### Data Types in C++

They are used to define type of variables and contents used. Data types define the way you use storage in the programs you write. Data types can be built in or abstract.

#### Built in Data Types

These are the data types which are predefined and are wired directly into the compiler. eg: int, char etc.

#### User defined or Abstract data types

These are the type, that user creates as a class. In C++ these are classes where as in C it was implemented by structures.

### Basic Built in types

|  |  |
| --- | --- |
| char | for character storage ( 1 byte ) |
| int | for integral number ( 2 bytes ) |
| float | single precision floating point ( 4 bytes ) |
| double | double precision floating point numbers ( 8 bytes ) |

*Example* :

char a = 'A'; // character type

int a = 1; // integer type

float a = 3.14159; // floating point type

double a = 6e-4; // double type (e is for exponential)

#### Other Built in types

|  |  |
| --- | --- |
| bool | Boolean ( True or False ) |
| void | Without any Value |
| wchar\_t | Wide Character |

### Enum as Data type

Enumerated type declares a new type-name and a sequence of value containing identifiers which has values starting from 0 and incrementing by 1 every time.

For Example :

enum day(mon, tues, wed, thurs, fri) d;

Here an enumeration of days is defined with variable d. *mon* will hold value 0, *tue* will have 1 and so on. We can also explicitly assign values, like, enum day(mon, tue=7, wed);. Here, *mon* will be 0, *tue* is assigned 7, so *wed* will have value 8.

### Modifiers

Specifiers modify the meanings of the predefined built-in data types and expand them to a much larger set. There are four data type modifiers in C++, they are :

1. long
2. short
3. signed
4. unsigned

Below mentioned are some important points you must know about the modifiers,

* **long** and **short** modify the maximum and minimum values that a data type will hold.
* A plain int must have a minimum size of **short**.
* Size hierarchy : short int < int < long int
* Size hierarchy for floating point numbers is : float < double < long double
* **long float** is not a legal type and there are no **short floating point** numbers.
* **Signed** types includes both positive and negative numbers and is the default type.
* **Unsigned**, numbers are always without any sign, that is always positive.