

Name of Experiment :-

A client class can drive one vehicle out of 5 types Bus, Truck, Motorcycle, Microbus, CNG. The type of vehicle will be selected at run time. The actions inside the vehicle are to change gear, speed up, speed down and press break. The business logic of each action is to display two statements: the name of the vehicle and actions like "Changing gear". Write a JAVA program to demonstrate the explained problem.

Introduction :-

We have to define a interface vehicle which will have 4 methods for 4 actions mentioned in the problem statement. Then we will have to define 5 classes that would implement the interface and define the methods. Then finally we will define a main class that will hold the main method.

Objectives :

- * to learn how interface work
- * to learn how to solve problems using interfaces
- * to learn when we can use interfaces correctly.

Analysis :-

After analysing our problem we have found following components of our problem:

- * an interface for ~~with~~ vehicles with all the actions as methods
- * 5 classes that would implement the interface and define the actions as their own.
- * a main class that will contain the main method which will use the vehicles and decide which vehicle to use at runtime.

From above analysis the conceptual class diagram is drawn below, in Figure-1:-

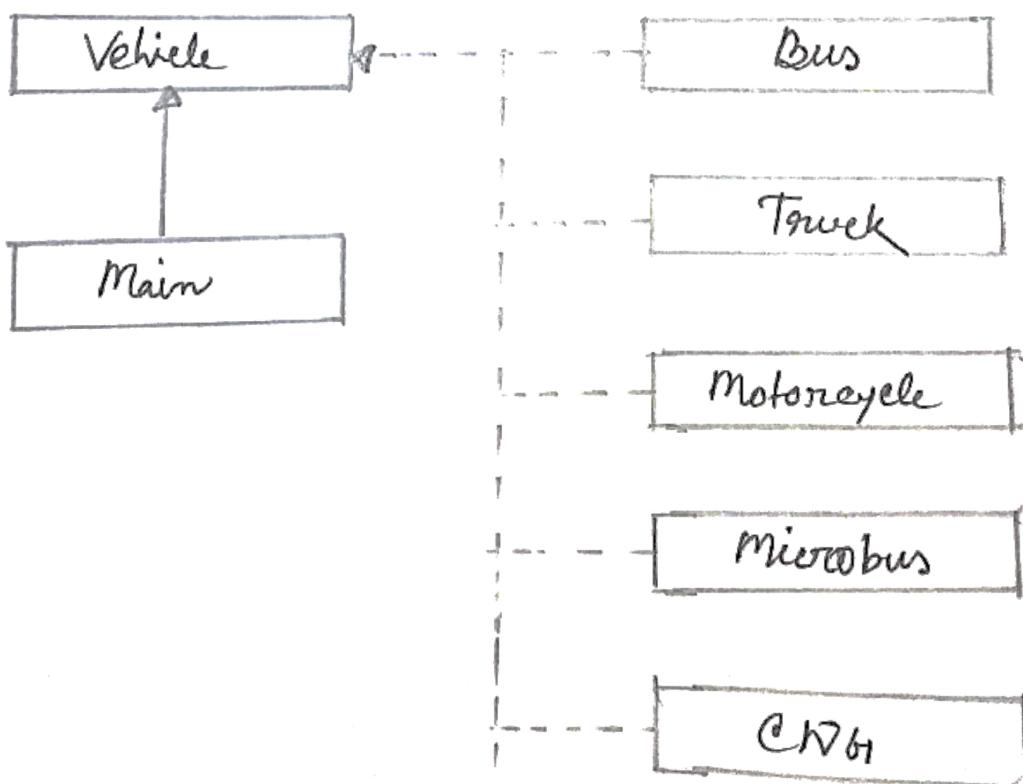


Figure-1: Conceptual class diagram.

Design :-

From above analysis our design description is given below.

* interface Vehicle :

□ Methods :

- changeGear : prints change gear message.
- speedUp : prints speed up message.
- speedDown : prints speed down message.
- pressBreak : prints pressed break message.

* class Bus :

□ Methods :

- change Gear : define it and prints the vehicle name with the action name.
- speed Up : prints the vehicle name with the action name.
- speed Down : prints the vehicle name with the action name.
- press Break : prints vehicle name with the action name.

* Class Truck :

□ Methods :

- * changeGear : prints the vehicle name and the action name.
- * speedUp : prints the vehicle name and the action name.
- * speedDown : prints the vehicle name and the action name.
- * pressBreak : prints the vehicle name and the action name.

* class Motorcycle :

□ Methods :

- * changeGear : prints the vehicle name and the action name.
- * speedUp : prints the vehicle name and the action name.
- * speedDown : prints the vehicle name and the action name.
- * pressBreak : prints the vehicle name and the action name.

* class Microbus :

□ Methods :

- * change Gear : prints the vehicle name and the action name
- * speed Up : prints the vehicle name and the action name
- * speed Down : prints the vehicle name and the action name
- * press Break : prints the vehicle name and the action name

* Class CNG:

Methods:

- * changeGear : prints the vehicle name and the action name.
- * speedUp : prints the vehicle name and the action name.
- * speedDown : prints the vehicle name and the action name.
- * pressBreak : prints the vehicle name and the action name.

* Class Main:

Methods

- * main : the main method of the program from where the program execution will start.

~~From above design description, the pseudocode of the methods are given below.~~

~~Bus :: changeGear () :~~

~~print Bus and changing gear message.~~

~~Bus :: speedUp () :~~

~~prints Bus and speeding up message.~~

~~Bus :: speedDown () :~~

~~print Bus and speeding down message.~~

From above design description the architectural class diagram is given in figure-2:

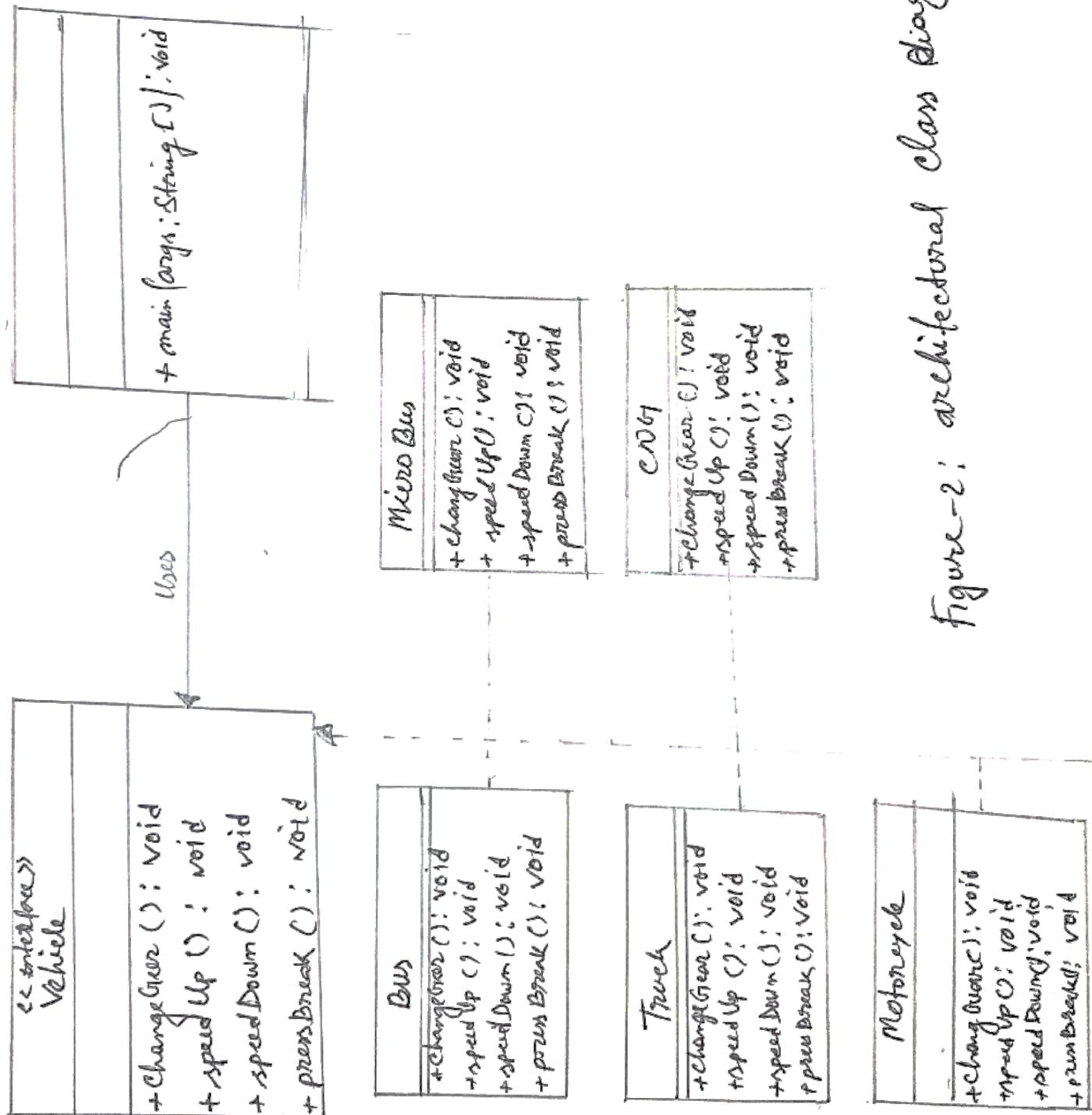


figure-2: architectural class diagram.

From the above description, the pseudocode of the methods are given below:-

Bus :: changeGear();

print Bus and changing gear message.

Bus :: speedUp();

print Bus and speeding up message.

Bus :: speedDown();

print Bus and speeding down message

CNG:: speed up ()

print CNG and speeding UP message.

CNG:: speed Down ()

print and speeding down message.

CNG:: pressBreak ()

print CNG and pressed break message.

Implementation :-

* Implementation is attached with report.

Conclusion :-

We defined an interface vehicle and also defined 5 classes that implements the interface. We also defined the main class that defined the main method.

Experiment: 10

Implementation:

File: Main.java

```
import java.util.Scanner;

interface Vehicle {
    public void changeGear();
    public void speedUp();
    public void speedDown();
    public void pressBreak();
}

class Bus implements Vehicle {
    public void changeGear() {
        System.out.println("Bus");
        System.out.println("Changing gear...");
    }
    public void speedUp() {
        System.out.println("Bus");
        System.out.println("Speeding up...");
    }
    public void speedDown() {
        System.out.println("Bus");
        System.out.println("Speeding down...");
    }
    public void pressBreak() {
        System.out.println("Bus");
        System.out.println("Pressed Break...");
    }
}

class Truck implements Vehicle {
    public void changeGear() {
        System.out.println("Truck");
        System.out.println("Changing gear...");
    }
    public void speedUp() {
        System.out.println("Truck");
        System.out.println("Speeding up...");
    }
    public void speedDown() {
        System.out.println("Truck");
        System.out.println("Speeding down...");
    }
    public void pressBreak() {
        System.out.println("Truck");
        System.out.println("Pressed Break...");
    }
}

class Motorcycle implements Vehicle {
    public void changeGear() {
        System.out.println("Motorcycle");
        System.out.println("Changing gear...");
    }
    public void speedUp() {
        System.out.println("Motorcycle");
        System.out.println("Speeding up...");
    }
    public void speedDown() {
        System.out.println("Motorcycle");
        System.out.println("Speeding down...");
    }
}
```

```
public void pressBreak() {
    System.out.println("Motorcycle");
    System.out.println("Pressed Break...");
}
}

class Microbus implements Vehicle {
    public void changeGear() {
        System.out.println("Microbus");
        System.out.println("Changing gear...");
    }

    public void speedUp() {
        System.out.println("Microbus");
        System.out.println("Speeding up...");
    }

    public void speedDown() {
        System.out.println("Microbus");
        System.out.println("Speeding down...");
    }

    public void pressBreak() {
        System.out.println("Microbus");
        System.out.println("Pressed Break...");
    }
}

class CNG implements Vehicle {
    public void changeGear() {
        System.out.println("CNG");
        System.out.println("Changing gear...");
    }

    public void speedUp() {
        System.out.println("CNG");
        System.out.println("Speeding up...");
    }

    public void speedDown() {
        System.out.println("CNG");
        System.out.println("Speeding down...");
    }

    public void pressBreak() {
        System.out.println("CNG");
        System.out.println("Pressed Break...");
    }
}

class Main {
    public static void main(String[] args) {
        System.out.println("Choose an option:\n1. Bus\n2. Truck\n3. Motorcycle\n4. Microbus");
        Scanner sc = new Scanner(System.in);
        int option = sc.nextInt();
        Vehicle vehicle = new Bus();
        switch (option) {
            case 1:
                vehicle = new Bus();
                break;
            case 2:
                vehicle = new Truck();
                break;
            case 3:
                vehicle = new Motorcycle();
                break;
            case 4:
                vehicle = new Microbus();
                break;
            case 5:
                break;
        }
    }
}
```



```
    vehicle = new CNG();
    break;
default:
    System.out.println("Wrong Option!\n");
    System.exit(-1);
}
```

```
    vehicle.changeGear();
    vehicle.speedUp();
    vehicle.speedDown();
    vehicle.pressBreak();
```

```
    sc.close();
```

```
}
```

```
}
```