**MODULE – 3(Core PHP)**

• **Write a PHP program to enter marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer, calculate percentage and grade by if else write a PHP program for find „Thursday‟ in week using switch Function.**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<title>Enter marks of subjects And Week </title>

</head>

<body>

<h2>enter marks of five subjects</h2>

<form action="" method="post">

<table>

<tr>

<td>Enter Mark of Physics :</td>

<td><input type ="text" name="physics"><br></td>

</tr>

<tr>

<td>Enter Mark of Chemistry :</td>

<td><input type ="text" name="chemistry"><br></td>

</tr>

<tr>

<td>Enter Mark of Biology :</td>

<td><input type ="text" name="biology"><br></td>

</tr>

<tr>

<td>Enter Mark of Mathematics :</td>

<td><input type ="text" name="mathematics"><br></td>

</tr>

<tr>

<td> Enter Mark of Computer :</td>

<td><input type ="text" name="computer"><br></td>

</table>

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

</form>

<?php

if(isset($\_POST['submit']))

{

$physics=$\_POST('physics');

$chemistry=$\_POST('chemistry');

$biology=$\_POST('biology');

$mathematics=$\_POST('mathematics');

$computer=$\_POST('computer');

$total=$physics+$chemistry+$biology+$mathematics+$computer;

$per=($total/500)\*100;

if($per>=75)

{

$grade="distinction";

}

else if($per>=60 && $per<75 )

{

$grade="First Class";

}

else if($per>=50 && $per<60 )

{

$grade="Second Class";

}

else

{

$grade="Pass Class";

}

echo "Physics :- $physics<br>";

echo "Chemistry :- $chemistry<br>";

echo "biology :- $biology<br>";

echo "mathematics :- $mathematics<br>";

echo "computer - $computer<br>";

echo "Total- $total<br>";

echo "Percentage - $per<br>";

echo "Grade - $grade<br>";

}

?>

<br><br>

// write a PHP program for find „Thursday‟ in week using switch Function.

<?php

$currentDay = date('N');

switch ($currentDay)

{

case 1:

$day = 'Monday';

break;

case 2:

$day = 'Tuesday';

break;

case 3:

$day = 'Wednesday';

break;

case 4:

$day = 'Thursday';

break;

case 5:

$day = 'Friday';

break;

case 6:

$day = 'Saturday';

break;

case 7:

$day = 'Sunday';

break;

default:

$day = 'Invalid day';

}

echo "Today is $day\n";

?>

</body>

</html>

• **Write a PHP program to check Leap years between 1901 to 2016 Using nested if.**

<html>

<head>

<title>Leap year</title>

</head>

<body align="center">

<h3><u>PHP program to check Leap years</u></h3>

<?php

$year1=1901;

$year2=2016;

echo "Leap years between 1901 to 2016 <br>";

for($i=$year1;$i<=$year2;$i++)

{

if($i % 2==0)

{

echo "$i <br>";

}

}

?>

</body>

</html>

**• Write a PHP program to find the largest of three numbers using ternary Operator.**

=><!DOCTYPE html>

<html>

<head>

    <meta charset="utf-8" />

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <title>Largest Number</title>

</head>

<body>

<h3><u><b><i>Largest of Three Number</i></b></u></h3>

<form action="" method="POST">

<table>

<tr>

<td>Enter First Number </td>

<td><input type="number" name="num1" required></td>

</tr>

<tr>

<td>Enter Second Number </td>

<td><input type="number" name="num2" required></td>

</tr>

<tr>

<td>Enter Third Number </td>

<td><input type="number" name="num3" required></td>

</tr>

</table><br>

<input type="submit" value="Submit" name="submit"><br><br>

</form>

</body>

</html>

<?php

function larg($n1,$n2,$n3)

{

    if($n1>$n2 && $n1>$n3)

    {

        return $n1;

    }

    else if($n2>$n1 && $n2>$n3)

    {

        return $n2;

    }

    else if($n3>$n1 && $n3>$n2)

    {

        return $n3;

    }

    else

    {

        return 0;

    }

}

?>

<?php

    $sum=isset($\_POST['num1']) && isset($\_POST['num2']) && isset($\_POST['num3']);

    if($sum)

    {

        $num=larg($\_POST['num1'],$\_POST['num2'],$\_POST['num3']);

        if($num==0)

        {

            echo "error";

        }

        else

        {

            echo "The Largest Number is : ".$num."" ;

        }

    }

?>

**• Write a program in PHP to print Fibonacci series. 0, 1, 1, 2, 3, 5, 8, 13, 21, o 34**

=><DOCTYPE html>

<html>

    <head>

        <title> Fibonacci Series </title>

    </head>

    <body>

        <h1> Fibonacci Series </h1>

        <?php

            $num=0;

            $n1=0;

            $n2=1;

            echo "<h3> Fibonacci Series for First 12 number :</h3>";

            echo "\n";

            echo $n1. ','.$n2.',';

            while ($num<10)

            {

                $n3=$n2+$n1;

                echo $n3.',';

                $n1=$n2;

                $n2=$n3;

                $num=$num+1;

            }

        ?>

    </body>

</html>

**• Write a program to find whether a number is Armstrong or not Write a program to print the below format :**

**5 9**

**2610**

**3711**

**4812**

=><!DOCTYPE html>

<html>

<head>

<title>Armstrong Number</title>

</head>

<body align="center">

<table align="center">

<form method="post" action="">

<tr>

    <td> Number:</td>

    <td><input type="text" name="number" /></td>

</tr>

<tr>

    <td>

        <input type="submit" name="submit" value="Check" />

    </td>

</tr>

</form>

</table>

<?php

   if(isset($\_POST['number']) && $\_POST['number']!='')

   {

        $number = $\_POST[ 'number' ]; // get the number entered by user $temp = $number;

        $sum = 0;

        while($sum != 0 )

        {

            $remainder = $temp % 10; //find reminder $sum = $sum + ( $remainder \* $remainder \* $remainder );

            $temp = $temp / 10;

        }

        if( $number == $sum )

        {

            echo "$number is an Armstrong Number";

        }

        else

        {

            echo "$number is not an Armstrong Number";

        }

    }

?>

</body>

</html>

**• Write a program for this Pattern:**

**\*\*\*\*\***

**\***

**\***

**\***

**\*\*\*\*\***

=><!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>Program for Pattern</title>

</head>

<body>

<?php

$Size=5;

for($i=0; $i<$Size; $i++)

{

for($j=0; $j<$Size; $j++)

{

if($i===0 || $i===$Size-1)

{

echo "\*";

}

else

{

if($j===0 || $j===$Size)

{

echo "\*";

}

else

{

echo "&nbsp; &nbsp;";

}

}

}

echo "<br>";

}

?>

</body>

</html>

**• What will be the values of $a and $b after the code below is executed? Explain your answer. $a = '1'; $b = &$a; $b = "2$b";**

=>Both $a and $b will be equal to the string “21” after then above code is executed.

**• How can you tell if a number is even or odd without using any Condition or loop?**

=><html>

    <head>

        <title>Even Or Odd Number</title>

    </head>

<body>

<form method="post">

Enter a number:

<input type="number" name="number">

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

</form>

</body>

</html>

<?php

if($\_POST){

    $num = $\_POST['number']; //divide entered number by 2

    if(($num % 2) == 0)                     //if the reminder is 0 then the number is even otherwise the number is odd

    {

        echo "$num is an Even number";

    }

    else

    {

        echo "$num is Odd number";

    }

}

?>

• **How can you declare the array (all type) in PHP? Explain with example Covert a JSON string to array.**

=> **Declare the array:**

1. Indexed Array:Arrays with numeric index. An example:-

$indexedArray = ["apple", "banana", "cherry"];

2. Associative Array: Arrays with named keys. An example:-

$assocArray = ["name" => "John", "age" => 30, "city" => "New York"];

3. Multidimensional Array: Arrays containing one or more arrays.

An example:-

$multiArray = array(

array("apple", "banana", "cherry"),

array("John", "Doe", 25),

array("city" => "New York", "country" => "USA")

);

**Covert a JSON string to array:**

json\_decode() function to convert a JSON string to an array in PHP.An example:-

$jsonString = '{"name":"John","age":30,"city":"New York"}';

$arrayFromJson = json\_decode($jsonString, true); // Convert JSON string to an associative array

echo $arrayFromJson['name']; // Output: John // Accessing elements in the array

echo $arrayFromJson['age']; // Output: 30

echo $arrayFromJson['city']; // Output: New York

**Note:-**  JSON decoding might fail if the provided JSON string is malformed. In that case, you can use json\_last\_error() and json\_last\_error\_msg() to get information about the error.

• **Write program to remove duplicate values from array**

=> <?php

$array\_value=array(10,20,30,40);

$array\_count=count($array\_value);

$duplicate\_array=array();

for($i=0;$i<$array\_count;$i++)

{

$duplicate\_array[$array\_value[$i]]=$array\_value[$i];

}

echo "<pre>";

print\_r($duplicate\_array);

?>

**• Get random values from array.**

=>Random values from an array in PHP, you can use the array\_rand() function.

an example:

<?php

$myArray = array("apple", "banana", "cherry", "date", "fig"); // Your array

$randomKey = array\_rand($myArray); // Get a random key from the array

$randomValue = $myArray[$randomKey]; // Use the random key to get the

corresponding value

echo "Random value from the array: " . $randomValue . "\n"; // Display the result

?>

In this example, array\_rand() is used to obtain a random key from the array, and then that key is used to access the corresponding value in the original array. The resulting $randomValue will be a random element from your original array.

If you want to get multiple random values from the array, you can specify the number of random keys you want to retrieve as a second parameter to array\_rand():

<?php

$numberOfRandomValues = 3;

$randomKeys = array\_rand($myArray, $numberOfRandomValues);

// $randomKeys is an array containing the random keys

// To get the corresponding values

$randomValues = array\_intersect\_key($myArray, array\_flip($randomKeys));

In this case, $randomValues will be an array containing the corresponding values for the randomly selected keys.

**• Write a PHP script which decodes the following JSON string.**

=>To decode a JSON string in PHP, you can use the json\_decode() function. Here's an example:

Suppose you have the following JSON string:

json

Copy code

$jsonString = '{"name":"John","age":30,"city":"New York"}';

You can decode it using the following PHP script:

php

Copy code

<?php

$jsonString = '{"name":"John","age":30,"city":"New York"}'; // JSON string

$decodedArray = json\_decode($jsonString, true); // Decode JSON string

if ($decodedArray === null) // Check if decoding was successful

{

echo "Error decoding JSON.";

}

else

{

echo "Decoded Array:\n"; // Display the decoded array

print\_r($decodedArray);

}

?>

In this script, the json\_decode() function is used to convert the JSON string into a PHP associative array. The second parameter true is used to specify that the result should be an associative array. The print\_r() function is then used to display the decoded array.

If the JSON string is valid, the script will output the decoded array. If there is an error in decoding, it will display an error message.

**• Use a for loop to total the contents of an integer array called numbers which has five elements. Store the result in an integer called total.**

=>

• **Declare a Multi Dimensioned array of floats called balances having Three rows and five columns.**

=> You can declare a multi-dimensional array of floats called balances with three rows and five columns in PHP like this:

<?php

$balances = [ // Declare a multi-dimensional array of floats

[1.2, 2.3, 3.4, 4.5, 5.6],

[6.7, 7.8, 8.9, 9.1, 10.2],

[11.3, 12.4, 13.5, 14.6, 15.7],

];

echo "Multi-Dimensional Array (balances):\n"; // Display the multi-dimensional array

for ($row = 0; $row < count($balances); $row++) {

for ($col = 0; $col < count($balances[$row]); $col++) {

echo $balances[$row][$col] . "\t";

}

echo "\n";

}

?>

In this example:

The balances array is declared with three rows and five columns.

Values in the array are floats.

The script uses nested for loops to iterate through the rows and columns, displaying the elements of the multi-dimensional array.

Feel free to modify the array values according to your requirements.

?>