Q1. Write a program to show Interface Example in java?

Ans: An interface is a fully abstract class. It includes a group of abstract methods (methods without a body).

We use the `interface` keyword to create an interface in Java.

```
interface Polygon {

  void getArea(int length, int breadth);

}

class Rectangle implements Polygon {

  public void getArea(int length, int breadth) {

    System.out.println("The area of the rectangle is " + (length * breadth));

  }

}

class Main {

  public static void main(String[] args) {

    Rectangle r1 = new Rectangle();

    r1.getArea(5, 6);

  }

}
```

Q2.Write a program a Program with 2 concrete method and 2 abstract method in java ?

```java
abstract class AbstractExample {
    abstract void display();
    void show()
    {
        System.out.println("Concrete method of abstract class");
    }
}


class SubClass extends AbstractExample {

    void display()
    {
        System.out.println("Abstract method implemented");
    }
}
public class AbstractClass{

    public static void main(String args[])
    {

        SubClass obj = new SubClass();

        obj.display();

        obj.show();

    }

}
```

Q3.Write a program to show the use of functional interface in java?

```
@FunctionalInterface

interface Square {
    int calculate(int x);
}

class Test {
    public static void main(String args[])
    {
        int a = 5;

        // lambda expression to define the calculate method
        Square s = (int x) -> x * x;

        // parameter passed and return type must be
        // same as defined in the prototype
        int ans = s.calculate(a);
        System.out.println(ans);
    }
}
```

Q4.What is an interface in Java?

Ans: An **Interface in Java** programming language is defined as an abstract type used to specify the behavior of a class. An interface in Java is a blueprint of a behaviour. A Java interface contains static constants and abstract methods.

Q5.What is the use of interface in Java?

Ans:

- t 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 infinite number of interface.
- It is also used to achieve loose coupling.

Q6.What is the lambda expression of Java 8?

Ans: Lambda Expressions implement the only abstract function and therefore implement functional interfaces lambda expressions are added in Java 8 and provide the below functionalities.

- Enable to treat functionality as a method argument, or code as data.
- A function that can be created without belonging to any class.
- A lambda expression can be passed around as if it was an object and executed on demand.

Q7.Can you pass lambda expressions to a method? When?

Ans: A lambda expression passed in a method that has an argument of type of functional interface. If we need to pass a lambda expression as an argument, the type of parameter receiving the lambda expression argument must be of a functional interface type.

Q8.What is the functional interface in Java 8?

Ans: A functional interface is an interface that contains only one abstract method. They can have only one functionality to exhibit. From Java 8 onwards, lambda expressions can be used to represent the instance of a functional interface. A functional interface can have any number of default methods.

Q9.What is the benefit of lambda expressions in Java 8?
Ans: Lambda expressions improve code readability and do not require interpretation. Lambdas allow you to write concise code. It encourages the use of functional programming. It simplifies variable scope and encourages code reusability.

Q10.Is it mandatory for a lambda expression to have parameters?
Ans: No need to declared type of parameter.