

```
        'role_id' => $role_details['id'],
        'resource_id' => $resource_details['id'],
    );
    if ( $this->rule_exists( $resource_details['id'], $role_details['id'] ) ) {
        if ( $access == false ) {
            // Remove the rule as there is currently no need for it
            $details['access'] = !$access;
            $this->_sql->delete( 'acl_rules', $details );
        } else {
            // Update the rule with the new access value
            $this->_sql->update( 'acl_rules', array( 'access' => $access ) );
        }
    }
    foreach( $this->rules as $key => $value ) {
        if ( $details['role_id'] == $value['role_id'] && $details['resource_id'] == $value['resource_id'] ) {
            if ( $access == false ) {
                unset( $this->rules[ $key ] );
            } else {
                $this->rules[ $key ]['access'] = $access;
            }
        }
    }
}
```

www.engineersworld.in



1. Introduction Of PHP	4
2. PHP Environment	4
3. Basic Structure Of PHP	4-5
4. Comment In PHP	5-6
5. echo	6
6. print() Function	6-7
7. variable	7
8. data type	8
9. operator	9-10
10. Decision Control Statement	
• What Is Decision Control Statement	11
• If Statement	11-12
• If else Statement	12-13
• If else if statement	14-16
• Switch case	17-19
11. Looping Statement	
• What Is Loop	20
• For loop	20
• While Loop	20-21
• Do-While Loop	21-22
• Foreach	22
12. Break Statement	23
13. Continue Statement	24
14. PHP With HTML	24
15. Array	25
16. String	
• What Is String	26
• String Concatenation	26
• strlen() Function	27
• strpos() Function	27
17. POST Array	27-28
18. GET Array	28-29
19. Function	
• What Is Function	30-31
• How To Pass Argument In Function	32
• Return value form Function	32
20. header function	33-34
21. include function	32
22. required Function	33
23. Mail function	34
24. File Uploading	35-36
25. Date And Time	
• time() Function	37
• getdate() Function	37-38

• date()	38-39
26. Session In Php	
• What Is Session	49
• session_start();	40
• \$_SESSION[] Variable	40-42
• session_destroy();	42-44
26. File Handling	
• What Is File	45
• What Is File Handling	45
• fopen() Function	45-46
• File Opening Mode	46-47
• fclose() Function	47
• fwrite() Function	47
• fgetc() Function	48
• fgets() Function	48-49
• filesize() Function	49
• file_exists() Function	50

Introduction of PHP :

PHP एक server side scripting Language है PHP का पूरा नाम Hypertext Preprocessor है जिसका use website को danamic Manage करने के लिए किया जाता है website को danamic बनाने में data Base का use किया जाता है php के साथ भी विभिन्न प्रकार के Database (mysql,orecal, PostgreSQL,,Microsoft sql server) आदि Database का उसे किया जा सकता है

PHP Environment :

PHP एक server side scripting Language है इसके program को रन करने के लिए एक server की आवश्यकता होती है इस लिए php का उसे करने के लिए आपको सबसे पहले अपने computer पर एक server को instal करना होगा

PHP के लिए बहुत से offline server उपलब्ध है जिनका use कर के आप अपनी PHP website को चैक क्र सकते है

PHP के लिए offline server निम्नलिखित है

- 1 . WAMPP (Winos,Apache,MySQL,PHP,Parl)
- 2 . XAMPP (x-window, Apache,MySQL,PHP,Parl)
- 3 . MAMPP (MAC, Apache,MySQL,PHP,Parl)

इन server में से किसी एक server का उसे कर के आप php को उसे कर सकते है इन सभी server को download करने के लिए लिंक आगे दी गई है जहा से आप इन server को download कर सकते है

Basic Structure Of PHP Program :

किसी भी php program को बनाने से पहले आपको एक php फाइल बनानी होती है जिसमे आप php program लिखेगे php फाइल का extencen .php होता है

php program को लिखने के लिए आपको php opening और closeing टैग का उसे करना पढता है

php program के opening और closeing टैग निम्नलिखित है

1. <?php ?> :

यह सबसे बेसिक टैग है जिसका use ज्यादा किया जाता है इसे `<?php` से शुरू करते हैं तथा `?>` पर अंत करते हैं इस के बिच में php program लिखते हैं

2. `<script language="php"></script>` :

is तरह के टैग का उसे के भी आप php प्रोग्राम को लिखा सकते हैं इसे `<script language="php">` से शुरू करते हैं तथा `</script>` पर अंत करते हैं यह बिलकुल javascript की तरह होते हैं

3. `<% %>` :

आप jsp की तरह भी php में opening और closing टैग का उसे कर सकते हैं इसे `<%` से शुरू तथा `%>` पर अंत किया जाता है तथा इसके बिच में php program को लिखा जाता है

Comment In PHP :

php में comment का use किया जाता है comment का use इस लिए किया जाता है की यदि हम कोई project बना रहे हैं तो उस project के किस भाग में कोन सा कार्य हो रहा है इस बात का संकेत दिया जाता है जिससे यदि उस project पर कोई और व्यक्ति कार्य करता है तो उसे project को समझने में आसानी होती है php में comment इसे statement होते हैं जो compile या interpreted नहीं होते हैं php में comment निम्नलिखित प्रकार की होती है

1. Single Line Comment :

single liine comment का use केवल एक सिंगल लाइन को comment करने के लिए किया जाता है इसका use करने के लिए जिस statement को comment करना है उसके पहले `//` लगा देते हैं जिससे वह statement comment हो जाता है

Example :

```
<?php
//This is a Hello Word Program
echo "Hello";
?>
```

2. Multi Line Comment :

इस comment का उसे एक से अधिक लाइन को comment करने के लिए किया जाता है इसे /* से शुरू करते हैं तथा */ पर अंत करते हैं Multi Line comment कहते हैं

Example :

```
<?php
/*This is a Hello Word Program
using echo statement */
echo "Hello";
?>
```

echo statement :

जब भी हम php का use कर के Screen पर Message को print करना चाहते हैं तो हम **echo** का उसे करते हैं echo का use कर के Message को Screen पर print किया जाता है

Example :

```
<?php
echo "Hello";
?>
```

print() Function :

print php का एक library Function है जिसका उसे screen पर Message को print करने के लिए किया जाता है

Syntax :

```
print("msg");
```

Example :

```
<?php
print("Hello");
?>
```

Variable :

जैसा की हम जानते हैं की किसी भी program में किसी न किसी प्रकार के data का use किया जाता है उस data को स्टोर करने के लिए variable का उसे

किया जाता है variable में data तब तक रहता है जब तक program चलता है program के अंत होने के बाद variable भी नष्ट हो जाता है variable कहलाते हैं

php में variable को डिक्लेअर करने का Syntax :

```
<?php
    $variable_name;
?>
```

php में variable की पहचान \$ डॉलर Symbol से होती है php में variable को create करने के लिए डॉलर symbol का उसे किया जाता है

Example :

```
<?php
    $a=10;
    $b=5;
    $c=$a+$b;
    echo "Sum =".$c;
?>
```

यह पर (.) dot एक concatenation Operator है जिसका उसे दो statement को जोड़ने के लिए करते हैं

Data Type :

php में data type उसी प्रकार के होते हैं जैसे की अन्य language में होते हैं data type का use data के type को प्रदर्शित करने के लिए किया जाता है

php में data type निम्नलिखित प्रकार के होते हैं

1. Integer :

इस data type का use पूर्णक Value को store करने के लिए किया जाता है

Integer value का Example :

```
2,101,1255;
```

इस तरह के value Integer value कहलाती है जो सिंपल Number हो

2. float :

इस data type का उसे डेसीमल value को store करने के लिए किया जाता है

डेसीमल value को real value भी कहते हैं जिन value में dot का उसे होता है वे value float value कहलाती हैं

float value का Example :

101.42 ,87.20

इस तरह की value float value होती है जिनको real value भी कहते हैं

3. character :

character का use character को store करने के लिए किया जाता है

character को single inverted ' ' में लिखते हैं

Example :

'e','n','n','g','i','n','e','e','r','s' 'w','o','r','l','d'

इस तरह से php में character का उसे किया जाता है

4 . boolean :

boolean data type में 2 value होती है true या false इन दो value के अनुसार ही boolean data type work करता है boolean data type में true का मतलब 1 होता है तथा false का मतलब 0 होता है

Operator In php :

php में operator का use data किसी न किसी प्रकार के operation को परफॉर्म करने के लिए किया जाता है php में अलग - अलग प्रकार के data पर operation परफॉर्म करने के लिए अलग - अलग प्रकार के operator होते हैं php में operator निम्नलिखित प्रकार के होते हैं

1. Arithmetic Operator

इसका उपयोग Methometric कैलकुलेशन करने के लिए किया जाता है इसके लिए दो या दो से अधिक operand की आवश्यकता होती है इसे Binnery Operator भी कहते हैं

Addition	+
Subtraction	-
Multiply	*

Division

/

2. Relational Operator

रिलेशनल ऑपरेटर का use दो operands की value को comparer करने के लिए किया जाता है

Grator Then	<
Less Then	>
Grator Then equal	>=
Less Then equal	<=
Not Equal	!=

3. Logical Operator

एक से अधिक शर्तों की परीक्षा करके किसी निर्णय तक पहुंचने के लिए इन operator का use किया जाता है

Logical Operator तिन प्रकार के होते हैं

AND	&&
OR	
NOT	!!

4. Assignment Operator

php में किसी value का कैलकुलेशन करने के बाद use किसी verable में असाइन करने के लिए Assignment ऑपरेटर का use किया जाता है

EX :

x=a+b ;

5. Conditional Operator

php में इस ओपरेटर को टर्नरी ऑपरेटर कहते हैं इसके साथ दो चिन्ह ? और : का use किया जाता है Conditional Operator

में कम से कम तिन operand की आवश्यकता होती है

Ex :

```
x= a :b;
```

Decision Control Statement :

php में कुछ Statement का use किया जाता है जिसमे कोई न कोई condition दी जाती है decision Control statement में condition के सत्य होने पर ही इन statement के अन्दर दिये गये statement का ब्लॉक एक्जीक्यूट होता है decision control statement में दी गई condition का मान true या false होता है यदि दी गई condition true है तब decision control statement के अन्दर दिये गये कथनों का ब्लॉक एक्जीक्यूट होता है अगर condition false है तब नियंत्रण decision control statement के अन्दर दिये गये कथनों में न जाकर प्रोग्राम के अगले कथनों पर चला जाता है इन कथनों का प्रयोग करने से true या false के आधार पर नियंत्रण के लिए दो मार्ग होते हैं दो मार्गों में से कोई एक मार्ग प्रोग्राम के execution के लिए condition पर आधारित होता है जिसके लिए दो condition का प्रयोग किया जाता है true और false

php में decision control statement निम्नलिखित है -

- if statement,
- if else statement,
- nested if Statement ,
- Switch Statement

If Statement :

if statement के सामान्य दो भाग होते हैं सामान्य if statement एवं if else statement php में इन्हें नियंत्रण या निर्णय करने वाले statement कहते हैं if statement पहले test expression की testing करता है फिर उसके true और false के आधार पर यह दो भागों में से किसी एक का चयन करता आगे बढ़ जा है इस तरह यह मुख्यतः द्विमार्गीय पद्धति का सहारा लेता है if statement का use निम्नलिखित प्रकार से किया जा सकता है

सामान्य: if statement में कंप्यूटर if से start होने वाले एक्सप्रेशन या शर्त की परीक्षा करता है यही true है तो वह statement block में जा कर परिणाम अंकित करता है यही false है तो वह use छोड़ कर वह दूसरे statement को test करता है इसका फ्लोचार्ट निचे दर्शाया गया है

Syntax :

```
if(condition)
{
    Statement;
}
```

Example Of If Statement :

```
<?php
$a=4;
if($a%2==0)
{
    echo "Even No";
}
?>
```

if-else statement

if...else statement if statement का ही एक विस्तृत रूप है इसमें condition के true आउट false होने पर एक statement या कई statement का ग्रुप एक्सीक्यूट होता है if else statement का सिंटेक्स निचे दिया गया है

```
if(condition)
{
    Statement;
}
Else
{
    Statement;
}
```

If else statement का फलोचार्ट निचे इया गया है इसमें यदि condition true है तो statement ब्लॉक 1 एक्सीक्यूट होगा नहीं तो condition false होने पर statement block 2 एक्सीक्यूट होगा if else statement का फलोचार्ट

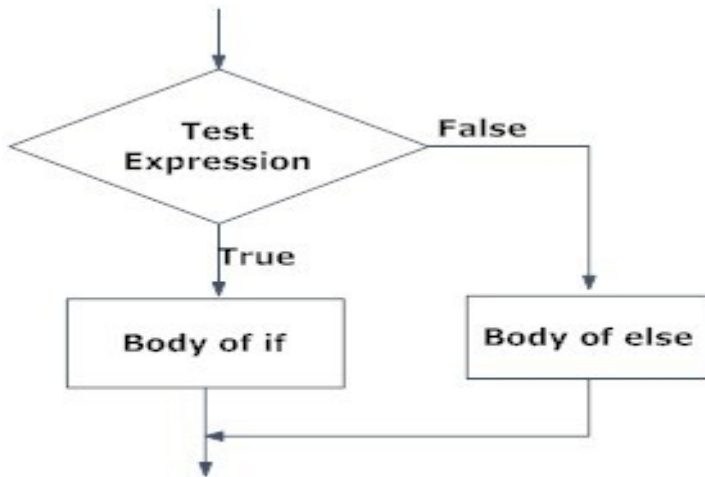


Fig: Operation of if...else statement

Example Of If else Statement

<?php

```
$a=7;
if($a%2==0)
{
    echo "Even No";
}
else
{
    echo "Odd No";
}
?>
```

Nested if else Statement

if और nested if else statement Control और decision statement है यह statement प्रोग्राम की दिशा निर्धारित करते है
nested if statement का use उस Condition में किया जाता है जब एक ही प्रोग्राम में if else के कई statement एक दुसरे से गुथे हुए रूप में प्रस्तुत किये गई हो तब इसे nested if else statement कहा जाता है इसका सामान्य syntax निचे दर्शया गय है

```
if(condition1)
{
    Statement;
}
else if(condition2)
{
    Statement;
}
else if(condition n)
{
    Statement;
}
```

nested if statement को निचे दिए गई फ्लोचार्ट के द्वारा भी समझ सकते है

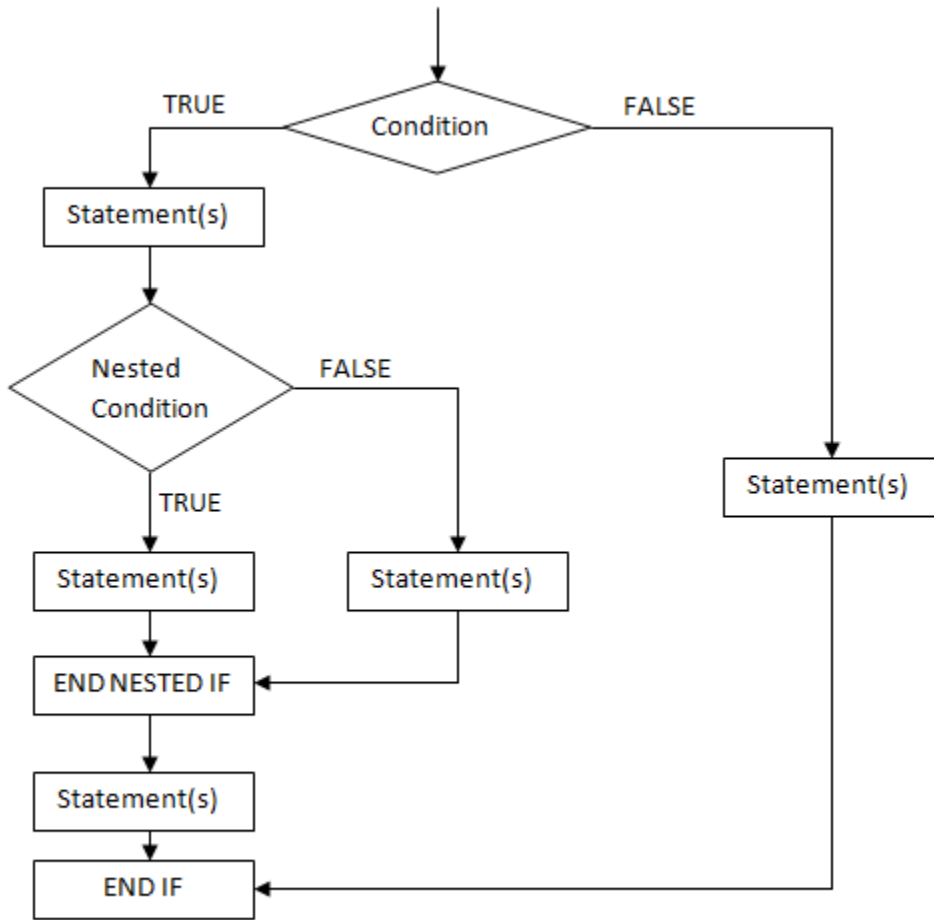


fig: Flowchart for nested if statement

Example Of nested If Statement

```
<?php  
$a=8;  
$b=10;  
$c=1;  
if($a>$b)  
{  
    echo "A Is Big";  
}  
else if($b>$c)
```

```
{
    echo "B Is Big";
}
else if($b>$c)
{
    echo "C is Big";
}
else
{
    echo "A Is Big";
}
?>
```

Switch Statement

php में कभी - कभी एक ही प्रोग्राम में if else का रूप इंतना सघन हो जाता है की प्रोग्रामर स्वय ही भ्रमित हो जाता है उन स्थिति में भ्रान्ति से बचने के लिए switch statement का प्रयोग किया जाता है इसमें विभिन्न कार्य करने के लीये statement को अलग - अलग Group में बाट दिया जाता है व हर group को एक case-label के अंतर्गत रखा जाता है switch statement में कोई संख्या या करेक्टर variable का मन दिया जाता है जिसमे सभी जिसे सभी case -labels के साथ मेच किया जाता है जिस case label का मान switch

statement में दिए गए मान से मेच करा जाता है उसी के अंतर्गत लिखे गए statement Execute किये हो जाते

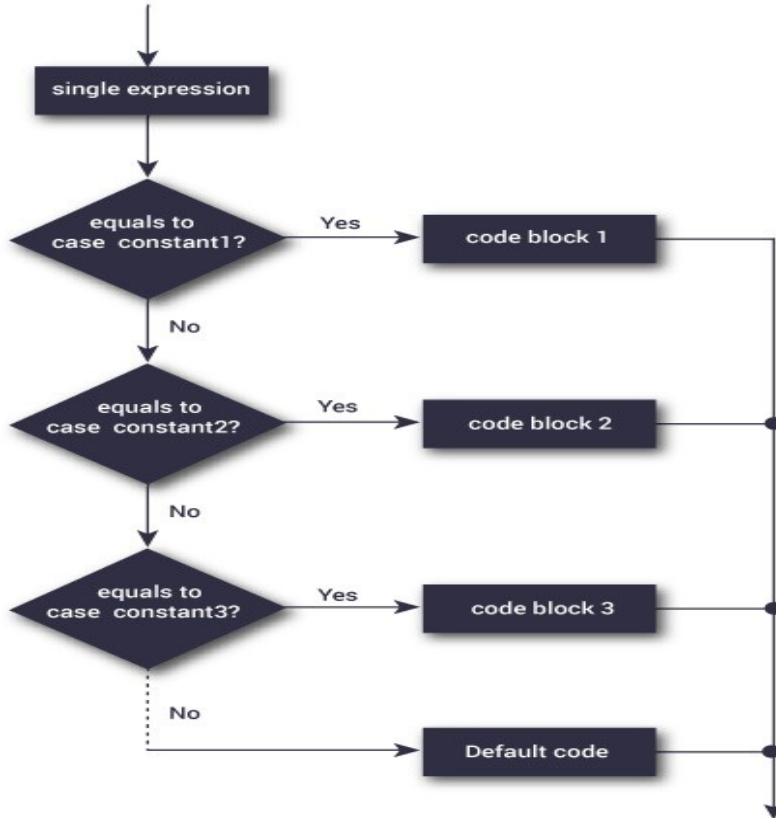
है कभी कभी प्रोग्राम में Case label के साथ default label भी रख देते हैं जब switch statement में दिया गया मान किसी भी case label से मैच नहीं होता है जब default label के अंतर्गत लिखे गए statement execute किये जाते हैं

switch statement का syntax निचे दिया गया है

```
switch(expression)
{
    Case label 1:
        Statement sequence 1:
        break;

        case label 2 :
        statement sequence 2:
break;
case label 3 :
    statement sequence 3:
break;
:
:
Case label n-1:
Statement sequence n-1;
break;
default:
statement sequence n;
}
```

Switch statement को अधिक स्पष्ट रूप से समझने के लिए फ्लोचार्ट निचे दर्शाया गया है



Example Of Switch Case :

```
<?php
$a=2;

switch($a)
{
    case 1:
        echo "A=1";
        break;
    case 2:
        echo "A=2";
        break;
    case 3:
        echo "A=3";
        break;
    default :
        echo "Wrong Number";
}
?>
```

Looping Statement

What Is Loop :

loop को Repetitive Control Statement भी कहते हैं इसका उसे क्र के किसी भी Condition को Number Of Times Ripit किया जाता है loop कहलाते हैं php में Loop 4 प्रकार के होते हैं

1. for loop
2. while loop
3. do-while loop

1. for loop :

यह सबसे कॉमन loop है जिसको ज्यादातर उसे किया जाता है इस loop को लिखना भी आसन होता है क्योंकि इसमें loop के सभी statement एक ही लाइन में लिखे जाते हैं initialization , condition , increment ,Decrement ये सभी एक ही लाइन में लिखे जाते हैं

syntax of for loop :

```
for(initialization;condation;update)
{
    Statement;
}
```

Example of for loop

```
<?php
for($a=0;$a<=10;$a++)
{
    echo $a."<br>";
}
?>
```

2. while loop :

while loop को entry control loop कहते हैं क्योंकि इसमें condition loop के Starting में ही चैक ककर ली जाती है यदि condition true होती है तो loop execute होता है नहीं तो control loop के बाद आ जाता है

while loop का syntax :

```
while(condition)
{
    statement;
    update;
```

```
}
```

Example of while loop

```
<?php  
$a=0;  
while($a<=10)  
{  
    echo $a."<br>";  
    $a++;  
}  
?>
```

3. do-while loop :

इस loop को exit control loop है इस loop में condition loop के अंत में चेक की जाती है इस loop में condition false होने पर भी एक बार तो loop execute होता है और फिर condition चेक होती है यदि condition false होती है तो control loop से बाहर आ जाता है और यदि condition true होती है loop Ripit हो जाता है

do-while loop का syntax निम्नलिखित है

```
do  
{  
    statement;  
    update;  
}while(condition);
```

Example of do-while loop

```
<?php  
$a=0;  
do  
{  
    echo $a."<br>";  
    $a++;  
}while($a<=10);  
?>
```

foreach Loop :

इस loop का उसे ज्यादातर array के साथ किया जाता है foreach loop कहलाता है

foreach loop का syntax :

```
foreach ($variable as $key => $value)
{
    statement;
}
```

Example of foreach Loop :

```
<?php
$a=array(1,2,3,4,5);
foreach ($a as $value)
{
    echo $value."<br>";
}
?>
```

Break Statement :

इस Statement का उसे किसी loop के Execution के बिच में loop से बाहर आने के लिए किया जाता है break statement का उसे करने से loop का Execution रुक जाता है तथा control loop से बाहर आ जाता है

Example Of Break Statement

Continue Statement :

इस statement का उसे loop को किसी condition के बाद continue करने के लिए किया जाता है

Example of Continue Statement

PHP With HTML :

जैसा की आप जानते हैं की php का उसे website बनाने में लिया जाता है और website का static layout html का use कर के बनाया जाता है और website को dynamic करने के लिए php का उसे किया जाता है इस लिए php को html कोड के बिच में लिखना पडता है html में php को लिखने ले लिए html डॉक्यूमेंट में आप php के opening और closing टैग लगा क्र php के कोड लिख सकते हैं

Example :

```
<!DOCTYPE html>
<html>
<head>
    <title>Example</title>
</head>
<body>
<?php
$a=10;
$b=15;
$c=$a+$b;
echo $c;
?>
</body>
</html>
```

Array

What Is Array :

Array एक User Define Data Type है इसका use कर के एक ही variable मेर Mulltiple value को store कर सकते है एक variable में एक प्रकार की value को store किया जाता है array कहलाता है

php में array 3 प्रकार के होते है

Numeric array :

इस तरह के तरह के variable में number string तथा object आदि store कर सकते है किन्तु इनकी index number में represent की जाती है तथा यह single row में होते है

Example of Numeric array :

```
<?php
$a=array(1,2,3,4,5);
foreach($a as $v)
```

```
{  
    echo "Value Is :".$v."<br>";  
}  
?>
```

string

string :

php एक string को character के group के रूप में प्रदर्शित करते हैं एक से अधिक character को मिला कर string बनाई जाती है string को "" Double envarted coma में लिखी जाती है तथा इसके अंत में एक null character होता है जो string के अंत की घोषणा करता है

उदहारण :

'a'

उपरोक्त उदहारण में a एक character है

उदहारण :

“engineersworld”

उपरोक्त उदहारण में engineersworld एक string है जो एक से अधिक character का समूह है

उदहारण :

<?php


```
$name="engineersworld";  
echo $fullname;  
?>
```

1. String Concatenation :

जब एक एक से अधिक string को जोड़ा जाता है तो इस process को String Concatenation कहते हैं इस तरह से एक से अधिक string को जोड़ने के लिए string के बिच में Concatenation operator (.) का use किया जाता है

उदहारण :

```
<?php  
$name="engineers";  
$lname="world";  
$fullname=$name.$lname;  
echo $fullname;  
?>
```

2. strlen() Function :

strlen() function का use किसी भी string की Length Calculate करने के लिए किया जाता है यह function एक पेरामीटर लेती है वह string जिसकी length Calculate करना है उसकी पेरामीटर में pass करते हैं तथा यह Function उस string की Length Return करता है

Example :

```
<?php  
$name="engineersworld";  
echo strlen($name);  
?>
```

3. Strpos() Function :

इस function का उसे किसी String में से किसी character की index value को पता करने के लिए किया जाता है यह Function 2 पेरामीटर लेती है पहला पेरामीटर एक string होती है तथा दूसरा पेरामीटर वह character होता है जिसकी index location पता करनी है

Example :

```
<?php
```

```
$name="engineersworld";  
echo strpos($name, 'w')  
?>
```

\$_POST[] Array :

यह एक global array होता है जिसका textfield से data को read करने के लिए करते हैं तथा form के द्वारा किये जाने वाले किसी भी प्रकार के operation को परफॉर्म करने में किया जाता है इस array का use तब किया जाता है जब html form की method post हो

उदाहरण :

```
<html>  
<head>  
    <title>$_POST</title>  
</head>  
<body>  
<form action="" method="post">  
<input type="text" name="name" placeholder="Enter Name" />  
<input type="submit" name="sub" value="submit">  
</form>  
<?php  
if(isset($_POST['sub']))  
{  
    $name=$_POST['name'];  
    echo $name;  
}  
>  
</body>  
</html>
```

\$_GET[] :

यह एक global array होता है जिसका textfield से data को read करने के लिए करते हैं तथा form के द्वारा किये जाने वाले किसी भी प्रकार के operation को परफॉर्म करने में किया जाता है इस array का use तब किया जाता है जब html form की method get हो

उदहारण :

```
<html>
<head>
    <title>$_POST</title>
</head>
<body>
<form action="" method="get">
<input type="text" name="name" placeholder="Enter Name" />
<input type="submit" name="sub" value="submit">
</form>
<?php
if(isset($_GET['sub']))
{
    $name=$_GET['name'];
    echo $name;
}
?>
</body>
</html>
```

isset Function () :

इस function का use किसी भी प्रकार के event को चैक करने के लिए किया जाता है की किस तरह तरह का event हुआ है

उदहारण :

```
<html>
<head>
    <title>$_POST</title>
</head>
<body>
<form action="" method="get">
<input type="submit" name="sub" value="submit">
</form>
```

```
<?php
if(isset($_GET['sub']))
{
    echo "Button Click";
}
?>
</body>
</html>
```

Function

What is Function :

php में तब कोई program बनाया जाता है तब उस program में कोई statement बार - बार Ripit होता है तो उसे function में लिखा दिया जाता है जिस कारन उस statement को बार बार लिखने की समस्या नहो रहती है एक बार function बना देने के बाद जब भी उस statement की आवश्यकता होती है तो उस function का नाम लिखा कर उस function को call किया जाता है

1. library function
2. user - define function

library function :

library function वह function होते हैं जो php में pre - Define होते हैं जिनको आवश्यकता अनुसार use किया जाता है

उदहारण :

print() , strlen() , include() आदि

user – define function :

user Define Function वह function होता है जो user अपनी आवश्यकता अनुसार बनाता है user define function कहलाता है

syntax :

```
function function_name(arg_list)
{
    body of function ;
}
```

Example :

```
<?php
function f1()
{
    $a=50;
    $b=5;
    $c=$a+$b;
    echo $c;
}
f1(); //Function Call
```

?>

How To Pass Argument in Function

php में function का use कर के उसमे argement pass करने का Example निम्नलिखित है

```
<?php
function fl($a,$b)
{
    echo $a+$b;
}
fl(5,4);
?>
```

Return Value From Function

```
<?php
function fl()
{
    $a=20;
    $b=10;
    return $a+$b;
}
$res=fl();
echo "Sum :". $res;
?>
```

Include() Function :

Php include() एक library function है जिसका use एक webpage में किसी दुसरे webpage को add करने के लिए किया जाता है include function एक पेरामीटर लेता है उस पेज का नाम जिसे include करना है include function का use कर के एक webpage पर एक ही webpage को एक से ज्यादा बार include किया जा सकता है include() function कहलाता है

syntax :

```
include("pagename");
```

Example :

```
<?php
include("p1.php");
echo "Current Page";
include("p1.php");
?>
```

require function() :

इस function का उसे एक webpage पर किसी और webpage को include करने के लिए इस function का उसे किया जाता है

syntax :

```
require("pagename");
```

Example :

```
<?php
require("p1.php");
echo "Current Page";
require("p1.php");
?>
```

header() function :

इस function का use program का control एक webpage से किसी दुसरे webpage पर ट्रांसफर करने के लिए किया जाता है header function का use कर के एक webpage से किसी दुसरे webpage पर Redirect किया जाता है

syntax :

```
header("Location:pagename");
```

Example :

```
<html>
<head>
    <title>Header Function</title>
</head>
<body>
    <h2>Current Page</h2>
<form action="" method="post">
```

```

        <input type="submit" name="sub" value="Next Page" />
</form>
<?php
if(isset($_POST['sub']))
{
    header("Location:p1.php");
}
?>
</body>
</html>

```

Mail Function In PHP :

Php में mail send करने के लिए एक pre-Define function दिया गया है जिसका उसे कर के आप mail send कर सकते हैं किन्तु इस mail function का use offline server पर नहीं किया जाता है इसके लिए आपको एक live server की आवश्यकता होती है जिसका use कर के आप mail send कर सकते हैं mail function का syntax निम्नलिखित है

syntax :

mail(to , Message , Header , Parameters);

to :

इस पैरामीटर में वह email दिया जाता है जिस email पर mail send करना है

from :

इस पैरामीटर में जिस mail से email send करना है उसका उसकी email है

subject :

इस पैरामीटर में email का subject लिखते हैं

Message :

इस पैरामीटर में mail का message लिखते हैं जिसे send करना है

Header :

इस पैरामीटर में mail का Header लिखते हैं

Example :

```

<?php
$to = "abc@example.com";
$subject = "This is subject";
$message = "This is simple message";
$header = "From:engineersworld@example.com \r\n";
$ma = mail ($to,$subject,$message,$header);
if( $ma == true )
{
    echo "Mail Send";
}
else
{
    echo "Mail Not Send";
}
?>

```

File Uploading :

Php में file को upload करने के लिए php में कुछ pre define function दिए गए हैं जिनका उसे कर के php में फाइल को upload किया जाता है file uploading का तात्पर्य यह है की आप इसका use कर के image, Song , Video अदि upload कर सकते है

\$_FILE[] array :

इस array का उसे कर के file को html control से php variable में transfar किया जाता है

move_uploaded_file () :

php में file को upload करने के लिए move_uploaded_file() function का use किया जाता है

Example :

```

<html>
<head>
<titleImage Uplode </title>
</head>

```

```
<body>
<form action="u.php" method="post"enctype="multipart/form-data">
<input type="file" name="image"/>
<input type="submit"name="uplode"value="uplode"/>
</form>
</body>
</html>
<?php
if (isset($_POST['uplode'])) {
$image=$_FILES['image']['name'];
$image_tmp=$_FILES['image']['tmp_name'];
    move_uploaded_file($image_tmp,"$image");
    echo "<img src='$image'width='200px'/>";

}
?>
```

Time And Date

Php में time और Date पता करने के लिए अलग अलग function का use करते हैं

php में time और date पता करने के लिए

प्रमुख function निम्नलिखित है

1. time() :

इस function का उसे कर के computer की Currenter date और time को पिक करते है और यह function कोई भी पेरामीटर नहीं लेती है और यह function एक Integer number return करता है यह function जो integer value return करता है वह time को Second में Convert की गई value है

Example of time() Function

```
<?php
$t1=time();
echo $t1;
?>
```

getdate() Function

getdate() Function php का एक pre-define function है जो current date और time से से सम्बंधित जानकारी asociativ array के रूप में return करता है

निम्नलिखित statement का उसे कर के आप getdate() function से अपनी आवश्यकता के अनुसार time तथा date पता कर सकते है

Key	Description	Example
seconds	Seconds past the minutes (0-59)	20
minutes	Minutes past the hour (0 - 59)	29
hours	Hours of the day (0 - 23)	22
mday	Day of the month (1 - 31)	11
wday	Day of the week (0 - 6)	4
mon	Month of the year (1 - 12)	7
year	Year (4 digits)	1997
yday	Day of year (0 - 365)	19
weekday	Day of the week	Thursday
month	Month of the year	January

Example

```
<?php
$date_array = getdate();
foreach ( $date_array as $key => $val )
```

```

{
print "$key = $val<br />";
}
$formated_date = "Today's date: ";
$formated_date .= $date_array[mday] . "<br>";
$formated_date .= $date_array[mon] . "<br>";
$formated_date .= $date_array[year];
print $formated_date;
?>

```

date() function :

यह function एक string return करता है यह string के Formated date होती है date function का syntax निम्नलिखित है
date(format , Timestamp);

date function में format के रूप में निम्नलिखित पेरामीटर pass करते हैं

Format	Description	Example
a	'am' or 'pm' lowercase	pm
A	'AM' or 'PM' uppercase	PM
d	Day of month,	20
D	Day of week (three letters)	Thu
F	Month name	January
h	Hour (12-hour format - leading zeroes)	12
H	Hour (24-hour format - leading zeroes)	22
g	Hour (12-hour format - no leading zeroes)	12
G	Hour (24-hour format - no leading zeroes)	22
i	Minutes (0 - 59)	23

Example :

```

<?php
print date("m/d/y G.i:s<br>", time());
print "Today is ";
print date("j of F Y, \a\t g.i a", time());
?>

```

Session In php

What is Session:

Session php की एक टेकनिक है जिसका use के के session को store किया जाता है session handling ज्यादातर तब की जाती है जब आप कोई लॉग इन system बनाते है जब कोई user उस system में login होता है तब उस user का session start होता है user के login होने से लेकर logout होने तक का समय एक session कहलाता है इसे हैंडल करने के लिए php में session का use किया है

session_start() :

यह php का एक function है जिसका उसे कर के php में session को start किया जाता है इस method को session के starting में call किया जाता है जहा से session start होता है तब user login होता है तब इस method के द्वारा ही session start होता है जब भी कोई user login होता है तो उसके नाम से server पर एक temporary फाइल create हो जाती है जिसमे user से रिलेटेड जानकारी रहती है और यह फाइल तब तक रहती है है जब तक user login रहता है

\$_SESSION [] :

यह के global variable है जिसका use session की इनफार्मेशन को store करने के लिए किया जाता है session को start करने के बाद user से रिलेटेड इनफार्मेशन को इस variable में store किया जाता है यह एक global variable है इस कारन इससे पुरे session में कही भी use किया जा सकता है

session handling को हम एक login system का example लेकर समझते है

login.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Login System</title>
</head>
<body>
<form action="" method="post">
<table>
    <tr>
        <td width="200px" height="50px" >Username</td>
        <td width="200px" height="50px" ><input type="text" name="un"
placeholder="Enter Name"></td>
    </tr>
    <tr>
        <td width="200px" height="50px" >Password</td>
        <td width="200px" height="50px" ><input type="password"
name="ps" placeholder="Enter Password"></td>
    </tr>
    <tr>
        <td><input type="submit" name="sub" value="Login"></td>
    </tr>
</table>
```

```
</form>
</body>
</html>
<?php
if(isset($_POST['sub']))
{
    $u=$_POST['un'];
    $p=$_POST['ps'];
    session_start();
    $_SESSION['uname']=$u;
    if($u=='admin' && $p=='a123')
    {
        header("Location:p1.php");
    }
    else
    {
        header("Location:login.php");
    }
}
?>
```

p1.php

```
<?php
session_start();
$n1=$_SESSION['uname'];
echo "Welcome ".$n1;
?>
```

session_destroy() function :

इस function का use session को destroy करने के लिए किया जाता है इस function को logout के समय किया जाता है session_destroy() का Example निम्नलिखित है

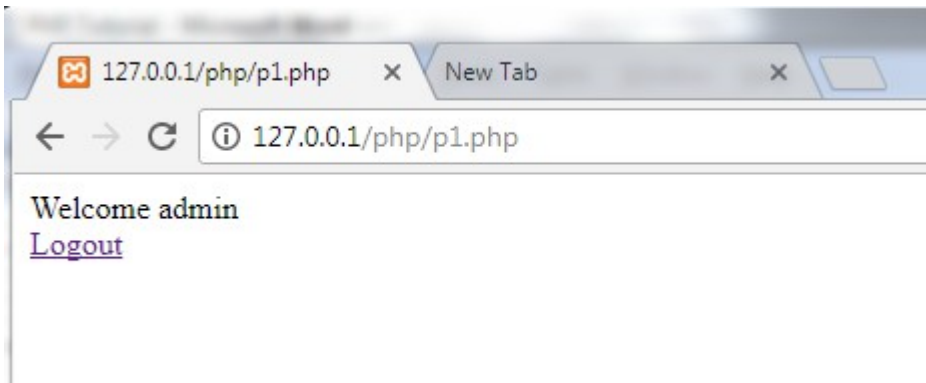
login.php

```
<!DOCTYPE html>
```

```
<html>
<head>
    <title>Login System</title>
</head>
<body>
<form action="" method="post">
<table>
    <tr>
        <td width="200px" height="50px" >Username</td>
        <td width="200px" height="50px" ><input type="text" name="un"
placeholder="Enter Name"></td>
    </tr>
    <tr>
        <td width="200px" height="50px" >Password</td>
        <td width="200px" height="50px" ><input type="password"
name="ps" placeholder="Enter Password"></td>
    </tr>
    <tr>
        <td><input type="submit" name="sub" value="Login"></td>
    </tr>
</table>
</form>
</body>
</html>
<?php
if(isset($_POST['sub']))
{
    $u=$_POST['un'];
    $p=$_POST['ps'];
    session_start();
    $_SESSION['uname']=$u;
    if($u=='admin' && $p=='a123')
    {
        header("Location:p1.php");
    }
    else
    {
        header("Location:login.php");
    }
}
?>
```

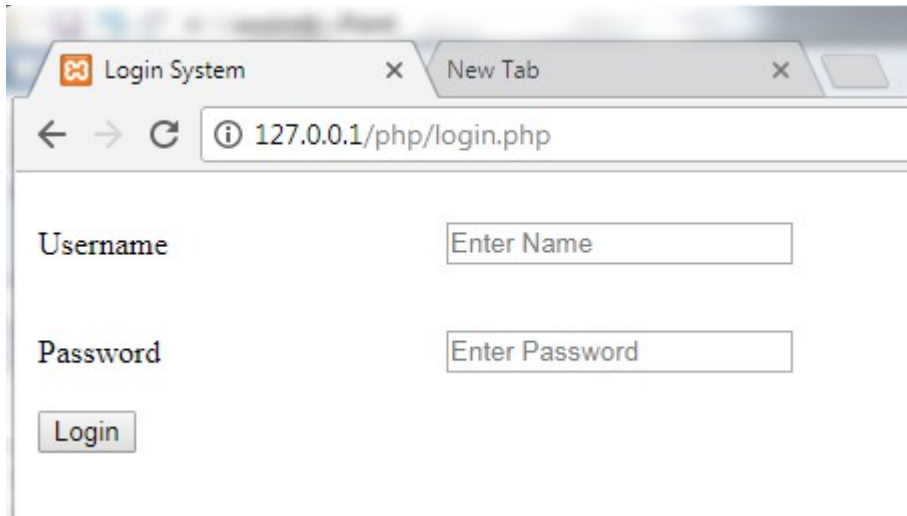

p1.php :

```
<?php
session_start();
$n1=$_SESSION['uname'];
echo "Welcome ".$n1;
?><br>
<a href="logout.php">Logout</a>
```



logout.php

```
<?php
session_start();
$_SESSION['uname'];
if(session_destroy())
{
    header("Location:login.php");
}
else
{
    header("Location:p1.php");
}
?>
```



File Handling

What Is File :

file का तात्पर्य एक ऐसे ऑब्जेक्ट से होता है जो किसी ना किसी प्रकार के data को कन्टेन्ट कर के रखती है

अर्थात दुसरे शब्दों में कहे तो फाइल का उपयोग किसी ना किसी प्रकार के data को स्टोर करने के लिए करते हैं और php में फाइल handling का use कर के फाइल में ही data को write का read किया जाता है

What Is File Handling :

जब भी हम php में कोई प्रोग्राम बनाते हैं तो उस प्रोग्राम में हम किसी ना किसी प्रकार के data का use करते हैं वह data तब तक ही कंप्यूटर की memory में रहता है जब तक प्रोग्राम एक्सीक्यूट होता है जैसे ही प्रोग्राम एंड होता है डाटा कंप्यूटर की memory से नष्ट हो जाता है

यदि हमें अगली बार data की आवश्यकता होती है तो हमें वह data नहीं मिल पाता है इस समस्या को खतम करने के लिए php में फाइल हैंडलिंग का use किया जाता है php में फाइल हैंडलिंग का use से php प्रोग्राम में use किया जाने वाला data फाइल के रूप में स्टोर किया जाता है ताकि बाद में प्रोग्राम के end होने के बाद भी यदि हमें उस data की आवश्यकता होती है तो उसे हम use कर सकते हैं
दुसरे शब्दों में कहे तो php में data को परमानेंट फाइल के रूप में स्टोर करने के लिए फाइल हैंडलिंग का उपयोग किया जाता है

fopen() Function :

fopen() इस Function का use किसी भी फाइल को open करने के लिए किया जाता है fopen() Function का use ककर के किसी भी फाइल को read का write किसी भी उद्देश्य से खोला जा सकता है fopen() Function 2 पैरामीटर लेता है

1. पहला पैरामीटर उस फाइल का नाम दिया जाता है जिसे open करना है
2. दूसरा पैरामीटर File opening Mode होता है file opening mode यह बताता है की फाइल को किस उद्देश्य से open किया गया है

fopen() Function उस फाइल का address return करता है जिसे open किया जाता है उसे एक variable में स्टोर किया जाता है

File Opening Mode :

File Opening Mode निम्नलिखित है

w :

इस opening मोड का use तब किया जाता है जब फाइल में data को write करना हो किन्तु w opening मोड का use करने पर फाइल में पहले से मौजूद data डिलीट हो जाता है तथा new data write होता है और यदि जिस फाइल में data write करना है वह फाइल निर्धारित पाथ पर मौजूद है तो new फाइल क्रिएट हो जाती है

syntax :

```
fp=fopen("a.txt","w");
```

r :

इस मोड का use फाइल से data को read करने के लिए किया जाता है

syntax :

```
fp=fopen("a.txt","r");
```

a :

इस मोड का use फाइल में data को write करने के लिए किया जाता है किन्तु इसका use करने पर फाइल में पहले से लिखा हुआ data डिलीट नहीं होता है बल्कि पुराने data के बाद new add हो जाता है

syntax :

```
fp=fopen("a.txt","a");
```

fclose() Function :

इस function का use फाइल को close करने के लिए किया जाता है इस function के एक्सीक्यूट हो जाने के बाद फाइल प्रोग्राम से कनेक्ट नहीं रहती है यह function 1 पैरामीटर लेता है वह फाइल का address होता है जो उस फाइल को पॉइंट करता है जिसे close करना है

syntax :

```
fclose(file_reffrence);
```

Example :

```
fclose(fp);
```

fwrite() Function :

इस Function का use file में data को write करने के लिए किया जाता है यहाँ Function 2 पैरामीटर लेता है पहला पैरामीटर उस file का reference होता है जिसमें data को write करना है तथा दूसरा पैरामीटर वह data होता है जिसे write करना है

syntax :

```
fwrite($file,$data);
```

Example :

```
<?php
$file = fopen("a1.txt","w");
$msg="Engineersworld.in";
fwrite($file,$msg);
fclose($file);
?>
```

fgetc() :

किसी भी फाइल से data की read करने के लिए fgetc() Function का use किया जाता है fgetc() Function एक बार में फाइल से केवल एक कैरेक्टर को ही read करता है

```
fgetc(file_reference);
```

Example :

```
<?php
$f1=fopen('a1.txt','r');
while(!feof($f1))
{
    $string=fgetc($f1);
    echo $string;
}
?>
```

fgets() Function :

इस फंक्शन का use फाइल से data को read करने के लिए किया जाता है यह फंक्शन एक बार में एक String को read करता है यह फंक्शन एक पैरामीटर लेता है वह पैरामीटर उस फाइल का reference होता है जिससे data को read करना है और यह फंक्शन string return करता है

syntax :

```
$string = fgets(File_Reffrence);
```

Example :

```
<?php  
$f1=fopen('a1.txt','r');  
$string=fgets($f1);  
echo $string;  
?>
```

filesize(“file_name”) :

इस फंक्शन का use किसी भी फाइल की size पता करने के लिए किया जाता है यह Function पैरामीटर में उस फाइल का नाम लेता है जिसकी size पता करना करना है और यह फंक्शन फाइल की size return करता है

syntax :

```
variable = filesize(“file name”);
```

Example :

```
<?php  
echo "<br>".filesize('a1.txt');  
?>
```

file_exists(“file name”) :

इस Function का use file उपस्थित है या नहीं यह पता करने के लिए किया जाता है यह function एक पैरामीटर लेता है उस file का नाम जिसके बारे में जानकारी पता करना है और यह फंक्शन एक boolean वैल्यू return करता है यदि फाइल उपस्थित होती है तो यह 1 return करती है और यदि फाइल उपस्थित नहीं होती है तो यह 0 return करती है

syntax :

```
variable = file_exists ("file name");
```

Example :

```
<?php  
echo "<br>".file_exists('a1.txt');  
?>
```