PHP

(Unit 1)

- What is PHP.
- What can PHP do
- Why PHP
- PHP Syntax
- Comment in PHP
- PHP is case sensitive
- PHP is loosely type language
- PHP variables
    - Rules
    - Output variables
    - Variable scope
    - Global variables
    - Local variables
    - Static keyword
    - Data types
- PHP Operators
- PHP Constant

(Unit 2)

- PHP Decision Statements

- **If**
- **If….else**
- **If ….else if ….else**
- **switch**

- **<u>Looping</u>**
  - **While**
  - **Do while**
  - **For**

- **<u>Array</u>**
  - **What is array**
  - **Create array**
  - **Loop through array**
  - **Multidimensional array**
  - **Sorting array'**
  - **Merging array**

- **<u>Function</u>**
  - **Why function**
  - **Defined function**
  - **Create function**
  - **Function argument**
  - **Function return type**
  - **Calling function**

- **Recursion function**

(Unit 3)

- **Class and Objects**
- **What is class & object .**
- **Creating class and object**
- **Object properties& methods**
- **Overloading , Inheritance , constructor , destructor**

- **String**
- **Creating & accessing string**
- **Formatting string**
- **Date and time**

(Unit 1)

1.What is PHP.?

- **PHP (PHP Hypertext Preprocessor)**
- **It is a server side scripting language**
- **Using php we create Dynamic webpage**
- **Php is provide a Security**
- **We can give a permission and we can restrict the user**
- **We can embedded this php in HTML code also .we can write content in HTML tag.**
- **Php is a not case sensitive language.**

2.What can PHP do

- **php can generate dynamic page content .**
- **php can create ,open ,read ,write ,delete and close file on server .**
- **php can collection of data .**
- **php can add ,delete ,modify data in your database.**
- **With php we are not limit to output html. You can output images,pdf file, flash movie.**

3.why php ?

- php runs on various platform (windows,linux,macos)

- php is compatible with almost all servers uses today (apaches , lls)

- php supports a wide range of database.

- php is easy to learn and runs efficiently on the servers side.

4.PHP Syntax

- **We save php file by .php**
- **We can start php tag like <?php (content) ?>**
- **Php file normally contains HTML tags and some php Scripting code ..**

<html>

<head>

</head>

<body>

<?php

(content)

?>

</body>

</html>

5.Comment in PHP

- A comment in php code is a line that is not read/ execute as part of the program. Its only purpose is to be read some one who looking at code.

- **Comment can be used to:**
  - **Let other understand what you doing.**

- **Remind yourself of what you did most programming have experienced coming back to there own work year or later and having to re-figure out what they did .comments can remaind you of what you were thinking when you wrote code.**

6.PHP is not case sensitive

- **In php , all keywords , class , functions are not case-sensitive.**
- **We can write the keywords class , functions are in either capital letter or small letter.**
- **“Hello” & “hello” both words are same but they are different meaning and use as a different variable or keyword or element in the php scripting language .**

7.PHP is loosely type language

- **We can did not have to tell php which type the variable is .**
- **Php automatically convert the variable to the correct data type,depending on its value.**
- **Other programming language the programmer must declare the name and type of variable before using it**

8.PHP variables

- creating (Declaring) variable in php, a variable starts with the $, followed by variable name.

Example :- $a=11;

- **Rules:-**

- **Variables starts with $**
- **A variable name always starts with letter or underscore .**
- **Variable name cant starts with number**
- **Variable name can only contain alphanumeric characters and underscore .**

- **Output variable :-**
  - **In php we use “echo”and "print" variable as a output variable for printing statement**

- **Variables scope**

- **Php variables can be declared anywhere in script.**
- **The scope of a variable is the part of script where the variable can be used.**
- **Php has different variable scope.**

a) Globle scope :-

- a variable declared out side the function is called as a globel variable

- a globle key word is used to access a globle variable from with in a function.

b) local scope :-

- a variable declared inside side the function or with in function is called as a local variable

c) Static keywords

Normally, when a function is completed /executed ,all of the variables are deleted . How ever , sometimes we want a local variables not to be deleted .we need it for future use.

To do this , use the static keyword when first declare the variable.

Example:-

```
static $a=10;
```

- **echo() and print() statement**

  - **Both statements are use to print out put.**
  - **echo statement has no return value but print have a return value.**
  - **echo is marginally faster than print statement.**
  - **Both statements are used with in () and also without ().**

(8.f) Data types

Variables can store a data of different data types can also do different things

- **Simple data types**

a. String :-

Its hold a string .

```
$x=”hello”;
```

b. Int :-

Range -2,147,483,648 to 0 to 2,147,483,647

List of digits , either +ve or –ve .

$x=1212

c. Float :-

Number with decimal point.

$x=12.1

$x=12.1221

d. Boolean :-

Boolean represent two possible value true or false .

- **Compound data types**

a. Array :-

Array stores multiple value in one single variable .

$a = array(“volvo”,”BMW”,”toyota”);

b. Object :-

Object is a data type which stores data & information on how to that data.

In php ,object must be explicitly declared.

First must declare class of object .

- **Special Data type**

a. null :-

Its has null value . A variable of data type null is variable has no value assigned to it.

Variable also be emptied by setting value null.

Example;-

$a=null;

b. Resource:-

Its not a actual data type . it is string of reference to function & resources external to php.

- **.PHP Operators**

a. Arithmetic :-

+	**addition**
-	**Substraction**
*****	**multiplication**
/	**division**
%	**module**
******	**Exponentional**

b. Assignment

X=y

X+=y

x-=y

x*=y

x/=y

x%=y

c. Comparison:-

== equal

=== equal boolen

!= not equals

<> not equals

!== not equal boolen

> grater than

< less then

>= grater equal

<= less equal

d. Increment & Decrement :-

++$x

$x++

$x-

-$x

e. Logical :-

And

Or

Xor

&&

||

!

f .String:-

. concatenation

.= concatenation assignment

g. Array operator:-

+ union

== equality

=== identity

!= inequality

<> inequality

!== non identity

h.bitwise operator

Bitwise Operators		
Example	**Name**	**Result**
`$a & $b`	And	Bits that are set in both *$a* and *$b* are set.
`$a | $b`	Or (inclusive or)	Bits that are set in either *$a* or *$b* are set.
`$a ^ $b`	Xor (exclusive or)	Bits that are set in *$a* or *$b* but not both are set.
`~ $a`	Not	Bits that are set in *$a* are not set, and vice versa.
`$a << $b`	Shift left	Shift the bits of *$a* *$b* steps to the left (each step means "multiply by two")
`$a >> $b`	Shift right	Shift the bits of *$a* *$b* steps to the right (each step means "divide by two")

10.PHP Constant

- constant is a identifier (names) for a simple value, the value can not be change during script .

- create constant

To create constant use define() function

define(name, value, case_insensitive);

- **Name = name of constant**
- **Value = value of constant**
- **case_insensitive = true or false . default is false**

(Unit 2)

11.PHP Decision Statements

- **If:**

Execute some code if one condition is true.

syntax :

```
if(condition){
}
```

Example:-

```
< ?php
        $t=date(“H”);
        if($t<"20”){
        echo”good day!”
        }?>
```

Output = good day!

- **If…else :**

Execute some code if one condition is true & another is false condition.

Example:-

```
< ?php
        $t=date(“H”);
        if($t<”20”){
        echo”good day!”
        }
        else{
        echo”good night!”
        }
?>
```

Output = good day!

- **If…elseif…else :**

Execute different code for more than two condition .

Example:-

```
< ?php
    $t=date(“H”);
    if($t<”20”){
    echo”good day!”
    }
    elseif(($t=”20”){
    echo”good evening!”
    }
    else{
    echo”good night!”} ?>
    Output = good day!
```

- **switch :**

Execute some code if one condition is true.

Example:-

```
< ?php
$favcolor=”red”;
switch($favcolor)
    case”red”:
```

```
        echo”my fav color if red”;
        break;
    case”yellow”:
        echo”my fav color if yellow”;
        break;
    case”green”:
        echo”my fav color if green”;
        break;
    default:
        echo”my fav color if black”;
                ?>
```

Output = red

12.LOOPING

1)While :-

Loop execute a block of code along the specified condition is true.

2) do…While :-

Loop execute a block of code once, it will check condition ,and repeat the loop while specified condition is true.

3) for :-

Loop is used when you know in advanced how many time the script should run.

4) forEach :-

It is use to read the values from array .

13.Array

- **What is array:-**
    - **An array is a special variable, which can hold more than one value at time.**
    - **An array can hold many values under a single name , and you can access the value by referring to an index**

- **Create an array:-**

In php, the array() function is used to create an array.

In php there are three types of array.

- **Indexed array:-**
    - **Array with a numeric index.**
- **Associate array:-**
    - **Array with a named key.**
- **Multidimensional array:-**
    - **Array containing more than one value.**

- **Indexed array:-**

  - Array with a numeric index.

  Two ways to create array .

  EXAMPLE:-

```
<? Php
    $cars = array("VOLVO","BMW","TOYTO");
    echo"I like".$cars[0];
    ?>
```

  Output= I like Volvo

- **Associative array:-**

  - Array with a named key.

  Two ways to create array .

  EXAMPLE:-

```
<? Php
    $cars = array("VOLVO"="1000","BMW"="2000","TOYTO"="2010");
   echo"key=".$x.",value=".$xvalue;
    echo "<br>"
?>
```

  Output= key= VOLVO, value =1000

  key =BMW, value =2000

key =TOYTO, value =2010

- **Get length of array:-**

  For getting length of array we use count() method.

EXAMPLE:-

```
<? Php
    $cars = array("VOLVO"="1000","BMW"="2000","TOYTO"="2010");
 echo count($cars);
?>
```

Output =3

- **Looping through array:-**

- **For loop indexed array :-**

EXAMPLE:-

```
<? Php
    $cars = array("VOLVO","BMW","TOYTO");
    $arraylength=count($cars);
     For($x=0;$x<$arraylength;$x++){
    echo"$cars($x)";
  echo"<br>";
}
?>
```

Output= Volvo

BMW

TOYTO

- **For each loop associated array :-**

EXAMPLE:-

```
<? Php
    $cars = array("VOLVO"="1000","BMW"="2000","TOYTO"="2010");
  foreach($cars as $x=$xvalue){
    echo"key=".$x.",value=".$xvalue;
  echo"<br>";
}
?>
```

Output= key= VOLVO, value =1000

key =BMW, value =2000

key =TOYTO, value =2010

- **<u>Multidimensional array:-</u>**

- **Multidimensional array which can contains one or more arrays.**
- **PHP understands multidimensional array that as two, three, four or more Level deep.**

Example:-

```
<? Php
$mbooks=array(
        array(
                "title"="grapes",
              "author"="johan",
                "pubyear"=1939),

        array(
                "title"="trial",
              "author"="franz",
                "pubyear"=1925),

        array(
                "title"="hobbit",
              "author"="charels",
                "pubyear"=1956),
        );
        echo"<pre>";
        print_r($mybooks);
        echo"</pre>";
        ?>
```


- **Accessing Array:-**

- **With help of array name**

Example : $authors=array(“john”,”franz”);

$myauther=$auther[0]; //output= john

- **With help of key name**

Example :-

$books= array(

“title”=”hobbit”,

“author”=”charels”,

“pubyear”=1956),

$mytitle =$books(“title”);//output:-“hobbit”

- **Sorting array:-**

- the element in array can be sorted alphabetical or numerical values in ascending or descending order .

Sort functions as follows;

- **Sort() = ascending order .**
- **rSort() = descending order .**
- **aSort() = ascending order according to value.**
- **kSort() = ascending order according to key.**
- **arSort() = descending order according to value.**

- **krSort() = descending order according to key.**

- **Program for rSort() ,Sort()**

```
<?php
    $cars=array("VOLVO","BMW","TOYTO");
    Sort($cars);
    $clength=count($cars);
   for($x=0;$x<clength;$x++){
  echo$cars[$x];
  echo"<br>";
}?>
```

Output=

sort() = BMW	rsort() = TOYTO
VOLVO	VOLVO
TOYTO	BMW

- **Program for aSort() ,kSort() ,arsort(),ksort()**

```
<?php
    $cars=array("VOLVO"="2000","BMW"="5000","TOYTO"="8000");
    kSort($cars);
  for($cars as $x=$xvalue)
```

```
        echo"key=".$x.",value=".$xvalue;
        echo"<br>";
    }?>
```

Output=

asort() = VOLVO =2000
BMW =5000
TOYTO=8000

arsort() = TOYTO =8000
BMW =5000
VOLVO =2000

krsort() = BMW=5000
VOLVO= 2000
TOYTO=8000

krsort() = TOYTO=8000
VOLVO=2000
BMW=5000

- **merging array:-**
  - **its combine two array and form one big arry called merging of array.**
  - **We use method/function array_merge()**
    - ***Merge two array;***

Example :- $mybook=array("title"="abc","author"="john")
$mybook1 =array("title"="xyz","author"="frazn")
$newarry= array_merge($mybook, $mybook1);

- ***Add key in aary;***

Example :- $mybook=array("title"="abc","author"="john")

$mybook = array_merge($mybook array("numpage"=654))

14.Function

A) why function ?

- **function avoid duplicate code.**
- **function make easier to eliminate error.**
- **function can be reduce length of scripting**
- **function help you to break down big project.**

B) Defined function

- **In php function we can create our own function**
- **A function is an a block of statement that can be used repeatedly in a program.**
- **function will not execute immediately when page is loading .**
- **A function can be execute by call to the function**

.

C) Create user defined function

- **the user defined function declaration starts with word function.**

Example:-

```
<? php
    Function writemsg(){
```

```
        echo”hello”;
        }
    writemsg();
?>
```

d) function Arguments:-

- Information can pass to the function with help of the arguments ,an argument is just like a variables .

- Argument is specified after the function name , inside the (). You can add as many arguments as you words . we can separate arguments with help of comma.

Example:-

```
<? php
        Function  writemsg($fame,$year){
        echo”$fame refers,born in $year<br>”;
        }
    writemsg(“ram”,”2002”);
    writemsg(“sham”,”2001”);
?>
```

Output= ram refers,born in 2002

sham refers,born in 2001

E) function return type:-

To return the value to a function we use a return() in php.

Example:-

```
<?php
    function sum($x,$y){
        $z=$x+$y;
        Return $z;
    }
echo"5+10".(5,10);
?>
```

Output= 5+10=15

F) Calling function:-

- for calling function we write the function name, followed by opening &Closing ().

- **function_name()**
- **if you need to pass arguments to function ,place them between (),separating with comma.**
  - **function_name(par1,par2)**

15.Class and Objects

a) What is class & object .

a. class :-

- **in the real word , object have a characteristics and behaviour**
- **a class is a unit of code that describes characteristics and behaviour of something or a group of things .**
- **class contains a variables , methods () , logics ,and executable statements .**

b. objects :-

- **the object is a specific instance of a class is like a variable .**
- **Class as a blueprint or factory for constructing object.**
- **a class specifies the characters that an object will have , but no necessarily that specific value to those characters.**
- **Object is construct using blueprint provided by a class, and characters have a specific value.**

B) creating class and object.

Class and object are easy to create in php.

- **class :-**
  - **To create class in php , we use a "class" keyword.**
  - **To create class we use a class keyword class followed by the classname and {}.**
  - **{} use as a block which contain a class body or executable code which is under that class**
  - **Class name always starts with a capital letter .**

- We can create number of class .

- object :-
  - To create object in php , we use a ‘new’ keyword.
  - To create object we use a new keyword class followed by the classname that you want to base object on .
  - We can assign a object to variable like another value.
  - Once the object created we can display using print_r().
  - We can create number of oject .

- **Object properties& methods .**

  - Properties :-
    - In php or OOP’s characteristic’s of class or object are known as properties.
    - Properties are much like variables , in that they have a name and value .
    - Some properties allow to change there values and some does not ..

  - Visibility Properties :-
    - There are three visibility properties in php
      - Public
      - Private
      - Protected

      (which is based on whether the variable or method is use and hide)Where

- **Accessing Properties :-**

  Property is access by using object with property name .

- **Method :-**
  - **In php or OOP's behaviour of class or object are known as method**
  - **The class action always associated with class .**
  - **Methods are much like functions, we can declare method using function statements.**
  - **The Method of class , along with properties are collectively known as member of class**

d) Overloading , inheritance , constructor , destructor .

- **Overloading :-**

- **Inheritance:-**
  - **Inheritance is a concept in which one class acquired a property of another class .**
  - **In inheritance childe class get or access property of its parent class.**
  - **We can use the keyword like extends to access the property of the parent class .**
  - **Using inheritance we can decrease the code length , we are break a code in simpler form and we can reuse a useful code without writing again and again .**

Example:-

```
<?php
    class Maths{
        $a = 10;
        $b = 11;
        echo”c=$a+$b”;
    }
    class Sub extends Maths{
        echo”d=$a-$b”;
    }
?>
```

Output=

C=21

D=-1

- **constructor:-**
  - **by creating constructor method in our class, we can cause other actions to be triggered when the object created .**
  - **to create a constructor simply add a method with special name _constructor() to your class .**
  - **php look for special method name when object is created .**

Example :-

```
<?php
    class Myname{
    function_constructor(){
```

```
echo" my name is john";
}
$obj=new Myname();
}
```

Output= my name is john

- **destructor:-**
  - **destructor is useful for tiding up on object before its removed from memory .**
  - **to create a denstructor simply add a method with special name _denstructor() to your class**
  - **php is object destructor just call before object is deleted .**

Example :-

```
<?php
    class Myname{
    function save(){
        echo" my name is john";
}
    function_destructor(){
        $this=save();
}
} $obj=new Myname();
    Unset($obj);
    $obj1=new Myname();
    die("something error");
```

16.String

- **Creating & accessing string.**

  - **Creating the string**

    Example :-

    $my_ String = “HELLO”;

  - **In above example we create a string with special property name as $my_String followed by the value .**
  - **Values always written in “ ” or in ‘ ’.**
  - **If you used string value in a “ ” it give you a couple of extra features .**
  - **If you used string value in a ‘ ’ it give string exactly typed .**
  - **Any variable name with in a string are parsed and replace with the variable value.**
  - **You can include special characters in string by escaping them.**

    List of the more common escape sequences that you can use with in double quotes things.

Sequence	meaning
\n	a line feed characters
\r	a carriage return characters.

\t	**A horizontal tab-character**
\v	**A vertical tab character**
\f	**a form feed character**
****	**opposed to start of escape sequence**
\$	**opposed to start the variable name**
*****	**as opposed to double quote making the end of string**

- **Formatting string-**

- **Date and Time**
  - **Timestamp:**
    - **is the number of seconds since January 1,1970 at 00;00:00 GMT .this is also known as the unix Timestamp.**
  - **Format the date:**

    -the first parameter in the date() function specifies how to format the date/time. It is uses letters to represent date and time formats.

    d= the day of month(01-31)

    m=current month , as number(01-12)

    y=current year in four digit

Example:-

```
<?php
    echo date("y/m/d");
```

?>

Output= 2018/7/08

- **Adding a Timestamp:**
    - **Date function specifies time span . this parameter is optional . if do not supply timestamp, the current time will be used .**
    - **Mktime() function will make timestamp for tomorrow**

- **Include():**
    - **The include() function take all the text in a specified file and copies it into the file that uses the include function.**

- **require():**
    - **The require() function is include(),except that it handles error differently.**
    - **The include() function generats a warning ehile the requires() function generats a fatal error.**

- **If you includes a file while include() function and error occurs you might grt an error message like the below.**