Experiment No. 04

Title: To study and implementing the concept of Inheritance.	
Aim: To study and implementing the concept of Inheritance.	

Theory:

Inheritance Basics

Inheritance is a major component of object-oriented programming. Inheritance will allow to you define a very general class, and then later define more specialized classes by simply adding some new details to the older more general class definition. This saves work, because the more specialized class *inherits* all the properties of the general class and you, the programmer, need onlyprogram the new features.

Inheritance in Java is a mechanism in which one object acquires all the properties and behaviors of a parent object.

The syntax of Java Inheritance

```
class Subclass-name extends Superclass-name
{
  //methods and fields
}
```

The **extends keyword** indicates that you are making a new class that derives from an existing class. The meaning of "extends" is to increase the functionality.

In the terminology of Java, a class which is inherited is called a parent or superclass, and the new class is called child or subclass.

//Inheritance

```
class A
{
        void meth1()
        {
                System.out.println("I am in Adcet");
        }
}
class B extends A
{
        void meth2()
        {
                System.out.println("I am in PL LAB");
        }
}
class demo
{
        public static void main(String ar[])
        {
                B
                      b=n
                ew B();
                b.meth1
                ();
                b.meth2();
        }
}
```

Types of inheritance in java

On the basis of class, there can be three types of inheritance in java: **single, multilevel and hierarchical**.

In java programming, multiple and hybrid inheritance is supported through interface only.

Single Inheritance :

When a class inherits another class, it is known as a ***single inheritance*.**

Multilevel Inheritance :

When there is a chain of inheritance, it is known as ***multilevel inheritance*.**

Hierarchical Inheritance :

When two or more classes inherit a single class, it is known as ***hierarchical inheritance*.**

Note: Multiple inheritance is not supported in Java through class.

Q) Why multiple inheritance is not supported in java?

To reduce the complexity and simplify the language, multiple inheritance is not supported in java.

Consider a scenario where A, B, and C are three classes. The C class inherits A and B classes. If A and B classes have the same method and you call it from child class object, there will be ambiguity to call the method of A or B class.

Since compile-time errors are better than runtime errors, Java renders compile-time error if you inherit 2 classes. So whether you have same method or different, there will be compile time error.

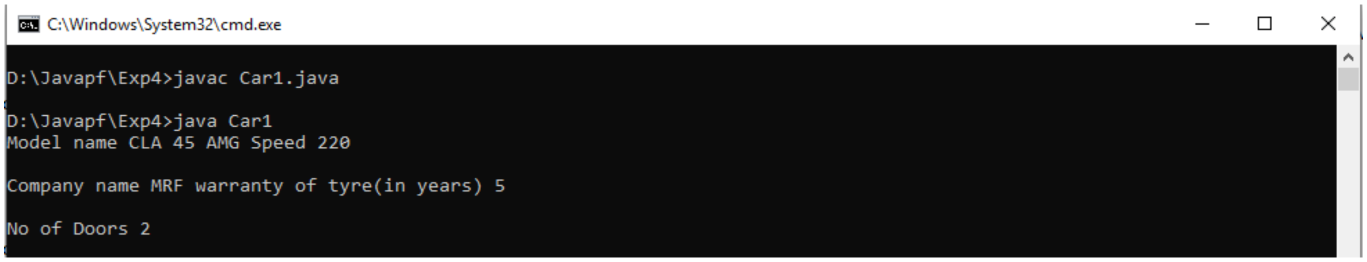
Statement: Create Separate Engine, Tyre, and Door Class. Create a Car class using these classes. Andshow functionality of each component in the car.

Program:

```
import java.util.*; class Engine
{
        int speed;
        String model_name;

        public void eng_Details(int e_speed,String m_name)
        {
                speed=e_speed; model_name=m_name;
                System.out.println("Model name "+model_name+" Speed "+speed);
        }
}
class Tyre extends Engine
{
        int warranty;
        String company_name;


        public void tyre_Details(int t_warranty,String c_name)
        {
                warranty=t_warranty; company_name=c_name;
                System.out.println("\nCompany name "+company_name+" warranty of tyre(in
years)"+warranty);
        }
}

class Door extends Tyre
{
        int door_no;

        public void door_Details(int n_door)
        {
                door_no=n_door;
                System.out.println("\n No of Doors "+door_no);
        }
}

class Car1
{
        public static void main(String[] args)
        {
                Engine e=new Engine();
                Tyre t=new Tyre();
                Door d=new Door();
                e.eng_Details(220,"CLA 45 AMG");
                t.tyre_Details(5,"MRF");
```

```
                    d.door_Details(2);
          }
}
```

Output:-



```
C:\Windows\System32\cmd.exe

D:\Javapf\Exp4>javac Car1.java

D:\Javapf\Exp4>java Car1
Model name CLA 45 AMG Speed 220

Company name MRF warranty of tyre(in years) 5

No of Doors 2
```

Conclusion: Thus we have studied the concept of inner class and inheritance and implement it.