Introduction to C++ set

- C++ sets are **special containers for storing unique elements** in order. Ordering of the elements should be in a **specified manner** in the C++ set as it puts most of its emphasis on the **key and pairs** of elements.

- Each **element should be different**, and once put in the C**++ set container** cannot be modified as elements will be treated as **constant**.

- Although these **elements** are considered **constant** and cannot be modified still any new element in the (**key and value format**) can be **inserted and retrieved** easily from the container.

Syntax

C++ Set is an associative container concerning other standard library components of C++ where all the elements play an important role.

This is represented using the following template :

```
template <class T_ype,
class Compare = less<T_ype>,
class Alloc = allocator<T_ype>
>class set;
```

The template includes the following parameter:

- **T_ype**: It signifies the type of element that is present in the container.
- **Compare**: A class for comparison is also introduced to take two arguments of the **same type** and then return a Boolean value after comparison. Usage of this is optional in the sense default value will be considered if it is less than the compared values.
- **Alloc**: Alloc is the class that is specifically used to allocate the values to the storage class and according to the storage class allocator.

How set function work in C++?

A set function is part of the standard library of C++, which is used for storing some unique elements and is then used for performing many operations on top of it.

- A **set function** is part of the standard library of C++, which is used for **storing some unique elements** and is then used for **performing many operations** on top of it.

Beneficial :

- ⇨ This is a function that allows programmers to use the C++ set easily whenever there is a **requirement based on key and value pairs.**
- ⇨ In a set, the **value present** is the **key** that is used for accessibility according to the requirement.
- ⇨ All the elements once present in **the container cannot be modified** as the value **become constant.**
- ⇨ Elements in the set **don't allow duplicate elements** as the value and value pairs.
- ⇨ The elements **can be inserted and retrieved** accordingly. However, the only modification is not possible.
- ⇨ Internally also **element can be sorted**, but with **some protocols or restrictions** like comparisons are generated when the internal object present gets compared.

Conclusion:

- C++ set plays a very important role like other standard library modules. It helps programmers to play around with the elements to

put them easily into **order with traversals** , **manipulation** and **retrieval**.

- It provides easy accessibility also when compared with the **un_ordered** subset in terms of easy accessibility.