Experiment 3

Student Name: Anjali Singh **UID:20BCS9239**
Branch: CSE **Section/Group:607/A**
Semester: 5th **Date of Performance:06/09/2022**
Subject Name: PBLJ Lab **Subject Code: 20CSP-321**

1. **Aim/Overview of the practical:** Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.

2. **Task to be done/ Which logistics used:**

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.

Some sample rates are given in the below tables:

Rate of FD interest for amounts below 1 Crore:

	Current Rates of interest	
Maturity Period	General	Senior Citizen
7 days to 14 days	4.50	5.00
15 days to 29 days	4.75	5.25
30 days to 45 days	5.50	6.00

45 days to 60 days	7	7.50
61 days to 184 days	7.50	8.00
185 days to 1 year	8.00	8.50

Rate of FD interest for amounts above 1 Crore:

Maturity Period	Interest Rate
7 days to 14 days	6.50
15 days to 29 days	6.75
30 days to 45 days	6.75
45 days to 60 days	8
61 days to 184 days	8.50
185 days to 1 year	10.00

Rate of RD interests:

	Current Rates of interest	
Maturity Period	General	Senior Citizen
6 months	7.50	8.00
9 months	7.75	8.25

12 months	8.00	8.50
15 months	8.25	8.75
18 months	8.50	9.00
21 months	8.75	9.25

SB Account interest rates:

Type of Account	Interest Rate
Normal	4%
NRI	6%

3. Algorithm/Flowchart (For programming based labs):

1. Make Account Class.
2. Using Method Overriding Create Interest Calculate.
3. Create FD, RD & SD.
4. Take input of amount and age and days for FD.
5. Take input of saving account
6. For RD take amount and month and age as input.
7. Create a Launcher class.

4. Steps for experiment/practical/Code:

```
package com.company.CWH;

import java.util.Scanner;
abstract class Account
{
    double interestRate, amount;
    abstract double calculateInterest();
}
```

```
class MyException extends Exception
{
    public MyException()
    {
        System.out.println("Invalid input!!");
        System.exit(0);
    }
}
class FDAccount extends Account
{
    Scanner sc = new Scanner(System.in);
    double interestRate, amount;
    int days, age=0;

    void input()
    {
        try
        {
            System.out.println("Enter amount (in Rs): ");
            amount=sc.nextInt();
            if(amount<0)
                throw new MyException();
            System.out.println("Enter maturity period (in days): ");
            days=sc.nextInt();
            if(days<0)
                throw new MyException();
            if(amount<10000000)
            {
                System.out.println("Enter age (in years): ");
                age=sc.nextInt();
                if(age<0)
                    throw new MyException();
            }
        }
        catch(MyException ex)
        {
        }
    }

    double calculateInterest()
    {
        double interest=0;
        if(age!=0)
        {
            if (age>60)
            {
                if(days>=7 && days<=14)
                    interestRate=5.00;
                else if (days>=15 && days<=29)
                    interestRate=5.25;
                else if (days>=30 && days<=45)
                    interestRate=6.00;
                else if (days>=46 && days<=60)
                    interestRate=7.50;
                else if (days>=61 && days<=184)
                    interestRate=8.00;
                else if (days>=185 && days<=365)
                    interestRate=8.50;
                else
```

```
                {
                    interestRate=0;
                    System.out.println("Invalid maturity period");
                }
            }
            else
            {
                if(days>=7 && days<=14)
                    interestRate=4.50;
                else if (days>=15 && days<=29)
                    interestRate=4.75;
                else if (days>=30 && days<=45)
                    interestRate=5.50;
                else if (days>=46 && days<=60)
                    interestRate=7.00;
                else if (days>=61 && days<=184)
                    interestRate=7.50;
                else if (days>=185 && days<=365)
                    interestRate=8.00;
                else
                {
                    interestRate=0;
                    System.out.println("Invalid maturity period");
                }
            }
        }
        else //amount>crore
        {
            if(days>=7 && days<=14)
                interestRate=6.50;
            else if (days>=15 && days<=29)
                interestRate=6.75;
            else if (days>=30 && days<=45)
                interestRate=6.75;
            else if (days>=46 && days<=60)
                interestRate=8.00;
            else if (days>=61 && days<=184)
                interestRate=8.50;
            else if (days>=185 && days<=365)
                interestRate=10.00;
            else
            {
                interestRate=0;
                System.out.println("Invalid maturity period");
            }
        }
        interest=(interestRate*amount)/100;
        return interest;
    }
}
class RDAccount extends Account
{
    Scanner sc = new Scanner(System.in);
    double interestRate, amount;
    int months, age=0;

    void input()
    {
        try
```

```
        {
            System.out.println("Enter amount (in Rs): ");
            amount=sc.nextInt();
            if(amount<0)
                throw new MyException();
            System.out.println("Enter maturity period (in months): ");
            months=sc.nextInt();
            if(months<0)
                throw new MyException();
            System.out.println("Enter age (in years): ");
            age=sc.nextInt();
            if(age<0)
                throw new MyException();
        }
        catch(MyException ex)
        {
        }
    }

    double calculateInterest()
    {
        double interest=0;
        if (age>60)
        {
            if(months==6)
                interestRate=8.00;
            else if (months ==9)
                interestRate=8.25;
            else if (months==12)
                interestRate=8.50;
            else if (months==15)
                interestRate=8.75;
            else if (months==18)
                interestRate=9.00;
            else if (months==21)
                interestRate=9.25;
            else
            {
                interestRate=0;
                System.out.println("Invalid maturity period");
            }
        }
        else
        {
            if(months==6)
                interestRate=7.50;
            else if (months ==9)
                interestRate=7.75;
            else if (months==12)
                interestRate=8.00;
            else if (months==15)
                interestRate=8.25;
            else if (months==18)
                interestRate=8.50;
            else if (months==21)
                interestRate=8.75;
            else
            {
                interestRate=0;
```

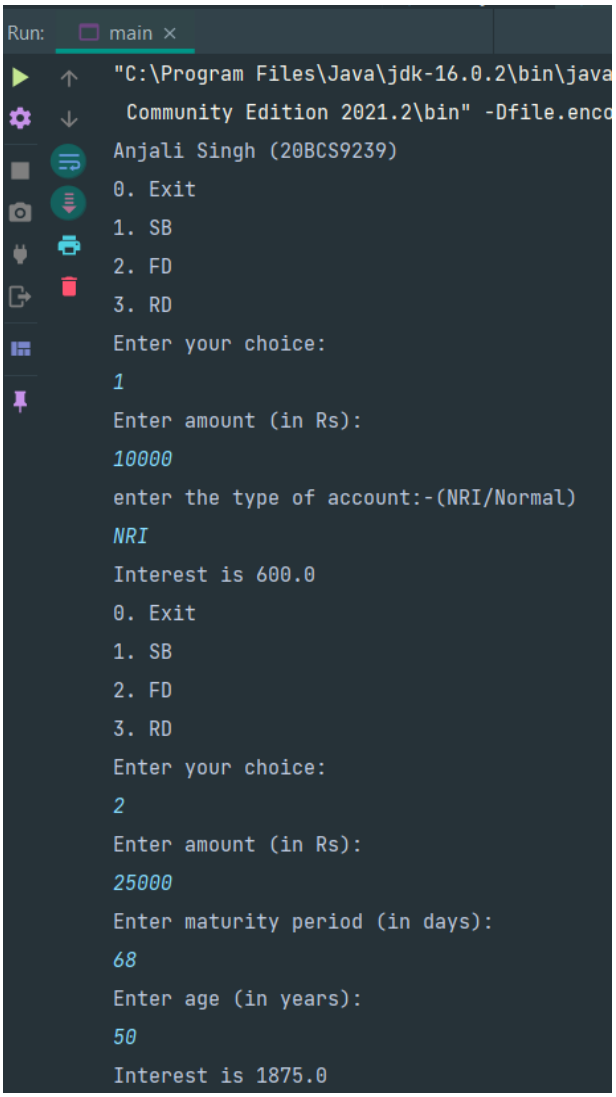
```
                System.out.println("Invalid maturity period");
            }
        }
        interest=(interestRate*amount)/100;
        return interest;
    }

}
class SBAccount extends Account
{
    String type;
    void input()
    {
        try
        {
            Scanner sc=new Scanner(System.in);
            System.out.println("Enter amount (in Rs): ");
            amount=sc.nextInt();
            if(amount<0)
                throw new MyException();
            System.out.println("enter the type of account:-(NRI/Normal) ");
            type=sc.next();
        }
        catch(MyException ex)
        {
        }
    }
    double calculateInterest()
    {
        if(type.equalsIgnoreCase("NRI"))
            interestRate =6.0;
        else
            interestRate =4.0;
        double interest = (interestRate*amount)/100;
        return interest;
    }
}



public class main
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Anjali Singh (20BCS9239) ");
        int c;
        double result;
        while(true)
        {
            System.out.println("0. Exit");
            System.out.println("1. SB");
            System.out.println("2. FD");
            System.out.println("3. RD");
            System.out.println("Enter your choice: ");
            c=sc.nextInt();
            switch(c)
            {
                case 0:
```

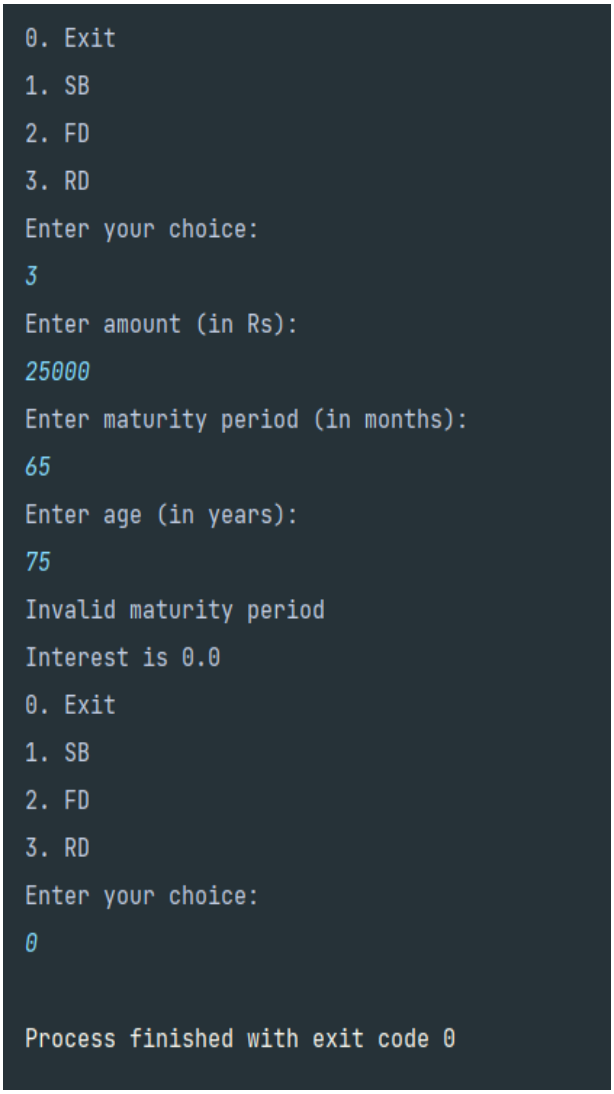
```
                    System.exit(0);
                case 1:
                    SBAccount sb = new SBAccount();
                    sb.input();
                    result=sb.calculateInterest();
                    System.out.println("Interest is "+result);
                    break;
                case 2:
                    FDAccount fd = new FDAccount();
                    fd.input();
                    result=fd.calculateInterest();
                    System.out.println("Interest is "+result);
                    break;
                case 3:
                    RDAccount rd = new RDAccount();
                    rd.input();
                    result=rd.calculateInterest();
                    System.out.println("Interest is "+result);
                    break;
            }
        }
    }

}
```

4. Observations/Discussions/ Complexity Analysis:

In this experiment we are creating an interest calculator which is going to calculate the interest for a given amount by the user according to account type, maturity period and age of the user.

5. Result/Output/Writing Summary:





Learning outcomes (What I have learnt):

1. I learn how to use intellij idea for executing java programs.
2. I learn basics related to java program implementation.
3. I learn to store information using array in java.
4. I learn about the concept about inheritance in java.
5. I learn how to use methods and how to call methods in java.