

## Experiment 3

**Student Name:** Lavanya

**Branch:** CSE

**Semester:** 6<sup>th</sup>

**Subject:** Java Lab

**UID:** 22BCS50012

**Section/Group:** 635/B

**DOP:** 17/1/2025

**Subject Code:** 22CSH-359

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

**Objective:** To develop a Java application that calculates interest for Fixed Deposits (FDs) and Recurring Deposits (RDs) using object-oriented programming principles. The application will use inheritance to define common properties and methods for accounts while providing specific implementations for FDs and RDs based on their respective conditions.

### **Algorithm:**

- **Create Account class** with attributes: `accountHolderName`, `principal`, `rateOfInterest`. Include methods for calculating interest (to be overridden) and displaying details.
- **Create FixedDeposit subclass** that calculates FD interest using: `principal * rateOfInterest * tenureInYears / 100`. Display FD details.
- **Create RecurringDeposit subclass** that calculates RD interest using: `(monthlyDeposit * months * (months + 1) / 2) * (rateOfInterest / (12 * 100))`. Display RD details.
- **In main method**, create instances of `FixedDeposit` and `RecurringDeposit` and display their details.

### **Code:**

```
class Account {  
  
    String accountHolderName;  
  
    double principal;  
  
    double rateOfInterest;  
  
    public Account(String accountHolderName, double principal, double rateOfInterest) {  
  
        this.accountHolderName = accountHolderName;  
  
        this.principal = principal;  
  
        this.rateOfInterest = rateOfInterest;}  
  
    public double calculateInterest() {  
  
        return 0;  
  
    }  
}
```

```
        public void displayDetails() {

            System.out.println("Account Holder: " + accountHolderName);

            System.out.println("Principal Amount: " + principal);

            System.out.println("Rate of Interest: " + rateOfInterest + "%");

        }

    }

    class FixedDeposit extends Account {

        int tenureInYears;

        public FixedDeposit(String accountHolderName, double principal, double rateOfInterest, int
            tenureInYears) {

            super(accountHolderName, principal, rateOfInterest);

            this.tenureInYears = tenureInYears;

        }

        public double calculateInterest() {

            return principal * rateOfInterest * tenureInYears / 100;

        }

        public void displayDetails() {

            super.displayDetails();

            System.out.println("Tenure (Years): " + tenureInYears);

            System.out.println("Interest Amount: " + calculateInterest());}}

    class RecurringDeposit extends Account {

        int months;

        double monthlyDeposit;

        public RecurringDeposit(String accountHolderName, double monthlyDeposit, double rateOfInterest,
            int months) {
```

```
super(accountHolderName, 0, rateOfInterest);

this.monthlyDeposit = monthlyDeposit;

this.months = months;
}

public double calculateInterest() {

    // RD interest formula:  $P(n(n+1)/2) * (r / 12 * 100)$ 

    double n = months;

    return (monthlyDeposit * n * (n + 1) / 2) * (rateOfInterest / (12 * 100));
}

public void displayDetails() {

    System.out.println("Account Holder: " + accountHolderName);

    System.out.println("Monthly Deposit: " + monthlyDeposit);

    System.out.println("Number of Months: " + months);

    System.out.println("Rate of Interest: " + rateOfInterest + "%");

    System.out.println("Interest Amount: " + calculateInterest());
}
}

public class InterestCalculator {

    public static void main(String[] args) {

        // Example FD account

        FixedDeposit fd = new FixedDeposit("Lavanya", 100000, 5.5,

        3); System.out.println("Fixed Deposit Details:");

        fd.displayDetails();

        System.out.println();
    }
}
```

```
RecurringDeposit rd = new RecurringDeposit("Lavanya_22BCS50012", 5000, 6.5, 12);  
  
System.out.println("Recurring Deposit Details:");  
  
rd.displayDetails();  
  
}  
  
}
```

## Output:

```
Fixed Deposit Details:  
Account Holder: Lavanya  
Principal Amount: 10000.0  
Rate of Interest: 5.0%  
Tenure (Years): 3  
Interest Amount: 1500.0  
  
Recurring Deposit Details:  
Account Holder: Lavanya_22BCS50012  
Monthly Deposit: 5000.0  
Number of Months: 12  
Rate of Interest: 6.5%  
Interest Amount: 2112.5  
  
Process finished with exit code 0
```



# DEPARTMENT OF COMPUTERSCIENCE & ENGINEERING

Discover. Learn. Empower.

## Learning Outcomes:

- **Inheritance:** Use of base and derived classes for shared attributes and methods.
- **Method Overriding:** Custom implementation of methods in subclasses.
- **Constructor:** Initializing object attributes using constructors.
- **Encapsulation:** Storing and manipulating data within objects.
- **Polymorphism:** Different behavior of `calculateInterest()` based on object type.
- **Interest Calculation:** Implementing FD and RD interest formulas.
- **Class Interaction:** Creating objects and calling methods to display details



# **DEPARTMENT OF COMPUTERSCIENCE & ENGINEERING**

---

Discover. Learn. Empower.