# do-while Loop

In this lesson, you will be introduced to the do-while loop.

**We'll cover the following**

## Introduction

Suppose we want to execute the body of a loop at least once even if the condition evaluates to false. How can we accomplish this task in C++?

**1** of 2

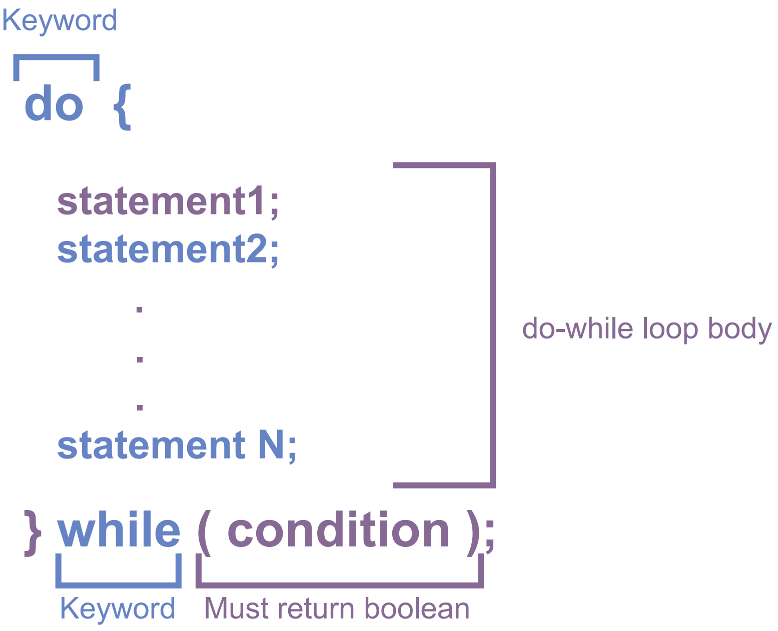
In the era of programming, we can use the `do-while` loop to implement such tasks.

> *The **do-while** loop is similar to the while loop, with the exception that, first, it executes the block of code and then checks the given condition.*

> 📝 The `do-while` loop is called an exit-controlled loop because it executes the body of the loop first, and then tests the given condition.

## Syntax

Let’s go over the syntax of the `do-while` loop.

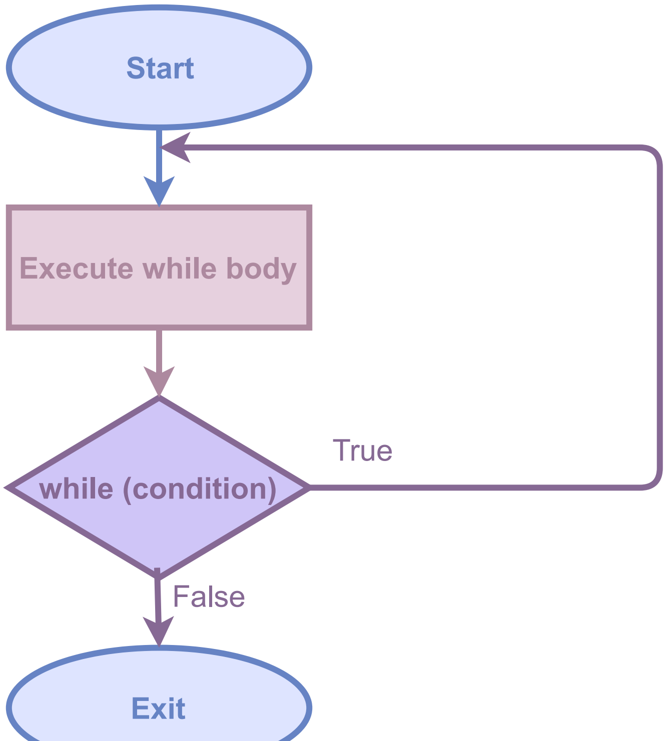


The general syntax of the do-while loop consists of a `do` keyword followed by curly brackets `{ }`, which contain statements to be executed. It is further followed by the `while` keyword and the condition to be checked.

> 📝 Like the `while` loop, the `do-while loop` does not know in advance how many times the loop body should be executed.

## Flowchart

Let’s look at the flowchart of the `do-while` loop.



- The `do-while` loop first executes the code in the body of the loop.

- After executing the loop block, it evaluates the given condition.

- If the condition evaluates to true, the code inside the body of the `do-while loop` is executed again. This process continues as long as the given condition remains true.

## Example program

Let’s translate the example given above into a C++ program.

Press the **RUN** button and see the output!

```cpp
#include <iostream>

using namespace std;

int main() {
  // Initialize the variable money
  int money = 0;
  // Initialize the variable icecream_price
  int icecream_price = 5;
  // Prints value of variables
  cout << "Intial money = " << money << endl;
  cout << "Ice-cream price = " << icecream_price << endl;
  // Start of the  do-while loop
  do {
    // Body of the do-while loop
    cout << "Buy an ice-cream" << endl;
    money = money - icecream_price;
    cout << "Remaining money = " << money << endl;
  } while (money >= icecream_price);
  // End of the  do-while loop
  cout << "You can't buy an ice-cream" << endl;

  return 0;
}
```

## Explanation

**Line No. 7:** Initializes the value of `money`.

**Line No. 9:** Initializes the value of `icecream_price`.

**Line No. 11:** Prints the value of `money` to the console.

**Line No. 12:** Prints the value of `icecream_price` to the console.

**Line No. 14:** Executes **Lines No. 16 to 19**.

**Line No. 16:** Prints `Buy an ice-cream` to the console

**Line No. 17** Subtracts an `icecream_price` from the `money`

**Line No. 18:** Prints the new value of `money` to the console

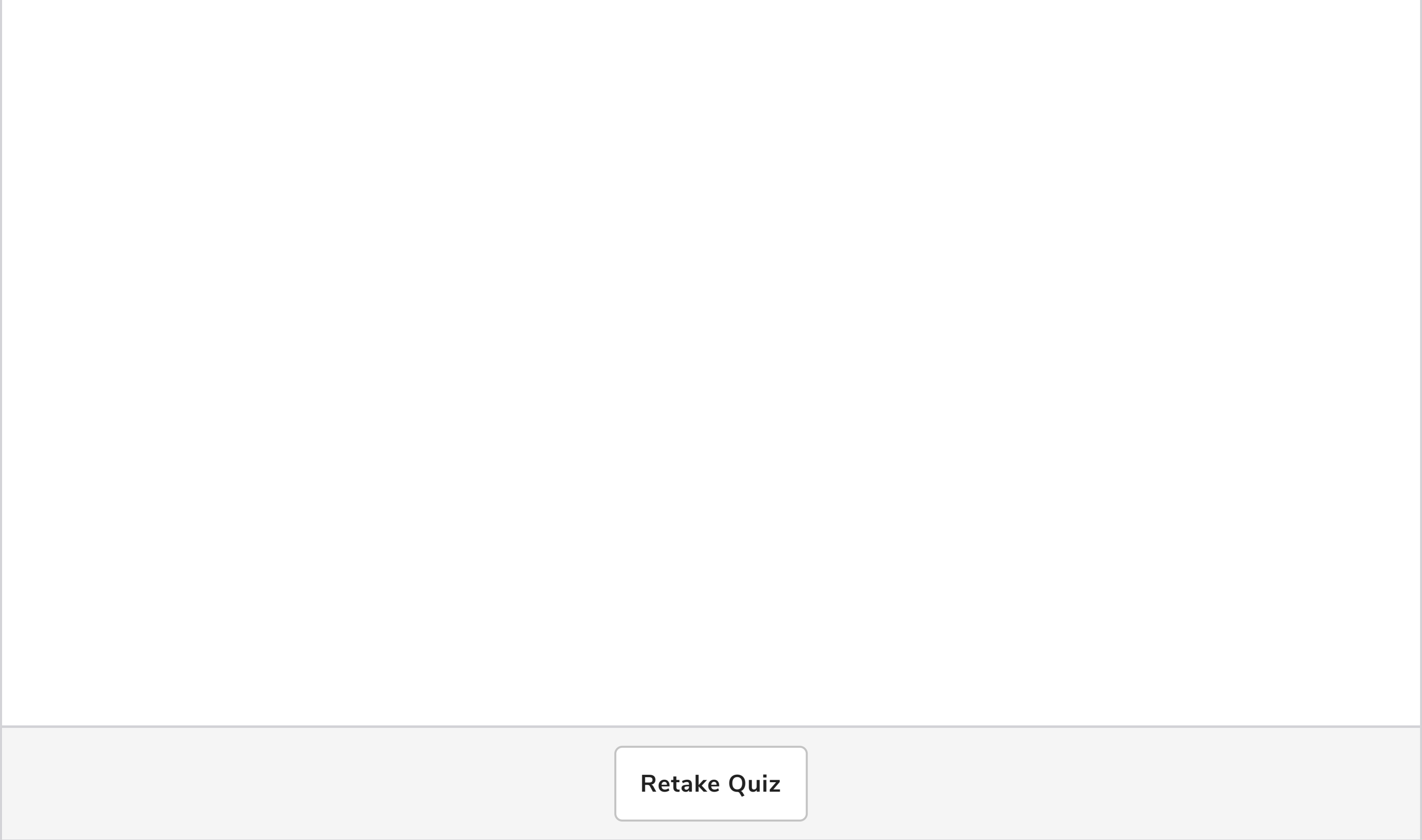
**Line No. 19:** Checks if the value of `money` is greater than `icecream_price`. If yes, then it jumps to **Line No. 14**. If no, then it executes **Line No. 21**.

**Line No. 21:** Prints `You can't buy an ice-cream` to the console

Q

Which of the following statements is not true about the `do-while` loop?

You can select multiple correct answers.



Let’s discuss the `for` loop in the upcoming lesson.

See you there!