THE CODE JOURNAL

SAMPLE 2020

VISIT : WWW.THECODEJOURNAL.IN
Email - query@thecodejournal.in

PHP Sample Paper Fully Solved 2020

Instructions:

(1)All questions are compulsory.

(2) Illustrate your answers with neat sketches wherever necessary.

(3) Figures to the right indicate full marks.

(4) Assume suitable data if necessary.

(5) Preferably, write the answers in sequential order.

Q.1) Attempt any FIVE of the following. **(2 x 5 = 10 marks)**

a) List any four advantages of PHP?

Answer: The advantages of PHP languages are as follows -:

1. Open Source:

PHP is open-source and free of cost, which helps developers to install it quickly and readily available for use. All the features and tools will be provided to the developer for that framework easily.

2. Platform Independent:

PHP is mainly supported by all the operating systems like Windows, Unix, Linux etc. The PHP based developed web applications can be easily run on any platform. It can be integrated with other programming language and database easily and there is no requirement of re-development. It helps in saving a lot of effort and cost.

3. Simple and Easy:

This advantage of PHP is simple and easy to learn and code. It is mainly organized code and clean, which helps the new developers also. The command functions of PHP can easily learn and understood. The one who knows any programming language can easily work on PHP. It is simple to learn, as its learning curve is not large. The syntax is simple and flexible to use.

4. Database: PHP is easily connected with the database and make the connection securely with databases. It has a built-in module that is used to connect to the database easily.

b) State the use of str_word_count along with its syntax.

Answers: The str_word_count() function counts the number of words in a string.

Syntax -:

```
str_word_count(string,return,char)
```

Parameter Values

Parameter	Description
string	Required. Specifies the string to check
return	Optional. Specifies the return value of the str_word_count() function. Possible values: • 0 - Default. Returns the number of words found • 1 - Returns an array with the words from the string • 2 - Returns an array where the key is the position of the word in the string, and value is the actual word
char	Optional. Specifies special characters to be considered as words.

PHP Sample Paper Fully Solved 2020

c) Define serialization.

Answer: The serialize() converts a storable representation of a value.

A serialize data means a sequence of bits so that it can be stored in a file, a memory buffer, or transmitted across a network connection link.

Syntax:

serialize(value1)

Parameter:

Name	Description	Required / Optional	Type
value1	The value to be serialized	Required	Mixed*

*Mixed: Mixed indicates that a parameter may accept multiple (but not necessarily all) types.

Return value:

A string.

Value Type: String.

Example:

```
<?php
$serialized_data = serialize(array('Math', 'Language', 'Science'));
echo  $serialized_data . '<br>';
?>
```

Output:

a:3:{i:0;s:4:"Math";i:1;s:8:"Language";i:2;s:7:"Science";}

d) Write the syntax for creating Cookie.

Answer: A cookie is often used to identify a user. A cookie is a small file that the server embeds on the user's computer. Each time the same computer requests a page with a browser, it will send the cookie too. With PHP, you can both create and retrieve cookie values.

A cookie is created with the `setcookie()` function.

Syntax

```
setcookie(name, value, expire, path, domain, secure, httponly);
```

Example -:

```
<?php
$cookie_name = "user";
$cookie_value = "John Doe";
setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/"); // 86400 = 1 day
?>
<html>
<body>

<?php
if(!isset($_COOKIE[$cookie_name])) {
   echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
   echo "Cookie '" . $cookie_name . "' is set!<br>";
   echo "Value is: " . $_COOKIE[$cookie_name];
}
?>

</body>
</html>
```

PHP Sample Paper Fully Solved 2020

e) Write syntax of Connecting PHP Webpage with MySQL.

Answer: PHP **mysqli_connect() function** is used to connect with MySQL database. It returns *resource* if connection is established or *null*.

Syntax

```
resource mysqli_connect (server, username, password)
```

f) Define GET and POST methods.

Answer: There are two ways the browser client can send information to the web server.

- The GET Method
- The POST Method

Before the browser sends the information, it encodes it using a scheme called URL encoding. In this scheme, name/value pairs are joined with equal signs and different pairs are separated by the ampersand.

The GET Method

The GET method sends the encoded user information appended to the page request. The page and the encoded information are separated by the **?** character.

- The GET method produces a long string that appears in your server logs, in the browser's Location: box.
- The GET method is restricted to send upto 1024 characters only.
- Never use GET method if you have password or other sensitive information to be sent to the server.
- GET can't be used to send binary data, like images or word documents, to the server.

- The data sent by GET method can be accessed using QUERY_STRING environment variable.
- The PHP provides **$_GET** associative array to access all the sent information using GET method.

The POST Method

The POST method transfers information via HTTP headers. The information is encoded as described in case of GET method and put into a header called QUERY_STRING.

- The POST method does not have any restriction on data size to be sent.
- The POST method can be used to send ASCII as well as binary data.
- The data sent by POST method goes through HTTP header so security depends on HTTP protocol. By using Secure HTTP you can make sure that your information is secure.
- The PHP provides **$_POST** associative array to access all the sent information using POST method.

g) State the use of "$" sign in PHP.

Answer: A dollar ($) sign in php acts like the dollar sign in bash, it refers to a variable, in bash you can use the dollar sign with commands to get their output and assign them to a variable like this:

```
variable_name=$(echo 'Hello World')
```

Besides, it makes the interpretation of php source code much faster than other languages when their source code is compiled and, it also provides you features that you won't find in other languages which are string inclusion and variable variables, take a look at the following:

```
<?php

// A variable with a string value
```

```
$greeting = 'Hello, ';

 // Now let's greet John.

echo "$greeting John";

// This will output "Hello, John" without quotations.
```

Q.2) Attempt any THREE of the following. (4 x 3 = 12 marks)

a) Write a program using foreach loop.

Answer: The foreach construct provides the easiest way to iterate the array elements. It works on array and objects both. The foreach loop though iterates over an array of elements, the execution is simplified and finishes the loop in less time comparatively. It allocates temporary memory for index iterations which makes the overall system to redundant its performance in terms of memory allocation.

```
<?php
$employee = array(
   "name" => "Robert",
   "email" => "robert112233@mail.com",
   "age" => 18,
   "gender" => "male"

);

// Loop through employee array
foreach($employee as $key => $element) {
   echo $key . ": " . $element . "<br>";
}
  ?>
```

Output:

```
name: Robert
email: robert112233@mail.com
age: 18
gender: male
```

PHP Sample Paper Fully Solved 2020

b) Explain Indexed and Associative arrays with suitable example.

Answer: An array is a data structure that stores one or more similar type of values in a single value. For example if you want to store 100 numbers then instead of defining 100 variables its easy to define an array of 100 length.

Associative Arrays

The associative arrays are very similar to numeric arrays in term of functionality but they are different in terms of their index. Associative array will have their index as string so that you can establish a strong association between key and values.

To store the salaries of employees in an array, a numerically indexed array would not be the best choice. Instead, we could use the employees names as the keys in our associative array, and the value would be their respective salary.

```
<html>
  <body>

    <?php
      /* First method to associate create array. */
      $salaries = array("mohammad" => 2000, "qadir" => 1000, "zara" => 500);

      echo "Salary of mohammad is ". $salaries['mohammad'] . "<br />";
      echo "Salary of qadir is ".  $salaries['qadir']. "<br />";
      echo "Salary of zara is ".  $salaries['zara']. "<br />";

      /* Second method to create array. */
      $salaries['mohammad'] = "high";
      $salaries['qadir'] = "medium";
      $salaries['zara'] = "low";

      echo "Salary of mohammad is ". $salaries['mohammad'] . "<br />";
      echo "Salary of qadir is ".  $salaries['qadir']. "<br />";
      echo "Salary of zara is ".  $salaries['zara']. "<br />";
    ?>

  </body>
</html>
```

Indexed Array

PHP indexed array is an array which is represented by an index number by default. All elements of array are represented by an index number which starts from 0.

PHP indexed array can store numbers, strings or any object. PHP indexed array is also known as numeric array.

```
<?php
$size=array("Big","Medium","Short");
echo count($size);
?>
```

c) Define Introspection and explain it with suitable example.

Answer: Introspection is a common feature in any programming language which allows object classes to be manipulated by the programmer. You'll find introspection particularly useful when you don't know which class or method you need to execute at design time.

Introspection in PHP offers the useful ability to examine classes, interfaces, properties, and methods. PHP offers a large number functions that you can use to accomplish the task. In order to help you understand introspection, I'll provide a brief overview of some of PHP's classes, methods, and functions using examples in PHP to highlight how they are used.

PHP Introspection Functions

In the first example, I'll demonstrate a handful of PHP's introspection functions. You can use these functions to extract basic information about classes such as their name, the name of their parent class, and so on.

- class_exists() – checks whether a class has been defined
- get_class() – returns the class name of an object

- get_parent_class() – returns the class name of an object's parent class
- is_subclass_of() – checks whether an object has a given parent class

Here is the example PHP code that contains the definition for Introspection and Child classes and outputs information extracted by the functions listed above:

```
<?php

class Introspection

{

 public function description() {

      echo "I am a super class for the Child class.n";

   }

}

class Child extends Introspection

{

   public function description() {

      echo "I'm " . get_class($this) , " class.n";

      echo "I'm " . get_parent_class($this) , "'s child.n";

   }

}

if (class_exists("Introspection")) {
```

```
    $introspection = new Introspection();

    echo "The class name is: " . get_class($introspection) . "n";

    $introspection->description();

}

if (class_exists("Child")) {

    $child = new Child();

    $child->description();

   if (is_subclass_of($child, "Introspection")) {

        echo "Yes, " . get_class($child) . " is a subclass of Introspection.n";

    }

    else {

        echo "No, " . get_class($child) . " is not a subclass of Introspection.n";

    }

}
```

PHP Sample Paper Fully Solved 2020

d) Differentiate between Session and Cookies.

SESSION	COOKIES
A temporary and interactive information interchange between two or more communicating devices or between a computer and user	Small pieces of data sent from a website and stored on the user's computer by the user's web browser while the user is browsing
Stored in the server side	Stored in the client's browser as text files
Can store a large amount of data	Can store a minimum amount of data
Provides more security because it is difficult to access session values	Provide minimum security because it is easier to access cookie values
Holds multiple variables	Do not hold multiple variables
Available until the browser is opened	Keep information until deleted by the user or set as per the timer
More reliable	Comparatively less reliable

PHP Sample Paper Fully Solved 2020

Q.3) Attempt any THREE of the following. **(4 x 3 = 12 marks)**

a) Differentiate between implode and explode functions.

Implode	Explode
Implode() function is used to merge 2 or more array values into one string value, kind of like concatenation.	Explode() function is used to split a string into an array with the help of a separator.
Syntax : implode(separator,array);	Syntax : explode(separator,string,limit)
Example <?php $arr = array('Welcome','To','Letsknowit'); echo implode(" ",$arr)." "; echo implode("-",$arr)." "; ?> **OUTPUT** : Welcome To Letsknowit Welcome-To-Letsknowit	Example <?php $str = "Welcome To Letsknowit"; print_r (explode(" ",$str)); ?> **OUTPUT :** Array ([0] => Hello [1] => Welcome [2] => TO [3] => Letsknowit)
The implode() function returns a string from elements of an array. implode() takes an array of strings and joins them together into one string using a delimiter (string to be used between the pieces) of your choice.	The explode function in PHP allows us to break a string into smaller text with each break occurring at the same symbol. This symbol is known as the delimiter.
The implode function returns a string from the elements of an array.	The explode() function breaks a string into an array.

b) Write a program for cloning of an object.

Answer:

```
<?php

  // Program to create copy of an object

// Creating class
class GFG {
```

```
    public $data1;
    public $data2;
    public $data3;
}

// Creating object
$obj = new GFG();

// Creating clone or copy of object
$copy = clone $obj;

// Set values of $obj object
$obj->data1 = "The";
$obj->data2 = "Code";
$obj->data3 = "Journal";

// Set values of copied object
$copy->data1 = "MSBTE";
$copy->data2 = "DIPLOMA ";
$copy->data3 = "WEBSITE";

// Print values of $obj object
echo "$obj->data1$obj->data2$obj->data3\n";

// Print values of $copy object
echo "$copy->data1$copy->data2$copy->data3\n";

?>
```

c) Define session and explain how it works.

Answer: A session creates a file in a temporary directory on the server where registered session variables and their values are stored. This data will be available to all pages on the site during that visit.

The location of the temporary file is determined by a setting in the **php.ini** file called **session.save_path**. Before using any session variable make sure you have setup this path.

PHP Sample Paper Fully Solved 2020

When a PHP script wants to retrieve the value from a session variable, PHP automatically gets the unique session identifier string from the PHPSESSID cookie and then looks in its temporary directory for the file bearing that name and a validation can be done by comparing both values.

A session ends when the user loses the browser or after leaving the site, the server will terminate the session after a predetermined period of time, commonly 30 minutes duration.

Starting a PHP Session

A PHP session is easily started by making a call to the **session_start()** function.This function first checks if a session is already started and if none is started then it starts one. It is recommended to put the call to **session_start()** at the beginning of the page.

Session variables are stored in associative array called **$_SESSION[]**. These variables can be accessed during lifetime of a session.

The following example starts a session then register a variable called **counter** that is incremented each time the page is visited during the session.

```
<?php
   session_start();

   if( isset( $_SESSION['counter'] ) ) {
      $_SESSION['counter'] += 1;
   }else {
      $_SESSION['counter'] = 1;
   }

   $msg = "You have visited this page ".  $_SESSION['counter'];
   $msg .= "in this session.";
?>

<html>

   <head>
      <title>Setting up a PHP session</title>
   </head>

   <body>
      <?php  echo ( $msg ); ?>
   </body>

</html>
```

PHP Sample Paper Fully Solved 2020

d) Write Update and Delete operations on table data.

Answer: The UPDATE statement is used to update existing records in a table:

```
UPDATE my_table
SET name= brad, age=21
WHERE name = prince
```

The DELETE statement is used to delete records from a table:

```
DELETE FROM my_table
WHERE name = brad;
```

Q.4) Attempt any THREE of the following. (4 x 3 = 12 marks)

a) State the variable function. Explain it with example.

Answer: PHP supports the concept of variable functions. This means that if a variable name has parentheses appended to it, PHP will look for a function with the same name as whatever the variable evaluates to, and will attempt to execute it. Among other things, this can be used to implement callbacks, function tables, and so forth.

Variable functions won't work with language constructs such as echo, print, unset(), isset(), empty(), include, require and the like. Utilize wrapper functions to make use of any of these constructs as variable functions.

```
<html>

<body>

<?php

function red()

{

        echo "Roses are red";

}
```

```
function blue()
{
        echo "Sky is blue"
}
function green()
{
        echo "Tress are green"
}
$fun_var = "red";
$fun_var();
?>
</body>
</html>
```

b) Explain the concept of Serialization with example.

Answer: Most often we need to store a **complex array** in the database or in a file from PHP. Some of us might have surely searched for some built-in function to accomplish this task. Complex arrays are arrays with elements of more than one data-types or array.
But, we already have a handy solution to handle this situation. We don't have to write our own function to convert the complex array to a formatted string. There are two popular methods of serializing variables.

- **serialize()**
- **unserialize()**

We can serialize any data in PHP using the serialize() function. The serialize() function accepts a single parameter which is the data we want to serialize and returns a serialized string.

```
<?php
  // a complex array
$myvar = array(
    'hello',
    42,
    array(1, 'two'),
    'apple'
);

// convert to a string
$string = serialize($myvar);

// printing the serialized data
echo $string;
  ?>
```

Output:

```
a:4:{i:0;s:5:"hello";i:1;i:42;i:2;a:2:{i:0;i:1;i:1;s:3:"two";}i:3;s:5:"apple";}
```

d) Explain Inserting and retrieving the query result operations.

Answer: After a database and a table have been created, we can start adding data in them.

Here are some syntax rules to follow:

- The SQL query must be quoted in PHP
- String values inside the SQL query must be quoted

- Numeric values must not be quoted
- The word NULL must not be quoted

The INSERT INTO statement is used to add new records to a MySQL table:

```
INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)
```

Example -:

```php
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";

if ($conn->query($sql) === TRUE) {
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}

$conn->close();
?>
```

Retrieving the Query Result -:

Data can be fetched from MySQL tables by executing SQL SELECT statement through PHP function mysql_query. You have several options to fetch data from MySQL.

The most frequently used option is to use function **mysql_fetch_array()**. This function returns row as an associative array, a numeric array, or both. This function returns FALSE if there are no more rows.

```
<?php
   $dbhost = 'localhost:3036';
   $dbuser = 'root';
   $dbpass = 'rootpassword';

   $conn = mysql_connect($dbhost, $dbuser, $dbpass);

   if(! $conn ) {
      die('Could not connect: ' . mysql_error());
   }

   $sql = 'SELECT emp_id, emp_name, emp_salary FROM employee';
   mysql_select_db('test_db');
   $retval = mysql_query( $sql, $conn );

   if(! $retval ) {
      die('Could not get data: ' . mysql_error());
   }

   while($row = mysql_fetch_array($retval, MYSQL_ASSOC)) {
      echo "EMP ID :{$row['emp_id']}  <br> ".
         "EMP NAME : {$row['emp_name']} <br> ".
         "EMP SALARY : {$row['emp_salary']} <br> ".
         "--------------------------------<br>";
   }

   echo "Fetched data successfully\n";

   mysql_close($conn);
?>
```

Q.5) Attempt any TWO of the following. **(6 x 2 = 12 Marks)**

a) Explain any three data types used in PHP.

Answer: Variables can store data of different types, and different data types can do different things.

PHP supports the following data types:

- String
- Integer

- Float (floating point numbers - also called double)
- Boolean
- Array
- Object
- NULL
- Resource

PHP Integer

An integer data type is a non-decimal number between -2,147,483,648 and 2,147,483,647.

Rules for integers:

- An integer must have at least one digit
- An integer must not have a decimal point
- An integer can be either positive or negative
- Integers can be specified in: decimal (base 10), hexadecimal (base 16), octal (base 8), or binary (base 2) notation

```
<?php
$x = 5985;
var_dump($x);
?>
```

PHP Float

A float (floating point number) is a number with a decimal point or a number in exponential form.

In the following example $x is a float. The PHP var_dump() function returns the data type and value:

```
<?php
$x 10.365;
var_dump($x);
?>
```

PHP Boolean

A Boolean represents two possible states: TRUE or FALSE.

```
$x = true;
$y = false;
```

b) Write a program to connect PHP with MySQL.

Answer:

Example (MySQLi Procedural)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Create connection
$conn = mysqli_connect($servername, $username, $password);

// Check connection
if (!$conn) {
   die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
?>
```

Example (MySQLi Object-Oriented)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Create connection
$conn = new mysqli($servername, $username, $password);

// Check connection
if ($conn->connect_error) {
   die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

PHP Sample Paper Fully Solved 2020

c) Explain the concept of overriding in detail.

Answer: In function overriding, the parent and child classes have the same function name with and number of arguments

Function overriding is same as other OOPs programming languages. In function overriding, both parent and child classes should have same function name with and number of arguments. It is used to replace parent method in child class. The purpose of overriding is to change the behavior of parent class method. The two methods with the same name and same parameter is called overriding.

```
<?php
  class Base {
    function display() {
      echo "\nBase class function declared final!";
    }
    function demo() {
      echo "\nBase class function!";
    }
  }
  class Derived extends Base {
    function demo() {
      echo "\nDerived class function!";
    }
  }
  $ob = new Base;
```

```
  $ob->demo();
  $ob->display();
  $ob2 = new Derived;
  $ob2->demo();
  $ob2->display();
?>
```

Output

This will produce the following output−

```
Base class function!
Base class function declared final!
Derived class function!
Base class function declared final!
```

Q.6) Attempt any TWO of the following. (6 x 2 = 12 Marks)

a) Explain web page validation with example.

Answer: Validation means check the input submitted by the user. There are two types of validation are available in PHP. They are as follows −

- **Client-Side Validation** − Validation is performed on the client machine web browsers.
- **Server Side Validation** − After submitted by data, The data has sent to a server and perform validation checks in server machine.

PHP Sample Paper Fully Solved 2020

Some of Validation rules for field

Field	Validation Rules
Name	Should required letters and white-spaces
Email	Should required **@** and **.**
Website	Should required a valid URL
Radio	Must be selectable at least once
Check Box	Must be checkable at least once
Drop Down menu	Must be selectable at least once

Valid URL

```
$website = input($_POST["site"]);

if (!preg_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~_|!:,.;]*[-a-z0-9+&@#\/%=~_|]/i",$website)) {

  $websiteErr = "Invalid URL";

}
```

Valid Email

```
$email = input($_POST["email"]);
```

```
if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
  $emailErr = "Invalid format and please re-enter valid email";
}
```

```
<html>
  <head>
      <style>
         .error {color: #FF0000;}
      </style>
   </head>

   <body>
      <?php
         // define variables and set to empty values
         $nameErr = $emailErr = $genderErr = $websiteErr = "";
         $name = $email = $gender = $comment = $website = "";

         if ($_SERVER["REQUEST_METHOD"] == "POST") {
            if (empty($_POST["name"])) {
               $nameErr = "Name is required";
            }else {
               $name = test_input($_POST["name"]);
            }
```

PHP Sample Paper Fully Solved 2020

```
if (empty($_POST["email"])) {
   $emailErr = "Email is required";
}else {
   $email = test_input($_POST["email"]);

   // check if e-mail address is well-formed
   if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
      $emailErr = "Invalid email format";
   }
}

if (empty($_POST["website"])) {
   $website = "";
}else {
   $website = test_input($_POST["website"]);
}

if (empty($_POST["comment"])) {
   $comment = "";
}else {
   $comment = test_input($_POST["comment"]);
}

if (empty($_POST["gender"])) {
```

PHP Sample Paper Fully Solved 2020

```
            $genderErr = "Gender is required";
         }else {
            $gender = test_input($_POST["gender"]);
         }
      }

      function test_input($data) {
         $data = trim($data);
         $data = stripslashes($data);
         $data = htmlspecialchars($data);
         return $data;
      }
   ?>

   <h2>Absolute classes registration</h2>

   <p><span class = "error">* required field.</span></p>

   <form method = "post" action = "<?php
      echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">
      <table>
         <tr>
            <td>Name:</td>
            <td><input type = "text" name = "name">
               <span class = "error">* <?php echo $nameErr;?></span>
```

```
            </td>
        </tr>

        <tr>
            <td>E-mail: </td>
            <td><input type = "text" name = "email">
                <span class = "error">* <?php echo $emailErr;?></span>
            </td>
        </tr>

        <tr>
            <td>Time:</td>
            <td> <input type = "text" name = "website">
                <span class = "error"><?php echo $websiteErr;?></span>
            </td>
        </tr>

        <tr>
            <td>Classes:</td>
            <td> <textarea name = "comment" rows = "5" cols =
"40"></textarea></td>
        </tr>

        <tr>
            <td>Gender:</td>
            <td>
```

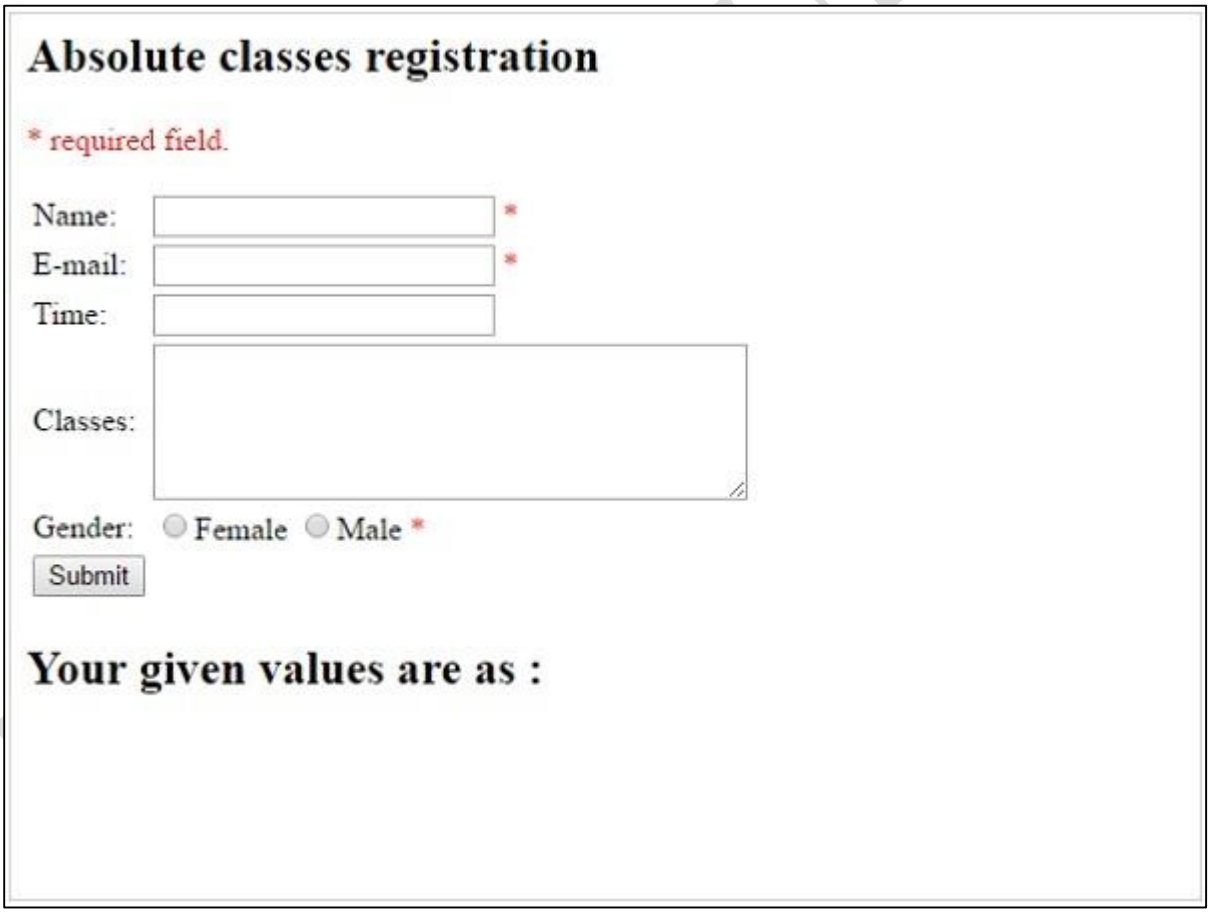
PHP Sample Paper Fully Solved 2020

```
                    <input type = "radio" name = "gender" value =
"female">Female

                    <input type = "radio" name = "gender" value = "male">Male

                    <span class = "error">* <?php echo $genderErr;?></span>

                </td>

            </tr>


            <td>

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

            </td>


         </table>


      </form>


      <?php

         echo "<h2>Your given values are as:</h2>";

         echo $name;

         echo "<br>";


         echo $email;

         echo "<br>";


         echo $website;

         echo "<br>";
```

PHP Sample Paper Fully Solved 2020

```
        echo $comment;

        echo "<br>";



        echo $gender;

    ?>



  </body>

</html>
```

Output :



PHP Sample Paper Fully Solved 2020

b) Write a program to create PDF document in PHP.

Answer: The FPDF class is written for PHP4 and was the first free PHP script for creating PDF files "on the fly". I've used this class 8 years ago to generate PDF documents in several ecommerce applications. The project website offers great documentation and also a lot of useful code examples. The latest version is from June 2011 and I hope the next version is written for PHP5. The original version doesn't support UTF-8, but since a while there is class extension named tFPDF which supports UTF-8.

```
<?php
require('fpdf.php');
$pdf = new FPDF();
$pdf->AliasNbPages();
$pdf->AddPage();
$pdf->SetFont('Times','',12);
$pdf->Cell(0,10,'Question 1: '.$_POST["first"],0,1);
$pdf->Cell(0,10,'Question 2: '.$_POST["second"],0,1);
$pdf->Cell(0,10,'Question 3: '.$_POST["third"],0,1);
$pdf->Cell(0,10,'Question 4: '.$_POST["fourth"],0,1);
$pdf->Cell(0,10,'Question 5: '.$_POST["fifth"],0,1);
$pdf->Cell(0,10,'Question 6: '.$_POST["sixth"],0,1);
$pdf->Cell(0,10,'Question 7: '.$_POST["seventh"],0,1);
$pdf->Cell(0,10,'Question 8: '.$_POST["eight"],0,1);
$pdf->Cell(0,10,'Question 9: '.$_POST["nine"],0,1);
$pdf->Cell(0,10,'Question 10: '.$_POST["ten"],0,1);
$pdf->Output(); ?>
```

PHP Sample Paper Fully Solved 2020

c) Elaborate the following:

i) __call() **ii) mysqli_connect()**

Answer: i) __call() -: PHP 5 includes a number of magic methods that a class can implement. Each magic method begins with __, and gets called automatically by PHP under certain conditions. Technically some are called overloading methods rather than magic methods, but they all effectively work by the same magic blue smoke.

Today we're just going to look at one of them: __call(). If a class implements __call(), then if an object of that class is called with a method that doesn't exist __call() is called instead.

```
<?php
class Caller {
  private $x = array('a', 'b', 'c');

  public function

__call($method, $args) {
    print "Method $method called:\n";
    var_dump($args);
    return $this->x;
  }
}
$foo = new Caller();
$a = $foo->test(1, 2, 3);
var_dump($a);
?>
```

ii) mysqli_connect() -: The mysqli_connect() function in PHP is used to connect you to the database. In the previous version of the connection mysql_connect() was used for connection and then there comes mysqli_connect() where i means improved version of connection and is more secure than mysql_connect().

Syntax:

```
mysqli_connect ( "host", "username", "password", "database_name" )
```

PHP Sample Paper Fully Solved 2020

Parameters used:

host: It is optional and it specify the host name or IP address. In case of local server localhost is used as a genral keyword to connect local server and run the program.

username: It is optional and it specify mysql username. In local server username is root.

Password: It is optional and it specify mysql password.

database_name: It is database name where operation perform on data. It also optional.

Example -:

```
?php
   mysqli_connect("localhost", "root", "", "GFG");

   if(mysqli_connect_error())
      echo "Connection Error.";
   else
      echo "Database Connection Successfully.";
?>
```