

Data Types

- When we wish to store data in a C++ program, such as a whole number or a character, we have to tell the compiler which type of data we want to store. The data type will have characteristics such as the range of values that can be stored and the operations that can be performed on variables of that type.

❖ C++ data types are of two types

1.FUNDAMENTAL TYPES

2.DERIVED TYPES

Fundamental data types are those that are not composed of other data types

There are five types of data type-

1. int data type (for integer)
2. Char data type (for character)
3. Float data type
4. Void data type

• *int data type*

The integer type is used for storing whole numbers. We can use signed, unsigned or plain integer values as follows:

```
signed int index = 41982;
```

```
signed int temperature = -32;
```

```
unsigned int count = 0;
```

```
int height = 100;
```

```
int balance = -67;
```

• **Char data type**

The character type is used to store characters - typically ASCII characters but not always. For example:

```
char menu Selection = 'q';
```

```
char user Input = '3';
```

•Float data type

Floating point types can contain decimal numbers, for example 1.23, -.087. There are three sizes, float (single-precision), double (double-precision) and long double (extended-precision). Some examples:

float Celsius = 37.623;

double Fahrenheit = 98.415;

long double account Balance = 1897.23

•Void data type

A data type that has no values or operators and is used to represent nothing



Derived data types are those that are defined in terms of other data types, called base types. **Derived** types may have attributes, and may have element or mixed content

There are five types of derived data types

1. Arrays
2. Structure
3. Pointer
4. References
5. Constants

• ARRAYS

C Arrays are the data structures which can be used to store consecutive values of the same data types. C Arrays can be declared for all c++ data types viz., int, float, double, char, struct, char etc., All the values are stored in consecutive memory locations. The values can be accessed by using the position of the stored value.

CONSTANT

The const keyword specifies that a variable's value is constant and tells the compiler to prevent the programmer from modifying it.

•Pointer

A **pointer** in **C** is said to "point to" the memory address that is stored in it. Also, when defining a **C** pointer variable, we must specify the type of variable to which it is pointing. For example, to define a pointer, which will store a memory address at which exists an int, we can do the following:

•REFERENCE

A *reference* is an alias or an alternative name for an object. All operations applied to a reference act on the object to which the reference refers. The address of a reference is the address of the aliased object.

•FUNCTIONS

Functions are building blocks of the programs. They make the programs more modular and easy to read and manage. All C++ programs must contain the function `main()`. The execution of the program starts from the function `main()`. A C++ program can contain any number of functions according to the needs

DATA TYPE

There are four types of user defined derived data type:

1. CLASS
2. UNION
3. STRUCTURE
4. ENUMERATION

• UNION

A **union** is a user-defined data or class type that, at any given time, contains only one object from its list of members (although that object can be an array or a class type).

• Structure

Structure is a collection of variables under a single name. Variables can be of any type: int, float, char etc. The main difference between structure and array is that arrays are collections of the same data type and structure is a collection of variables under a single name

• **CLASS**

A class in C++ is an encapsulation of data members and functions that manipulate the data. The class can also have some other important members which are architecturally important.

• **ENUMERATIONS**

An enumeration is a user-defined type consisting of a set of named constants called enumerators.