

# C - The if-else Statement

The **if-else** statement is one of the frequently used **decision-making statements** in C. The **if-else** statement offers an alternative path when the condition isn't met.

The **else** keyword helps you to provide an alternative course of action to be taken when the Boolean expression in the **if** statement turns out to be false. The use of **else** keyword is optional; it's up to you whether you want to use it or not.

## Syntax of if-else Statement

Here is the syntax of **if-else** clause –

```
if (Boolean expr){  
    Expression;  
    . . .  
}  
else{  
    Expression;  
    . . .  
}
```

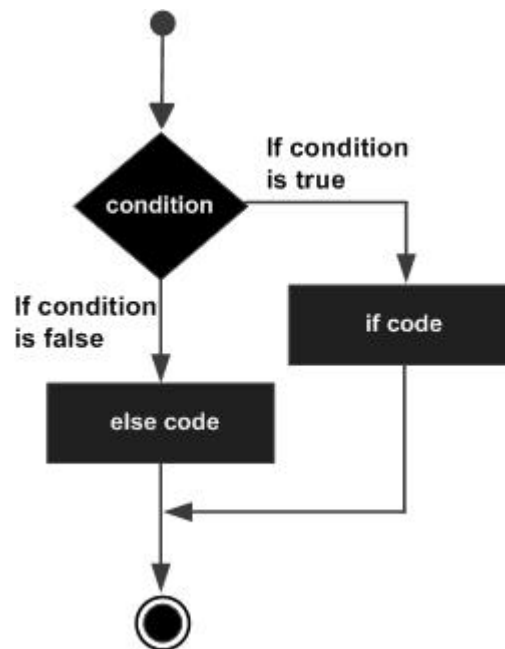
The **C compiler** evaluates the condition, and executes a statement or a block of statements following the **if statement** if it is true.

If the programming logic needs the computer to execute some other instructions when the condition is false, they are put as a part of the **else** clause.

An **if** statement is followed by an optional **else** statement, which executes when the Boolean expression is false.

## Flowchart of if-else Statement

The following flowchart represents how the **if-else** clause works in C –



Note that the curly brackets in the **if** as well as the **else** clause are necessary if you have more than one statements to be executed. For example, in the following code, we don't need curly brackets.

```
if (marks<50)
    printf("Result: Fail\n");
else
    printf("Result: Pass\n");
```

However, when there are more than one statements, either in the **if** or in the **else** part, you need to tell the compiler that they need to be treated as a compound statement.

Explore our **latest online courses** and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

## C if-else Statement Examples

### Example: Tax Calculation Using if-else Statement

In the code given below, the tax on employee's income is computed. If the income is below 10000, the tax is applicable at 10%. For the income above 10000, the excess income is charged at 15%.

</>

Open Compiler

```
#include <stdio.h>

int main() {
    int income = 5000;
```

```
float tax;
printf("Income: %d\n", income);

if (income<10000){
    tax = (float)(income * 10 / 100);
    printf("tax: %f \n", tax);
}
else {
    tax= (float)(1000 + (income-10000) * 15 / 100);
    printf("tax: %f", tax);
}
}
```

## Output

Run the code and check its output –

```
Income: 5000
tax: 500.000000
```

Set the income variable to 15000, and run the program again.

```
Income: 15000
tax: 1750.000000
```

## Example: Checking Digit Using if-else Statement

The following program checks if a char variable stores a digit or a non-digit character.

[Open Compiler](#)

```
#include <stdio.h>

int main() {
    char ch='7';

    if (ch>=48 && ch<=57){
        printf("The character is a digit.");
    }
    else{
```

```
    printf("The character is not a digit.");  
}  
return 0;  
}
```

## Output

Run the code and check its output –

The character is a digit.

Assign any other character such as "\*" to "ch" and see the result.

The character is not a digit.

## Example: if-else Statement Without Curly Braces

Consider the following code. It intends to calculate the discount at 10% if the amount is greater than 100, and no discount otherwise.

[Open Compiler](#)

```
#include <stdio.h>  
  
int main() {  
    int amount = 50;  
    float discount;  
    printf("Amount: %d\n", amount);  
  
    if (amount >= 100)  
        discount = amount * 10 / 100;  
    printf("Discount: %f \n", discount);  
    else  
        printf("Discount not applicable\n");  
  
    return 0;  
}
```

## Output

The program shows the following errors during the compilation –

error: 'else' without a previous 'if'

The compiler will execute the first statement after the **if** clause and assumes that since the next statement is not **else** (it is optional anyway), the subsequent `printf()` statement is unconditional. However, the next **else** is not connected to any **if** statement, hence the error.

## Example: if-else Statement Without Curly Braces

Consider the following code too –

[Open Compiler](#)

```
#include <stdio.h>

int main() {
    int amount = 50;
    float discount, nett;
    printf("Amount: %d\n", amount);

    if (amount<100)
        printf("Discount not applicable\n");
    else
        printf("Discount applicable");
        discount = amount*10/100;
        nett = amount - discount;
        printf("Discount: %f Net payable: %f", discount, nett);

    return 0;
}
```

## Output

The code doesn't give any compiler error, but gives incorrect output –

```
Amount: 50
Discount not applicable
Discount: 5.000000 Net payable: 45.000000
```

It produces an incorrect output because the compiler assumes that there is only one statement in the **else** clause, and the rest of the statements are unconditional.

The above two code examples emphasize the fact that when there are more than one statements in the **if** or **else** else, they must be put in curly brackets.

To be safe, it is always better to use curly brackets even for a single statement. In fact, it improves the readability of the code.

The correct solution for the above problem is shown below –

```
if (amount >= 100){
    discount = amount * 10 / 100;
    printf("Discount: %f \n", discount);
} else {
    printf("Discount not applicable\n");
}
```

## The else-if Statement in C

C also allows you to use **else-if** in the programs. Let's see where you may have to use an **else-if** clause.

Let's suppose you have a situation like this. If a condition is true, run the given block that follows. If it isn't, run the next block instead. However, if none of the above is true and all else fails, finally run another block. In such cases, you would use an **else-if** clause.

## Syntax of else-if Statement

Here is the syntax of the **else-if** clause –

```
if (condition){
    // if the condition is true,
    // then run this code
} else if(another_condition){
    // if the above condition was false
    // and this condition is true,
    // then run the code in this block

} else{
    // if both the above conditions are false,
    // then run this code
}
```

## Example of else-if Statement

Take a look at the following example –

&lt;/&gt;

Open Compiler

```
#include <stdio.h>

int main(void) {
    int age = 15;

    if (age < 18) {
        printf("You need to be over 18 years old to continue\n");
    } else if (age < 21) {
        printf("You need to be over 21\n");
    } else {
        printf("You are over 18 and older than 21 so you can continue \n");
    }
}
```

## Output

Run the code and check its output –

You need to be over 18 years old to continue

Now, supply a different value for the variable "age" and run the code again. You will get a different output if the supplied value is less than 18.