## TY. B. Tech. (Sem-II)

**CS3215 : Web Technology**

**TITLE : - Write a program to calculate Electricity bill in PHP (a)You need to write a PHP program to calculate electricity bill using if-else conditions. (b) Conditions For first 50 units – Rs. 3.50/unit For next 100 units – Rs. 4.00/unit For next 100 units – Rs. 5.20/unit for units above 250 – Rs. 6.50/unit You can use conditional statements.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Roll. No.** | **Gr. No.** | **Div** | **Batch** | **Name** |
| **62** | **12120238** | **D** | **1** | **Tanaya Naik** |

**Theory:**

**Learning PHP Script**

The primary difference between a scripting language and a programming language is in their execution – programming languages use a compiler to convert the high-level programming languages into machine language, on the other hand, scripting languages use an interpreter.

**Is PHP an alternative to JavaScript?**

PHP is server-side scripting language whereas JavaScript is a client-side scripting language. PHP doesn't execute within browser whereas JavaScript executes within browser. PHP supports database whereas JavaScript doesn't support databases. PHP accepts both upper case and lower case variables while JavaScript doesn't.

**What is PHP?**

PHP is a open source server side scripting language, and a powerful tool for making dynamic and interactive Web pages.

PHP is an acronym for "PHP: Hypertext Preprocessor"

PHP 7 is the latest stable release.

PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.

PHP scripts are executed on the server

PHP is free to download and use

**What is a PHP File?**

PHP files can contain text, HTML, CSS, JavaScript, and PHP code

PHP code is executed on the server, and the result is returned to the browser as plain HTML

PHP files have extension ".php"

**What Can PHP Do?**

PHP can generate dynamic page content

PHP can create, open, read, write, delete, and close files on the server

PHP can collect form data

PHP can send and receive cookies

PHP can add, delete, and modify data in your database

PHP can be used to control user-access

PHP can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even Flash movies. You can also output any text, such as XHTML and XML.

**Why PHP?**

PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)

PHP is compatible with almost all servers used today (Apache, IIS, etc.)

PHP supports a wide range of databases

PHP is free. Download it from the official PHP resource: www.php.net

PHP is easy to learn and runs efficiently on the server side

**Basic PHP Syntax**

A PHP script can be placed anywhere in the document.

A PHP script starts with <?php and ends with ?>:

<?php

// PHP code goes here

?>

The default file extension for PHP files is ".php".

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

**PHP Case Sensitivity**

In PHP, keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are not case-sensitive.

**Note:** However; all variable names are case-sensitive!

$color, $COLOR, and $coLOR are treated as three different variables.

# PHP Comments

<?php  
// This is a single-line comment  
  
# This is also a single-line comment  
?>

<?php  
/\*  
This is a multiple-lines comment block  
that spans over multiple lines  
\*/  
?>

**Displaying Output –** echo/echo() and print/print() statements are often used to output data to the screen.

The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions. echo can take multiple parameters while print can take one argument. echo is marginally faster than print

<?php

echo "<h2>PHP is Fun!</h2>";

echo "Hello world!<br>";

echo "I'm about to learn PHP!<br>";

echo "This ", "string ", "was ", "made ", "with multiple parameters."; ?>

But

<?php

print "<h2>PHP is Fun!</h2>";

print "Hello world!<br>";

print "I'm about to learn PHP!<br>";

print "This ", "string ", "was ", "made ", "with multiple parameters.";

?> gives error

<?php  
$txt1 = "Learn PHP";  
$txt2 = "W3Schools.com";  
$x = 5;  
$y = 4;  
  
print "<h2>" . $txt1 . "</h2>";  
print "Study PHP at " . $txt2 . "<br>";  
print $x + $y;  
?>

**Creating (Declaring) PHP Variables**

In PHP, a variable starts with the $ sign, followed by the name of the variable:

<?php  
$txt = "Hello world!";  
$x = 5;  
$y = 10.5;  
?>

**PHP Variables Scope**

PHP has three different variable scopes:

* local
* global
* static

A variable declared outside a function has a GLOBAL SCOPE and **can only be accessed outside a function:**

<?php  
$x = 5; // global scope  
  
function myTest() {

echo "<p>Variable x inside function is: $x</p>"; // will not dispaly  
}

myTest();  
  
echo "<p>Variable x outside function is: $x</p>";  
?>

A variable declared **within** a function has a LOCAL SCOPE and can only be accessed within that function:

<?php  
function myTest() {  
  $x = 5; // local scope  
  echo "<p>Variable x inside function is: $x</p>";  
}  
myTest();  
  
// using x outside the function will generate an error  
echo "<p>Variable x outside function is: $x</p>";  
?>

**PHP The global Keyword**

The global keyword is used to access a global variable from within a function.

<?php  
$x = 5;  
$y = 10;  
  
function myTest() {  
  global $x, $y;  
  $y = $x + $y;  
}  
  
myTest();  
echo $y; // outputs 15  
?>

**PHP also stores all global variables in an array called $GLOBALS[index].** The index holds the name of the variable. This array is also accessible from within functions and can be used to update global variables directly.

<?php  
$x = 5;  
$y = 10;  
  
function myTest() {  
  $GLOBALS['y'] = $GLOBALS['x'] + $GLOBALS['y'];  
}  
  
myTest();  
echo $y; // outputs 15  
?>

**PHP The static Keyword**

Normally, when a function is executed, all of its variables are deleted. However, sometimes we want a local variable NOT to be deleted. We need it for a further job.

<?php  
function myTest() {  
  static $x = 0;  
  echo $x;  
  $x++;  
}  
  
myTest();  
myTest();  
myTest();  
?>

**Code:**

Index.php

<html>

<head>

    <title>Electricity Bill Calculation</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <center><h1>Electricity Bill Calculation</h1></center>

    <form action="" method="POST">

        <label for="units">Enter units consumed:</label>

        <input type="number" name="units" id="units" min="0" max="1000" required>

        <input type="submit" value="Calculate">

    </form>

</body>

</html>

<?php

if(isset($\_POST['units'])){

    $units = $\_POST['units'];

}

$bill = 0;

if(isset($units))

{

    if ($units <= 50) {

        $bill = $units \* 3.50;

    } elseif ($units <= 150) {

        $bill = 50 \* 3.50 + ($units - 50) \* 4.00;

    } elseif ($units <= 250) {

        $bill = 50 \* 3.50 + 100 \* 4.00 + ($units - 150) \* 5.20;

    } else {

        $bill = 50 \* 3.50 + 100 \* 4.00 + 100 \* 5.20 + ($units - 250) \* 6.50;

    }

    echo "<span>"."Total units consumed: " . $units . "<br>";

    echo "Electricity bill: Rs. " . $bill . "</span>";

}

?>

**Style.css**

@import url('https://fonts.googleapis.com/css2?family=Source+Code+Pro:wght@300&display=swap');

@import url('https://fonts.googleapis.com/css2?family=Poppins&display=swap');

\*{

    margin: 0;

    padding: 0;

    box-sizing: border-box;

    font-family: 'Poppins', sans-serif;

}

body{

    background: url("bg.jpg");

    background-size: cover;

    z-index: -1;

}

h1{

    margin-top: 50px;

}

label{

    font-size: 24px;

    font-weight: bold;

    color: #333;

}

#units{

    border-radius: 20px;

}

form {

    display: flex;

    flex-direction: column;

    align-items: center;

    margin-top: 50px;

}

input[type=number] {

    padding: 10px;

    font-size: 16px;

    border: 2px solid #ccc;

    border-radius: 5px;

    margin-bottom: 20px;

    text-align: center;

    width: 200px;

}

input[type=submit] {

    background-color: #4CAF50;

    color: white;

    font-size: 24px;

    padding: 10px 20px;

    border: none;

    border-radius: 15px;

    cursor: pointer;

    transition: background-color 0.3s ease;

}

input[type=submit]:hover {

    background-color: #3e8e41;

}

span {

    margin-top: 20px;

    display: flex;

    justify-content: center;

    align-items: center;

    font-size: 24px;

    font-weight: bold;

    color: #333;

}

**Output**



