

## 1. Control / Conditional statement in php

Ans : प्रोग्रामिंग भाषा में, प्रोग्राम के execution के flow को नियंत्रित करने के लिए जो स्टेटमेंट्स या structures प्रयोग किये जाते हैं उन्हें control statements(structures) कहते हैं.

Program में condition के हिसाब से statements को control करने के लिए control statements का इस्तेमाल किया जाता है | आप different decisions के लिए अलग-अलग कार्य करना चाहते हैं और तब आपको ऐसा करने के लिए अपने कोड में conditional स्टेटमेंट का उपयोग करना पड़ता है

**if statement:-** if statement एक control statement है जिसका प्रयोग एक विशेष कंडीशन को test करने के लिए किया जाता है. इसमें कंडीशन केवल एक बार execute होती है जब कंडीशन सत्य होती है.

**if-else statements:-** किसी विशेष कंडीशन को test करने के लिए if-else statement का प्रयोग किया जाता है. यदि कंडीशन सत्य है तो if statement को execute किया जाता है अगर कंडीशन असत्य है तो else statement को execute किया जाता है.

**switch statements:-** switch statement एक multi-way branch स्टेटमेंट है. यह statement भी एक selection स्टेटमेंट जो कि एक प्रोग्राम के execution के लिए विभिन्न paths को डिफाइन करता है. Switch case statement में expression होता है और उससे related कुछ cases होते हैं | जो case उस expression या declare किये हुए variable से match होती है तब वो output में print होता है | अगर कोई भी case expression से match नहीं होती तो वो default का statement output में print करेगा | आपको हर statement के बाद break लगाना पड़ता है, इसका मतलब वो उसके पहले का statement ही print करेगा | अगर आप break नहीं लगाते तो वो पहला और दूसरा ये दोनों statement को print करेगा | default case के बाद break नहीं लगाते | यह if-else स्टेटमेंट के लिए विकल्प के तौर पर कार्य करता है.

Simple IF	If else	Switch Case
<pre>&lt;?php \$a = 5; \$b = 6;  if( \$a&lt;\$b ) {     echo "a is less than b"; } ?&gt;</pre>	<pre>&lt;?php \$a = 5; \$b = 5;  if( \$a&lt;\$b ) {     echo "a is less than b"; } else{     echo "a is not less than b"; } ?&gt;</pre>	<pre>&lt;?php \$favcolor = "red";  switch (\$favcolor) {     case "red":         echo "Your favorite color is red!";         break;     case "blue":         echo "Your favorite color is blue!";         Break;     default:         echo "Your favorite color is neither red, blue, nor green!"; } ?&gt;</pre>

## 2. Array in its Types in PHP

Ans :

Array के जरिये आप same type की value को एक ही variable में store करवा सकते हो। यह special तरह का variable होता है जसमें एक time में बहुत सी values को store करवाया जा सकता है। उदाहरण के लिए मान लीजिये यदि आपने 100 variables में values को define करना है तो, आपको एक- एक करके values को variable में insert करना होगा। वहीं array के जरिये 100 values को insert करना बहुत आसान हो जाता है।

Array एक विशेष प्रकार का डाटा स्ट्रक्चर(data structure) हैं जो की सामान साइज (fixed size) वाले डाटा कलेक्शन(collection) को स्टोर करके रखता हैं। Array में हम कितना भी डाटा स्टोर करके आसानी से use कर सकते हैं।

PHP में array को array() function द्वारा use किया जाता है।

PHP में Arrays तीन तरह के होते हैं और array को access करने के लिए हमें foreach या array index का use करते हैं।

- **Numeric Array**
- **Associative Array**
- **Multidimensional Array**

**Numeric Array :** इस array में आप numbers, string और objects को store करवा सकते हो। लेकिन index हमेशा numbers से शुरू होता है और default array 0 से शुरू होता है। हर array के element को comma से separate किया जाता है।

<pre>&lt;?php \$a=array(10,20,30); print_r(\$a);  ?&gt;</pre>	<p><b>Output :</b></p> <pre>Array (     [0] =&gt; 10     [1] =&gt; 20     [2] =&gt; 30 )</pre>
---	--

**Associative Array :** Associative Array भी numeric array की तरह होता है। फर्क बस इतना है की Associative array में index को अलग से define किया जा सकता है। आप array के index को string में लिख सकते हो। index को define करने के लिए => इस sign के साथ value का इस्तेमाल किया जाता है। हर array के element को comma से separate किया जाता है

<pre>&lt;?php \$a=array('roll'=&gt;10,'name'=&gt;'raj',30); print_r(\$a);  ?&gt;</pre>	<p><b>Output :</b></p> <pre>Array (     [roll] =&gt; 10     [name] =&gt; raj     [0] =&gt; 30 )</pre>
--	---

**Multidimensional Array :** PHP में Multidimensional Array को Nested Array भी कहा जाता है | Multi Dimensional Array में दो या दो से अधिक arrays होते हैं |

<pre>&lt;?php \$a=array(array('name'=&gt;'raj','class'=&gt;11),           array('name'=&gt;'mohan','class'=&gt;12)); print_r(\$a);  ?&gt;</pre>	<b>Output :</b> Array ( [0] => Array ( [name] => raj [class] => 11 )  [1] => Array ( [name] => mohan [class] => 12 ) )
---	--

### 3. Create Simple student registration form.

Enter name:	<input type="text"/>
Enter password:	<input type="password"/>
Enter Email:	<input type="text"/>
Enter Gender:	<input type="radio"/> male <input type="radio"/> female
Select Country:	<input type="text" value="india"/>
<input type="button" value="register"/>	

Ans :

जब HTML Form को create किया जाता है तब उसे validate भी किया जाता है | Validate करने का मतलब यही है कि जो भी User इस form को fill कर रहा है वो सही information डाले |

अगर User से valid information send करनी हो तो कुछ चीजों पर ध्यान रखना पड़ता है |

**Name :** Name के सिर्फ letters और whitespace valid होंगे |

**UserName :** UserName में सिर्फ letters और numbers ही valid होंगे |

**Email :** Email में letters या nummmbers उसके बाद @ symbol उसके बाद letters उसके बाद . (dot) और आखिरी में letters

**URL :** URL valid होगा |

**Mobile Number :** सिर्फ Number valid और length 10 digit तक होगी |

**Gender :** Radio तो एक select होगा |

**Use : label, textbox, radio button, drop down list, submit button, password**

#### 4.Process of sending email in PHP

Ans : PHP में mail() function ये बहुत ही महत्वपूर्ण हिस्सा है | PHP के mail() function plain text, HTML या कुछ attachments messages भेजे जाते हैं |

#### Syntax for mail() function

**mail(to, subject, message, headers, some\_parameters)**

##### Parameters :

**to :** यहाँ पर email receiver का email address आता है |

**subject :** यहाँ पर email का subject आता है | Subject के लिए कोई newline character नहीं होता |

**message :** यहाँ पर mail का message आता है, जो receiver को भेजा जाएगा | लेकिन message में हर एक line 70 characters की होनी चाहिए | अगर line 70 characters से बढ़कर होती है तो ,

wordwrap() function का इस्तेमाल किया जाता है | हर 70 characters के बाद CR-Carriage Return LF-Line Feed(\r\n) का उपयोग line seperate करने के लिए किया जाता है |

**headers :** यहाँ पर sender के लिए कुछ headers दिए जाते हैं जैसे कि, From, Cc(Carbon copy), Bcc(Blind carbon copy)| हर एक header को CRLF(\r\n) से seperate किया जाता है |

**some\_parameters :** यहाँ पर mail() function के लिए कुछ additional paramters आते हैं |

```

<?php
if (isset($_POST['email']))
{

$from = "abcd@gmail.com";
$to = $_POST['email'];
$subject = $_POST['subject'];
$comment = $_POST['comment'];
$headers = 'From: <'.$from.'>'. "\r\n";

mail($to, $subject, $comment, $headers);
    echo "Email sent successfully.";
}
else{
?>

<form method="post">
Email: <input name="email" type="text" /><br />
Subject: <input name="subject" type="text" /><br />

Message:<br />
<textarea name="comment" rows="20"
cols="50"></textarea><br />
<input type="submit" value="Submit" />
</form>
<?php
}
?>

```

Email:

Subject:

Message:

## 5. Write a code of operation in PHP MYSQL

### 1. Insert data 2. Update data 3. Delete data

Ans :

<pre>// database connection code &lt;?php \$server = "localhost"; \$user = "root"; \$password = ""; \$db = "info";  \$conn = mysqli_connect(\$server, \$user, \$password, \$db);  if(\$conn){     echo "Connected successfully."; } else{     echo mysqli_connect_error(); }</pre>		
<pre>//insert data  \$tb_insert = "INSERT INTO student(student_name, percentages, grade) VALUES ('Rakesh', '86.45', 'A')";  if (mysqli_query(\$conn, \$tb_insert)) {     echo "Record inserted successfully."; } else {     echo "Error inserting record : " . mysqli_error(\$conn); }  mysqli_close(\$conn); ?&gt;</pre>	<pre>//update data  \$tb_update = "UPDATE student SET student_name = 'Mukesh', percentages = 50.30 WHERE id = 1";  if (mysqli_query(\$conn, \$tb_update)) {     echo "Table Updated successfully."; } else {     echo "Table Updating failed : " . mysqli_error(\$conn); }  mysqli_close(\$conn); ?&gt;</pre>	<pre>//delete data  \$tb_update = "DELETE FROM student WHERE id=2";  if (mysqli_query(\$conn, \$tb_update)) {     echo "Some row deleted successfully."; } else {     echo "Row deleting failed : " . mysqli_error(\$conn); }  mysqli_close(\$conn); ?&gt;</pre>