UNIT-1

PHP

① Introduction, overview, characteristics, Advantage, disadvantages of PHP

② Declaring Variables in PHP

③ Data types in PHP [Integer, Boolean, Double, String, Array, object, NULL, Resource]

④ Operators in PHP [precidence & Associativity]

⑤ Expressions in PHP.

⑥ Session handling in PHP

⑦ Cookies in PHP

⑧ Control Structures [Conditional statements (if, ifelse, elseif, Switch)
Control loops (while, do while, for, foreach)]

⑨ file handling in PHP (open, read, write, close)

⑩ handling file uploads

⑪ Form handling (or) form Controls (Get and post method)

⑫ Listing Directories

⑬ Scope and life time of a Variable in PHP

⑭ Connecting to database [MYSQL as a reference]

UNIT-1:

CH.NAGENDRA
SAI

1.

1) Introduction to PHP.

⇒ PHP was developed in the year 1994 by Apache group.

⇒ It is a widely used open Source general purpose Scripting language that is Especially used for web development

⇒ PHP Stands for hyper text preprocessor Earlier it is called as personal homepage.

⇒ PHP is a Server Side Scripting language. It is mainly used for form handling, database Access, web development.

⇒ It is free to download & use.

Syntax:-
```
<? php
  // code is written here
?>
```

Example:-
```
<? php
  echo "Hello";
?>
```
it is Statement which is used to generate output

* A php Script Starts with <?php and ends with ?>

* In above example we use built in PHP function "echo" to output "Hello" on web page

* php statements ends with Semi colon (;)

Overview of php:

⇒ php is a Server Side Scripting language Embedded in XHTML. It is an Alternative to CGI, ASP, ASP.NET, JSP.

JSP ↳ Java Server page

CGI ↓ Common Gateway Interface

ASP → Active Server page

⇒ the Extension to php files are .php, .php3, phtml

⇒ php makes use of dynamic typing that means there is no need to declare Variable type in php.

⇒ php has large number of library functions which makes it flexable to develop code of php

⇒ php processor works in two modes

ⓐ If php processor finds XHTML tags in the php Script then code is Simply Copied to output file.

ⓑ But when php processor finds php code in Script then code is Simply understand and output is Copied to output file.

characteristics:

① open Source: we Can download php with free of Cost in Interne

② Simplicity: Since php donot include library functions like C, C++, So its Structure is Simple.

(3) Efficiency: It uses Resource Allocation mechanisms, object oriented programming, Session management features, So it eliminates the unnecessary memory Allocation.

(4) Security: It Support Encryption functions for Security

(5) Flexability: It is very flexable language because it Can be embedded with HTML, CSS, JavaScript, XML and many other languages. and the php Code Can be run on any devices like phone, Tabs, laptops etc..

(6) object Oriented : php Support oop's features.

Advantages:

(1) It is Supported by All operating Systems like windows, linux, etc..

(2) It is Integrated with other programming languages likes HTML, CSS, Java Scripts etc..

(3) It is easy to Connect with the database to Store and retrieve data from database

(4) It is fast programming language Compared to other programming languages.

(5) php Frameworks & tools are used to protect web Applications from Attacks & threats

Disadvantages

① It is not Suitable for large Applications because its ~~maintanan~~ maintanance is difficult

② Error handling of php framework is not good.

② Declaring Variables in PHP :-

⇒ Variables are the Entities that are used for Storing the values. PHP is a dynamically typed language i.e, PHP has no type declaration.

⇒ The value Can be Assigned to the Variable in following manner,

Syntax: $variable_name = value;

⇒ If the value is not Assigned to the Variable then by default the value is "NULL". and this unassigned Variables are called unbound Variables and its null value is converted to the value 0.

Example:

```
<?php
$txt = "hello";
$x = 5;
$y = 10.5;
?>
```

→ when you Assign text Value to a Variable then put Quotes around it.

Rules that must be followed while using Variables:

① php Variables Starts with $ sign by Variable name and php Variable names are Case Sensitive. i.e php treat Capital letters and Small letters differently.

Example: $x =10;

$a =10; $A =10;

both are different

② Variables must start with letter or under Score.

Examples { $x =10 ;
$_b =10;

③ Variables must not Start with number.

④ Variables must Contain only Alpha numeric Values.

⑤ there should not be Space in the name of the Variable.

Example 2:

```
<html>
<body>
 <?php
  $x=5;
  $y=10;
     echo $x +$y;
 ?>
</body>
</html>
```

output :- 15

```
<!DOCTYPE html>
<html>
<body>
<?php
$txt = "Avanthi College";
echo "I love". $txt;
?>
</body>
</html>
```

output

```
I love Avanthi College.
```

③ DATATYPES IN PHP :-

there are four Scalar types that are used in PHP. they are

- ⓐ Integer type
- ⓑ Boolean type [~~(or)~~]
- ⓒ Double type [(or) float]
- ⓓ String type

} Scalar types

~~ⓐ Integer~~ ⇒ Variables can store data of different types, and different datatypes can do different things.

⇒ PHP also supports datatypes such as Compound type and Special type

1) Array
2) object

> Compound types

3) NULL
4) Resource

> Special types

llege.

1) Integer type:

It is used for displaying Integer Values. It is similar to long datatype in 'c'. Its size is 32 bit.

Example: $x = 5 ;

=) An integer must have at least on digit. and it must be non decimal and it Can be either positive or negative.

2) Double type:

It is used for displaying Real values.

Example: $x = 10.5;

It includes number with decimal point. It is not compulsory to have ~~numbers~~ digits before & after decimal point.

3) Boolean types:

there are two types of Values that Can be defined by Boolean type they are true & false true is Represented by 1 and false is Represent as 0.

Example: $x = true;
$x = false;

⇒ If Boolean Values are used in Context of double type then False is Represented as 0.0 and true is Represented as 1.1

String type:-

⇒ there is no character datatype in PHP. If character has to be represented then, it is Represented using String datatype.

⇒ String is Collection of characters. In php String is ~~denoted~~ defined by Single or double Quotes

⇒ The Simple php Script that Stores the String in a Variable is given below.

```
<?php
$A = "I like php";
echo $A;
?>
```

```
o/p:- I like php
```

nts as o

double type

e is Represented

If character

nted using

np String

String in a

php

following are the Various functions used for String handling

1. Strlen (string) :- find length of string
2. Strcmp (string1, string2) :- this function Compares String1 and String2
3. Strtolower (String1) :- Converts characters In string1 to lower case
4. Strtoupper (String1) :- " " " " upper case
5. trim (String 1); :- this function Eliminates the white Space from both ends of String.

1) **NULL** :- NULL is a Special datatype in PHP which can have only one value : NULL.

⇒ A variable of data type NULL is a Variable that has no Value assigned to it.

⇒ if a Variable is created without a Value, it is automatically assigned a value of NULL.

⇒ Variables Can also be emptied by Setting the value to NULL

Example:

```
<?php
$x = "sai";
$x = null;
Var_dump($x);
?>
```

It is used to dump information about Variable. ← Var_dump($x);

output: null

Resource in PHP :-

⇒ the Special resource type is not an Actual datatype. it is the Storing of a reference to functions and resources external to PHP.

Object type :-

⇒ php is object oriented programming language ie we can Create objects in php. object is Real world Entity. and this object is Compound data type.

⇒ class is a template for object and object is an instance of class. In php we can Create number of classes in a class and we can include number of objects in it. This object Contains properties and Action. In class we need to write class name Starting with Capital letter

Example:
```
<?php
class Fruit {
{
   ≡   } code is written
         here
}
?>
```
Capital letter

To declare an object of a class we need to use new statement

```
class Myclass
{
  ≡
}
$obj = new Myclass;
```

```
<?php
$obj = new stdClass;
$obj -> name = "Deepak";
$obj -> age = 21;
$obj -> marks = 75;
print_r ( $obj );
?>
```

object of stdclass is null to begin with. we can add properties dynamically

output:-

```
Std Class object (
    [name] => Deepak
    [age] => 21
    [marks] => 75
)
```

Arrays:-

⇒ Arrays in php is a type of datastructure that allows us to store multiple elements of similar datatype under a single variable there by saving us the effort of creating a different variables for every data.

⇒ Array is a collection of similar type of elements and in php we can include elements of mixed type together in single Array.

There are two ways to create Arrays in php

using Construct Array:-

```
$mylist = ~~[Array]~~
    array [10, 20, 30, 40, 50];
```

Assigning value directly to Array

```
$mylist[0] = 10;
```

An empty array Can also be Created using the array Construct.

Example: `$mylist = array();` → array function is used to Create Array

we Can have mixed type of elements in the Array.

Example: `$mylist = array ["Archana", 519, 89.23];`

there are three types of arrays in PHP.

1. Numeric Array:- Arrays with a numeric index.
2. Associative Array:- Arrays with named Keys.
3. multidimensional Array:- Arrays Containing one or more Arrays.

Example to Count Size of Array. (or) Length of array:-

```
<!DOCTYPE html>
<html>
<body>
<?php
$bikes = array ("KTM", "Bullet", "Duke");
echo Count($bikes);
?>
</body>
</html>
```

```
o/p : 3
```

Example 2 :-

```
<! DOCTYPE html>
<html>
<body>
<? php
  $bikes = array("KTM", "Bullet", "Duke");
   echo "I like". $bikes[0]. ",". $bikes[1]. "and".
                                   $bikes[2]. ".";
?>
</body>
</html>
```

O/p:- I like KTM, Bullet and Duke.

3. OPERATORS IN PHP :- 7 types

1. Arithmatic operators
2. Assignment operators
3. Logical operators
4. Equality operators
5. Comparision operators
6. Increment & decrement operators
7. Bitwise operator

① **Arithmatic operators** :- there are 5 arithmatic operators that are used to perform mathematic operators they are +, -, *, /, %

operators	Name	Example
+	Addition	$x + $y
—	Subtraction	$x - $y
*	multiplication	$x* $y
/	Division	$x / $y
%	modulus	$x % $y

(2) Assignment operators (denoted by Symbol "=")

operator	Name	Example
=	Assignment	$x = 2
+=	Addition Assignment	$x += 2
-=	Subtraction "	$x -= 2
*=	multiplication "	$x *= 2
/=	Division "	$x /= 2
%=	modulus "	$x %= 2

(3) Logical operator:-

Used to perform decision, i.e, Used to take decision

operator	Name	Example
&&	AND	$x && $y (or) $x AND $y
||	OR	$x || $y (or) $x OR $y
!	NOT	$x ! $y (or) $x NOT $y
XOR Exclusive OR	Exclusive OR	$x XOR $y

(4) EQuality operator:-

Used to Compare two Values

operator	Name	Example
==	Equal to	$x == $y
!=	Not Equal to	$x != $y
===	is identical to	$x === $y

(5) Comparision operator :-

<, >, <=, >=, ?:

?: → ternary

Example: $x < $y

(6) Increment & decrement operators:

++ , --

++ → Increment

-- → Decrement

Example: ++$x → pre Increment

$x++ → post Increment

(7) Bit wise operator:

&,	^,	|,	~,	<<,	>>
AND	XOR	OR	NOT	shift left	shift Right

Example: $x & $y

Precidence and Associativity

Associativity → means order of execution

Precidence → means priority i.e, it represent which Symbol has high priority and which Symbol has low priority

from highest priority to lowest

highest → [ ]

*, /, %

+, -

>, >=, <, <=

==, !=

&& [AND]

|| (OR) ← Lowest

operator	Associativity
+, -	R to L
*, /, %	L to R
<, <=, >, >=	L to R
= =, !=	L to R

4. EXPRESSIONS IN PHP :-

⇒ Expressions are the most important building blocks of PHP, almost anything we write is an Expression.

⇒ An Expression is defined as "Anything that has a Value"

Ex: $a = 5

⇒ Anything that appears to the right of assignment operator (=) is an expression.

Syntax:

```
$x=100; // 100 is an expression
$a = $b+$c ; // $b+$c is an expression
$c = add ($a, $b); // add ($a,$b) is expre
~~$val~~ =
$var = $x != $y ; // $x != $y is expression
```

⇒ Expression with ++ and -- operators are Called Increment and decrement operators. Both prefix and postfi ++ operators increment value of operand by 1, where as -- operators decrement by 1.

pre increment

```
<?php
$x=10;
echo++$x;

?>
```

o/p: 11

post increment

```
<?php
$x=10;
echo $x++;
echo $x;
?
```

o/p 10
11
12

pre decrement

```
<?php
$x=10;
echo --$x;

?>
```

o/p: 9

post decrement

```
<?php
$x=10;
echo $x--;
echo $x;
?>
```

o/p 10
9

⇒ php Regulor Expression are Sequence of choracters. Using Regular Expression you can find a particular String, Replace one String by another, we Can Split Single String into many parts etc..

⇒ Ternary operator has three operands. First one is logical expression. if it is TRUE, Second operand expression is evaluated otherwise third one is evaluated. Syntax: echo (condition) ?"true": "false";

$x=4

$x<5

```
<?php
$x=70;
echo ($x<20) ? pass : fail;
?>
```

5 SESSION HANDLING IN PHP :-

⇒ when you work with an Application, you open it, do Some changes and then you close it. This entire process is Known as Session.

⇒ the Computer Knows who you are, It knows when you Start the Application and when you end. But in the internet there is one problem. the web Server does'not Know who you are, or what you do, because the http address doesnot mantain State.

⇒ Session Variables Solve this problem by Storing user informati to be used across multiple pages by default, Session variables are present until user closes the browser. So, Session Variables hold information about one Single user, and are available to all pages.

⇒ A Session is Started with Session_Start() function. this function checks if a Session is already Started and if non is Started then it Start Session. This Session_Start() is put at the beginning of page.

⇒ A PHP Session Can be destroyed by using Session_destroy() function. This function doesnot need any arguments and Single call Can destroy a Single Session Variables.

```
<?php
   session_start();   → to Start Session
?>
```

⇒ Unset function is used to unset a Session Variables

```
<? php
Unset ($_ SESSION['counter']);
?>
```

```
<?php
Session_destory();
?>
```

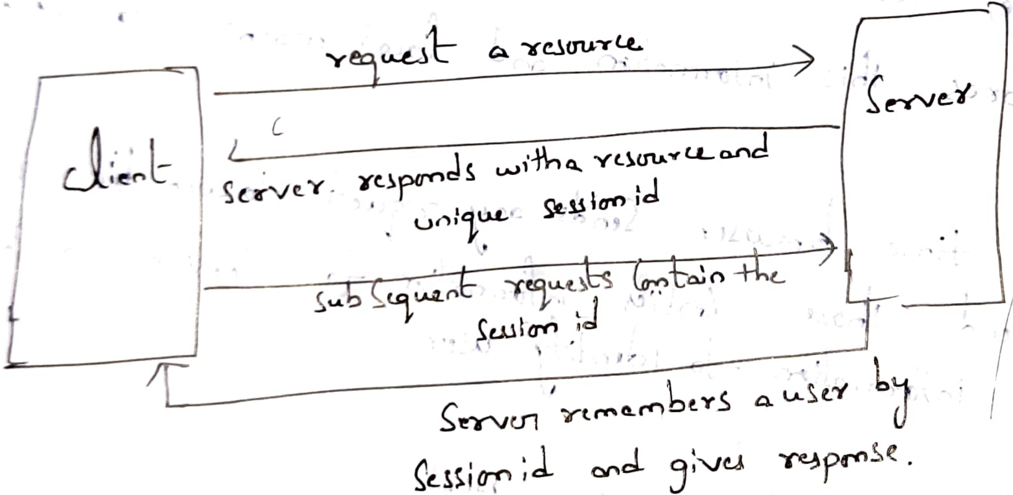

request a resource
Client
Server
Server responds with a resource and unique session id
subsequent requests Contain the Session id
Server remembers a user by Session id and gives response.

fig: Session handling.

6. COOKIES IN PHP :-

⇒ A Cookies is often used to identify a user. A Cookies is a Small file that Server embeds on the user's Computer. It is used for tracking purpose. 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 Setcookies() function

Syntax: Set Cookie(name, value, expire, path, domain);

=> only name parameter is required. All other parameters are optional.

=> PHP transparently Support HTTP Cookies. there are three Steps Involved in identifying & Returning ~~using~~ users.

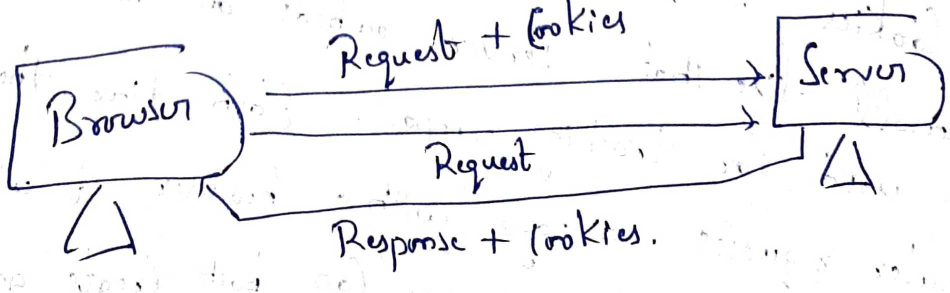
ⓐ Server Script Sends Set of Cookies to the browser. For example name, age, identification number etc..

ⓑ Browser Stores this information and local machine for future user

ⓒ when next time browser Sends any request to web Server then it Send those Cookies information to Server and Server uses that Information to identify user.

Example:

```
<?php
    $ Cookie_name = "user";
    $ Cookie_ value= "John";
    Set Cookies ( $cookie_ name, $ Cookie_value,
                  time() + (86400 * 30) );
                           30day
?>
```



7. CONTROL STRUCTURES IN PHP:

allows us to Control the flow of execution of Code. Generally Code is executed Sequentially.

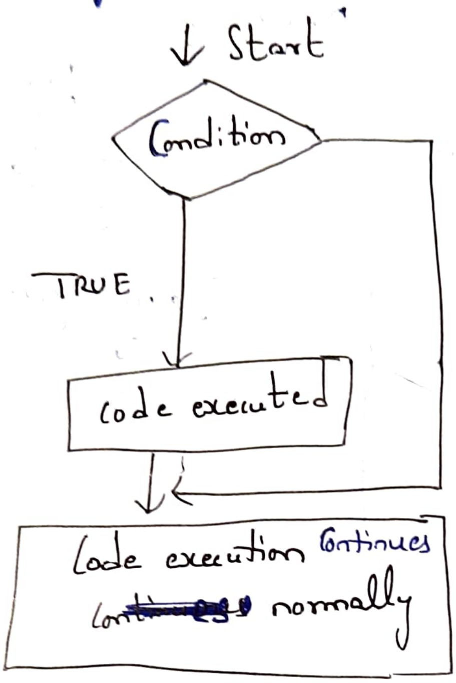


fig:- Structure of Control structures.

⇒ there are two types of Control Structures in PHP they are

ⓐ Conditional Statements

ⓑ Control loops

ⓐ Conditional Statements:

Conditional Statements are used to perform different actions based on different Conditions. There are four Conditional Statements in php they are

① If Statement
② if else statement
③ If..., elseif... else Statement (elseif statement)
④ Switch Statement.

① If Statement:

It executes some code if condition is true.

Syntax:

```
if (condition)
{
code to be executed if condition is true;
}
```

Program:

```
<?php
$age=50;
if ($age>30)
{
echo "your age is more than 30";
}
?>
```

O/p: your age is more than 30

② If else Statement:

It executes some code if a condition is true and another code if that condition is false

Syntax:

```
if (condition)
{
code to be executed if condition is true;
}
else
{
code to be executed if condition is false;
}
```

Program:

```
<?php
$t= ~~date~~ 10;
if ($t<"20")
{
echo "Have a good day!";
}
else
{
echo "Have a good night!";
}
?>
```

③ [redacted].....elseif....[redacted] Statement:

Us this Statement to Specify a new Condition to test, if first Condition is false

Syntax:

```
if (condition)
    {
code to be executed if condition is true;
    }
 else if (condition)
    {
code to be executed if first Condition
  is false and this is true;
    }
   else
    {
 Code to be executed if all Conditions
   are false
    }
```

Program:

```
<?php
$t = 40;
if ($t <"10")
{
echo " good morning!";
}
else if ($t < "20")
 {
 echo " good Afternoon!";
 }
 else
 {
 echo "good night!";
 }
 ?>
```

④ Switch Statement:

⟹ Switch Statement is a multiple choice Selection Statement

Syntax:

①

```
Switch (n)
   {
case label1:
code to be executed if n=label1;
   break;
case label2:
code to be executed if n=label2;
   break;
        :
default:
code to be executed if n is different
          from all labels;
   }
```

Program:

```
<?php
~~$ fav~~ $colour = "red";
Switch ($colour)
  {
Case "red";
echo "your colour is red!";
break;
Case "blue":
echo "your colour is blue!";
break;
case "green":
echo "your colour is green;
break;
default:
echo "your colour is not present;
  }
?>
```

~~exp~~ ~~expression~~

~~Start~~

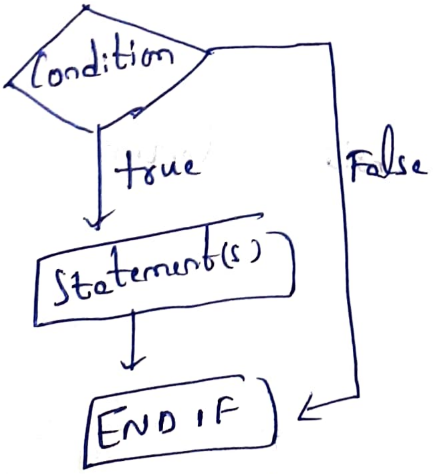


fig: if statement flowchart

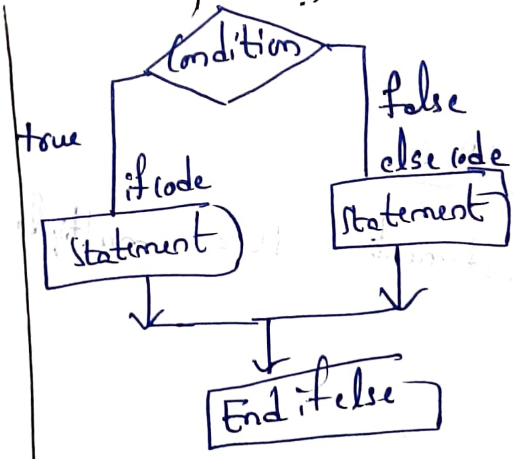


fig: if else flowchart

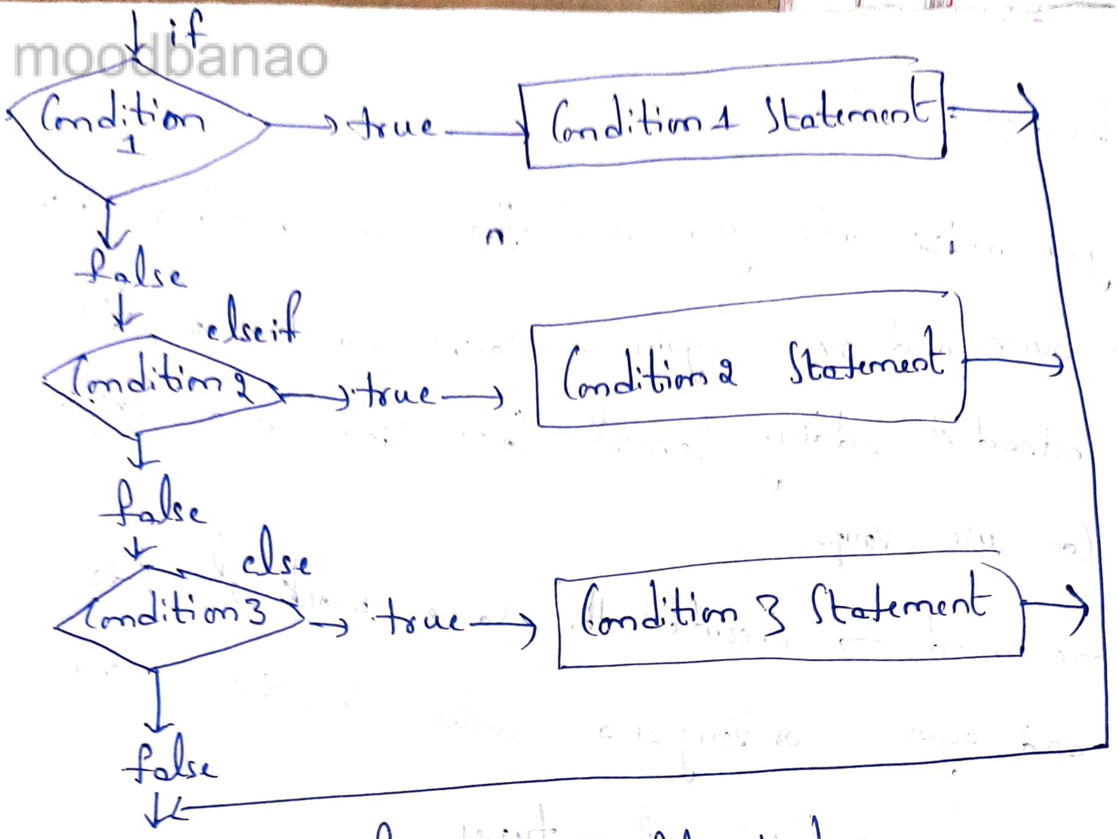
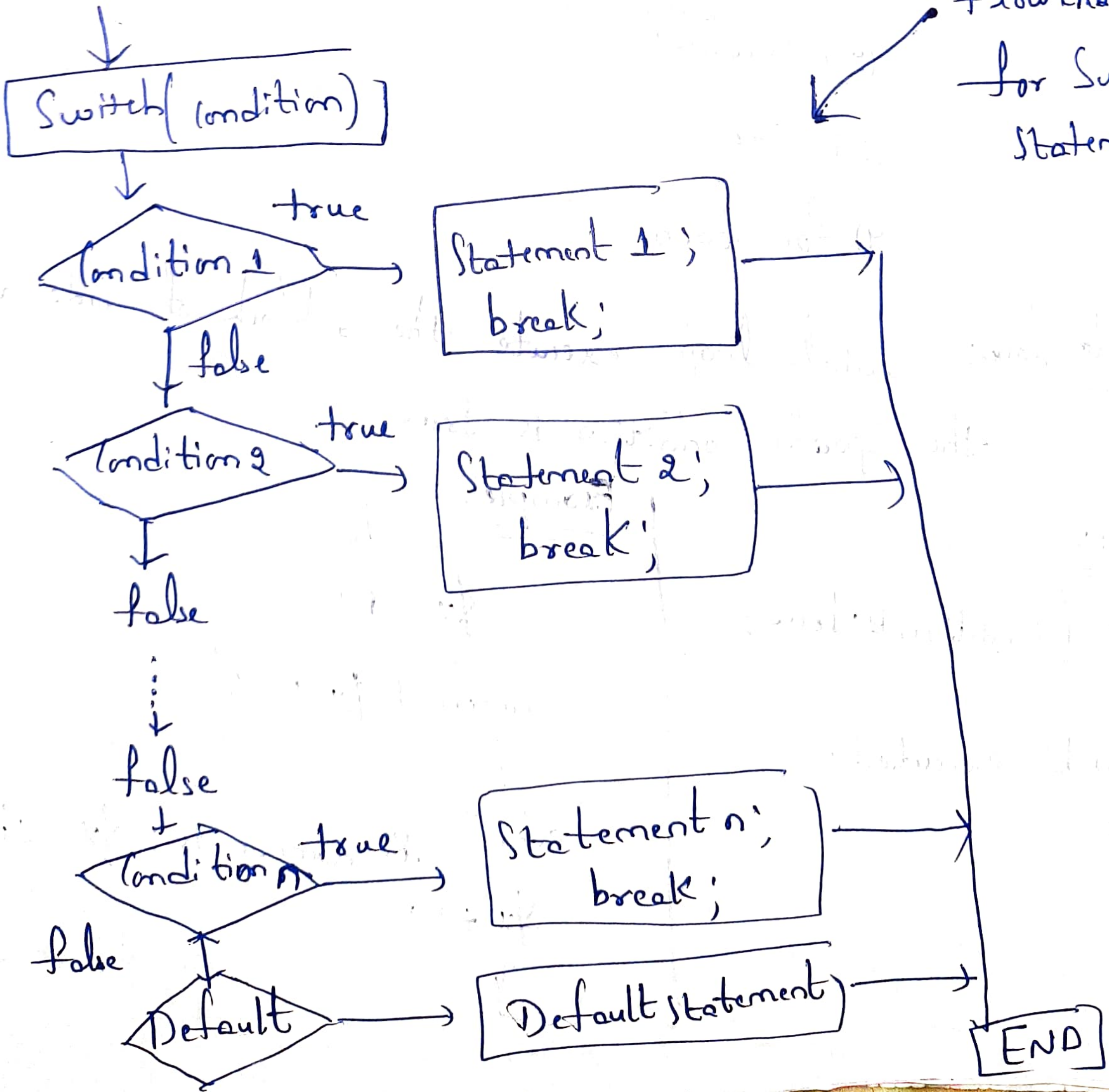


fig:- elseif flowchart



flowchart for Switch Statement.

(1) (2) Control loops:

⟹ when you write Code, you want the Same block of Code to run over and over again a Certain number of times. So, instead of adding Several Code lines in a Script. we Can use loops.

⟹ loops are used to execute the Same block of Code again and again, as long as a Certain Condition is true.

⟹ loops are classified into four types

1. while
2. do while
3. for
4. for each

(1) while loop:- while loops execute the a block of Code as long as the Specific Condition is true.

Syntax:

```
while (Condition is true)
{
Code to be executed;
}
```

Example:

```
<?php
   $x = 1;
  while ($x <= 5)
  {
 echo " The number is : $x <br>";
  $x ++;
  }
  ?>
```

output: 1,2,3,4,5

(2) do-while loops:

do while loops through a block of code once, and then repeats the loop as ~~th~~ long as the Specific Condition is true

Syntax:

```
do
{
Condition to be executed;
}
while( Condition is true);
```

Program:

```
<?php
$x = 1;
do
{
echo " The number is $x <br>";
$x++;
}. while ( $x<=5);
?>
```

O/p: ~~1,2,3,4~~,5

```
this number is 1
"       "      2
"       "      3
"       "      4
               5
```

(3) For loop:

For loop through block of code a Specific number of times

Syntax:

```
for( Init Counter ; test Counter ; Increment Counter)
{
Code to be executed for each iteration;
}
```

Example:

```
<?php
for ( $x=0 ; $x <=10 ; $x++);
{
echo " the number is $x: <br> " ;
}
?>
```

output:

```
the number is 0
"    "     "  1
"    "     "  2
        :
"  ........  10
```

(4) for each:

⇒ for each loops through a blok of code for each element in an array

⇒ for each loop works only on Array, and is used to loop through each Key/value pair in an array

Syntax:

```
for each ($array and $value)
{
Code to be executed;
}
```

Program:

```
<?php
$colors = array ("red", "green",
          "blue", "yellow");

for each ($colors as $value)
{
echo "$value <br>";
}
?>
```

O/P:

```
red
green
blue
yellow
```

8 FILE HANDLING IN PHP

File handling is an important part of any web Application For different tasks we need to open and process a file.

⇒ php has several functions for Creating, reading, uploadi editing etc..

File handling in PHP is Similar as file handling done by using any programming languages like c.

PHP has many functions to work with normal files. Those functions are

① fopen():
↓
file open.

⇒ PHP fopen() function is used to open file. It takes two parameters.

function

Contains name of file to be opened.

mode

It includes in which mode file should be opened

Example:

```
<?php
$file=fopen("f.txt","w");
}
```

modes:

r → read only
w → write only
r+ → read/write
w+ → Read/write.

a → Append
a+ → Read/Append

② fread():

After file is opened using fopen() the Contents of data are read using fread().

It takes two arguments

① pointer to file

② file size [in bytes]

Syntax: $filedata = fread ($file, $size);

⇒ we Can also use other functions like fgets() & fgetc() to read functions from file

③ fwrite()

⇒ New file Can be Created or text Can be appended to an existing file using fwrite() function.

⇒ fwrite() and fputs () are functions used to write data into file

modes: w, r+, w, w+, x, x+...... etc..

④ fclose():

file is closed using fclose() function. Its argument is file which needs to be closed.

Example: fclose ($file);

⑤ ~~Dele~~ Unlink():

Used to delete the file. only one parameter is Accepted i.e, file name. Example: unlink($file) if file is Successfully deleted it display TRUE else it Return false.

9 HANDLING FILE UPLOADS :

⇒ PHP allows you to upload Single and multiple files through few lines of code

⇒ PHP file upload features allows you to upload binary and text files both. moreover, you Can have the full Control over the file which you have uploaded by using PHP.

⇒ PHP also Supports authentication & file operation functions

PHP $_FILES :-

The PHP global $_FILES Contains all the information of file By the help of $_FILE global, we Can get filename, filetype, filesize, temp file name and errors associated with file.

Here, in this example, we are assuming the file name is filename.

$_FILES ['filename'] ['name'] :- It is used to return the name of the file

$_FILES ['filename'] ['type'] :- " " " MIME type " " (multipurpose Internet mail extensions)

$_FILES ['filename'] ['size'] :- " " Size " "

$_FILES ['filename'] ['tmp_name'] :- " " " temporary " "

$_FILES ['filename'] ['error'] :- " " errors associated with the file

move_uploaded_file() function:

⇒ the move_uploaded_file() function moves the uploaded file to a new location. the move_uploaded_file() function checks internally if the file is uploaded through POST Request. It moves the file if it is uploaded through post Request.

Syntax: move_uploaded_file (String $filename, String ~~$[illegible]~~ $destination)

10. FORM HANDLING (OR) FORM CONTROLS :-

⇒ we Can Create and use forms in PHP.

⇒ To get the data present in the form, we need to use PHP Superglobals. They are

① $_GET ~~form~~ i.e, GET method } used to collect form data.
② $_POST ~~form~~ i.e, POST Method

⇒ we Can Send form request by using get() or post() methods.

⇒ To retrieve data by using get method we need to use $_GET
" " post " " " $_PO

⇒ " " "

, Internal

PHP Post form :-

post request is widely used to Submit form that have large amount of data Such as file uploads, image uploads, login forms, registration forms etc.

⇒ The data passed through post request is not Visible on the URL browser. So it is Secured. you Can Send large amount of data through post request.

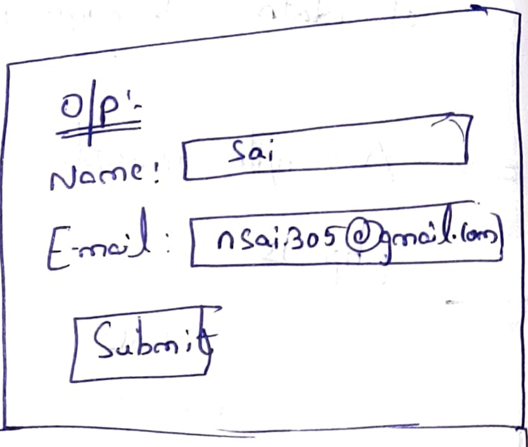
⇒ A Simple example to receive data from post request in php is

file.html.

```
<html>
<body>
<form action = " Data.php" method = "post" >
Name : <input type = " text " name = " name " > <br>
Email : < input type = "text " name = "email " > <br>
<input type = "Submit" Value = "Submit">

</form>
</body>
</html>
```

```
<html>
<body>
welcome
<?php
echo $_Post ["name"];
?>
<br>
your email address is :
<?php
echo $_POST ["email"];
?>
</body>
</html>
```

← Data.php file.



o/p:-

Name: Sai

E-mail: nsai305@gmail.com

Submit

```
welcome Sai
your email address is :
nsai305@gmail.com
```

PHP Get method:

This Get Request is the default form Request. the data passed through get request is visible on the URL browser, So it is not Secure. you need to Send limited amount of data through get Request.

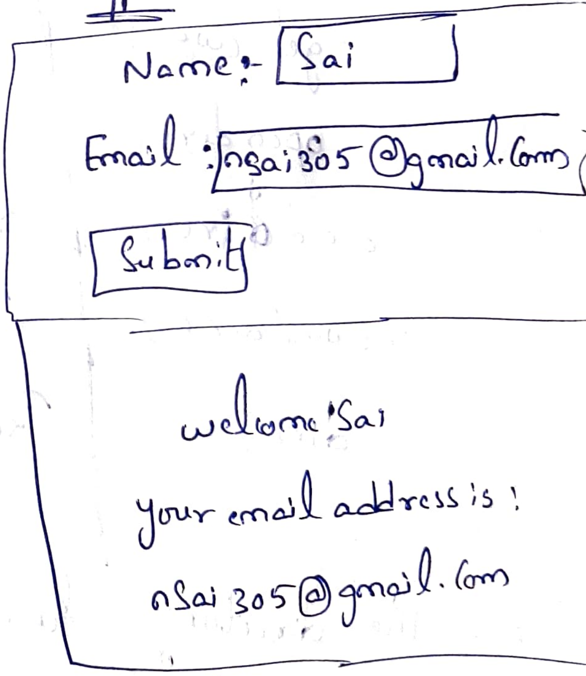
file.html

```
<html>
<body>
<form action = " Data . php   "      method = " get . " >

Name : < input type = " text  "     name = " name   " > <br>

Email ; < input type = " text  "    name = " email " > <br>
        < input type = " Submit " ~~action~~ = "Submit" >

   </form>
    </body>
     </html>
```

Data.php file.

```
<html>
 <body>
 welcome
   <?php
   echo $_get ["name"];

   ?>
   <br>
   your email address is :
    <?php
    echo $_get ["email"] ;
     ?>
   </body>
   </html>
```

O/P:



Name :- Sai

Email : nsai305@gmail.Com

Submit

welcome Sai

your email address is :

nSai 305@gmail.Com

11. LISTING DIRECTORIES

⇒ PHP provides a Support of directory functions that ~~are~~ allows you to retrieve information about directories and its Contents.

⇒ Various directory functions are given below.

Function	description
ch dir()	change Current directory
chroot()	" " root "
close dir()	close Current working directory
get cwd()	Return " "
open dir()	open directory handle
readdir()	read an entry of directory
rewinddir()	Reset a directory handle.
dir()	Returns an instance of the directory class

⇒ The directory functions are part of PHP Core. No installation is required to use these functions.

⇒ The opendir and readdir are the two important directory functions that are commonly used while listing the Contents of a directory. Let us discuss Syntax of the

1) **opendir:**

parameters

Syntax: opendir (path, Context);

path → denotes path of Specific directory

Context → (optional) used to denote behaviour of stream

2) **Read dir:**

Syntax: readdir (handle);

handle → optional parameter.
It denotes directory handle resource which is opened with opendir( )

3) **close dir:**

Syntax: closedir (handle);

handle → optional parameter.

It denotes directory handle resource which is opened with opendir( )

12. SCOPE AND LIFETIME OF VARIABLES IN PHP

Scope of a Variables:-

Variables are of ~~two~~ three types

1. local Variables :- Variables declared inside a function.
2. global Variables :- " " outside " "
3. Static

⟹ Global Variables Can be accessed anywhere in the Program. To access global Variables in a function, it must be declared in a global decleration of function using global Keyword. otherwise, the Variable acts as a local Variable.

```
<?php
$a=1;
$b=2;
function Sum()
{
global $a, $b;
$b = $a + $b;
}
Sum();
echo $b;
?>
```

if you want to define global Variable Inside a function we use global Keyword.

output:- 3

Static

Normally, when a function is Completed / executed. all of its Variables are deleted. However, Sometimes we want a local variable NOT to be deleted. To do this we use Static Keyword.

Ex:- Static $x=10;

Lifetime of Variable:

=> The life time of local Variable in php function begin when Variable is first used and ends when the function execution terminates i.e after termination of function ~~exect~~ execution, the value of local Variable is lost.

=> The life time of a Static local Variable in a function begin when the Variable is ~~function~~ first used in function, and ends when the Script Execution terminates

∴ This Static local Variables is Specified using 'Static Keyword'.

13. CONNECTING TO DATABASE [MYSQL as a reference]

=> MYSQL is open Source database product and Can be downloaded from the ~~wed~~ internet.

=> MYSQL is a Kind of database in which the records are Stored in an entity Called table. In the table the data is arranged in the rows and Columns. we Can query a database to retrieve particular information. Query is a request or a question for database.

=> PHP is a Server side Scripting language and it has an ability to Create dynamic pages with Customized features

Using PHP- MYSQL user friendly and interactive web sites Can be Created.

⇒ Both PHP and MYSQL are open Source technologies that work hand-in-hand to Create rich internet Applications. and it is highly Secure.

⇒ PHP-MYSQL are Stable technologies and have Cross platform Capability. Hence the web application developed using these technologies become portable.

⇒ Since HTML is embedded with PHP, there is no need to write Separate Code for web-Scripting.

⇒ PHP has got a Strong Support for developing e-Commerce applications using the technologies Such as Ecommerce, Drupal and Soon.

⇒ The most popular websites being developed using PHP and MY SQL technologies are

① Facebook

② wordpress

③ wiki pedia

④ Yahoo.

⇒ mysql_connect() function is used to establish a Connection between PHP and MYSQL.

Functions in MYSQL:

to Create data base: function mysql_query is used.

for errors in Commands: " mysql_error() " "

Selecting database: " mysql_Select_db() " "

Listing Database: " " mysql_list_db() " .

⇒ Tables are present inside the database .
function mysql_list_table function is used to display tables present in database, we Can also use mysql_query. we Can perform Various operations like

① Create a table
② Inserting data into the table
③ Altering table
④ Deleting table.