Chapter – 5
INHERITANCE: EXTENDING CLASSES

Q1. What is Inheritance ?

Ans. It is a special feature of OOPS. Inheritance is capability to inherit the properties of one class in to another class.

The derive new class is called **derived class (sub class)** and old class is called based class (**super class).**

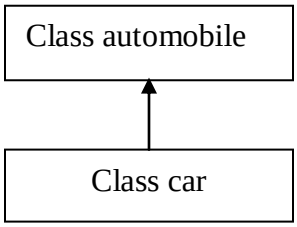
The Class whose properties of data members are **inherited,** is called **Base Class or Super Class** and the class that inherits these properties, is called **Derived Class or Sub Class.**

Exp1:- **If Class A** inherits the data of **Class B** then we can say **A is Sub Class and B is Super Class.**

Q2. What are the different types of inheritance ?

Ans. Type of Inheritance

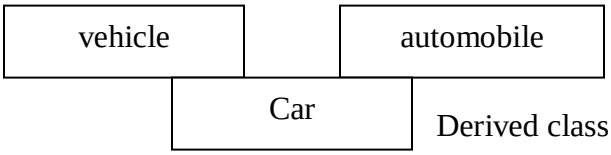
1. Single Inheritance



it is a Base class (super)

it is a sub class (derived)

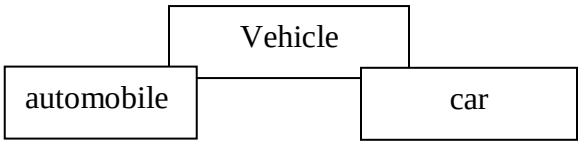
2. Multiple Inheritance



Base class

Derived class

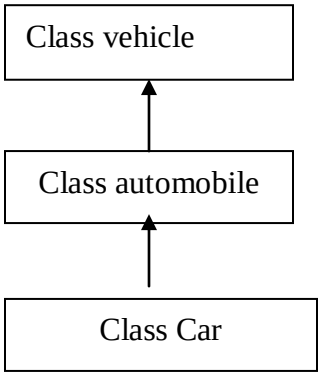
3. Hierarchical Inheritance



Base class (super)

Sub Class(derived)

4. Multilevel Inheritance



it is a Base class (super) of automobile

it is a sub class (derived) of automobile and base class of class car

derived class(sub) of class automobile

5. Hybrid Inheritance

It is combination of two or more forms of inheritance.

Q2. What do you mean by Base and Derived Class ?

Ans. A derived class (or sub class) has to identify the class from which it is derived i.e. its base class (or super class)

class derived-class-name : visibility-mode base-class-name
{
members of derived class;
};

class is a key word and **visibility mode is a access speicifier (i.e. public, private or protected), : (colon) is used separation**

colon (:) indicates derived class (sub class) is based on base class(super class)

Example

```
class car : public automobile
{
members
};
```

Q3. What is Multiple Inheritance ?

Ans. Multiple Inheritance means deriving a class from more than one base class.

```
Class kvschool
        {
                int rollno1;
                void num1();
        Public:
                {
                float kvroll;
                void num2();
        };
class kvstud
        {
                int rollno2
                void num3();
        Protected:
                float kvroll1;
                void num4();
        };
class kvclass : public kvschool, public kvstud
        {
                int kvroll2;
        protected:
                void display();
        };

        // to complete the program deifine display () – in class kvclass
                i.e. kvclass ::display()
```

Q4. Define the needs and objectives of Inheritance.

Ans The major needs and objectives of inheritance are:

(i) *It ensures the closeness with the real world models.*
(ii) *It extend the functionality of an existing class.*
(iii) *It establishes "a kind of" relationship.*
(iv) *It helps in reuse of an existing class by a subclass (reusability).*
(v) *It implements transitive nature(if a class Y inherits properties from class X, then all subclassY will automatically inherit the properties of X)*
(vi) *The redundancy can be reduced by abstracting a super class from few sub classes.*
(vii) *It is conept of reusability.*

Q5. Give the following definitions, answer the questions that follow:-

```
#include <iostream.h>
class book
{
char title[20];
char author[20];
int noof pages;
public:
        void read();
        void show();
};
class textbook: private textbook
{
int noofchapters, noof assignments;
protected:
int standard;
void readtextbook();
void showtextbook();
};
class physicsbook: public textbook
{
char topic[20];
public:
void readphysicsbook();
void showphysicsbook();
};
```

(i) Name the members, which can be accessed from the member functions of class physicsbook.
(ii) Name the members, which can be accessed by an object of Class textbook.
(iii) Name the members, which can be accessed by an object of Class physicsbook.
(iv) What will be the size of an object (in bytes) of class physicsbook.

Ans

(i) standard , readtextbook(),showtextbook() and topic;
(ii) readtextbook() and showtextbook()
(iii) readphysicsbook(), showphysicsbook(), readtextbook() and showtextbook()
(iv) The size of object of physicsbook= size of book + size of Textbook + size of physicsbook.
= 42+6+20 = 68 bytes

Q6. Consider the following declarations and answer the questions given below:

```
Class vehicle
{
int wheels;
protected:
int passenger;
public:
void inputdata( int, int);
void outputdata();
};
class heavyvehicle: protected vehicle
{
int dieselpetrol;
protected:
int load;
public:
void readdata( int, int);
void writedata();
};
class bus:private heavyvehicle
{
char marks[20];
public:
void fetchdata(char);
void displaydata();
};
```

(i) Name the class and derived class of the class **heavyvehicle.**

(ii) Name the data members that can be accessed from function **displaydata()**

(iii) Name the data members that can be accessed by an object of **bus class**

(iv) Is the member function outputdata() accessible to the objects of **heavyvehicle class.**

Ans

(i) **base class = vehicle, derived class = bus**

(ii) The data members passenger, load, make are available to function display data

(iii) No data members can be accessed by the object of bus calss.

(iv) No member functions outputdata () is not accessible to the objects of heavy vehicle class.

Q7 . What type of C++ class members (data members and member functions) are not inherited?

Ans **Data member** : Static data members of the base class are not inherited by the derived class

Member functions: Constructors and destructors of base class are not inherited