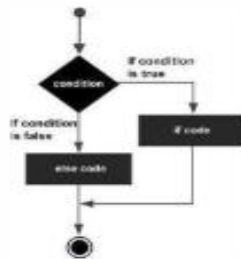


If Condition

If the boolean expression evaluates to **true**, then the **if** block of code is executed, otherwise **else** block of code is executed.

Flow Diagram



```

// if-else statement
if (condition)
{
    then-statement;
}
else
{
    else-statement;
}
// Next statement in the program.

// if statement without an else
if (condition)
  
```

Chapter 8 Relational Operators - page 62

The following table shows all the relational operators supported by C#. Assume variable **A** holds 10 and variable **B** holds 20, then

Operator	Description	Example
==	Checks if the values of two operands are equal or not, if yes then condition becomes true.	(A == B) is not true.
!=	Checks if the values of two operands are equal or not, if values are not equal then condition becomes true.	(A != B) is true.
>	Checks if the value of left operand is greater than the value of right operand, if yes then condition becomes true.	(A > B) is not true.
<	Checks if the value of left operand is less than the value of right operand, if yes then condition becomes true.	(A < B) is true.
>=	Checks if the value of left operand is greater than or equal to the value of right operand, if yes then condition becomes true.	(A >= B) is not true.
<=	Checks if the value of left operand is less than or equal to the value of right operand, if yes then condition becomes true.	(A <= B) is true.

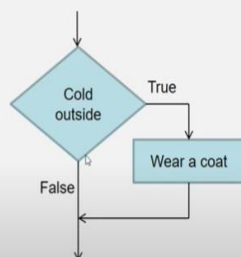
A Simple Decision Structure

- The flowchart is a single-alternative decision structure
- It provides only one alternative path of execution
- In C#, you can use the **if** statement to write such structures. A generic format is:

```

if (expression)
{
    Statements;
    Statements;
    etc.;
}
  
```

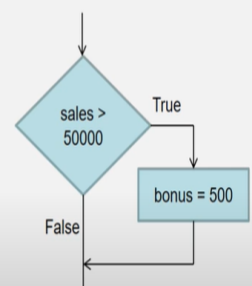
- The *expression* is a Boolean expression that can be evaluated as either true or false



if Statement with Boolean Expression

```

if (sales > 50000)
{
    bonus = 500;
}
  
```



First Decision Program on Student Programming and Maths grades

Write a program to ask the user to input the students name and their Programming and Maths results in their last exam. The program will compare the results and display an appropriate message for the following conditions.

RELATIONAL OPERATORS determine whether a specific relationship exist between two values E.G.

<

less than > greater than == equal to != not equal to <= less than or equal to >= greater than or equal to

Firstly Program and test the following conditions and outputs

- If the programming grade is equal to the maths grade output student name, the grades and an appropriate message, program and test

```
// Processing & Output

if (program_grade == maths_grade)
{
    // Print out appropriate message to student displaying both grades
}
```

- Then continue the add the following conditions and test. If the programming grade is less than 40 output student name, the grade and an appropriate message, write code and test

```
if (program_grade < 40)
{
    // Print out appropriate message to student displaying program grade
}
```

-
- If the maths grade is less than 40 output student name, the grade and an appropriate message, write code and test

```
if (maths_grade < 40)
{
    // Print out appropriate message to student displaying maths grade
}
```

-
- If the programming grade is less than the maths grade is less than 40 output student name, the grades and an appropriate message, write code and test

```
if (program_grade < maths_grade)
{
    /// Print out appropriate message to student displaying both grades
}
```

-
- If the programming grade is greater than the maths grade is less than 40 output student name, the grades and an appropriate message, write code and test

```
if (program_grade > maths_grade)
{
    // Print out appropriate message to student displaying both grades
}
```

-
- If the programming grade is greater than or equal to 40 output student name, the grade and an appropriate message, write code and test

```
if (program_grade >= 40)
{
    // Print out appropriate message to student displaying program grade
}
```

-

First Decision Program

- If the maths grade is greater than or equal to 40 output student name, the grade and an appropriate message, write code and test

```
if (maths_grade >= 40)
{
    // Print out appropriate message to student displaying maths grade
}
```

-
- If the maths grade and the programming grade are less than 40 output student name, the grades and an appropriate message, write code, and test

```
if (program_grade < 40 && maths_grade <40)
{
    // Print out appropriate message to student displaying program grade
}
```

-
- If the maths grade and the programming grade are greater than or equal to 40 output student name, the grades and an appropriate message, write code, and test

```
if (program_grade >= 40 && maths_grade >= 40)
{
    // Print out appropriate message to student displaying program grade
}
```

-

- An ***if-else*** statement will execute one block of statement if its Boolean expression is true or another block if its Boolean expression is false
- It has two parts: an *if* clause and an *else* clause
- In C#, a generic format looks:

```

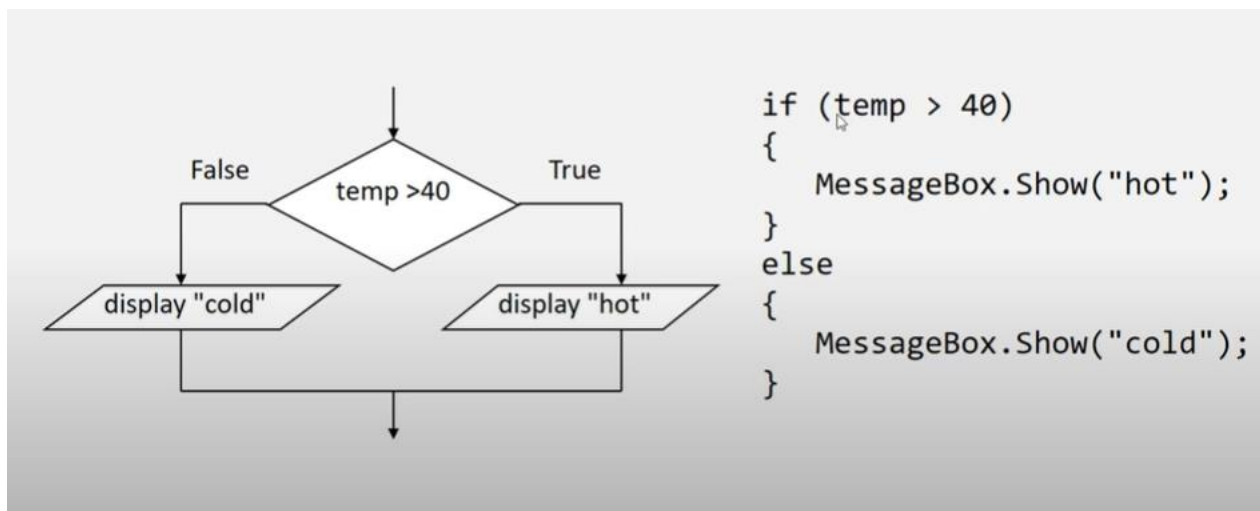
if (expression)
{
    statements;
}
else
{
    statements;
}

```

2

Second Decisions Program Temperature

Write a program and get the user to enter the Temperature. If the temperature is greater than 40 degrees then display the temperature and an appropriate message if not also display the temperature and an appropriate message



```

// Processing & Output
if (temp > 40)
{
    // Print out temperature and appropriate message of hot
}
else
{
    // Print out temperature and appropriate message of cold
}

```

² <https://www.youtube.com/watch?v=3wSL84Ayyvg>