

CHAPTER 7**PHP LOOPING STATEMENTS****PART I**

1. foreach
2. Bounded loops
3. All of them
4. 5
5. Counter is: 20
6. None of the Above
7. All the above
8. $12^2 = 144$
9. (b) *****
10. Infinite loop

PART II**1. Define Looping Structure in PHP.**

- Loop structures in PHP is an iterative control structures that involves executing the same block of code a specified number of times.
- Loops that iterate for fixed no of times is also called as Bounded loops.

2. Define for loop in PHP.

- The for loop is used when you know how many times you want to execute a statement or block of statements.

Syntax:

```
for (init counter; test counter; increment counter)
```

```
{
    code to be executed;
}
```

3. What is foreach loop in PHP?

- foreach loop is exclusively available in PHP and is mainly used for looping through the values of an array.
- The loop iteration depends on each KEY Value pair in the Array.
- For each, loop iteration the value of the current array element is assigned to \$value variable and the array pointer is advanced by one, until it reaches the end of the array element.

4. List out Looping Structure in PHP.

- PHP supports four types of loops:
for Loop
foreach Loop
while Loop
do While Loop

5. Write Syntax of For loop in PHP.

Syntax:

```
for (init counter; test counter; increment counter)
{
    code to be executed;
}
```

6. Write Syntax of foreach loop in PHP.

Syntax:

```
foreach ($array as $value)
{
    code to be executed;
}
```

7. Write Syntax of while loop in PHP.

Syntax:

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

8. Write Syntax of do while loop in PHP.

Syntax:

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

9. Compare for loop and for each loop.

for	Foreach
The for loop is used when you know how many times you want to execute a statement or block of statements.	foreach loop is exclusively available in PHP and is mainly used for looping through the values of an array. The loop iteration deepens on each KEY Value pair in the Array.
Syntax: for (init counter; test counter; increment counter) { code to be executed; }	Syntax: foreach (\$array as \$value) { code to be executed; }

10. Explain the use of for each loop in PHP.

- The foreach loop works only on arrays, and is used to loop through each key/ value pair in an array.
- The foreach construct provides an easy way to iterate over arrays

PART III

1. Write the features of Looping Structure.

- Loop structures in PHP is an iterative control structures that involves executing the same block of code a specified number of times.
- Loops that iterate for fixed no of times is also called as Bounded loops.
- This helps the user to save both time and effort of writing the same code multiple times.
- PHP supports four types of loops:
 - for Loop
 - foreach Loop
 - while Loop
 - do While Loop

2. **Write the purpose of Looping Structure in PHP.**

- Loop structures in PHP is an iterative control structures that involves executing the same block of code a specified number of times. Loops that iterate for fixed no of times is also called as Bounded loops.
- This helps the user to save both time and effort of writing the same code multiple times.
- Looping structures are useful for writing iteration logics.
- PHP supports four types of loops:
for Loop
foreach Loop
while Loop
do While Loop

3. **Differentiate Foreach and While loop.**

foreach	while
foreach loop is exclusively available in PHP and is mainly used for looping through the values of an array. The loop iteration depends on each KEY Value pair in the Array.	The while loop executes a block of code as long as the condition specified in the while statement evaluates to true.
Syntax: foreach (\$array as \$value) { code to be executed; }	Syntax: while (condition is true) { code to be executed; }

4. **Write short notes on do while Loop.**

- The do.. while loop is always very similar to the while loop but executes the block of code at least once before evaluating the condition.
- Here the condition is evaluated only at the time of exit of each iteration.
- do...while loop does unconditional checking (atleast once) and is also called as exit check loops.
- **Syntax:**
do

```

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

```

5. Differentiate While and Do while loops.

While	do..while
The while loop executes a block of code as long as the condition specified in the while statement evaluates to true.	The do.. while loop is always very similar to the while loop but executes the block of code at least once before evaluating the condition. Here the condition is evaluated only at the time of exit of each iteration.
while loop does conditional checking and is also called as entry check loops	do...while loop does unconditional checking (atleast once) and is also called as exit check loops
Syntax: while (condition is true) { code to be executed; }	Syntax: do { code to be executed; } while (condition is true);

PART IV

1. Explain Looping Structure in PHP.

- Loop structures in PHP is an iterative control structures that involves executing the same block of code a specified number of times.
- Loops that iterate for fixed no of times is also called as Bounded loops.
- This helps the user to save both time and effort of writing the same code multiple times.
- PHP supports four types of loops:
for Loop
foreach Loop
while Loop
do While Loop
- **for loop:** The for loop is used when you know how many times you want to execute a statement or block of statements.

Syntax:

```
for (init counter; test counter; increment counter)
```

```
{
    code to be executed;
}
```

- **foreach loop:** foreach loop is exclusively available in PHP and is mainly used for looping through the values of an array. The loop iteration depends on each KEY Value pair in the Array.

- **Syntax:**

```
foreach ($array as $value)
```

```
{
    code to be executed;
}
```

- **while loop:** The while loop executes a block of code as long as the condition specified in the while statement evaluates to true.

- **Syntax:**

```
while (condition is true)
```

```
{
    code to be executed;
}
```

- **do..while loop:** The do.. while loop is always very similar to the while loop but executes the block of code at least once before evaluating the condition. Here the condition is evaluated only at the time of exit of each iteration.

- **Syntax:**

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

2. **Discuss in detail about foreach loop.**

5. **Explain working of loops in array.**

- When we write a code, we want the same block of code to run over and over again.
- Instead of adding several almost equal code-lines in a script, we can use loops to perform a task.

- foreach loop is exclusively available in PHP and is mainly used for looping through the values of an array.
- The loop iteration deepens on each KEY Value pair in the Array.
- For each, loop iteration the value of the current array element is assigned to \$value variable and the array pointer is advanced by one, until it reaches the end of the array element.
- **Syntax:**

```
foreach ($array as $value)
{
    code to be executed;
}
```
- The foreach construct provides an easy way to iterate over arrays. foreach works only on arrays and objects, and will issue an error when you try to use it on a variable with a different data type or an uninitialized variable.
- **Example:**

```
<?php
$Student_name = array("Magilan", "Iniyan", "Nilani", "Sibi", "Shini");
foreach ($Student_name as $value)
{
    echo "$value <br>";
}
?>
```

3. Explain the process of do while loop.

- The do.. while loop is always very similar to the while loop but executes the block of code at least once before evaluating the condition.
- Here the condition is evaluated only at the time of exit of each iteration.
- do...while loop does unconditional checking (atleast once) and is also called as exit check loops.
- **Syntax:**

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

- **Example:**

```
<?php
$Student_count = 10;
$student_number=1;
do
{
echo "The student number is: $student_number<br>"; $student_number++;
} while($student_number<= $Student_count)
?>
```

4. Explain concepts of for loop with example.

- The for loop is used when you know how many times you want to execute a statement or block of statements.

- **Syntax:**

```
for (init counter; test counter; increment counter)
{
    code to be executed;
}
```

- **Parameters:**

init counter: Initialize the loop initial counter value

Test counter: Evaluated for every iteration of the loop. If it evaluates to TRUE, the loop continues. If it evaluates to FALSE, the loop ends.

Increment counter: Increases the loop counter value.

- **Example:**

```
<?php
for ($i = 0; $i<= 5; $i++)
{
    echo "The number is: $i<br>";
}
?>
```