C++ conditional statements and operators

In C++, there are two conditional statements for making decisions. The two conditional statements are**if else** and **switch case**.

**if/else statement**

**If statement** will evaluate the condition. If it is true, it will execute the statements that follow it, otherwise, it will execute the statements in **else**block.

if statement without else:

**if** (*condition*) {

Statement

     Statement

}

or

if statement with else:

**if** (*condition*) {

  Statement

  Statement

}

**else** {

  Statement

  Statement

}

If you have more than one condition to check, you can use multiple **else if**statements:

**if**(condition1){

   Statement

   Statement

}

**else if**(condition2){

   Statement

   Statement

}

**else if**(condition3){

  Statement

  Statement

}

....

**else**{

  Statement

  Statement

}

Example:

int age;  
cout << "Enter your age: ";  
cin >> age;  
if (age >= 18)  
   cout << "\nYou can play the game";

else cout<< "\n You are not allowed to play the game";

The below table is the list of operators which can be used in **if else**statement.

|  |  |
| --- | --- |
| Sign | Meaning |
| == | Equal |
| != | Not equal |
| > | Greater than |
| >= | Greater than or equal to |
| < | Less than |
| <= | Less than or equal to |

If you want to put more than one statement in an**if else** statement then you must write them in curly brackets.

Example:

int age;  
cout << "Enter your age: ";  
cin >> age;  
if (age == 18)  
{  
   cout << "\nYou are 18 years old. ";  
   cout << "\nYou are allowed to play the game";  
}

Note: if/else statements can be nested

Example:

string user[20], password[30];  
cout << "Enter username: ";  
cin >> user;  
cout << "\nEnter password: ";  
cin >> password;  
if (user =="dara")  
   if (password == "12345")  
      cout << "\nWelcome"<<"  "<<user;

    else cout<<"\nInvalid login! Try again";

**Switch statement**

The**switch statement** can be used to test for multiple values of a variable. The **break** keyword lets the program stop when the desire value is met. The **default**keyword lets the program execute the statements that follow it when the desire value is not met.

switch(variable)

{

case val1: statement1

        break;

case val2: statement2

        break;

………………………

default: statement

}

Example:

int age;  
cout << "Enter your age: ";  
cin >> age;  
switch (age)  
{  
   case 1: cout << "You are 1 year old";  
           break;  
   case 2: cout << "You are 2 years old";  
           break;  
   case 3: cout << "You are 3 years old";  
           break;  
   default: cout << "You are older than 3 years";  
}