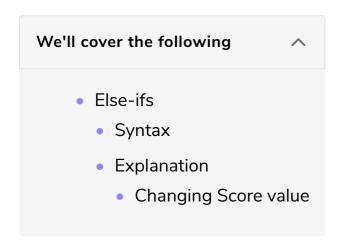# if-elseif-else Statement

This lesson discusses if-elseif-else statements in detail using examples.

Else-ifs

Else-ifs are the complex **nested** form of `if-else` block. They are used when you have to check multiple conditions and there's a different action that needs to be performed corresponding to every condition. Here the `else` statement gets replaced by **one** or **many** `else if` statements and there could be an optional ending `else` statement.

Syntax

Let's take a look at an example below to better understand the concept of **nested ifs**.

```php
<?php
$score = 50; //change the value of score to see other results
if ($score > 100)
{ // If score is greater than 100
    echo "Error: the score is greater than 100!\n";
}
else if ($score < 0)
{ // Else If score is less than 0
    echo "Error: the score is less than 0!\n";
}
else if ($score >= 50)
{ // Else if score is greater or equal to 50
    echo "Pass!\n";
}
else
{ // If none above, then score must be between 0 and 49
    echo "Fail!\n";
}
?>
```

Explanation

- All the statements in the code above will run in order from top to bottom until a *condition* has been met.

- In line **4** we check if `score` is greater than **100**.

- In line **7** we check if `score` is less than **0**.

- In line **10** we check if `score` is greater than or equal to **50**.

- In line **13** the `else` condition will execute if all the conditions above evaluate to *false*.

Changing Score value

- If you change the `score` to **110** the output displayed would be **Error: the score is greater than 100!**

- If you change the `score` to **-20** the output displayed would be **Error: the score is less than 0!**

- If you change the `score` to **48** the output displayed would be: **Fail!**

As useful as these `else if` conditions are, there's an easier way to write conditionals in PHP, i.e., using the **switch** statements which we will discuss in detail in the next lesson!