**C if...else Statement**

## C if Statement

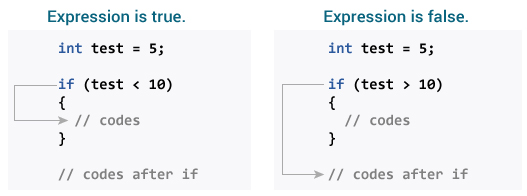
The syntax of the if statement in C programming is:

1. if (test expression)
2. {
3. // statements to be executed if the test expression is true
4. }

### How if statement works?

The if statement evaluates the test expression inside the parenthesis ().

* If the test expression is evaluated to true, statements inside the body of if are executed.
* If the test expression is evaluated to false, statements inside the body of if are not executed.



To learn more about when test expression is evaluated to true (non-zero value) and false (0), check [relational](https://www.programiz.com/c-programming/c-operators#relational) and [logical operators](https://www.programiz.com/c-programming/c-operators#logical).

### Example 1: if statement

1. // Program to display a number if it is negative
2. #include <stdio.h>
3. int main()
4. {
5. int number;
6. printf("Enter an integer: ");
7. scanf("%d", &number);
8. // true if number is less than 0
9. if (number < 0)
10. {
11. printf("You entered %d.\n", number);
12. }
13. printf("The if statement is easy.");
14. return 0;
15. }

**Output 1**

Enter an integer: -2

You entered -2.

The if statement is easy.

When the user enters -2, the test expression number<0 is evaluated to true. Hence, You entered -2 is displayed on the screen.

**Output 2**

Enter an integer: 5

The if statement is easy.

When the user enters 5, the test expression number<0 is evaluated to false and the statement inside the body of if is not executed

## C if...else Statement

The if statement may have an optional else block. The syntax of the if..else statement is:

1. if (test expression) {
2. // statements to be executed if the test expression is true
3. }
4. else {
5. // statements to be executed if the test expression is false
6. }

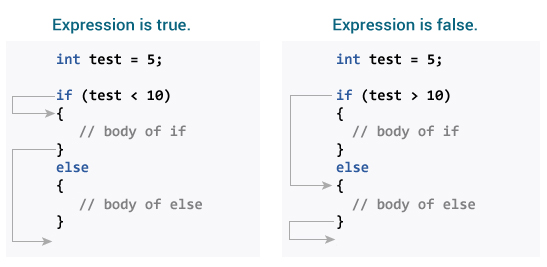
### How if...else statement works?

If the test expression is evaluated to true,

* statements inside the body of if are executed.
* statements inside the body of else are skipped from execution.

If the test expression is evaluated to false,

* statements inside the body of else are executed
* statements inside the body of if are skipped from execution.



### Example 2: if...else statement

1. // Check whether an integer is odd or even
2. #include <stdio.h>
3. int main()
4. {
5. int number;
6. printf("Enter an integer: ");
7. scanf("%d", &number);
8. // True if the remainder is 0
9. if (number%2 == 0)
10. {
11. printf("%d is an even integer.",number);
12. }
13. else
14. {
15. printf("%d is an odd integer.",number);
16. }
17. return 0;
18. }

**Output**

Enter an integer: 7

7 is an odd integer.

When the user enters 7, the test expression number%2==0 is evaluated to false. Hence, the statement inside the body of else is executed.

## C if...else Ladder

The if...else statement executes two different codes depending upon whether the test expression is true or false. Sometimes, a choice has to be made from more than 2 possibilities.

The if...else ladder allows you to check between multiple test expressions and execute different statements.

### Syntax of nested if...else statement.

1. if (test expression1)
2. {
3. // statement(s)
4. }
5. else if(test expression2)
6. {
7. // statement(s)
8. }
9. else if (test expression3)
10. {
11. // statement(s)
12. }
13. .
14. .
15. else
16. {
17. // statement(s)
18. }

### Example 3: C if...else Ladder

1. // Program to relate two integers using =, > or < symbol
2. #include <stdio.h>
3. int main()
4. {
5. int number1, number2;
6. printf("Enter two integers: ");
7. scanf("%d %d", &number1, &number2);
8. //checks if the two integers are equal.
9. if(number1 == number2)
10. {
11. printf("Result: %d = %d",number1,number2);
12. }
13. //checks if number1 is greater than number2.
14. else if (number1 > number2)
15. {
16. printf("Result: %d > %d", number1, number2);
17. }
18. //checks if both test expressions are false
19. else
20. {
21. printf("Result: %d < %d",number1, number2);
22. }
23. return 0;
24. }

**Output**

Enter two integers: 12

23

Result: 12 < 23

## Nested if...else

It is possible to include an if...else statement inside the body of another if...else statement.

### Example 4: Nested if...else

This program given below relates two integers using either <, > and = similar to the if...else ladder's example. However, we will use a nested if...else statement to solve this problem.

1. #include <stdio.h>
2. int main()
3. {
4. int number1, number2;
5. printf("Enter two integers: ");
6. scanf("%d %d", &number1, &number2);
7. if (number1 >= number2)
8. {
9. if (number1 == number2)
10. {
11. printf("Result: %d = %d",number1,number2);
12. }
13. else
14. {
15. printf("Result: %d > %d", number1, number2);
16. }
17. }
18. else
19. {
20. printf("Result: %d < %d",number1, number2);
21. }
22. return 0;
23. }

If the body of an if...else statement has only one statement, you do not need to use brackets *{}*.

For example, this code

1. if (a > b) {
2. print("Hello");
3. }
4. print("Hi");

is equivalent to

1. if (a > b)
2. print("Hello");
3. print("Hi");