**Chapter6 Inheritance**

**Inheritance**

* Inheritance is a process in which one object acquires all the properties and behaviours of its parent object automatically.
* The class which inherits the members of another class is called derived class.
* the class whose members are inherited is called base class.
* The derived class is the specialized class for the base class.

**Advantage of Inheritance**

**Code reusability:**

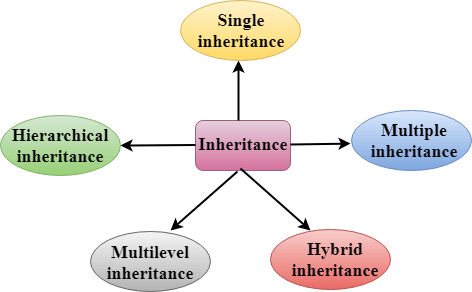
 Now you can reuse the members of your parent class. So, there is no need to define the member again.

So, less code is required in the class.

## Types Of Inheritance

**C++ supports five types of inheritance:**

* Single inheritance
* Multiple inheritance
* Hierarchical inheritance
* Multilevel inheritance
* Hybrid inheritance

****

## Derived Classes

A Derived class is defined as the class derived from the base class.

**SYNTAX:**

**class** derived\_class\_name:: visibility mode base\_class\_name

{

    // body of the derived class.

}

**Where ,**

**derived\_class\_name:** It is the name of the derived class.

**visibility mode:** The visibility mode specifies whether the features of the base class are publicly inherited or privately inherited. It can be public or private.

**ACCESS SPECIFIER**

Access specifiers define how the members (attributes and methods) of a class can be accessed.

There are three access specifiers:

**Public**: When the member is declared as public, it is accessible to all the functions of the program.

**Private**: When the member is declared as private, it is accessible within the class only.

**Protected**: When the member is declared as protected, it is accessible within its own class as well as the class immediately derived from it.

## Single Inheritance

**Single inheritance** is defined as the inheritance in which a derived class is inherited from the only one base class.

C++ Inheritance Where 'A' is the base class, and 'B' is the derived class.

**EXAMPLE**

#include <iostream>

**using** **namespace** std;

**class** Account

 {

**public**:

**float** salary = 60000;

 };

**class** Programmer: **public** Account

 {

**public**:

**float** bonus = 5000;

   };

**int** main(**void**)

{

     Programmer p1;

     cout<<"Salary: "<<p1.salary<<endl;

     cout<<"Bonus: "<<p1.bonus<<endl;

**return** 0;

}

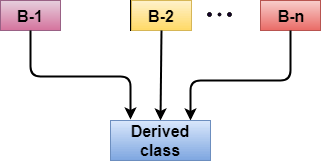
## Multilevel Inheritance

**Multilevel inheritance** is a process of deriving a class from another derived class.

## C++ Inheritance

## Multiple Inheritance

**Multiple inheritance** is the process of deriving a new class that inherits the attributes from two or more classes.



## Hybrid Inheritance

Hybrid inheritance is a combination of more than one type of inheritance.

## C++ Inheritance

## Hierarchical Inheritance

Hierarchical inheritance is defined as the process of deriving more than one class from a base class.

## C++ Inheritance