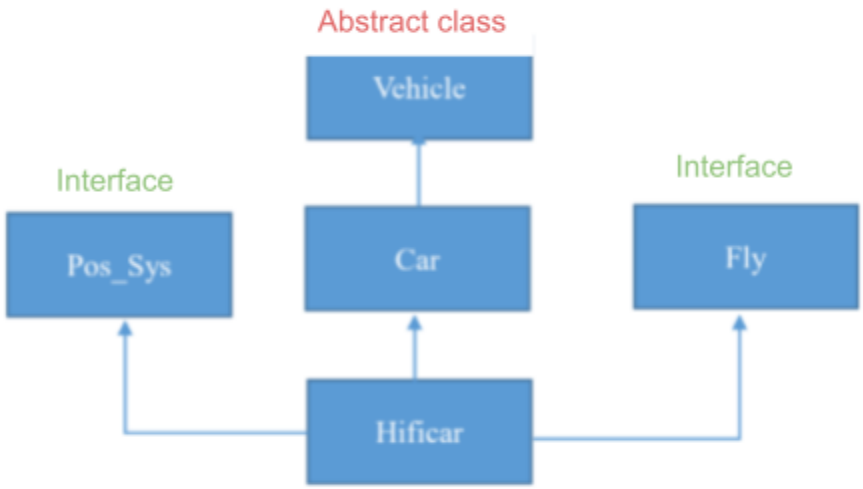
PPL LAB#8 08/05/2021
SEJAL KSHIRSAGAR 2372

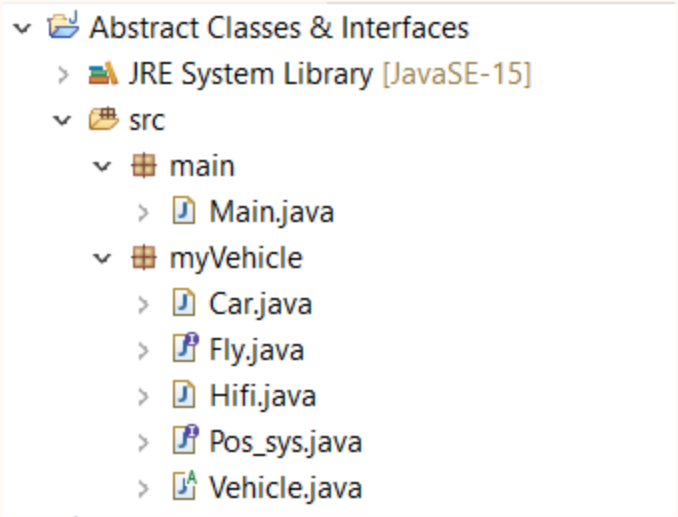
Title: Implement the following concepts by constructing suitable classes in Java - a. Abstract classes and abstract methods b. Interfaces.

Objective: Implementation using Abstract Class, Abstract methods and Interface in Java.

Sample Code: Implement following Hierarchy



JAVA PROGRAM:



Main.java

```
package main;

import myVehicle.Car;
import myVehicle.Vehicle;
import myVehicle.Hifi;
import myVehicle.Fly;
import myVehicle.Pos_sys;

public class Main {

public static void main(String[] args) {

        //Vehicle V1 = new Vehicle(); //Cannot instantiate abstract class object

        Car C1 = new Car(5, 'a');
        C1.start();
        C1.stop();
        C1.reg_vehicle();

        Hifi H1 = new Hifi();
        H1.change_gears(4);
        H1.reg_vehicle();
        H1.ferrari();
        H1.start();
        H1.stop();
        H1.start(3);
```

```
        H1.flyfast(80);
        H1.flyslow(20);
        H1.search_location("Moon");
        H1.show_road_ahead();

        //runtime polymorphism or dynamic method dispatch
        //- with reference of base class calling methods of derived class
        Vehicle V1[] = new Vehicle[2]; //array of references
        Car C2 = new Car(4, 'n');
        Hifi H2 = new Hifi();
        V1[0] = C2;
        V1[1] = H2;
        for(int i=0; i<2; i++) {
                V1[i].start();
                V1[i].stop();
        }

        //using references of interfaces
        Fly F1 = H2;
        F1.flyfast(90);
        F1.flyslow(10);
        Pos_sys P1 = H2;
        P1.show_road_ahead();
        P1.search_location("Sun");
}

}
```

Car.java

```
package myVehicle;

public class Car extends Vehicle {

int no_gear;
char ac;

public Car() {
        no_gear = 0;
        ac = 0;
}

public Car(int ng, char a) {
        no_gear = ng;
        ac = a;
}

@Override
public void start() {
        System.out.println("Car has started...");
}

@Override
public void stop() {
        System.out.println("Car has stopped...");
```

```
}

public void go_reverse() {
        System.out.println("Car is going in reverse...");
}

public void start(int gears) {
        System.out.println("Car has started. It is in "+gears+" gears.");
}

}
```

Fly.java

```
package myVehicle;


public interface Fly {


public void flyfast(int s);


public void flyslow(int s);

}
```

Hifi.java

```
package myVehicle;

public class Hifi extends Car implements Pos_sys, Fly {

public Hifi(){
	super(5, 'a');
}

@Override
public void flyfast(int s) {
	System.out.println("Hifi flying fast with speed: "+s);
}

@Override
public void flyslow(int s) {
	System.out.println("Hifi flying slow with speed: "+s);
}

@Override
public void show_road_ahead() {
	System.out.println("Hifi show road ahead.");
}

@Override
public void search_location(String l) {
```

```
        System.out.println("Hifi location: "+l);
}

public void start() {
        System.out.println("Hifi has started...");
}

public void stop() {
        System.out.println("Hifi has stopped...");
}

public void ferrari() {
        System.out.println("Hifi car Ferrari.");
}

public void start(int gears) {
        System.out.println("Hifi has started. It is in "+gears+" gears.");
}

public void change_gears(int gears) {
        gears++;
}
}
```

Pos_sys.java

```
package myVehicle;

public interface Pos_sys {

public void show_road_ahead() ;

public void search_location(String l) ;
}
```

Vehicle.java

```
package myVehicle;


public abstract class Vehicle {


int reg_no;

double hp;

double cost;


Vehicle(){

        reg_no = 0;

        hp = 0;

        cost = 0;

}


Vehicle(int rn, double p, double c){

        reg_no = rn;

        hp = p;

        cost = c;

}
```

```
//concrete method
public void reg_vehicle() {
        System.out.println("Registration number: "+reg_no);
        System.out.println("HP: "+hp);
        System.out.println("Cost: "+cost);
}

//abstract methods
public abstract void start();
public abstract void stop();
}
```

OUTPUT:

Car has started...

Car has stopped...

Registration number: 0

HP: 0.0

Cost: 0.0

Registration number: 0

HP: 0.0

Cost: 0.0

Hifi car Ferrari.

Hifi has started...

Hifi has stopped...

Hifi has started. It is in 3 gears.

Hifi flying fast with speed: 80

Hifi flying slow with speed: 20

Hifi location: Moon

Hifi show road ahead.

Car has started...

Car has stopped...

Hifi has started...

Hifi has stopped...

Hifi flying fast with speed: 90

Hifi flying slow with speed: 10

Hifi show road ahead.

Hifi location: Sun
