Experiment 3

Student Name: AMAN RAJ **UID: 22BCS12582**
Branch: BE-CSE **Section/Group: 901_A**
Semester:6th **Date of Performance: 25/1/25**
Subject Name: Project Based Learning in Java with Lab **Subject Code: 22CSH-359**

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

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:**

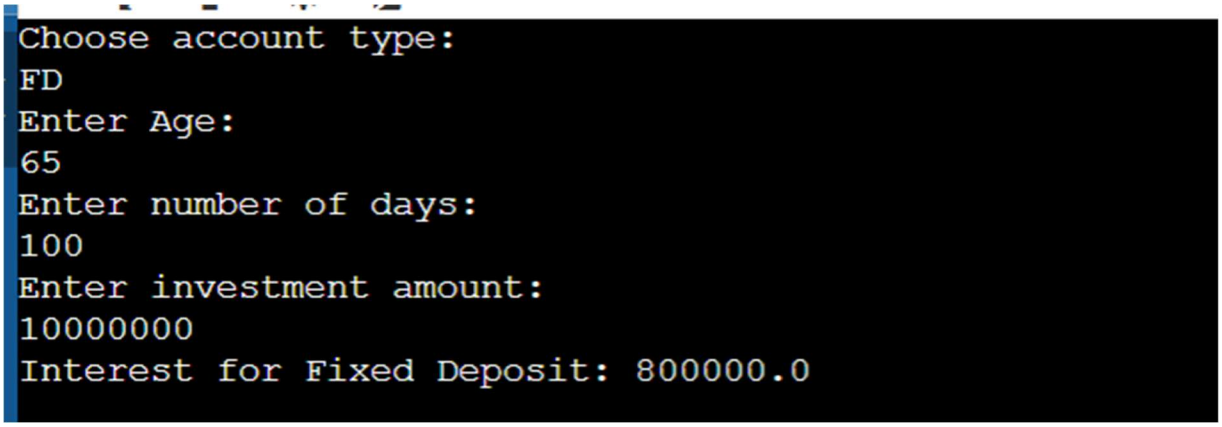
```
import java.util.*;

public class InterestCal {
   static double interest;
   static float interestAmount;

   public static float FD(String age, int days, int investment) {
      if (investment < 1000000) {
         if (age.equals("Senior Citizen")) {
            if (days >= 7 && days <= 14) {
               interest = (5 / 100.0) * investment;
            } else if (days >= 15 && days <= 29) {
               interest = (5.25 / 100.0) * investment;
            } else if (days >= 30 && days <= 45) {
               interest = (6 / 100.0) * investment;
            } else if (days > 45 && days <= 60) {
               interest = (7.5 / 100.0) * investment;
            } else if (days >= 61 && days <= 184) {
               interest = (8 / 100.0) * investment;
            } else if (days > 184 && days <= 365) {
               interest = (8.5 / 100.0) * investment;
            }
         } else if (age.equals("General")) {
            if (days >= 7 && days <= 14) {
               interest = (4.5 / 100.0) * investment;
            } else if (days >= 15 && days <= 29) {
               interest = (4.75 / 100.0) * investment;
            } else if (days >= 30 && days <= 45) {
```

```
                interest = (5.5 / 100.0) * investment;
            } else if (days > 45 && days <= 60) {
                interest = (7 / 100.0) * investment;
            } else if (days >= 61 && days <= 184) {
                interest = (7.5 / 100.0) * investment;
            } else if (days > 184 && days <= 365) {
                interest = (8 / 100.0) * investment;
            }
        }
    } else if (investment > 1000000) {
        if (days >= 7 && days <= 14) {
            interest = (5 / 100.0) * investment;
        } else if (days >= 15 && days <= 29) {
            interest = (5.25 / 100.0) * investment;
        } else if (days >= 30 && days <= 45) {
            interest = (6 / 100.0) * investment;
        } else if (days > 45 && days <= 60) {
            interest = (7.5 / 100.0) * investment;
        } else if (days >= 61 && days <= 184) {
            interest = (8 / 100.0) * investment;
        } else if (days > 184 && days <= 365) {
            interest = (8.5 / 100.0) * investment;
        }
    }
    return (float) interest;
}

public static float RD(String age, int month, int investment) {
    if (age.equals("Senior Citizen")) {
        if (month == 6) {
            interest = (8 / 100.0) * investment;
        } else if (month == 9) {
            interest = (8.25 / 100.0) * investment;
        } else if (month == 12) {
            interest = (8.5 / 100.0) * investment;
        } else if (month == 15) {
            interest = (7.5 / 100.0) * investment;
        } else if (month == 18) {
            interest = (8 / 100.0) * investment;
        } else if (month == 21) {
            interest = (8.5 / 100.0) * investment;
        }
    } else if (age.equals("General")) {
        if (month == 6) {
            interest = (7.5 / 100.0) * investment;
        } else if (month == 9) {
            interest = (7.75 / 100.0) * investment;
        } else if (month == 12) {
            interest = (8 / 100.0) * investment;
```

```
            } else if (month == 15) {
                interest = (8.25 / 100.0) * investment;
            } else if (month == 18) {
                interest = (8.5 / 100.0) * investment;
            } else if (month == 21) {
                interest = (8.75 / 100.0) * investment;
            }
        }
        return (float) interest;
    }

    public static float SB(String accountType, int investment) {
        if (accountType.equals("Normal")) {
            interest = (4 / 100.0) * investment;
        } else if (accountType.equals("NRI")) {
            interest = (6 / 100.0) * investment;
        }
        return (float) interest;
    }

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

        System.out.println("Choose account type: ");
        String accountType = sc.next();

        if (accountType.equalsIgnoreCase("FD")) {
            System.out.println("Enter Age: ");
            String age = sc.next();
            System.out.println("Enter number of days: ");
            int days = sc.nextInt();
            System.out.println("Enter investment amount: ");
            int investment = sc.nextInt();
            float interest = FD(age, days, investment);
            System.out.println("Interest for Fixed Deposit: " + interest);
        } else if (accountType.equalsIgnoreCase("RD")) {
            System.out.println("Enter Age : ");
            String age = sc.next();
            System.out.println("Enter number of months: ");
            int months = sc.nextInt();
            System.out.println("Enter investment amount: ");
            int investment = sc.nextInt();
            float interest = RD(age, months, investment);
            System.out.println("Interest for Recurring Deposit: " + interest);
        } else if (accountType.equalsIgnoreCase("SB")) {
            System.out.println("Enter Account type : ");
            String accType = sc.next();
            System.out.println("Enter investment amount: ");
            int investment = sc.nextInt();
```

```
            float interest = SB(accType, investment);
            System.out.println("Interest for Savings Bank: " + interest);
        } else {
            System.out.println("Invalid account type");
        }
        sc.close();
    }
     }
```

4. Output:



5. Learning Outcomes:

- **Understand object-oriented programming concepts such as classes, objects, attributes, and methods.**
- **Learn to implement and manipulate arrays in Java.**
- **Gain experience in designing and managing an inventory system.**
- **Practice developing and using methods for object interactions.**
- **Improve debugging and testing skills through a structured main program.**
- **Enhance problem-solving abilities by implementing business logic for a rental system.**