PHP (20 MARKS)

Q1 What do you understand by the term PHP? List any four features of PHP.

PHP (PHP Hypertext Preprocessor) is an open-source server-side scripting environment that can be used to create and run dynamic and interactive web-server application.

Features of PHP:

- PHP is a loosely typed language
- 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 provides HTTP authentication method

Q2. Name any two equivalent tools of PHP.

ASP JSP PERL CGI

Q3. What do you understand by the term PHP interpreter?

PHP interpreter is software on server to interpret the PHP code and create equivalent HTML code to be sent to browser

Q4. How can you add PHP code to your HTML file?

PHP code can be added to an HTML file by including the code in:

- the tags **<script language=php>** and **</script>**
- the delimiters **<?PHP** and **?>**

Q5. What are delimiters? What delimiters are used to surround PHP scripts?

A delimiter is a character or a sequence of characters that marks the beginning or end of a unit. The delimiters **<?PHP** and **?>** are used to surround PHP scripts.

Q6. What are comments? How to Add Comments in PHP

Comments are non-executable lines which are ignored by interpreter and not visible when code executed

Single Line comments it can be given by putting **//** before the text makes it a single line comment

Multiline Comments It can be given between **/*** ***/**

Q7. **Are keyword and Variables are case sensitive?**

All **keywords** (e.g. if, else, while, echo, etc.), classes, functions, and user- defined functions are **NOT case-sensitive**. In the example below, all three echo statements below are legal (and equal):

Example

```
<?php   ECHO "Hello World!<br>";
        echo "Hello World!<br>";      ?>
```

All **variable** names are **case-sensitive**. In the example below, only the first statement will display the value of the $color variable (this is because $color, $COLOR, and $coLOR are treated as three different variables):

Example

```
<?php       $color = "red";
            echo "My car is " . $color . "<br>";
            echo "My house is " . $COLOR . "<br>"; // error $COLOR  ?>
```

Q8 **What is the significance of the PHP function echo?**

Echo is a string function which is used to send content to the browser.

Q9. **Difference between Echo & Print**

Echo	Print
Can take multiple parameters separated by comma	can take only one parameter.
does not return any value	always returns 1
Echo Is faster than print	Slower

Q10. **What is a variable?**

A variable is a named memory location that can be used for storing data.

Q11. **Identify correct Variable name a) $123 $a123**

$123 wrong Variable name should start with Alphabet or _ after $

$a123 correct

Q12. How can we define constants in PHP?

In PHP we can define constants with the keyword **define** .

define(name, value, case-insensitive)

Parameters:

name: Specifies the name of the constant

value: Specifies the value of the constant

case-insensitive: Specifies whether the constant name should be case-**insensitive**. **Default is false**

```
<?PHP
define('SiteName','vinodsrivastava.com');   // SiteName is constant name
        echo  'You are visiting' .Sitename;  ?>
```

OUTPUT : You are visiting vinodsrivastava.com

Note : Constant name does not contain $.

Constants are automatically global and can be used across the entire script.

Q13. Explain constant() function

constant() function returns the value of a constant identifier

```
<?php
        //define a constant
        define("GREETING","GOOD MORNING");
        echo constant("GREETING");
?>
        Output:   GOOD MORNING
```

Q14. Explain the following output

```
$name= “Faips”;
Echo  “$name;
         output    Faips
Echo ‘$name’;
        output     $name
```

when using **single quotes**, we cannot use the echo statement to output a variable(echo '$name';). It will simply output $name

While using double quotes, you can output, via echo, variables(echo "$name";)

Q15. How to delete a variable

Variable can be deleted using **unset($Variablename)**

Q16 Give example of 6 basic data type supported by PHP

1. integer   2. double   3 boolean   4 string   5 array   6 object

Q17 What is Type conversion?

Type casting is converting one data type to another type. It is done automatically done by PHP. PHP has two functions

1. **gettype() to find data type of variable**

```
$ var=19
Echo gettype($var); // output   integer
```

2. **settype() to set the data type of variable**

```
$ var=”12E”
settype($var,”integer”); // $var is now set to 12(integer)
```

Q18.   Explain ternary operator with example

The condition operator **? :** is ternary operator, because it uses three operands. This operator to check a condition and produce a logical result either TRUE or FALSE. The syntax is

Variable= (conditional expression) **?** expression 1 if TRUE **:** expression 2 if FALSE;

Example

```
$x=10;
$y=15;
$ max= ($x>$y)? $x : $y;
Echo “ Maximum value is=” . $max;
Output          Maximum value is=15;
```

Q19. Differentiate between ‘/’ and ‘%’ operator

‘/’ operator	‘%’ operator
‘/’ operator is used to find quotient	‘%’ operator is used to find remainder
‘/’ operator can be used with integer and float both	‘%’ operator cannot used with float

Q20. Explain the following string function with example

Function	Description	Example
rtrim() **Syntax** • **rtrim(string);** • **rtrim(string, text to be trimmed from left);**	Removes whitespace or other characters from the right side of a string	**Ex1** <?php $str = “###VKS Learning”;”; echo **rtrim**(“$str”); //# means space ?> Output: **VKS Learning** **Ex2** <?php $str = “VKS Learning” echo **rtrim**(“$str”,”ing”); ?> Output: **VKS Learn**
ltrim() **Syntax** • **rtrim(string);** • **rtrim(string, text to be trimmed from right);**	Removes whitespace or other characters from the left side of a string	**Ex1** <?php $str = ““VKS Learning”####”; echo **ltrim**(“$str”); //# means space ?> Output: **VKS Learning** **Ex2** <?php $str = “VKS Learning”; echo **ltrim**(“$str”,”VK”); ?> Output: **S Hosting**
trim() **Syntax** • **trim(string);** • **trim(string, text to be trimmed from both side);**	Removes whitespace or other characters from both sides of a string	**Ex1** <?php $str = “##Wordpess Hosting####”; echo **trim**(“$str”); //# means space ?> Output: **VKS Learning** **Ex2** <?php $str = “Wordpess Hosting”; echo **trim**(“$str”,”VKing”); ?> Output: **S Host**
strlen() **Syntax** • **strlen(string);**	Returns the length of a string	<?php $str = “VKS Learning”; echo **strlen**(“$str”); ?> Output: **12**

strtolower() **Syntax** • **strtolower(string);**	Accept a string as parameter and returns the string with all alphabets in lowercase.	<?php $str = “VKS Learning”; echo **strtolower**(“$str”); ?> Output: **vks learning**
strtoupper() **Syntax** • **strtoupper(string);**	Accept a string as parameter and returns the string with all alphabets in uppercase.	<?php $str = “VKS Learning”; echo **strtoupper**(“$str”); ?> Output: **VKS LEARNING**

substr()
Syntax
substr (string,start);
substr (string,start,n);
	Returns a part of a string
	extract a string from a particular position
		Ex1
		<?php
		$str = “VKS Learning”;
		echo **substr** (“$str” 4);
		?>
		Output: **LEARNING**
		Ex2
		<?php
		$str = “VKS Learning”;
		echo **substr** (“$str” 4,5);
		?>
		Output: **LEARN**

strcmp()
Syntax
strcmp (string1,string2);
	It returns output either greater than zero, less than zero or equal to zero. If string 1 is greater than string 2 then it returns greater than zero. If string 1 is less than string 2 then it returns less than zero. It returns zero, if the strings are equal. (case-sensitive)	**Ex1**
		<?php
		echo strcmp(“Learn”,”LEARN);?>
		output: **1** as string 1 **>** string 2
		Ex2
		<?php
		echo strcmp(“Learn”, “Learn”); ?>
		//Both the strings are equal
		output: **0** as string **=** string 2
		Ex3
		<?php
		echo strcmp(“Cloud”,”Hosty”);?>
		output: -1 as string 1 **<** string 2
		Ex4
		<?php
		echo strcmp(“a”,”b”); ?>//compares alphabetically
		output: -1

strncmp()
Syntax
strcmp (string1, string2,n);
	function compare two strings of the first n characters.	**Ex1**
		<?php
		echo strncmp("Hello","Hello",5);
		?>
		Output: **0**
		Ex2
		<?php
		echo **strncmp**("Hello","hELLo",5);
		?>
		Output: **-1**

stripslashes()
Syntax
stripslashes (string);
	Unquotes a string quoted with addslashes()	<?php
		echo **stripslashes**("Who\'s Peter Griffin?");
		?>
		Output: **Who's Peter Griffin?**

strncasecmp()
 	
Syntax
strncasecmp (string1, string2,n);
	String comparison of the first n characters (case-insensitive)	**Ex1**
		<?php
		echo strncasecmp("Hello","Hello",5);
		?>
		Output: **0**
		Ex2
		<?php
		echo strncasecmp("Hello","hELLo",5
		);
		?>
		Output: **0**

strrev()
 	
Syntax
strrev (string);
	Reverses a string	<?php
		$str = “VKS”;
		echo **strrev** (“$str”);
		?>
		Output: **SKV**

str_word_count()
Syntax
str_word_count (string);
	display of the number of words in any specific string	<?php
		$str = “VKS Learning”;
		echo **str_word_count** ($str);
		?>
		Output: **2**

strpos();
Syntax
strpos (string,text);
	Strpos() enables searching particular text within a string. It works simply by matching the specific text in a string. If found, then it returns the specific position. If not found at all, then it will return “False”.	<?php
		$str = “VKS Learning”;
		echo **strpos** ($str, “Learn”);
		?>
		Output: **2**

strpos() **Syntax** **Str_replace(string to be replaced,text,string)**	The str_replace() function replaces some characters with some other characters in a string.	<?php $str = “VKS Learning”; echo **Str_replace** (“VKS” “FAIPS”,$str); ?> Output: **FAIPS Learning**

Strchr()
 	
Syntax
Strchr(string,search)
	This function is used to find the first occurrence of string and return all the characters from the search position to the end of the string	<?php
		$str = “VKS Learning”;
		echo **Strchr** ($str, “L”);
		?>
		Output: **Learning**

Q21. Explain the following date function with example

<table>
<tr><th>Function</th><th>Description & Example</th></tr>
<tr><td><b>date()</b>

<b>Syntax</b>
<b>date(format, timestamp)</b></td><td><b>The function displays a formatted local time/date.</b>
<b>Format</b>
<b>d</b> -> Day of the Month 2 digit -> 01-31
<b>D</b>-> 3 Letter textual day -> Mon-Sun
<b>j</b> -> day of the month without leading zero-> 1-31
<b>l</b>-> A full textual day-> Monday-Sunday
<b>z</b>-> the day of year (starting from 0)->0-365
<b>Y -></b> 4 digit year <b>-> 2017</b>
<b>Y -></b> 2 digit year -> 17
<b>F -></b> Full textual month-> January- December
<b>m</b> -> numeric representation of month-> 01-12
<b>M</b> -> 3 Letter Textual Month-> Jan-Dec
S-> English suffix for day or month-> st,nd,rd,th
a-> am or pm in lower case / A-> AM or PM uppercase

<b>Example</b>
echo <b>date</b>("d/m/Y")."<br />"; // <b>output</b> 10/11/2017
echo <b>date</b>("m/d/Y")."<br />"; // <b>output</b> 11/10/2017
echo <b>date</b>("m-d-Y")."<br />"; // <b>output</b> 11-10-2017
echo <b>date</b>("l jS \of F Y")."<br />"; // <b>output</b> Friday 10th of November 2017
echo <b>date</b>('l jS \of F Y h:i:s A'); <b>// output Friday 10th of November 2017 02:43:54 PM</b></td></tr>
<tr><td><b>getdate()</b>

<b>Syntax</b>
<b>getdate(timestamp)</b></td><td>The function accepts a timestamp and returns an associative array that contains the date and time information for a Unix timestamp.
An associative array. Details are as follows:
<table>
<tr><th>Key</th><th>Description</th><th>returned values</th></tr>
<tr><td>seconds</td><td>Numeric value of</td><td>0 to 59</td></tr>
<tr><td>minutes</td><td>Numeric value of</td><td>0 to 59</td></tr>
<tr><td>hours</td><td>Numeric value of</td><td>0 to 23</td></tr>
<tr><td>mday</td><td>Numeric value of the</td><td>1 to 31</td></tr>
<tr><td>wday&amp;</td><td>Numeric value of the</td><td>0(for Sunday) through</td></tr>
<tr><td>mon</td><td>Numeric value of a</td><td>1 through 12</td></tr>
<tr><td>year</td><td>Four-digit year.</td><td>1999 or 2003</td></tr>
<tr><td>yday</td><td>Numeric value of day</td><td>0 through 365</td></tr>
<tr><td>weekday</td><td>Full name of the day</td><td>Sunday through</td></tr>
<tr><td>month</td><td>Full name of a month.</td><td>January -December</td></tr>
</table>
$today=getdate();
$d=$today['mday'];
$m=$today['mon'];
$y=$today['year'];
Echo "Today is : .$d-$m-$y"; // Output: <b>Today is : 10-11-2017</b></td></tr>
</table>

gettimeofday()
 	
Syntax
gettimeofday(return_float)
	The function is used to get current time information.
	Example :
	<?php
	print_r(gettimeofday());
	?>
	Output :
	Array ([sec] => 1281420398 [usec] => 985536 [minuteswest] => 0 [dsttime] => 0)
	Note : Since the current time is not static, therefore the output may vary.

strtotime()
 	
Syntax
strtotime(time,now)
	Parse any english textual datetime description into Unix timestamp
	
	Echo “Today is:”, **date**(“l,jS F Y”);
	output **Today is:Friday 10th November 2017**
	echo “ Next Friday: “. **date**(“jS F Y”,**strtotime**(‘next Friday’));
	output **Next Friday**: **17th November 2017**

mktime() **Syntax** **mktime(hour, minute, second, month, day, year, is_dst)**	The function returns the Unix timestamp for a date. This timestamp is a long integer containing the number of seconds between Unix Epoch(January 1 1970 00:00:00 GMT) and the time specified. Echo **date**(“M-d-Y”,**mktime**(0,0,0,10,11,2017)); Output **Nov-11-2017** Echo **date**(“M-d-Y”,**mktime**(0,0,0,10,11,2017)+1); Output **Nov-12-2017** Echo **date (‘M-d-Y H: i: s” mktime(23,46,20,6,25,2017));** Output: **Jun-25-2017 23:46:20**

Cal_days_in_months()
 	
Syntax
Cal_days_in_months(calendar,month,year)
	This function returns the numbers of days in the month for the specified year.
	
	$nd= **Cal_days_in_months**(CAL_GEREGORAIN 10,2017);
	Echo $nd;
	Output : 31

Q22. Explain ord() function

This function is used to find the ASCII value of first character of a string.

echo **ord**(“CBSE”);// ASCII value of ‘**C’ is 67**
Output: **67**
 	
Echo **ord**(‘\n’)// ASCII value of **newline is 10**
Output: **10**
	$v=**ord**(“”);// ASCII value of **space is 32**
	Echo $v
	Output: **32**

Q 23 Name three logical, unary, binary and relational operator

Logical	**AND, OR, NOT**	**Unary**	**+, - , --, ++**
Binary	**+,*, / , %**	**Relational**	**>, <, ==, ===**

Q24. Explain two optional statement used in switch case

break it is used to exit the switch case immediately skipping all statements after it

default: if none of the case matches then default case is executed.

Q25. Rewrite the following program code using switch statements

```
If($d==1)
        $day= "Monday";
else If($d==2)
        $day= "Tuesday";
else If($d==3)
        $day= "Wednesday";
else
        $day= "_";
```

Ans

```
        switch($d)
        {
                case 1:   $day= "Monday";   break;
                case 2:   $day= "Tuesday";  break;
                case 3:   $day= "Wednesday"; break;
                default:   $day="_";
        }
```

Q26

1. **Change the for Loop to while loop without Affecting output of above**
<?php
$i=1; $x=0;
for($i=1; $i<10; $i*=2)
{ $x++;
echo $x;
}
echo "<BR>" . $i ;
?>
	Ans
	<?php
	$i=1; $x=0;
	while($i<10)
	{
	$x++;
	echo $x;
	$i*=2
	}
	echo "<BR>" . $i ;
	?>

Give Output and change the for loop to do while loop
for($x=10,$c=20;$c>=10;$c=$c-2)
echo $x++."
";
	$x=10;$c=20;
	do {
	echo $x++ . "
	";
	$c=$c-2;
	}**while**($c>=10);
	Output
	10
	11
	12
	13
	14

Give Output and change the for loop to while loop **$name="MULTI";** **for($x=0 ; $x <= strlen($name) ; $x++)** **echo substr($name , $x , $x+1) . " ";**	**$name="MULTI"; $x=0 ;** **while($x <= strlen($name))** **{ echo substr($name , $x , $x+1) ." "; x++; }** **M** **MU** **MUL** **MULT** **MULTI**

Q27. Differentiate between

1.

While	do..while
while specifies an entry-controlled loop	do..while specifies an exit controlled loop.
Loop does not execute even once if the condition if false in the beginning.	Loop executes at least once irrespective of the initial value of the condition.

2.

Deterministic loop	Non-deterministic loop
Number of iterations of the loop can be predicted.	Number of iterations of the loop cannot be predicted.
The looping condition does not depend on the user	The looping condition depends on the user input.
e.g. for loop and foreach loop	While loop and do..while loop.

3.

for loop	foreach loop
Can be used for arrays or otherwise.	Can be used for arrays only.
Need to specify the number of iterations of the loop.	Need not specify the number of iterations of the loop. The loop runs for the number of elements in the specified array.

4.

Entry controlled loop	Exit controlled loop
Looping condition is specified in the beginning of the loop.	Looping condition is specified at the end of the loop.
Loop does not execute even once if the condition if false in the beginning.	Loop executes at least once irrespective of the initial value of the condition.
Examples: for loop and while loop	Example: do..while loop

.5

PHP file	HTML file
Has the extension .php	Has the extension .html
Contains PHP code and may or may not contain HTML code.	Contains HTML code but does not contain PHP code.
Cannot be executed by the browser	Can be executed by the browser

6

date()	getdate()
Returns date and time in specified format.	Returns date and time in the form of an associative array.
Can take two arguments – date format and timestamp	Can take only one argument – timestamp

7.

Local Variable	Global variable
A local variable is defined inside a function.	A global variable is defined outside any function.
A local variable can be used only in the function in which it is defined.	A global variable can be accessed in any block of the script.

Q28. What is Array?Explain two types of one dimensional array in PHP with example

An array is a type of variable that groups a series of values and places them in a single variable.

There are two types of one dimensional arrays in PHP:

- **Indexed array** – An array with a numeric index
  ***$numbers** = **array**(5, 8, 3, 32, 17);*
- **Associative array** - An array where each ID key is associated with a value.
  $f_color = **array**('Banana'=>"Yellow", "Afapple"=>"Red", "Pear"=>"Green");

Q29. What is the use of return statement?

return statement is used to return a value and the control jumps out of current function and give back the return value to the code that called function.

Syntax $value=function_name();

Q30. Write a PHP code using user defined function sum() which return the sum of two number passed as parameter

```
<? PHP
        function sum($a, $b)
        {
                $sum=$a+$b;
                return($sum);
        }
        $result=sum(10,20);
        echo "sum=" .$result  // output  sum=30
?>
```

Q31. What is parameter? Name two type of parameter explain with examples

Parameter is used to transfer value from calling to called function

Two type of parameter: **Formal & Actual Parameter**

Formal Parameter	Actual Parameter
Parameter declared in function definition	Parameter used in function invocation
Formal parameter can only be variable	Parameter can be variable, constant or expression

<? PHP
function sum($a, $b)
{
$sum=$a+$b;
return($sum);
}
$result=sum(10,20);
echo "sum=" .$result
// output sum=30
?>
$a, $b are formal parameter
	<? PHP
	function sum($a, $b)
	{
	$sum=$a+$b;
	return($sum);
	}
	$x=5;$y=10;
	$result=sum($x,$y);
	echo "sum=" .$result."
	"; //output sum=**15**
	$result1=sum(10,20);
	echo "sum=" .$result1."
	"; //output sum=**30**
	?>
	$x, $y are actual parameter

Q32 Differentiate between Local & Global Variable with Example

Local Variable/Scope	Global Variable/Scope
Variable declared inside a function/method are local variable	Variable declared outside any function/method are Global variable
Function can be used within the function where it is declared so have local scope only	Function can be used by all script and function on the webpage so have global scope

<? PHP
function sum()
{
$a=10;$b=20;
$sum=$a+$b;
return($sum);
}
$x=5;$y=10;
$result=sum($x,$y);
echo “sum=” .$result.”
”;
?>
$a, $b, $sum are local variable
	<? PHP
	$x=5;$y=10;
	function sum()
	{ global $x, $y; // to use global value of $x, $y
	$sum=$x+$y;
	return($sum);
	}
	$result=sum($x,$y);
	echo “sum=” .$result.”
	”;
	?>
	$x, $y are Global variable

Q33 Differentiate between Call by Value & Call by reference with Example

Call by Value	Call by reference
It is a copy of corresponding actual parameter	It is an alias of corresponding actual parameter
Change in value parameter does not change actual parameter	Change in reference parameter, updates actual parameter

<? PHP
function change($a)
{
$a=2* $a;
}
$x=5;
change($x);
echo “x=” .$x.”
”;
?>
Output: x=5
reference parameter $a will not updated the value of actual parameter $x
	<? PHP
	function change(**&**$b)
	{
	$b=2* $b;
	}
	$x=5;
	change($x);
	echo “x=” .$x.”
	”;
	?>
	Output: x= 10
	reference parameter $b has updated the value of actual parameter $x

Q34. What are PHP Superglobals

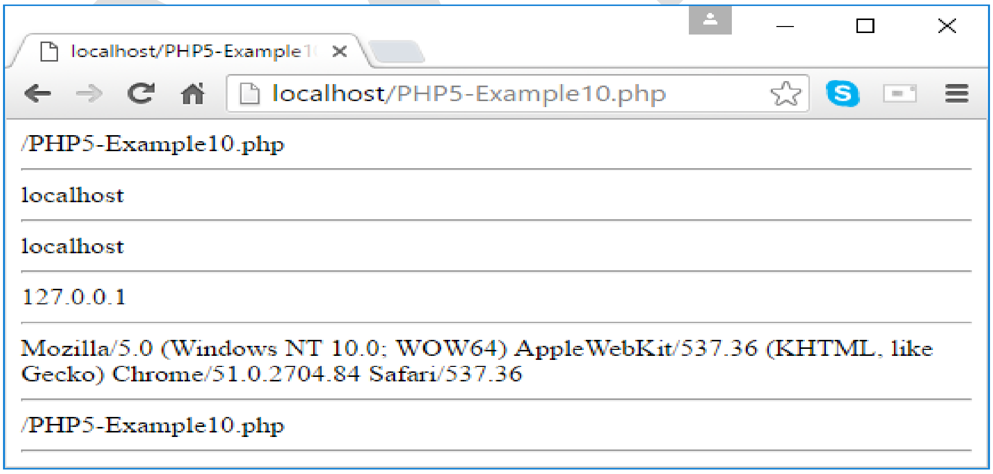
PHP provides several predefined variables, called “superglobals”, that are always accessible regardless of scope – we can access them from any part of the script.

The PHP superglobal variables are:

SNo	Superglobal	Description
1.	$GLOBALS	It is an associative array which contains references to all the currently defined global variables. The variable names are the keys of the array.
2.	$_SERVER	It holds the information about the headers, paths, and script locations.
3.	$_REQUEST	It is used to collect data after submitting an HTML form. It is An associative array that by default contains the contents of $_GET, $_POST and $_COOKIE.
4.	$_POST	It is an associative array of variables passed to the current script when a form is submitted via the HTTP POST method. The variable names are the keys of the array.
5.	$_GET	It is an associative array of variables passed to the current script via the URL parameters, i.e., when a form is submitted via the HTTP GET method.
6.	$_FILES	It is an associative array of items uploaded to the current script via the HTTP POST method.
7.	$_ENV	It is an associative array of variables passed to the current script via the environment method. It represents the data available to a PHP script from the environment in which PHP is running.
8.	$_COOKIE	It is an associative array of variables passed to the current script via HTTP Cookies.
9.	$_SESSION	It is an associative array containing session variables available to the current script.

Q35. Give output

```
<?PHP
    echo $_SERVER['PHP_SELF']."<HR>";
    echo $_SERVER['SERVER_NAME']."<HR>";
    echo $_SERVER['HTTP_HOST']."<HR>";
    echo $_SERVER['REMOTE_ADDR']."<HR>";
    echo $_SERVER['HTTP_USER_AGENT']."<HR>";
    echo $_SERVER['SCRIPT_NAME']."<HR>";
?>
```



```
/PHP5-Example10.php
localhost
localhost
127.0.0.1
Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/51.0.2704.84 Safari/537.36
/PHP5-Example10.php
```

$_SERVER['PHP_SELF'].	It return the file name path of currently executing script
$_SERVER['SERVER_NAME']	it return of the host server
$_SERVER['HTTP_HOST']	it return the host name from where the user is viewing the current page

$_SERVER['REMOTE_ADDR'] it return the IP Address from where the user is viewing the current page

$_SERVER['HTTP_USER_AGENT']. it return the Browser application from where the user is viewing the current page

Q36. What is the difference between GET and POST Method

GET Method

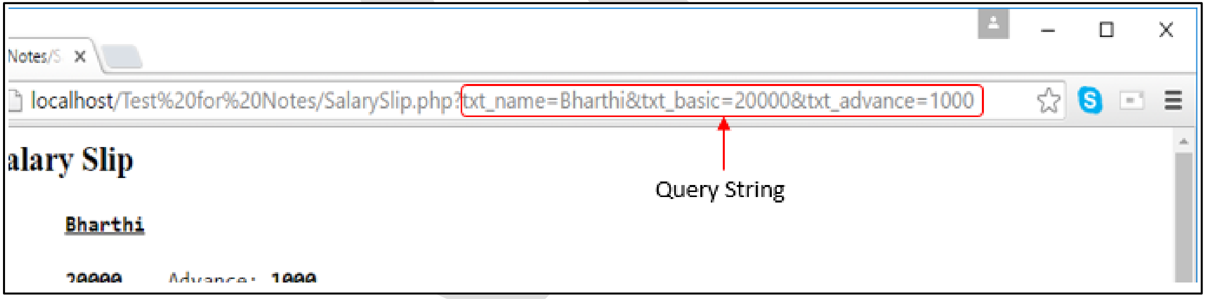
1) All the name value pairs are submitted as a query string in URL.
2) It's not secured.
3) Length of the string is restricted about 256.
4) If method is not mentioned in the Form tag, this is the default method used.

POST Method:

1) All the name value pairs are submitted in the Message Body of the request.
2) Length of the string (amount of data submitted) is not restricted.
3) $_Post is secured because Name-Value pairs cannot be seen in location bar of the web browser.
4) If post method is used and if the page is refreshed it would prompt before the request is resubmitted.
5) If the service associated with the processing of a form has side effects (for example, modification of a database or subscription to a service), the method should be POST.

Q37. Explain the working of QueryString using an example.

When a form is submitted using GET method, the form's data is sent to the server in the form of "variable=value" pairs. A variable and its value are separated by equality sign (=) and different "variable=value" pairs are separated by ampersand sign (&).This complete set of "variable=value" pairs is called Query String and is visible in the URL. Example:



localhost/Test%20for%20Notes/SalarySlip.php?txt_name=Bharthi&txt_basic=20000&txt_advance=1000

alary Slip

Bharthi

20000 Advance: 1000

The query string is specified by the values following the question mark (?).

When a form is submitted by GET method, this query string is visible in the URL. A PHP script retrieves this string in **$_SERVER["QUERY_SRING"].**

Query Strings are used to pass information from a browser to a web server. Several different processes can generate a query string. Query strings are generated by sending a form, by a user typing a query in the address box of the browser or in the anchor tag.

Q38 Explain $_SERVER["REQUEST_METHOD"]

It is used to tell whether data is sent using GET method or POST method when form is submitted

Q39. What is cookie?

Cookie is a small piece of data sent from a website and stored in a user's web browser while user is browsing that website. Every time the user loads the website, the browser send the cookies back to the server

Q40. Name the method used to create a cookie in PHP

setcookie() method is used to create a cookie

Q41. A PHP user "admin" set a cookie for 20 days with its value "password"

```
<?PHP
        $cookie_name= "admin";
        $cookie_value= "password";
        setcookie($cookie_name, $cookie_value, time()+(86400*20),"/");
?>
```

Q42. How to retrieve the values stored in cookie name "Name1"

$_COOKIE[ ] super global variable is used to retrieve the value of cookie

```
<?php
If(isset($_COOKIE['Name1']))
{
$v= ($_COOKIE['Name1']))
echo "Hello"+$v
}
else
echo "invalid"
?>
```

Q43. Name two global variable in PHP to store data across page

$_COOKIE & $_SESSION are used to store data across pages.

Q44. Explain session_start() & session_destroy()

session_start() function used to create a new session and generate a unique session ID. It should be started before before the script generate any output .

session_destroy() function destroy a session completely. It deletes all session varables

unset($_SESSION["variable"]) is used to delete a specific session variable.

Q45. What is the difference between session and cookies

There are some difference between session and cookies thath are as following:-
1 : **Session** are temporary and **Cookies** are permanent.
2 : **Session** data is store on server while **Cookies** are store on user's computer.
3 :**Cookies** contents can be easily modify but to modify **Session** contents is very hard.
4 :**Cookies** could be save for future reference but **Session** couldn't when user close the browser Session data also lost.

Q46. How to access Session Variable named "User"

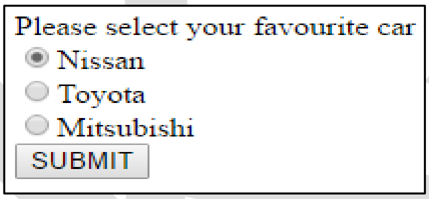
```
<?PHP
        session_start();
                echo  $_SESSION[“User”];
?>
```

Q47. Write a code to create a page counter to check how many visitors have visited the page

```
        <?php
                session_start();
        if (!isset ($_SESSION['views']))
        {
           $_SESSION['views'] = 1;
        }
        else
                $_SESSION['views'] = $_SESSION['views']+1;
        ?>

        Echo “This page has been viewed” . $_SESSION['views'] . “Times”;
```

Q48. Create the following form and based on the user selection print a message in the format given below:



Please select your favourite car

● Nissan

○ Toyota

○ Mitsubishi

SUBMIT

Your favourite car is: Nissan

```
<?php
$car = $carErr = "";   $OK=false;
if ($_SERVER["REQUEST_METHOD"] == "POST")
{    if (!empty($_POST["Car"])) {
                $car = $_POST["Car"];
                $OK=true;     }
     else
                $carErr = "  **Please select a Car";
}
?>
<html>  <body>
<form name = f1 method=POST action = "<?php echo $_SERVER["PHP_SELF"]; ?>" >
Please select your favourite car <?php echo $carErr; ?> <br>
<input type = radio name = Car value = Nissan>Nissan </BR>
<input type = radio name = Car value = Toyota>Toyota </BR>
<input type = radio name = Car value = Mitsubishi>Mitsubishi </BR>
<input type = submit name = submit value = "SUBMIT">
</form>
<?php
     if ($OK==true)         // - 6
                echo "Your favourite car is ".$car;  ?>
</body> </html>
```

Q49. Explain <fome action= “<?php echo $_SERVER[“PHP_SELF”]?>” statement

$_SERVER["PHP_SELF"] is used to display the path of currently used script file name. So the above statement action attributes states that after the form submission form data will be handle by the PHP Script present in the same file**.**

Q50. Write the command(s) to open a text file called data.txt which exists in the directory C:\temp.

$file **= fopen**("c:\temp\data.txt", "r");

Q51 Explain the PHP fopen() function and PHP fclose() functions.

fopen() : fopen() function is used to open a file. The syntax of fopen() is:

Syntax: **fopen**(filename, mode);

fopen() takes two parameters:

i) filename: Specifies the name of the file to be opened.

ii) mode: specifies the mode in which the file should be opened.

fclose() : **fclose()** is used to close an open file. fclose() takes a single parameter (the file reference) and returns a Boolean value indicating the successful or failed closure of file.

Syntax: fclose(file);

Q52. What are different modes for file opening in PHP?

Different file opening modes in PHP are:

SNo.	Mode	Description
1.	r	'Read only' mode.
2.	r+	Read/Write mode.
3.	w	Write only mode.
4.	w+	Read/Write mode.
5.	a	Append mode.
6.	a+	Read/Append mode.
7.	x	Creates the file and opens it in Write only mode.
8.	x+	Creates the file and opens it in Read/Write mode.

Q53. What is the difference between PHP fgests() and PHP fgetc()?

fgets() reads one line of text from the specified file

syntax fgets(FileObject,Length)

FileObject is an object(file pointer) that represent a file

Length it represent no of bytes to read.

This function reads one line at a time, up to 254 characters. Each line ends with newline. If length argument is omitted, PHP defaults to length of 1024

fgetc() read one character from the specified file.

syntax fgetc(FileObject)

Q54. Explain fread() & filesize() function

fread() function reads the entire file or specific size from file. It reads a number of bytes from a file,up to the end of the file.

syntax fgets(FileObject,Length)

filesize() It returns the size of file in bytes

syntax filesize(FileObject) or filesize("filename.txt")

Q56. Write PHP code to read a text file abc.txt and display alternate lines of the file on the screen.

```
<?php
        $file = fopen("abc.txt","r");
        while (!feof($file))
        {       $line = fgets($file);
                echo $line."<BR>";
                $line = fgets($file);   }
        fclose($file);
?>
```

Q57 Write PHP code to read a text file abc.txt and display the contents of the file on the screen replacing all spaces in the file by the # symbol.

```
   <?php
        $file = fopen("abc.txt","r");
        while (!feof($file))
        {       $char = fgetc($file);
                if ($char == ' ')
                        echo "#";
                else
                         echo $char;
                if (ord($char) == 13)
                        echo "<BR>";  }
      fclose($file);
?>
```

Q58. Write PHP code to read a text file str.txt and display all the lines beginning with letter "A".

```
      <?php
                $file = fopen("str.txt","r");
                while (!feof($file))
                {       $line = fgets($file);
                        if ($line[0] == 'A')
                                echo $line."<BR>";          }
                fclose($file);
        ?>
```

Q59. Write PHP code to read a text file named mega.txt and display the contents of the file after converting all the alphabets to upper case alphabets.

```
              <?php
                $file = fopen("mega.txt","r");
                while (!feof($file))
                {
                        $char = fgetc($file);
                                if (ord($char) == 13)
                                        echo "<BR>";
                                else
                                        echo strtoupper($char);
                }
                fclose($file);
        ?>
```

Q60. Write PHP code to read a text file xyz.txt and count the number of words, lines, and characters in the file.

```
    <?php
    $words = $lines = $char = 0;
    $file = fopen("xyz.txt","r");
    while (!feof($file))
    {
      $data = fgets($file);
      $lines++;  }
      fclose($file);
      $file = fopen("xyz.txt","r");
      while (!feof($file))
      { $data = fgetc($file);
            $char++;
      if ($data == ' ')
            $words++;
      if (ord($data) == 13)
            {
            $words++;
            $char--;
            $data = fgetc($file);
            }
      }
      fclose($file);
      $words++;
      echo "<P>File contains $words Words, $lines lines, and $char characters";
?>
```

Q61. Write PHP code to read a text file data.txt and count the number of vowels, consonants and other characters present in the file.

```
<?php
$file = fopen("data.txt","r");
      $vowels = $conso = $others = 0;
      while (!feof($file))
      {
            $char = fgetc($file);
            if (ord($char) != 13 && ord($char) != 10)
            {   $char = strtoupper($char);
            if($char=='A' or $char=='E' or $char=='I'  or $char=='O' or $char == 'U')
                  $vowels++;
            else if ($char >= 'A' and $char <= 'Z')
                  $conso++;
            else
                  $others++;
            }
       }
      fclose($file);
      echo "File contains $vowels vowels, $conso consonants, and $others other characters"; ?>
```

Q62. How does $_FILES variable work in PHP?

$_FILES is a superglobal which stores the details of the uploaded files. The details of the each file include

i) file name,
ii) file type,
iii) file size (in bytes),
iv) temporary location of the file on the server,

error code in case some error occurred during of the file upload

Q63. What is the purpose of move_uploaded_file() in PHP?

When a file is uploaded, it gets stored in a temporary area on the server until it is moved. The file has to be moved from that temporary area, or else it will be destroyed. The function **move_uploaded_file() moves an uploaded file to a new location on the server.**

Q64. Explain the attribute enctype? what will its value if you want to upload a file

The enctype attribute specifies how the form-data should be encoded when submitting it to the server.

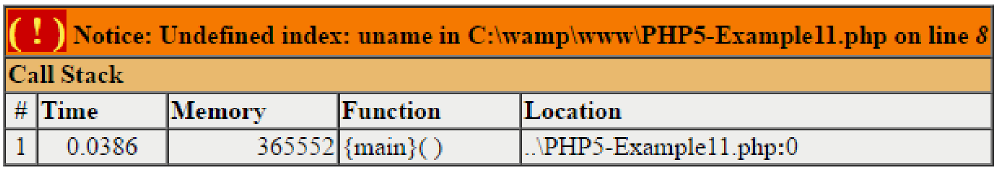
Note: The enctype attribute can be used only if method="post".

```
<form action="/upload.php" method="post" enctype="multipart/form-data">
```

Q65. How is PHP error handling done?

PHP error handling is done in two ways:

(i) **Default error handling:** The default error handling in PHP is very simple – an error message with filename, line number and a message describing the error is sent to the browser. An example is shown below:



```
( ! ) Notice: Undefined index: uname in C:\wamp\www\PHP5-Example11.php on line 8
Call Stack
# | Time   | Memory | Function  | Location
1 | 0.0386 | 365552 | {main}( ) | ..\PHP5-Example11.php:0
```

(ii) **Custom error handling:** To customize the error handling, PHP provides custom error handling techniques and these are:

- PHP die() method
- Defining custom error handling functions and error triggers

Q66. What is PHP die() method?

PHP **die() method** is used to to overrule the default error handling in PHP. die() method is used to terminate the further execution of PHP script and give a specified error message to the user.

Example:

```
<?php
if (!file_exists("myfile.txt"))
        die("File not found");
else
        $file = fopen("myfile.txt","r");
?>
```

This code will check if the file “myfile.txt” exists. If the file does not exist, then it will give a message “File not found” to the user and terminate the further execution of the script.

Q67. What is connection object? Write a PHP Code to establish a connection

Connection object is used to establish and managing connection between your application and data source.

mysqli_connect() function is used to established a connection between your application and data source.

```
        <?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";
     ?>
```

Q68 How to close database connection:

PHP provides **mysqli_close()** function to close a database. Although any open connection is automatically closed when script ends, it is always a good practice to close the database when the use of database is over.

Syntax: **mysqli_close**($dbhandle);

Q69 Explain mysqli_query():

The mysqli_query() function is a "catch all" function that can run about any MySQL query that is given to it. It can be used to create and destroy a database and tables, insert data, delete data, update data, and extract data from the database. This function takes two parameters (connection and sql) and returns TRUE on success and FALSE on failure.

Syntax: mysqli_query(connection, sql);

Where,

Argument	Description
Connection	An optional parameter that specifies the connection reference to be used. If not specified then the last opened connection of mysqli_connect() will be used.
Sql	Contains the SQL query to be executed.

Q70 How does the PHP execute SQL statements?

To get the PHP execute the SQL statement using **mysqli_query()** method

Q71. What is use of fetch_assoc() Method

This method returns an associative array that corresponds to the fetched row, the keys of the element are the name of the columns.

Q72. How to Create a database:

A database can be created using **mysqli_query()** function with suitable query.

Example:

```
<?php
   $servername = "localhost";
   $username = "root";
   $pwd = "faipskwt";
   $conn = mysqli_connect($servername,$username, $pwd);
   if (!$conn)
          die("Could not connect".mysqli_error($conn));

$sql = "Create database Library";
   if (mysqli_query($conn, $sql ))
          echo "Database created successfully";
   else
          echo "database not created ".mysqli_error($conn);

   mysqli_close($conn);
?>
```

Q73. Explain the usage of mysqli_select_db() method

It is important to select a database as there may be multiple databases residing on a server but in PHP a single connection can be linked to a single database at a time. The function mysqli_select_db() returns TRUE on success and FALSE on failure.

Syntax: mysqli_select_db(connection, databasename);

Q74. Study the give code below:

```
<?php
$myconn=__________________ ('localhost', 'root', '1234', 'myDB'); //1
if(!$myconn)
   die('could not connect: '. mysqli_error());
__________="CREATE TABLE  EmpList (EmpID  INT,  EmpName CHAR(20),   //2
/EmpSal INT,  DOJ  DATE)";
if(mysqli_query(____________ ,  $mysql))  //3
          echo " Table created";
else
          echo "No Table created";
______________($myconn);//4
?>
```

i. Fill in the blanks to complete the above code fragment to create a table called 'EmpList' in the database 'myDB'

1 **mysqli_connect** **2 $mysql** **3 $myconn** **4 mysqli_close**

ii. What does **mysqli_query( )** function do? **Same as Ans 69**

iii Write the data types used to create the table.

EmpID **INT**, EmpName **CHAR**, EmpSal **INT,** DOJ **DATE**

vi. Write the maximum character length of the data type CHAR(n) **255**

v. Which data type can be used if the character length exceeds more than 255.

VARCAHER

vi. Write the format of DATE data type. **YYYY-MM-DDQ**

Q75. Study the give code and answer the following question:

```
<?php
$myCon=mysqli_connect('localhost', 'root', '1234', 'myDB');
if(!$myCon)
        die('could not connect: '. mysqli_error());

$sql="UPDATE CLASSLIST SET NAME='John' WHERE AGE=17";
        if(mysqli_query($myCon, $sql))
        {
                if(mysqli_affected_rows($MyCon)>0)
                        echo "Record updated successfully";
                else
                        echo "No update";
        }
        else
        echo "Error update record..." . mysqli_error($myCon);
 mysqli_close($myCon);
?>
```

i. Write the name of the database connection. **$myCon**

ii. Write the SQL command used in the given code. Explain the use of it.
UPDATE CLASSLIST SET NAME='John' WHERE AGE=17
Update command is used to update the value of a column in this SQL command will change all names to john whare age is 17

iii. Write any two database functions used in the given code.
1. mysqli_query(($myCon, $sql)
2. mysqli_close($myCon)

iv. How many parameters are used in **mysqli_connect( )** function. Explain
same as Ans 67

v. Why is the **mysqli_affected_rows( )** function used?
mysqli_affected_rows() function returns the number of affected rows in the SELECT, INSERT, UPDATE, REPLACE, or DELETE query.

vi. Write the function used to execute SQL command. **mysqli_query()**

vii. Explain the use of **die( )** method in the given code. **Same as Ans 66**

Q76. Differentiate between mysqli_num_rows( ) and mysqli_affected_rows( )

mysqli_num_rows( )	mysqli_affected_rows( )
Return the no of rows return by select query in result set	**Return the no of rows afftected by query(UPDATE, INSERT,DELETE,**
It require resultset as argument	**It require connection object as arugment**

Q77. Name the function used for the following.

a. To changes the default database for the connection.
mysqli_select_db(database)

b. To closes a previously opened database connection.
mysqli_close(connectionobject)

c. To returns the last error description for the most recent function call.
mysqli_error()

d. To returns the number of affected rows in the previous MySQL operation.
mysqli_affected_rows( )

Q78 Write any four data type of MySQL to create a table.
INT, CHAR, DATE, VARCHAR, DECIMAL,BLOB

Q79. Explain 4 argument used in establishing a connection with mysqli_connect()

Argument	Description
Server	An optional parameter that contains the server or host name running the database server. If this is not specified then the default value is **localhost:3306**.
User	An optional parameter used to specify the username accessing the database. If this is not specified then default is the name of the user that owns the server process.
Password	An optional parameter used to specify the password of the user accessing the database. If it is not specified then default is an empty password.
Databasename	An optional parameter used to specify the database to be used when performing queries.

Q80. Name the function used for the following.

ii. Function is used in fread( ) function as second parameter to read and display the entire contents of the text file. **filesize()**

iii. Function is used to read single line from a file. **fgets()**

iv. Function is used to check if the end-of-file has been reached. **feof()**

v. Function is used to close an open file. **fclose()**

vi. Function is used to write to a file. **fwrite()**