do...while Loop

This lesson discusses the do-while loops in PHP. It uses coding examples to show their implementation and explains how they work.
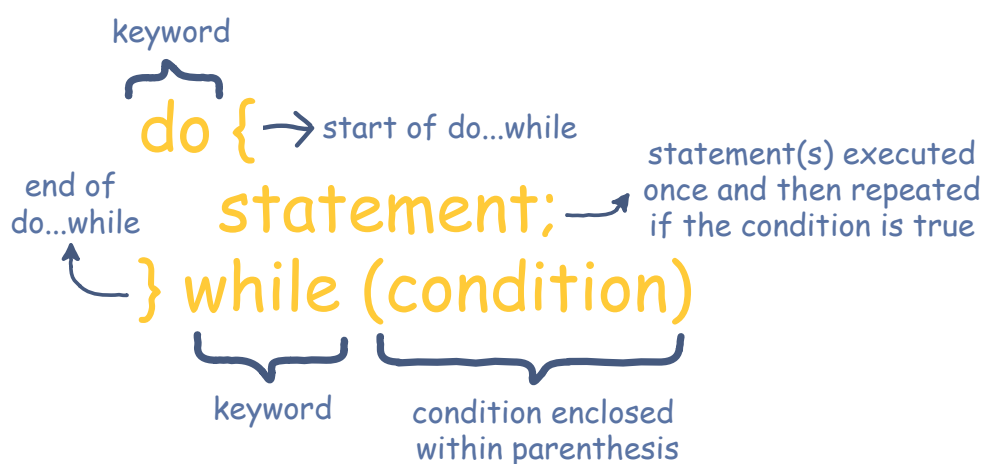
What is the do...while Loop?

The **do...while** loop is nearly identical to the `while` loop, but instead of checking the *conditional* statement before the loop starts, the **do...while** loop checks the *conditional* statement **after** the *first* run, before moving onto another iteration.

The *syntax* is as follows:

while vs. do...while

The `while` loop runs the code inside it if and only if the condition inside the parenthesis is true. The `do...while`, however, runs the loop **at least** once before

checking the conditional.

> **Note:** The **do...while** loop is still haunted by *infinite* loops, so exercise the same caution with the **do...while** loop as you would with the `while` loop. Its usefulness is much more limited than the `while` loop, so use this only when necessary.
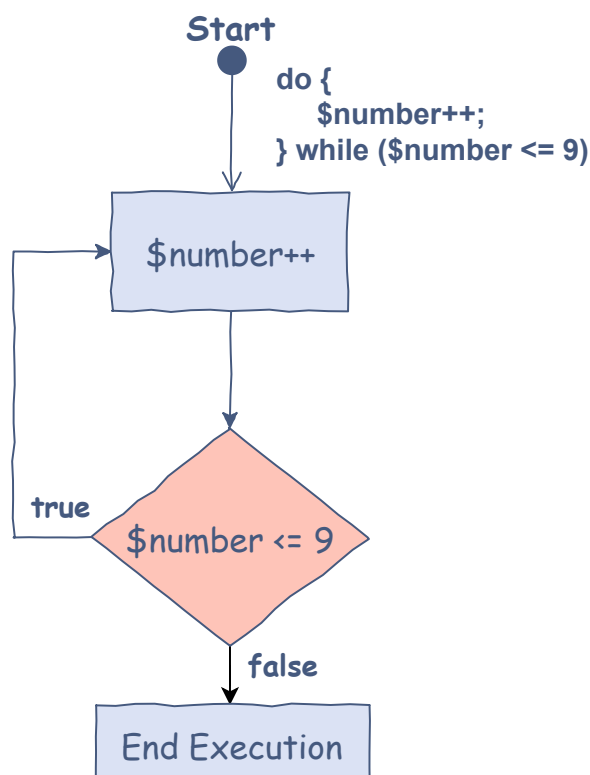
Example

Below is an example showing how to implement the **do...while** loop in PHP.

```php
<?php
$number = 5;
do
{
    echo "Value of number is: $number\n";
    $number++;
}
while ($number <= 9); // the contition is being checked after the first run
?>
```

Explanation

The *illustration* below will help you better understand the logic of the above code.

When is do-while used?

A *do-while* loop is used where your loop should execute **at least one** time.

For example, let's consider a scenario where we want to take an *integer* input from the user until the user has entered a **positive** number. In this case, we will use a **do-while** as we have to run loop **at-least once** because we want input from user at-least once. This loop will continue running until the user enters a **positive** number.

That's all the major stuff you needed to know about the workings of the **do...while** loops in PHP. Let's learn about `for` loops in the next lesson.