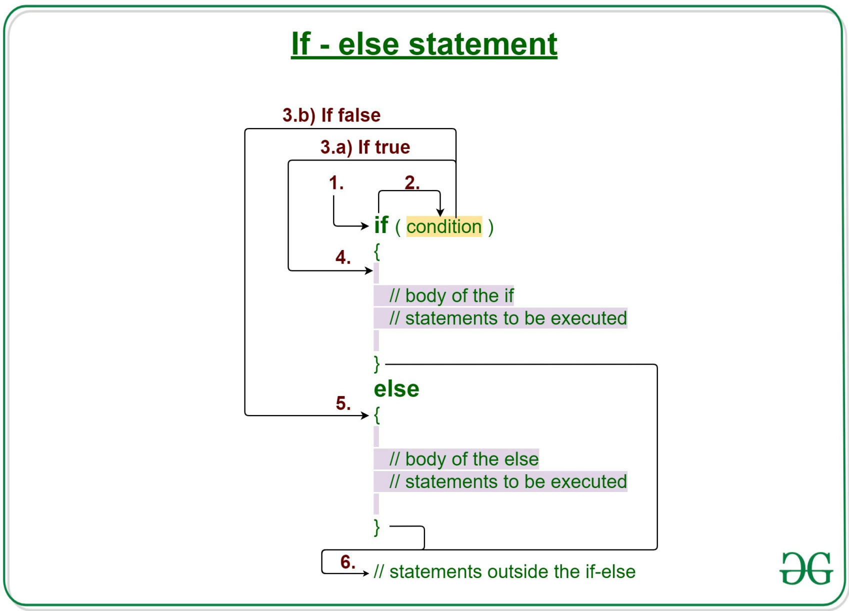
If Else Syntax in C++

Decision Making helps to write decision driven statements and execute a particular set of code based on certain conditions.

The *if* statement alone tells us that if a condition is true it will execute a block of statements and if the condition is false it won't. But what if we want to do something else if the condition is false. Here comes the C/C++ **else statement**. We can use the else statement with if statement to execute a block of code when the condition is false.



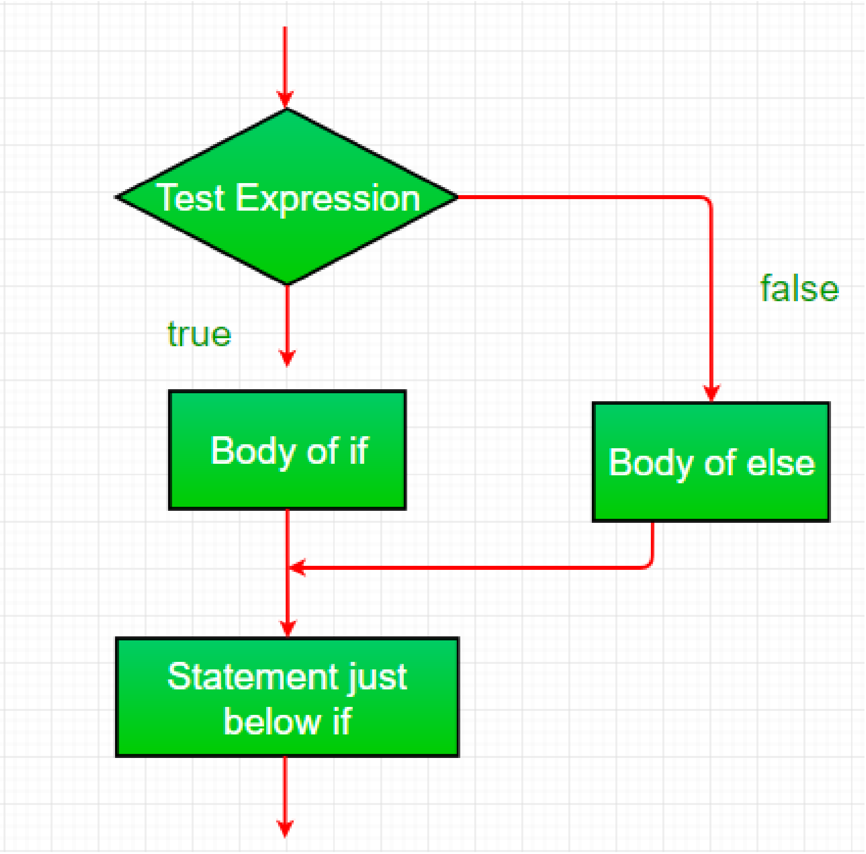
Syntax:

```
if (condition)
{
        // Executes this block if
        // condition is true
}
else
{
        // Executes this block if
        // condition is false
}
```

Working of if-else statements

1. Control falls into the if block.
2. The flow jumps to Condition.
3. Condition is tested.
    a. If Condition yields true, goto Step 4.
    b. If Condition yields false, goto Step 5.
4. The if-block or the body inside the if is executed.
5. The else block or the body inside the else is executed.
6. Flow exits the if-else block.

Flowchart if-else:



Example:

```
// C++ program to illustrate if-else statement
#include <iostream>
using namespace std;
```

```
int main()
{
    int i = 20;

    // Check if i is 10 if (i == 10)
        cout << "i is 10";

    // Since is not 10
    // Then execute the else statement
    else        cout << "i is 20\n";

    cout << "Outside if-else block";

    return 0;
}Output

i is 20
Outside if-else block
```