tr>
 </table>

 <input type="submit" value="Submit">
</form>
</body>
</html>
```

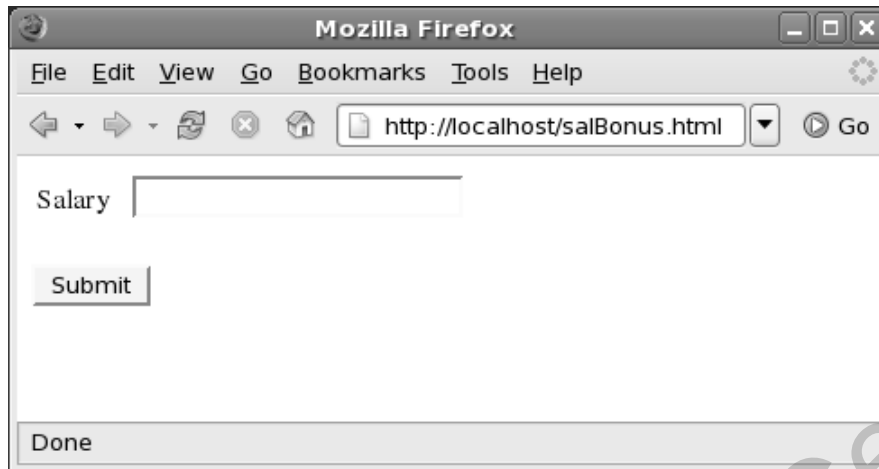


- ◆ Accepting and displaying the salary of an employee based on the salary and bonus
  - ◆ **salBonus.php** – process the salary and calculate the bonus

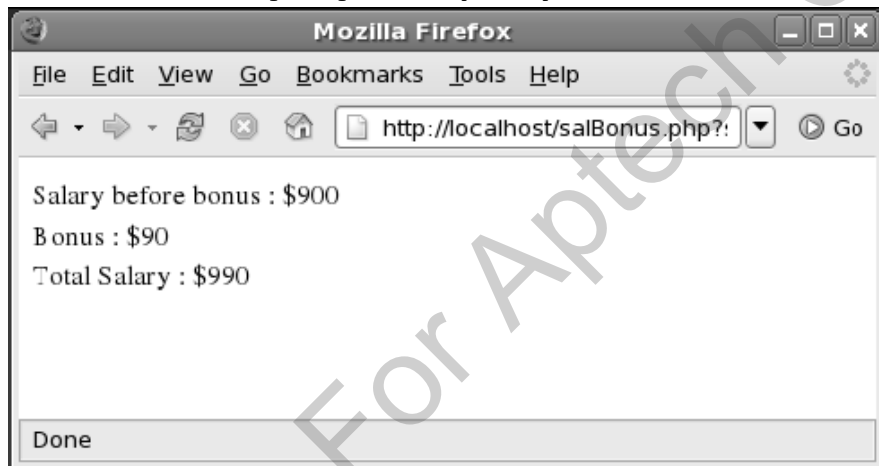
## Snippet

```
<?php
$sal = $_GET['sal'];
echo "Salary before bonus : $";
echo $sal;
echo "
";
if ($sal > 850)
{
 $bonus = $sal * .1;
 echo "Bonus : $$bonus";
 echo "
";
 $sal = $sal + $bonus;
 echo "Total Salary : $$sal";
}
?>
```

**salBonus.html** displays the following output:



**salBonus.php** displays the following output:



The code accepts and displays the bonus at the rate of 10% and total salary when the salary of the employee is greater than \$850.

- ◆ Is used along with `if` statement
- ◆ Is executed when a specified condition is `false`

## Syntax

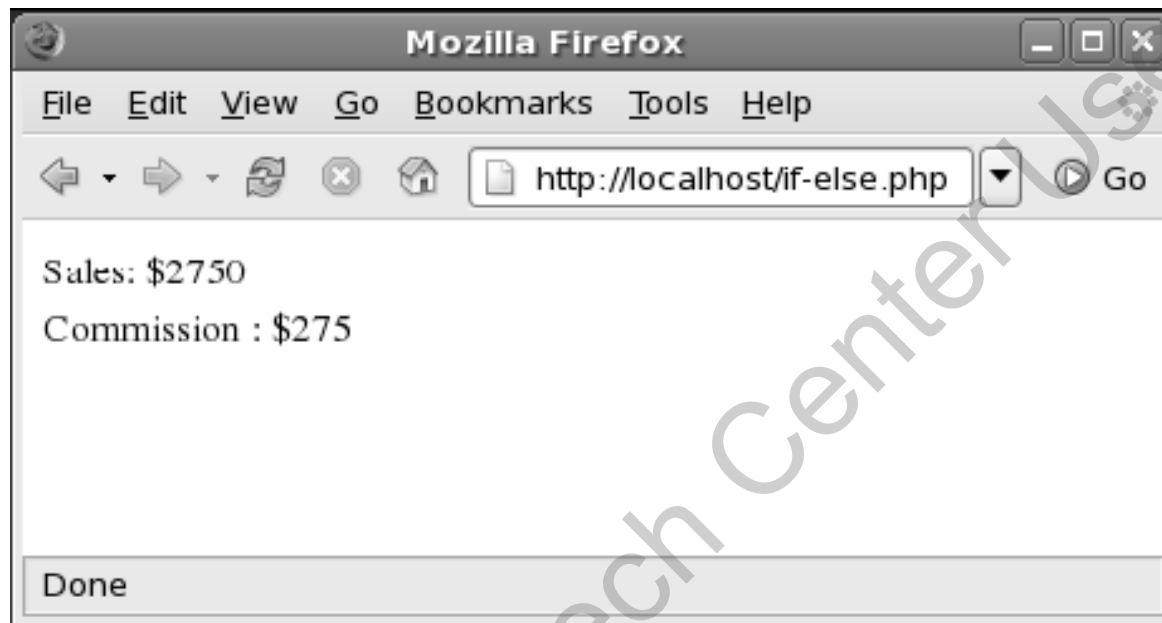
```
if(truth expression)
{
 Statements to be executed if the condition evaluates
 to true;
}
else
{
 Statements to be executed if the condition evaluates
 to false;
}
```

- ◆ Displaying a block of code with an if...else statement
  - ◆ Enter the following code:

## Snippet

```
<?php
$sales = 21050;
if($sales > 2000)
{
 $comm = $sales * .1;
 echo "Sales: $$sales
 Commission : $$comm";
}
else
{
 $comm = $sales * .05;
 echo "Sales: $$sales
 Commission : $$comm";
}
?>
```

Displays the following output:



When the sales amount exceeds \$2000, the program executes the body of the `if` statement and calculates commission at the rate of 10%.

When the sales amount is less than \$2000, the program executes the body of the `else` statement.

- ◆ `else if` clause is:
  - ◆ Used along with `if` statement
  - ◆ An optional clause that allows testing alternative conditions

- ◆ Demonstrating the use of `else if` clause
  - ◆ **saleComm.html** - accepts the sales amount from the user

## Snippet

```
<html>
<body>
<form action="SaleComm.php" method="GET">
<table>
<tr>
<td>Total Sales : </td>
<td><input type="text" name="sal"></td>
</tr>
</table>

<input type="submit" value="Submit">
</form>
</body>
</html>
```

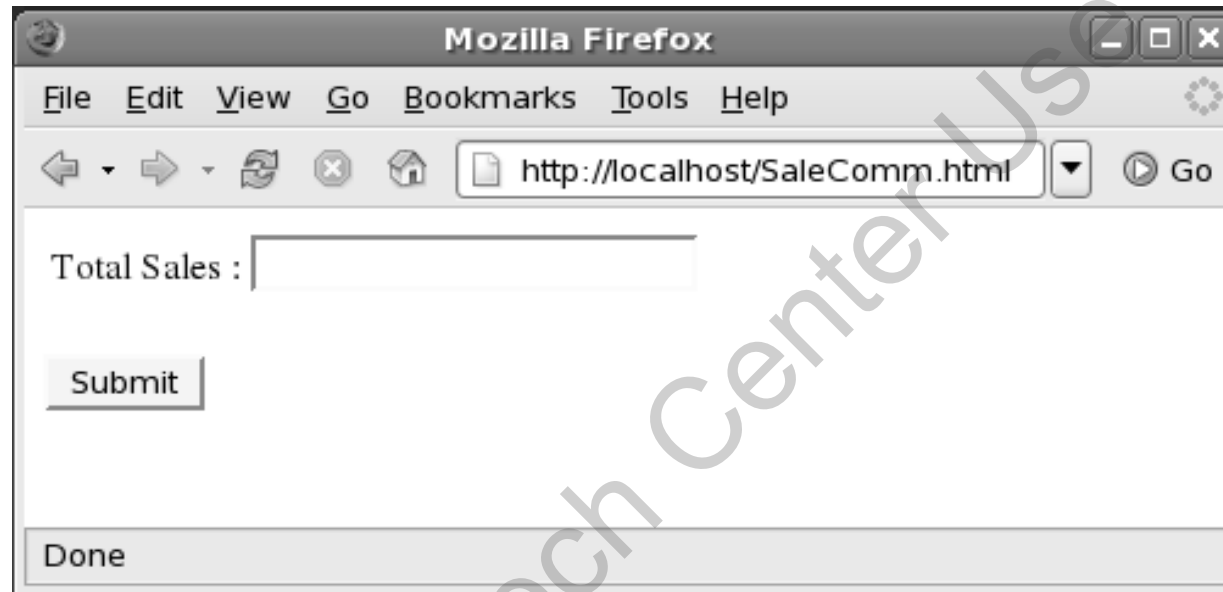
- ◆ **saleComm.php** - processes the sales amount and calculates the commission

## Snippet

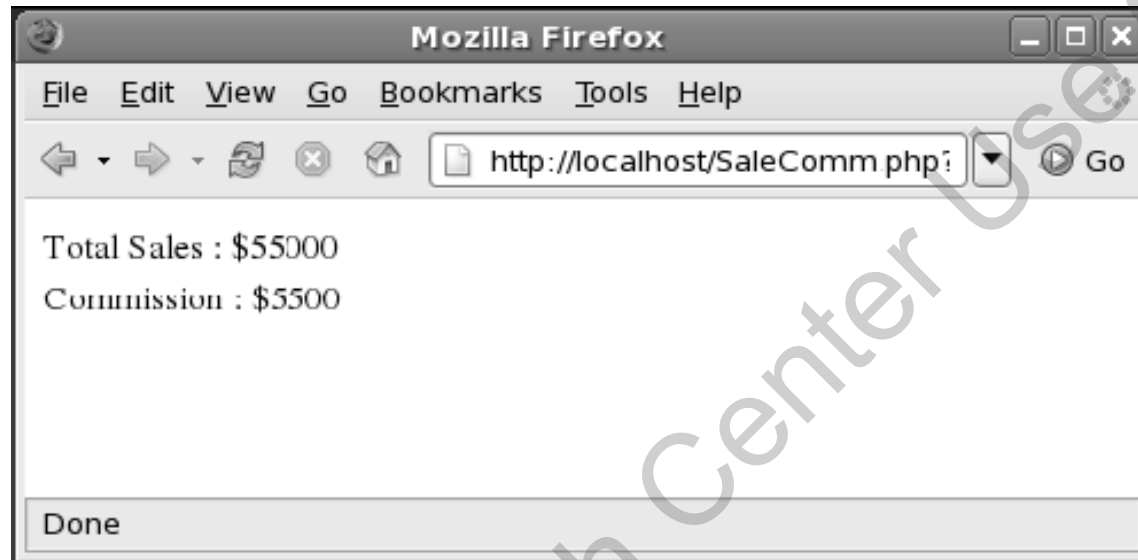
```
<?php
$sal=$_GET['sal'];
echo "Total Sales : $";
echo $sal;
echo "
";
if ($sal > 50000){
 $comm = $sal * .10;
 echo "Commission : $$comm";
 echo "
"; }
else if ($sal > 20000 and $sal <= 50000){
 $comm = $sal * .010;
 echo "Commission : $$comm";
 echo "
"; }
else if ($sal < 20000){
 $comm = $sal * .05;
 echo "Commission: $$comm";
 echo "
";
}
?>
```



**saleComm.html** displays the following output:



**saleComm.php** displays the following output:



In the code, commission is calculated according to the sales amount the user enters.

- ◆ An `if` statement within an `if` statement or an `else` statement is known as nested `if` statement

For Aptech Center Use Only

- ◆ Calculating the electricity charges based on the units of electricity consumed
  - ◆ **elecBill.html** - accepts the number of units consumed

## Snippet

```
<html>
<body>
<form action="elecBill.php" method="GET">
<table>
<tr>
<td>Electricity Units Consumed : </td>
<td><input type="text" name="units"></td>
</tr>
</table>

<input type="submit" value="Submit">
</form>
</body>
</html>
```

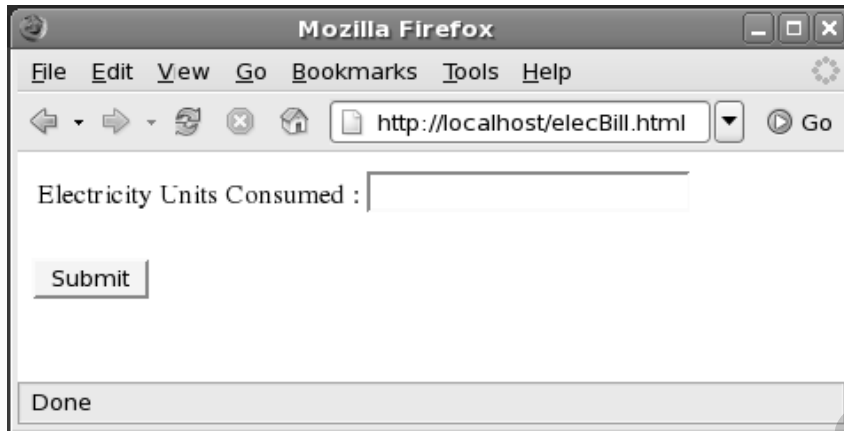
- ◆ Calculating the electricity charges based on the units of electricity consumed
  - ◆ **elecBill.php** - calculates total electricity bill

## Snippet

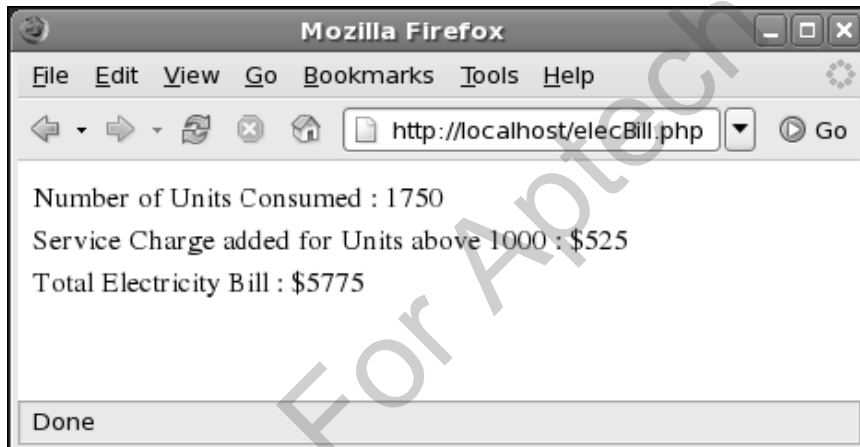
```
<?php
$units=$_GET['units'];
echo "Number of Units Consumed : ";
echo $units;
echo "
";
if ($units > 1000)
{
 $rate = $units * 3;
 $service = $rate * .1;
 echo "Service Charge added for Units above 1000 : $$service";
 echo "
";
 $totalbill = $rate + $service;
 echo "Total Electricity Bill : $$totalbill";
}
```

```
else
{
 if ($units > 500 and $units <= 1000)
 {
 $rate = $units * 2;
 echo "Total Electricity Bill : $$rate";
 }
else
{
 $rate = $units * 1.5;
 echo "Total Electricity Bill : $$rate";
}
}
?>
```

**elecBill.html** – displays the following output:



**elecBill.php** – displays the following output:



In the code, if the user enters 1500 as input, PHP code in Code first calculates the rate and stores the value in the `$rate` variable.

To calculate the service charge to be levied on the electricity bill, Code **elecBill.php** calculates the service charge at the rate of 10% and stores the value in the `$service` variable.

It then stores the total amount in the `$totalbill` variable.

- ◆ Used as an alternative to a lengthy `if . . . else` construct
- ◆ Consists of an expression that is compared to all possible case expressions listed in its body
- ◆ On finding a match, it executes the block of code ignoring any further case lines
- ◆ Uses a `break` statement to halt the execution of the switch statement and transfer the control to the code following switch



## Syntax

```
switch(variable){
 case value1:
Code executes if condition equals value1
 break;
 case value2:
Code executes if condition equals value2
 break;
 .
 .
 .
 default:
Code executes if the variable does not matches any
specified value
}
```

## Where,

- ◆ case keyword is followed by a case constant
- ◆ default is a special case executed when none of the case constants is matching

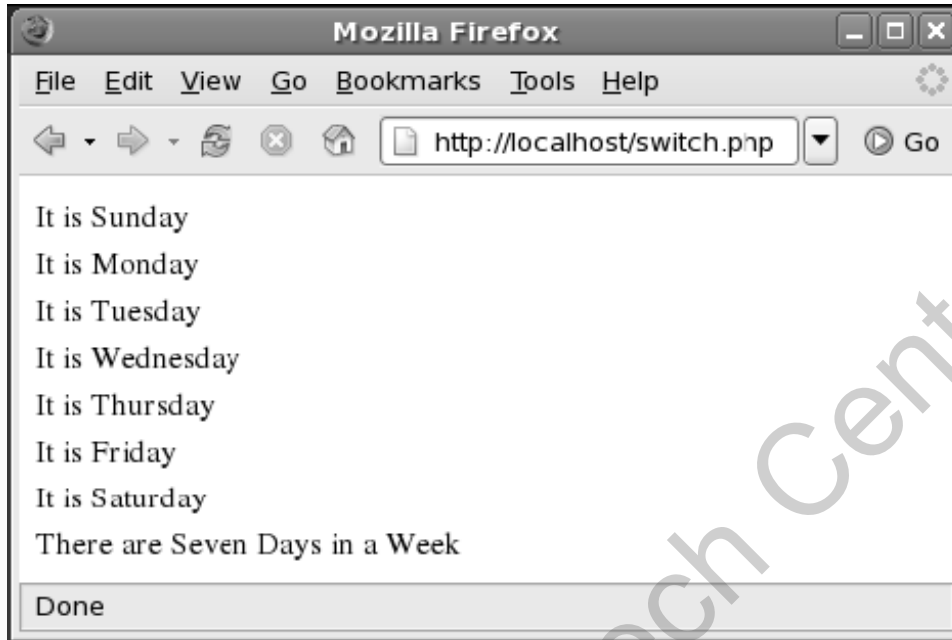
- ◆ Displaying a switch statement without any `break` statement
  - ◆ Enter the code in a script named **switch.php**

## Snippet

```
<?php
$day = 1;
switch ($day)
{
 case 1:
 echo "It is Sunday";
 echo "
";
 case 2:
 echo "It is Monday";
 echo "
";
```

```
case 3:
echo "It is Tuesday";
echo "
";
case 4:
echo "It is Wednesday";
echo "
";
case 5:
echo "It is Thursday";
echo "
";
case 6:
echo "It is Friday";
echo "
";
case 10:
echo "It is Saturday";
echo "
";
default:
echo "There are Seven Days in a Week";
echo "
";
}
?>
```

Displays the following output:



In the code, the weekday is 1, the program displays the message related to `case 1`.

Due to the absence of a `break` statement, it also displays the messages related to the subsequent cases until it reaches the end of the switch statement.

- ◆ Displaying a switch statement without any break statement
  - ◆ Enter the code in a script named **break.php**

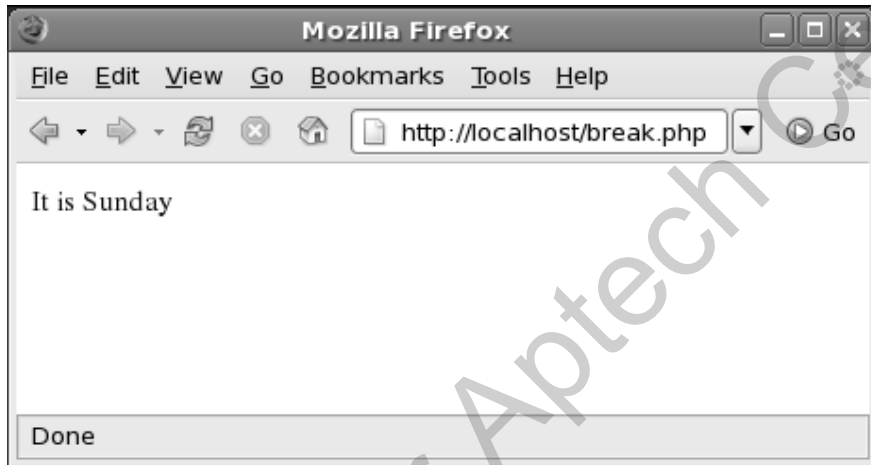
## Snippet

```
<?php
$day = 1;
switch ($day)
{
 case 1:
 echo "It is Sunday";
 echo "
";
 break;
 case 2:
 echo "It is Monday";
 echo "
";
 Break;
```

```
case 3:
echo "It is Tuesday";
echo "
";
break;
case 4:
echo "It is Wednesday";
echo "
";
break;
case 5:
echo "It is Thursday";
echo "
";
Break;
case 6:
echo "It is Friday";
echo "
";
break;
case 10:
echo "It is Saturday";
echo "
";
Break;
```

```
default:
echo "There are Seven Days in a Week";
echo "
";
break;
}
?>
```

Displays the following output:



In the code, the weekday is 1, the program displays only the message related to case 1.

If the value assigned is any other value apart from numbers 1 to 10, the program displays 'There are Seven Days in a Week.'

- ◆ It is also known as a conditional operator
- ◆ It simplifies complex conditions into one-line statements
- ◆ It is considered as an alternative for the `if...else` statement

## Syntax

```
truth_exp ? expr1 : expr2;
```

Where,

- ◆ `truth_expr` is evaluated and if it is `true`, `expr1` is evaluated
- ◆ If it is `false`, `expr2` is evaluated

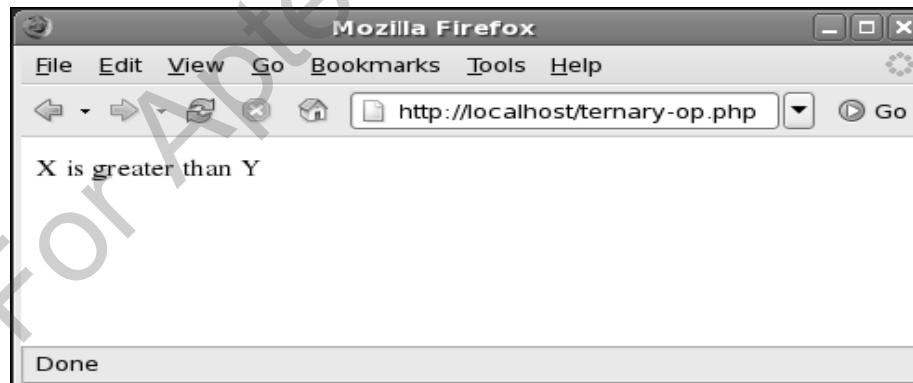


- ◆ Displaying the output using the ternary operator
  - ◆ Enter the code as shown in code in the script named **ternary-op.php**

## Snippet

```
<?php
$x = 100;
$y = 50;
$disp = ($x > $y) ? "X is greater than Y" : "Y is
greater than X";
echo $disp;
?>
```

Displays the following output:



- ◆ Conditional statements execute a set of statements only when a specified condition is satisfied and modify the order of flow in a program
- ◆ The if statement executes a block of code only when the specified condition is true
- ◆ In a nested if statement, you can include an if statement within another if statement or an else statement
- ◆ A switch...case statement checks a single variable against multiple values and executes a block of code based on the value it matches

- ◆ The break statement is used to transfer the control to the statements following the switch...case statement
- ◆ The default statement is used when none of the case statements matches the value of the switch variable
- ◆ Ternary operator is also known as conditional operator. It simplifies complex conditions into one-line statements