DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING

CU
CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

PROJECT 2

Student Name: Devesh Rajpoot **UID: 23BCS13411**
Branch: BE CSE
Semester: 5th sem
Subject Name:JAVA

1.Project Title: Interest Calculator

2.Java Code:

InvalidInputException.java

```
public class InvalidInputException extends Exception {
    public InvalidInputException(String message) {
        super(message);
    }
}
```

Account.java

```
public abstract class Account {
    protected double interestRate;
    protected double amount;

    public Account(double amount) throws InvalidInputException {
                if (amount < 0) throw new InvalidInputException("Amount cannot be
negative.");
        this.amount = amount;
    }

    public abstract double calculateInterest() throws InvalidInputException;
}
```

SBAccount.java

```
public class SBAccount extends Account {
    private String accountType;

        public SBAccount(double amount, String accountType) throws
InvalidInputException {
        super(amount);
        this.accountType = accountType;
    }

    @Override
    public double calculateInterest() {
        interestRate = accountType.equalsIgnoreCase("NRI") ? 6.0 : 4.0;
        return (amount * interestRate) / 100;
    }
}
```

FDAccount.java

```
public class FDAccount extends Account {
    private int noOfDays;
    private int age;

        public FDAccount(double amount, int noOfDays, int age) throws
InvalidInputException {
        super(amount);
        if (noOfDays <= 0 || age <= 0) throw new InvalidInputException("Days or age
cannot be zero or negative.");
        this.noOfDays = noOfDays;
        this.age = age;
    }

    @Override
    public double calculateInterest() {
        if (amount < 10000000) {
            if (noOfDays >= 7 && noOfDays <= 14)
                interestRate = age >= 60 ? 5.0 : 4.5;
            else if (noOfDays <= 29)
                interestRate = age >= 60 ? 5.25 : 4.75;
            else if (noOfDays <= 45)
                interestRate = age >= 60 ? 6.0 : 5.5;
            else if (noOfDays <= 60)
                interestRate = age >= 60 ? 7.5 : 7.0;
            else if (noOfDays <= 184)
                interestRate = age >= 60 ? 8.0 : 7.5;
            else
                interestRate = age >= 60 ? 8.5 : 8.0;
        } else {
            if (noOfDays >= 7 && noOfDays <= 14)
                interestRate = 6.5;
            else if (noOfDays <= 29)
                interestRate = 6.75;
            else if (noOfDays <= 45)
                interestRate = 6.75;
            else if (noOfDays <= 60)
                interestRate = 8.0;
            else if (noOfDays <= 184)
                interestRate = 8.5;
            else
                interestRate = 10.0;
        }
        return (amount * interestRate) / 100;
    }
}
```

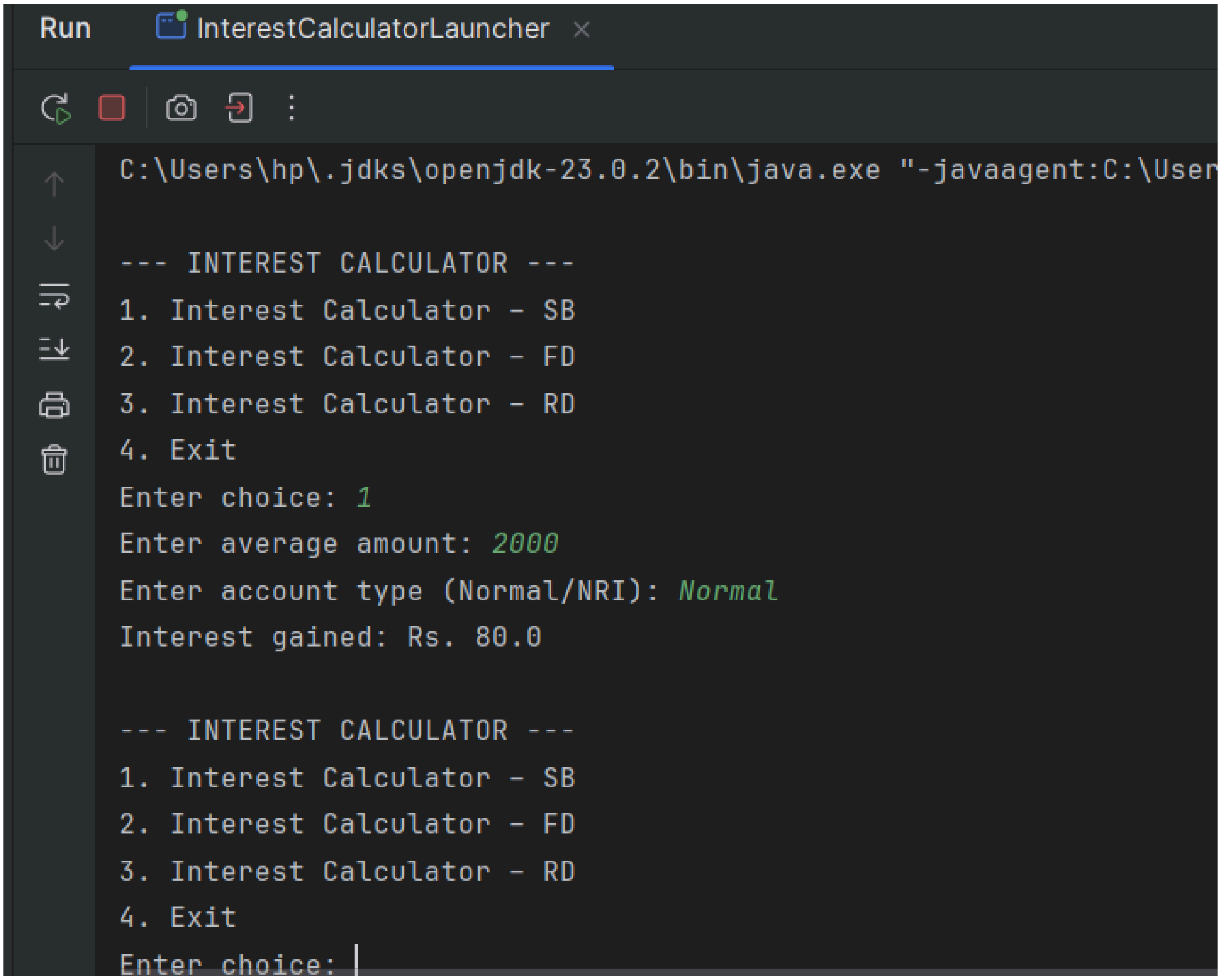
RDAccount.java

```java
public class RDAccount extends Account {
    private int noOfMonths;
    private double monthlyAmount;
    private int age;

    public RDAccount(int noOfMonths, double monthlyAmount, int age) throws InvalidInputException {
        super(monthlyAmount * noOfMonths);
        if (noOfMonths <= 0 || monthlyAmount < 0 || age <= 0)
            throw new InvalidInputException("Invalid RD input values.");
        this.noOfMonths = noOfMonths;
        this.monthlyAmount = monthlyAmount;
        this.age = age;
    }

    @Override
    public double calculateInterest() {
        if (noOfMonths == 6)
            interestRate = age >= 60 ? 8.0 : 7.5;
        else if (noOfMonths == 9)
            interestRate = age >= 60 ? 8.25 : 7.75;
        else if (noOfMonths == 12)
            interestRate = age >= 60 ? 8.5 : 8.0;
        else if (noOfMonths == 15)
            interestRate = age >= 60 ? 8.75 : 8.25;
        else if (noOfMonths == 18)
            interestRate = age >= 60 ? 9.0 : 8.5;
        else if (noOfMonths == 21)
            interestRate = age >= 60 ? 9.25 : 8.75;
        else
            interestRate = 0;

        return (amount * interestRate) / 100;
    }
}
```

InterestCalculatorLauncher.java

```
import java.util.Scanner;

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

        while (running) {
            System.out.println("\n--- INTEREST CALCULATOR ---");
            System.out.println("1. Interest Calculator – SB");
            System.out.println("2. Interest Calculator – FD");
            System.out.println("3. Interest Calculator – RD");
            System.out.println("4. Exit");
            System.out.print("Enter choice: ");
            int choice = sc.nextInt();

            try {
                switch (choice) {
                    case 1:
                        System.out.print("Enter average amount: ");
                        double sbAmount = sc.nextDouble();
                        sc.nextLine();  // consume newline
                        System.out.print("Enter account type (Normal/NRI): ");
                        String type = sc.nextLine();
                        SBAccount sb = new SBAccount(sbAmount, type);
                        System.out.println("Interest gained: Rs. " + sb.calculateInterest());
                        break;

                    case 2:
                        System.out.print("Enter FD amount: ");
                        double fdAmount = sc.nextDouble();
                        System.out.print("Enter number of days: ");
                        int days = sc.nextInt();
                        System.out.print("Enter age: ");
                        int fdAge = sc.nextInt();
                        FDAccount fd = new FDAccount(fdAmount, days, fdAge);
                        System.out.println("Interest gained: Rs. " + fd.calculateInterest());
                        break;

                    case 3:
                        System.out.print("Enter monthly RD amount: ");
                        double monthly = sc.nextDouble();
                        System.out.print("Enter number of months (6, 9, 12, 15, 18, 21): ");
                        int months = sc.nextInt();
                        System.out.print("Enter age: ");
                        int rdAge = sc.nextInt();
                        RDAccount rd = new RDAccount(months, monthly, rdAge);
                        System.out.println("Interest gained: Rs. " + rd.calculateInterest());
                        break;

                    case 4:
                        running = false;
                        System.out.println("Thank you!");
                        break;

                    default:
                        System.out.println("Invalid choice.");
                }
            } catch (InvalidInputException e) {
                System.out.println("ERROR: " + e.getMessage());
            }
        }

        sc.close();
    }
}
```

3.OUTPUT:



```
C:\Users\hp\.jdks\openjdk-23.0.2\bin\java.exe "-javaagent:C:\User

--- INTEREST CALCULATOR ---
1. Interest Calculator - SB
2. Interest Calculator - FD
3. Interest Calculator - RD
4. Exit
Enter choice: 1
Enter average amount: 2000
Enter account type (Normal/NRI): Normal
Interest gained: Rs. 80.0

--- INTEREST CALCULATOR ---
1. Interest Calculator - SB
2. Interest Calculator - FD
3. Interest Calculator - RD
4. Exit
Enter choice:
```