# PHP Loops

INTRODUCTION PHP

# Loops

In PHP, we have the following looping statements:

**while** - loops through a block of code as long as the specified condition is true

**do...while** - loops through a block of code once, and then repeats the loop as long as the specified condition is true

**for** - loops through a block of code a specified number of times

**foreach** - loops through a block of code for each element in an array

# Loops

## WHILE

```php
<?php
$x = 1;

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

## DO WHILE

```php
<?php
$x = 1;

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

Escola Pia
Santa Anna-Mataró

# Loops

## FOR

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

## FOREACH

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

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

Escola Pia
Santa Anna-Mataró

# Loops: BREAK

break ends execution of the current for, foreach, while, do-while or switch structure.

break accepts an optional numeric argument which tells it how many nested enclosing structures are to be broken out of. The default value is 1, only the immediate enclosing structure is broken out of.

```php
$arr = array('one', 'two', 'three', 'four', 'stop', 'five');
while (list(, $val) = each($arr)) {
    if ($val == 'stop') {
        break;    /* You could also write 'break 1;' here. */
    }
    echo "$val<br />\n";
}
```

Escola Pia
Santa Anna-Mataró