

VARIABLES . DATA TYPES

We can define a variable as a portion of memory to store a determined value. Each variable needs an identifier that distinguishes it from the others.

Identifiers

A valid identifier is a sequence of one or more letters, digits or underscore characters (_).

Neither spaces or punctuation marks or symbols can be part of an identifier. Only letters, digits and single underscore characters are valid.

The identifiers cannot match any *reserved keywords*.

Keywords in C++

```
else
for
if
int
long
namespace
or
return
void
while
```

An identifier written in capital letters is not equivalent to another one with the same name but written in small letters.

Identifier of a variable

```
AB ≠ ab
M1_START ≠ m1_start
```

Fundamental data types

Name	Description
<i>int</i>	Integer
<i>float</i>	Floating point number
<i>char</i>	Character or small integer
<i>void</i>	

Declaration of variables

In order to use a variable in C++ , we must first declare it specifying which data type we want it to be.

```
int a,b,c;
```

A variable can be global (declared in the main body of the source code) or local (declared within the body of a function or a block)

a) global variables

```
#include <iostream>
using namespace std;

// declaring global
variables:
    int a, b;
    int result;

int main ()
{
```

b) local variables

```
#include <iostream>
using namespace std;

int main ()
{
    // declaring local
variables:
    int a, b;
    int result;
```