

### Include Headers

**#include** <headername>

**#include** <iostream> Several standard stream objects

**#include** <string> std::basic\_string class template

**#include** <math> Behaves as if each name from <cmath> is placed in global namespace, except for names of mathematical special functions

**#include** <fstream> std::basic\_fstream, std::basic\_ifstream, std::basic\_ofstream class templates and several typedefs

**#include** <cctype> Behaves as if each name from <cctype> is placed in global namespace

### Namespace

**using namespace** std;

### Data Types

**int** integer type (-32768 to 32767)

**char** single character type

**float** Floating-point types (single precision)

**double** Floating-point types (double precision)

**bool** *true* or *false*

**void** type with an empty set of values

### Comments

//Single line

/\*Multiple lines\*/

### Pointers

**int \*ptr;** Pointer definition

**ptr = &var1;** *ptr* set to address of *var1*

**var2 = \*ptr;** set *var2* to value of *var1*

### Arithmetic Operators

**+, -, \*, /** Addition, subtraction, multiplication, division

**%** Modulus (rest of division)

**++** Add 1 to variable (example: i++)

**--** Subtract 1 to variable (example: i--)

**+=, -=, \*=, /=** Add/subtract/multiply/divide a value from the variable (example: i+=2; is equal to i=i+2;)

### Relational Operators

**<** Less than

**<=** Less or Equal than

**>** Greater than

**>=** Greater or Equal than

**==** Equal than

**!=** Not Equal than

### Logical Operators

**||** OR

**&&** AND

**!** NOT

### If Else

```
if (condition) {
    \Do something
} else if (condition) {
    \Do something
} else {
    \Do something
}
```

*condition* must be a condition that return a boolean value *true* or *false*

**else/else if** blocks are optional

### Switch

```
switch (variable) {
case constant1:
    \Do something
break;
case constant2:
    \Do something
break;
default:
    \Do something
break;
}
```

**default** is optional

### While Loop

```
while (condition) {
    \Do something
}
```

Loop as long as the *condition* is *true*, when it becomes *false* it exits



By **vittochan** (vittochan)  
[cheatography.com/vittochan/](https://cheatography.com/vittochan/)

Published 8th January, 2023.  
Last updated 9th January, 2023.  
Page 1 of 2.

Sponsored by **Readable.com**  
Measure your website readability!  
<https://readable.com>

### Do-While Loop

```
do {  
    \\Do something  
} while (condition);
```

Loop as long as the *condition* is *true*, when it becomes *false* it exits

### For Loop

```
for (initialize, condition, update) {  
    \\Do something  
}  
example:  
for (int i=0; i<10; i++) {  
    \\Do something  
}
```

Loop as long as the *condition* is *true*, when it becomes *false* it exits



By **vittochan** (vittochan)  
[cheatography.com/vittochan/](https://cheatography.com/vittochan/)

Published 8th January, 2023.  
Last updated 9th January, 2023.  
Page 2 of 2.

Sponsored by **Readable.com**  
Measure your website readability!  
<https://readable.com>