Interface
Assignment

Q1. What is an interface in Java?

Ans. The interface in Java is a mechanism to achieve abstraction. Traditionally, an interface could only have abstract methods (methods without a body) and public, static, and final variables by default. It is used to achieve abstraction and multiple inheritances in Java. In other words, interfaces primarily define methods that other classes must implement. Java Interface also represents the IS-A relationship.
In Java, the abstract keyword applies only to classes and methods, indicating that they cannot be instantiated directly and must be implemented.

When we decide on a type of entity by its behaviour and not via attribute we should define it as an interface.

Example

```
interface Animal {
  public void animalSound();
  public void sleep();
}

class Pig implements Animal {
  public void animalSound() {
    // The body of animalSound() is provided here
    System.out.println("The pig says: wee wee");
  }
  public void sleep() {
        System.out.println("Zzz");
  }
}

class Main {
  public static void main(String[] args) {
    Pig myPig = new Pig();
    myPig.animalSound();
```

```
    myPig.sleep();
  }
}
```

Q2. Which modifiers are allowed for methods in an interface ?Explain?
Ans. In Java, all methods in an interface are *public* even if we do not specify *public* with method names. Also, data fields are *public static final* even if we do not mention it with fields names. Therefore, data fields must be initialized.
Example- :

```
interface Test {
  int x = 10;
   void foo();
}
```

Only public and abstract modifiers are allowed for methods in interfaces.

Q3. What is the use of interfaces in java? Or, Why do we use an interface in java?
Ans. Uses of Interfaces in Java

Uses of Interfaces in Java are mentioned below:

- *It is used to achieve total abstraction.*
- *Since java does not support multiple inheritances in the case of class, by using an interface it can achieve multiple inheritances.*
- *Any class can extend only 1 class, but can any class implement an infinite number of interfaces.*
- *It is also used to achieve loose coupling.*
- *Interfaces are used to implement abstraction.*

Q4. What is the difference between abstract class and interface in java?
Ans.

Points	Abstract Class	Interface
Definition	Cannot be instantiated; contains both abstract (without implementation) and concrete methods (with implementation)	Specifies a set of methods a class must implement; methods are abstract by default.
Implementation Method	Can have both implemented and abstract methods.	Methods are abstract by default; Java 8, can have default and static methods.
Inheritance	class can inherit from only one abstract class.	A class can implement multiple interfaces.
Access Modifiers	Methods and properties can have any access modifier (public, protected, private).	Methods and properties are implicitly public.

Variables	Can have member variables (final, non-final, static, non-static).	Variables are implicitly public, static, and final (constants).

1) To achieve security - hide certain details and only show the important details of an object (interface).

2) Java does not support "multiple inheritance" (a class can only inherit from one superclass). However, it can be achieved with interfaces, because the class can implement multiple interfaces. Note: To implement multiple interfaces, separate them with a comma (see example below).