Experiment 3

Student Name: Harshad Fozdar
UID: 22BCS10263
Branch: BE-CSE
Section/Group: 22BCS_DL-901
Semester: 6
Date of Performance: 23/01/2025
Subject Name: Project Based Learning in Java with Lab
Subject Code: 22CSH-359

1. **Aim:** Design an Interest Calculator to calculate interest based on the type of the account and the status of the account holder.

2. **Objective:** Calculate interest based on the type of the account and the status of the account holder. The rates of interest changes according to the amount (greater than or less than 1 crore), age of account holder (General or Senior citizen) and number of days if the type of account is FD or RD.

3. **Implementation/Code:**

```
import java.util.Scanner;
abstract class Account {
   double interestRate;
   double amount;
   abstract double calculateInterest();
}
class FDAccount extends Account
   { int noOfDays, ageOfACHolder;
   FDAccount(double amount, int noOfDays, int ageOfACHolder)
      { this.amount = amount;
      this.noOfDays = noOfDays;
      this.ageOfACHolder = ageOfACHolder;
   }
   double calculateInterest()
      { if (noOfDays < 0) {
         System.out.println("Invalid input:
         Time"); return 0;
      }
```

```
        if (amount < 0) {
            System.out.println("Invalid input: Amount");
            return 0;
        }
        if (ageOfACHolder < 0)
            { System.out.println("Invalid input:
            Age"); return 0;
        }
        if (amount >= 10000000) {
            if (noOfDays >= 7 && noOfDays <= 14) interestRate = 6.50;
            else if (noOfDays >= 15 && noOfDays <= 44) interestRate = 6.75;
            else if (noOfDays >= 45 && noOfDays <= 60) interestRate = 8.00;
            else if (noOfDays >= 61 && noOfDays <= 184) interestRate = 8.50;
            else if (noOfDays >= 185 && noOfDays <= 365) interestRate =
            10.00;
        }
        else {
            if (ageOfACHolder >= 60) {
                if (noOfDays >= 7 && noOfDays <= 14) interestRate = 5.00;
                else if (noOfDays >= 15 && noOfDays <= 29) interestRate = 5.25;
                else if (noOfDays >= 30 && noOfDays <= 45) interestRate = 6.00;
                else if (noOfDays >= 45 && noOfDays <= 60) interestRate = 7.50;
                else if (noOfDays >= 61 && noOfDays <= 184) interestRate = 8.00;
                else if (noOfDays >= 185 && noOfDays <= 365) interestRate =
                8.50;
            }
            else {
                if (noOfDays >= 7 && noOfDays <= 14) interestRate = 4.50;
                else if (noOfDays >= 15 && noOfDays <= 29) interestRate = 4.75;
                else if (noOfDays >= 30 && noOfDays <= 45) interestRate = 5.50;
                else if (noOfDays >= 45 && noOfDays <= 60) interestRate = 7.00;
                else if (noOfDays >= 61 && noOfDays <= 184) interestRate = 7.50;
                else if (noOfDays >= 185 && noOfDays <= 365) interestRate =
                8.00;
            }
        }
        return (amount * interestRate / 100);
    }
}
```

```
class SBAccount extends Account
   { String accountType;
   SBAccount(double amount, String accountType)
      { this.amount = amount;
      this.accountType = accountType;
   }
   double calculateInterest()
      { if (amount < 0) {
         System.out.println("Invalid input: Amount");
         return 0;
      }
      interestRate = accountType.equalsIgnoreCase("Normal") ? 4.00 : 6.00;
      return (amount * interestRate / 100);
   }
}

class RDAccount extends Account
   { double monthlyAmount;
   int noOfMonths;
   int
   ageOfACHolder;
   RDAccount(double amount, double monthlyAmount, int noOfMonths, int ageOfACHolder)
{
      this.amount = amount;
      this.monthlyAmount = monthlyAmount;
      this.noOfMonths = noOfMonths;
      this.ageOfACHolder = ageOfACHolder;
   }
   double calculateInterest()
      { if (noOfMonths < 0) {
         System.out.println("Invalid input:
         Time"); return 0;
      }
      if (amount < 0) {
         System.out.println("Invalid input: Amount");
         return 0;
      }
```

```
        if (monthlyAmount < 0) {
           System.out.println("Invalid input: Monthly Amount");
           return 0;
        }
        if (ageOfACHolder >= 60) {
           if (noOfMonths == 6) interestRate = 8.00;
           else if (noOfMonths == 9) interestRate = 8.25;
           else if (noOfMonths == 12) interestRate = 8.50;
           else if (noOfMonths == 15) interestRate = 8.75;
           else if (noOfMonths == 18) interestRate = 9.00;
           else if (noOfMonths == 21) interestRate = 9.25;
        }
        else {
           if (noOfMonths == 6) interestRate = 7.50;
           else if (noOfMonths == 9) interestRate = 7.75;
           else if (noOfMonths == 12) interestRate = 8.00;
           else if (noOfMonths == 15) interestRate = 8.25;
           else if (noOfMonths == 18) interestRate = 8.50;
           else if (noOfMonths == 21) interestRate = 8.75;
        }
        return (amount * interestRate / 100);
     }
}

public class IntCalc {
   public static void main(String[] args)
      { Scanner input = new Scanner(System.in);
      int choice = 0;
      while (choice != 4)
         { System.out.println("22BCS13216");
         System.out.println("Select an account type: \n1. SB\n2. FD\n3. RD\n4.
         Exit"); choice = input.nextInt();
         switch (choice)
            { case 1:
               System.out.println("Enter the Average amount in your account:");
               double sbAmount = input.nextDouble();
```

```
                System.out.println("Enter Account Type (Normal/NRI):");
                String type = input.next();
                SBAccount sb = new SBAccount(sbAmount, type);
                System.out.println("Interest: " + sb.calculateInterest());
                break;
            case 2:
                System.out.println("Enter the FD amount:");
                double fdAmount = input.nextDouble();
                System.out.println("Enter the number of days:");
                int days = input.nextInt();
                System.out.println("Enter age:");
                int age = input.nextInt();
                FDAccount fd = new FDAccount(fdAmount, days,
                age);          System.out.println("Interest:          "          +
                fd.calculateInterest()); break;
            case 3:
                System.out.println("Enter the Monthly deposit amount:");
                double rdAmount = input.nextDouble();
                System.out.println("Enter the number of months:");
                int months = input.nextInt();
                System.out.println("Enter age:");
                int rdAge = input.nextInt();
                RDAccount rd = new RDAccount(rdAmount, rdAmount, months, rdAge);
                System.out.println("Interest gained is: " + rd.calculateInterest());
                break;
            case 4:
                System.out.println("Exit");
                break;
            default:
                System.out.println("Invalid  option");
        }
    }
  }
}
```

4. Output

```
PS D:\22bcs13216\6\java\code> java IntCalc
22BCS13216
Select an account type:
1. SB
2. FD
3. RD
4. Exit
1
Enter the Average amount in your account:
10000
Enter Account Type (Normal/NRI):
normal
Interest: 400.0
22BCS13216
Select an account type:
1. SB
2. FD
3. RD
4. Exit
2
Enter the FD amount:
100000
Enter the number of days:
100
Enter age:
69
Interest: 8000.0
```

5. Learning Outcome:

- Object-Oriented Programming concepts like Abstraction, Polymorphism & Inheritance
- Constructor, Control Flow & Conditional Statements