Experiment-3

Student Name: Om Mishra   UID:22BCS16609
Branch: BE-CSE   **Section/Group: 901/A**
Semester: 6th   **Date of Performance: 25.1.25**
Subject Name: Project Based Learning in Java   **Subject Code: 22CSH-359**

1. **Aim:** Create an application to calculate interest for FDs, RDs based on certain conditions using inheritence

2. **Objective:** To design and implement a Java program that calculates interest for various account types (FD, RD, SB) using object-oriented principles, focusing on abstraction, method overriding, and dynamic input validation.

3. **Implementation/Code:**

```java
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        try {
            System.out.println("Choose account type: 1. FD  2. RD");
            int choice = sc.nextInt();

            if (choice == 1) {
                System.out.print("Enter principal amount: ");
                double principal = sc.nextDouble();
                System.out.print("Enter interest rate: ");
                double rate = sc.nextDouble();
                System.out.print("Enter time in years: ");
                int time = sc.nextInt();

                FDAccount fd = new FDAccount(principal, rate, time);
                System.out.println("FD Interest: " + fd.calculateInterest());

            } else if (choice == 2) {
                System.out.print("Enter monthly deposit: ");
                double monthlyDeposit = sc.nextDouble();
                System.out.print("Enter interest rate: ");
                double rate = sc.nextDouble();
                System.out.print("Enter duration in months: ");
                int months = sc.nextInt();

                RDAccount rd = new RDAccount(monthlyDeposit, rate, months);
                System.out.println("RD Interest: " + rd.calculateInterest());
```

```java
            } else {
                System.out.println("Invalid choice.");
            }

        } catch (Exception e) {
            System.out.println("Error occurred: " + e.getMessage());
        } finally {
            System.out.println("Interest calculation complete.");
            sc.close();
        }
    }
}

abstract class Account {
    protected double principal;
    protected double rate;
    protected int time;

    public abstract double calculateInterest();
}
class FDAccount extends Account {
    public FDAccount(double principal, double rate, int time) {
        if (principal <= 0 || rate <= 0 || time <= 0) {
            throw new IllegalArgumentException("FD values must be greater than 0.");
        }
        this.principal = principal;
        this.rate = rate;
        this.time = time;
    }

    @Override
    public double calculateInterest() {
        return (principal * rate * time) / 100;
    }
}
class RDAccount extends Account {
    private int months;

    public RDAccount(double monthlyDeposit, double rate, int months) {
        if (monthlyDeposit <= 0 || rate <= 0 || months <= 0) {
            throw new IllegalArgumentException("RD values must be greater than 0.");
        }
        this.principal = monthlyDeposit;
        this.rate = rate;
        this.months = months;
    }
```

```
    @Override
    public double calculateInterest() {

        return (principal * months * (months + 1) * rate) / (2 * 12 * 100);
    }
}
```

4. Output:

```
Choose account type: 1. FD   2. RD
1
Enter principal amount: 100
Enter interest rate: 10
Enter time in years: 10
FD Interest: 100.0
Interest calculation complete.
```

5. Learning outcomes:

1. Understand the concept of abstract classes and method overriding in Java.
2. Learn to implement real-world scenarios using object-oriented principles.
3. Develop skills to validate user input for different account types.
4. Gain knowledge of calculating interest dynamically based on conditions.
5. Enhance problem-solving abilities by applying conditional logic effectively.