ABHISHEK DHAR
23201
E-10

classmate
Date
Page

ASSIGNMENT - 05

Title → Interface in Java

Aim → Design and develop context for given case study and implement interface for vehicles, Consider vehicles, car, bicycle, bike. All vehicle have common functionalities such as gear change, speed up and apply brakes. Make an interface and put comman functionalities. Bicycle, bike, car should be implemented this functionalities, in their own way.

Objective → To understand interface in Java.

Theory →

1) What is interface in Java?

→ It is a reterance in Java which is collection of abstract methods.
→ A class implements an interface, thereby inheriting abstract methods.

2) Syntax to declare interface →

→ Writing an Interface is similar to write a class. Interface contains behaviours that class implements.

→ Syntax →
```
interface interface_name {
  // methods declared in body
}
```

```
class class-name implements interface_name {
  // methods are defined in class
 }
```

→ A class uses implements keyword to implement interface.

3) Example of interface →

```
interface shape {
   public void area ()
 };

class square implements Shape {
   public void area () {
     System.out.println ("Area ="+ (side)**2)
 }

class interface demo {
  public static void main (String [] args) {
   square s = new square ();
    s.area ();
  }
 }
```

4) Relationship b/w classes & interface →

→ A class can inherit an interface using implements keyword.

→ Extends keyword can be used to inherit one interface to another.

$$\begin{tikzcd}
\boxed{\text{class}} & \boxed{\text{interface}} & \text{interface} \\
\boxed{\text{class}} \arrow[u, "\text{extends}"'] & \boxed{\text{class}} \arrow[u, "\text{implements}"'] & \boxed{\text{interface}} \arrow[u, "\text{extends}"']
\end{tikzcd}$$

5) Variables in interface →

→ Variables in interface are static because java interface cannot be instantiated on their own. Value of variable must be assigned in static context in which no instance exists.

→ Final modifier ensures that value assigned to interface variable is a true constant that cannot be re-assigned

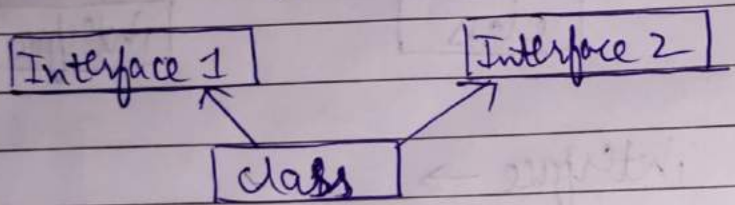
→ Interface can declare only constants

* Advantages of interface →

- Interface in Java support multiple inheritance
- Interface helps to achieve abstraction.
- Used to achieve loose - coupling.

* Multiple inheritance in Java by interface →

→ Classes in java do not support multiple inheritance, but a class can implement more than one interfaces.

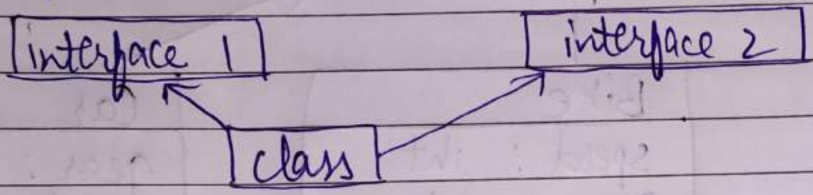


* Example →

```
interface petal {
 public void petals();
}

interface colour {
 public void colours();
}

class flower implements petal, colour {
  public void petals() {
   System.out.println("Flower has petals");
  }
}

class Multiple_demo {
 public void static main (String [] args) {
  flower t = new flower();
  t.petals();
  t.colours();
 }
}
```

- Interface can inherit to one or more interfaces using extends keyword.
- But, interface using extends does not provide implementation.
- Interface also supports multiple inheritance in Java.

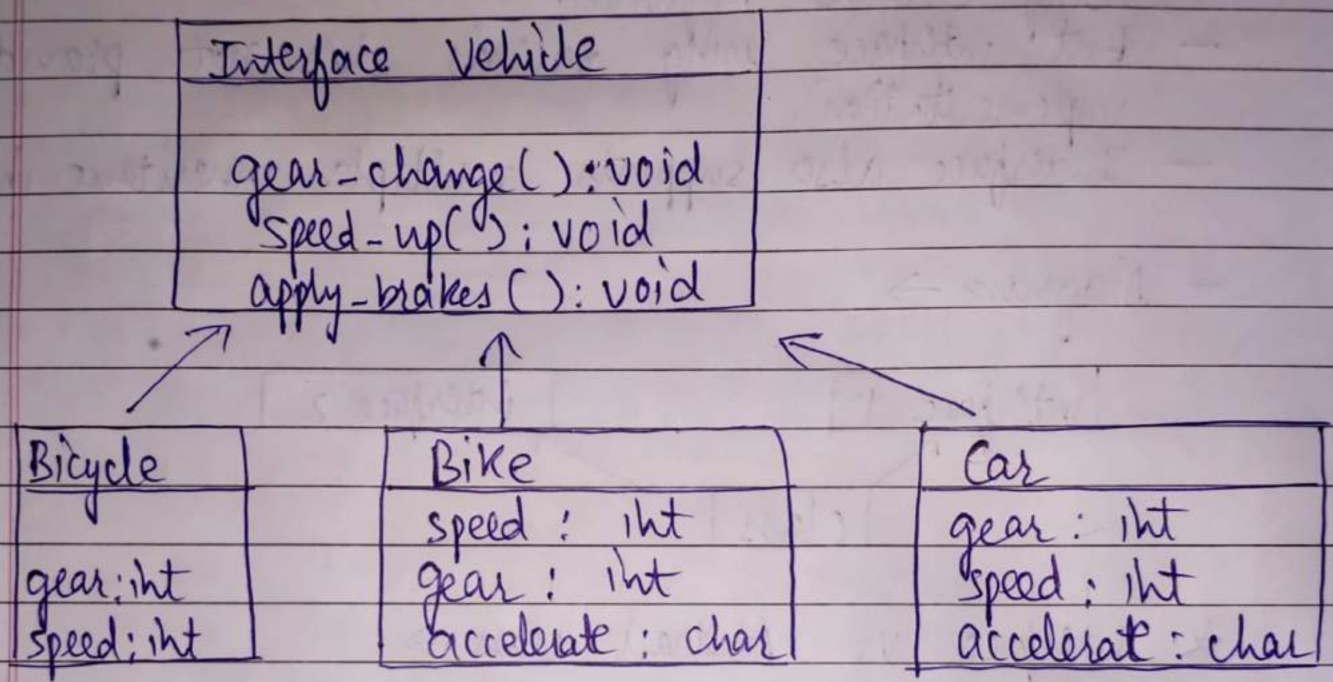
* Diagram →



* Interface vs Abstract class →

Interface	Abstract class
1) Contains method decelaration only not implementation.	1) Contains one or more abstract methods which are implemented by its child classes.
2) It is pure abstract class which starts with interface keyword.	2) It is a class prefixed with abstract keyword followed by class destination
3) contains all abstract methods and final variable declarations	3) It can also contain non-abstract methods
4) Useful in situation where properties should be implemented.	4) Useful where some general methods are used and some methods are implemented in child class.

* Class Diagram->



* Input->

- Choice of gear, whether to speed up and to apply brakes is given as input after giving choice of vehicle.

* Output->

After giving choice another menu ~~k~~ is displayed.

Enter your choice
1) Gear change
2) Speed up
3) Apply brakes

For Bike →

Choice: 1 → (Gear change)
A menu is displayed for change of gear.

Choice: 2 → (speed up)
User is asked to enter 'A' to accelerate bike.

Choice: 3 → (apply brakes)
Brakes are applied and bike stops

Similar outputs are generated for other two vehicles.

* Validations :-

1) Switch case is used for gear-change function for readability.

2) Operating_menu() function is used to reduce duplication in code.

* Test cases →

→ Implemented and pasted in soft copy.