Page No. 1
Date 2 12 19

UNIT 2.

PHP -- Serverside.

Comparison

	C/C++	Python	PHP
to define variable	datatype variable; int a;	variable; a;	$ variablename; a;
to initialize	variable = value; a=10	a=10.	$a=10;
to print	cout << "Hello";	print (" ")	echo " ";
to give input & store it in variable	cin >> " ";	a=input (" ");	$_POST [" "]; or $_REQUEST [" "];

⇒ dynamic - is always changing e.g. flipkart.
static - does not change e.g.

⇒ API - Application Programming Interface. connected to database 2 types 1. 3 party 2 for self

⇒ POST if Get or Post
REQUEST only if Get

⇒ Scope of variable - After declaration till where its visibility is.
from delaration till end of page.

⇒ pre processor directive - pre defined syntax c++ #include<conio.h>

What is PHP?

→ i. Php is a recursive acronym for hypertext pre processor.

ii. Php is an extremely popular, open source, scripting language most commonly used on web servers to produce dynamic pages. It allows web developers to create dynamic contents to interact with data bases.

iii. Php is basically used for developing webpage software application.

iv. Php known as server side language because Php does not get executed on your computer but the scripts are executed on computer. The results are then handed over to you & displayed in your browser.

SQL - Structured Query language.

Page No.
Date

v. Php code may be embedded into html code.
vi Php is free to download & use. Php file may contain text, html, javascript & php code. Php file have an extension '.p'
vii. It is integrated with a number of databases MySQL, Oracle, Sybase, Microsoft hpL server & many more
viii Php runs on different platforms like windows, linux, unix, mac OS.
ix Php is compatible with almost all server used today apache, IIS etc.
x. Php supports a large number of protocols such as POP3, IMAP, LDAP.

Characteristic of Php
i. Simplicity.
ii. Efficiency.
iii. Security.
iv. Flexibility.
v. Familarity.

what can Php do?
→ i. Php can generate dynamic webpages.
ii. Php can create, open, read, write, delete & close files on the server.
iii. Php can collect form data.
iv. Php can send & receive cookies.
v. Php can add, delete, modify data in your database.
vi. Php can be used to control user excess
vii. Php can encrypt data.

Page No.
Date

Php syntax.

i) Php parsing engine needs a way to differentiate Php code from other elements in the page.

ii) The Mechanism for doing so is known as escaping to Php.

iii) These are four ways to do this.

1. Canonical Php tags. <?php...>

The most universal Php tag style is <?...?>

If you use this style you can be positive that your tags will always be correctly interpreted.

2. Short Open tags.

<?...?> Short tags are shortest option you must do one of two things to enable php to recognise the tag I. choose --enable-short-tag configuration option when you are building php.

II. Set the short_open_tag setting in your php.ini file.

3. ASP style tags

<%...%> ASP style tags mimics the tags used by active server pages (ASP) to execute the code block. To use ASP style tag you will need to set the configuration in your php.ini file.

4. HTML script tags

<script language="PHP">...:</script> PHP scripts can be placed anywhere in the document.

e.g.

```
<html>
<body>
<?php
    echo "<h1>Hello World</h1>";
?>
</body>
</html>
```

Comments in PHP.

⇒ i. A comment in PHP code is a line that is not read or executed as part of your program its.

ii. Its only purpose is to be read by someone who is looking at the code.

iii. There are two types of comment.

1. **Single line comment**

They are generally used for short explanations or notes relevant to the local code. Single line comment may start with //.

2. **Multiline comment**:

They are generally used to print multiple lines in a single print statement. they are used to provide more detailed explanation when necessary. Multiline comment start with /* & ends with */.

e.g.

```
<html>
<body>
<?php
echo "<h1>Hello World</h1>";
// This is single line comment
/* This
is
multiline
comment */
?>
</body>
</html>
```

Page No.
Date

Variables in php

i. variable are used to store values.

ii. In Php a variable starts with the '$' sign, followed by name of variable.

iii. syntax:

```
$variablename;
$name=fycs
```

Naming rules of variable.

i. All variables in php are denoted with a leading '$' sign followed by name of variable.

ii. All variable must start with a letter or '_'

iii. A variable name cannot start with a number.

iv. A variable name can contain alphanumeric characters & '_' ~~letters~~ (A to Z, 0 to 9 & _) (a to z)

v. Variable names are case sensitive

vi. A variable name can have a short name like x & y or a more descriptive name like age, total marks, etc.

vii. Variable are assigned with the assignment operator with the variable on left side & the expression to be evaluated on the right.

viii Variables in php do not have intrinsic type, a variable does not know in advance whether it will be used to store number or string of characters.

ix Php does a good job of automatically converting types from one to another when necessary.

Page No.
Date

php echo and print statement.

i. Echo and print statements are interchangeably used to output data on the screen.

Echo	Print
i. Echo has no ~~return~~	i. Print has a return value of 1.
ii. Echo can take multiple parameters	ii. Print can take one argument.
iii. Echo is marginally fast	iii. Print is comparitively slower than echo.
iv. Echo statement can be used without parenthesis.	iv. Print statement can be used with or without parenthesis.
~~v. e.g. "Hello world", "Hello.World"~~ echo "Hello World" echo "Hello"."World"	v. e.g
v. Echo "Hello world" echo "Hello"." "."World"	v. Print "Hello World". Print "Hello"." "."World".

<html>
<head>
</head>
<body>
<form a
Enter a
Submit<input
</form
<?php

?>
</body>
</html>

Page No.
Date

Operators.

Array in php.

An array can store many values at once within a single variable.

ii. Arrays are complex variables that store multiple or a group of values under single variable name.

e.g. suppose you want to store colors in your php script, storing colors one by one in a variable would look like

```
$color1="red";
$color2="blue";
$color3="Yellow";
```

iii. But what if you want to store different color with different shades in a variable and this time not 3 but hundreds of color.

iv. It is quite difficult to store each color name in a separate variable. The solution to this problem is to create a php array

v. There are 3 types of arrays in php.

1. **Number / Index array:**

An array with numeric key.

2. **Associative array:**

An array where each key has its own specific value.

3. **Multidimensional Array:**

An array containing one or more arrays within itself

e.g.

```
$arrayname=array("Abc","PQR","MNO");
echo $arrayname[0]; //Abc
echo $arrayname[1]; //PQR
echo $arrayname[2]; //MNO
```

Page No.
Date

1. i. An Index or numeric array store each array element with a numeric index

ii. In this array indexes are automatically assigned & starts with zero & values can be of any type.

iii. There are 2 ways to create a numberic array:

1. The Index are automatically assigned.

e.g $arr= array ("Abc","PQR","MNO",10);

```
echo $arr[0]; // Abc
echo $arr[1]; // PQR
echo $arr[2]; // MNO
```

eg $arr[0]="ABC";
$arr[1]="PQR";
$arr[2]="MNO";

2. Index are manually assigned.

2. Associative array.

i. In this array the keys assigned to value can be user defined string.

ii. Here, array uses keys instead of numeric indexes.

e.g. $arr = array("Name" => "ABC", "Age" => 65);

```
echo $arr["Name"];
echo $arr["Age"];
```

$arr["Name"]="Abc";
$arr["Age"]=65;

Page No.
Date

i++ e.g i=2
j=i++
j=2+1
j=2 & i=3 first it assign j=i then '+'

++i e.g. i=2
j=++i
j=3
i=3 first it '+' then it assign i=3

~~j=2~~

i-- e.g. i=2
j=i--
j=2 j=2-1
i=1

--i e.g i=2
j=--i
j=2-1=j=1
i=1

if
if else
nested
while

for loop.

Syntax: (i=0 ; i<=10 , i++) {

(i=0 : initialise, i<=10 : condition, i++ : increment/decrement)

1. initialise.
2. condition if satisfied goes to body execute then increm. if not satisfies then breaks loop.

for each loop

Syntax:

```
for each (array name as value) {
    Body
}
```

e.g.

```
foreach ( $color as $value){
    echo $value;
}
```

* form handling

$_REQUEST → method: POST/GET

$_POST → method: POST

$_GET → method: GET

Get

name: txt1

akb → it add in url it does not considered the & after part.

name: txt2

POST considerés action

REQUEST considers both GET & POST.

Page No.
Date

functions

syntax:

```
function functionname (){
        Body of function
    }
```

i. function without parameter.(argument)

e.g.

```
function functionname(){
    $c=$a+$b;
}
```

ii function with parameter(argument)

e.g.

```
function add ($a,$b){
    $c =$a+$b;
    }
```

iii function with default argument.

iv function with return value

```
function functionname( ){
    return ;
```

```
e.g. fun cal ( ){
    $int =300(--)
    return $int;
```

Page No.
Date

Multidimension array: is an array in which each element can also be an array & each element in the sub array can be an array or further contains array within itself and so on.

The dimension of an array indicates the number of indices you need to select an element.

i. for a two dimensional array you need to two indices to select an element.

ii. for a three dimensional array you need three indices to select an array:

e.g. $contact =

```
array (
    array (
        "name" =>       "name" => "ABC"
        "ABC",          "email" => "abc@gmail.com"),
        "email" =>      "n
        "abc@gmail.com"),
```

1. Write a program to check if the given number is positive or negative.
2. Write a program to execute m^n
3. Write a program to sort an array.
4. Write a program to calculate the number of occurence of element in an array
5. Write a program to compute & return the squareroot of a given number.
6. Write a program to find a single number in an array that doesn't occur twice.
7. Write a program to reverse a number

Page No.
Date

Element

for only

```
<?php
$a = array(4,5,1,1,3,2,4,3,4,6,4,5);
$n=1;
$c=0;
foreach ($a as $i){
    if ($n==$i){
        $c++;
    }
}
echo $n." occurs ".$c." times";
?>
```

for all elements

```
<?php
$a = array (4,5,1,1,3,2,4,3,4,6,4,5);
$arr=array();
$index=0;
if(!in_array($i, $arr)){
foreach ($a as $i){
  $c=0;
  foreach ($a as $j){
      if($i==$j){
        $c++;
      }
  }
  echo $i." occurs ".$c." times";
  $arr[$index]=$i;
  $index++
}
}
```

* Php function.

A function is a self contained block of code that performs a specific task.

1. Creating & invoking function.

* Syntax:

```
function functionname (){
    body of function
}
```

Description:

Declaration of a users define function starts with the word function, followed by the name of function you want to create, followed by parenthesis .i.e.() and finally place functions code between curly brackets ie {}.

Example:

```
function display (){
 echo "Hello World";
}
```

2. Php function with parameters:

i. You can specify parameters when you define your function to accept input values at runtime.

ii. The parameter works like place holder variable within a function. They are replaced at runtime by the values (known as arguments) provided to the function at the time of invocation.

Syntax:

```
function functionname ($parameter1, $parameter2,.. $parameter n){
    body of function
}
```

Description

You can define as many parameters as you want however, for each parameter you specify a corresponding argument needs to be passed to the function when it is called.

Example:

```
function add ($num1, $num2) {
    $sum = $num1 + $num2;
    echo "sum of the two numbers $num1 & $num2 is :" . $sum;
}
```

Calling a function:

1. add 10, 20;

Note: An argument is a value that you pass to a function, and a parameter is the variable within the function that receives the argument.

3. Function with optional parameters & default value.

You can also create functions with optional parameters - just insert the parameter name followed by an equal sign, followed by default value

Syntax

```
function functionname ($parameter1, $parameter2 = $value) {
    body of function
}
```

Page No.
Date

Example

```
function customfont ($font, $size=1.5){
    echo "<p style='font-family:$font; font-size:
        $size'> Hellow World </p>"
}
```

Calling function

```
customfont ("Arial", 2);
custom font ("times");
```

4. Php function with returning value

A function can return a value back to the script that call the function using the return statement.

The value maybe of any type including arrays and objects.

Syntax

```
function functionname ($param1, $param2,... $paramn){
    body of function
    return ;
}
```

Example

```
function add ($num1, $num2){
    $sum = $num1 + $num2;
    return $sum;
}
```

Page No.
Date

* Php GET, POST, REQUEST or Passing information with php or form handling with php

A web browser communicates with the server typically using one of the two HTTP (Hyper text transfer protocol) method - GET, POST

i. The HTTP GET and POST methods are used to send information to the server
ii Before the browser sends the information it encodes it using a scheme called URL encoding.
iii In this scheme name/value pairs are joined with equal sign and different pairs are separated by ampersand (&)

1. GET method
   i. In GET method data is send as URL parameters that are usually strings of name and value pairs separated by ampersand (&)
ii It sends the encoded use information appended to page request. The page and encoded information are separated by ?.
In general a URL with GET data will look like
HTTP://www.example.com/action.php?name=abc&age=22
The underlined part in URL are the GET parameters and the value part are those parameters which are after the "=" sign.

iii More than one pa
the URL by conca
data via GET me
The php
access all the se

Advantages of GE
i Since the data se
in the URL it is
specific query st

Disadvantage of us
i. The GET method
information such
this are fully vi
potentially store
visited place page
ii Because the GET
variable the length
iii GET can't be
word documents to
iv The GET metho
only.

Example:

<form a
First n

</form
get GET.p
<? php
echo

iii) More than one parameter = value can be embedded in the URL by concatenating (&) ampersand. One can send simple text data via GET method

The php provides $_GET associative array to access all the sent information using GET method

Advantages of GET method.

i) Since the data sent by the get method are displayed in the URL it is possible to bookmark the page with specific query string value

Disadvantage of using GET method.

i) The GET method is not suitable for passing sensitive information such as username and password because this are fully visible in the URL query string as well as potentially stored in the client browser's memory as a visited page.

ii) Because the GET method assigns data to a server environment variable the length of URL is limited

iii) GET can't be used to send binary data like image or word documents to the server

iv) The GET method is restricted to send 1024 characters only.

Example:

```
<form action="getphp" method="GET">
First name <input type="text" name="firstname" />
<input type="submit" value="submit" />
</form>
```

get.php

```
<?php
echo "Welcome".
```

Page No.
Date

POST method

i. In POST method the data is sent to the server as the package in a separate communication with processing script.

ii. Data sent through POST method will not be visible in the URL.

iii. The php provides $_POST associative array to access all the information sent by POST.

Advantages of POST method

i. It is more secure than GET because user entered information is never visible in the URL.

ii. There is much larger limit on the amount data that can be passed and one can send text data as well as binary data (uploading a file) using POST.

Disadvantages of POST method.

Since the data sent by the POST is not visible in URL it is not possible to bookmark the page with specific query

Example:

```
<form action="post.php" method="POST">
<input type="text" name="firstname" />
<input type="submit" value="submit" />
</form>
```

post.php

```
<?php
echo
"Welcome";
$_POST["firstname"];
?>
```

Php $_REQUEST variable

i. It contains the contents of both $_GET & $_POST.

ii. The Php $_REQUEST variable can be used to get the result from form data sent with both GET and POST method