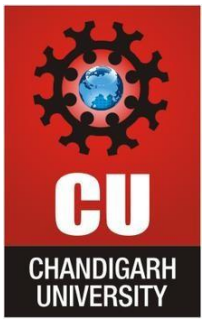
CHANDIGARH UNIVERSITY

UNIVERSITY INSTITUTE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING



Submitted By: Yash Kumar	**Submitted To: Ma'am Neeru Sharma**
Subject Name	**Project based learning in Java lab**
Subject Code	**20CSP-321**
Branch	**Computer Science**
Semester	**5th Semester**

LAB INDEX

Sr. No.	Program	Date	Evaluation				Sign.
			LW (12)	VV (10)	FW (8)	Total (30)	
01.	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	28/08/ 2022					

Experiment Title - 03

Student Name: Yash Kumar **UID: 20BCS9256**
Branch: CSE **Section/Group: 616 'B'**
Semester: 5th **Date of Performance: 28/04/2022**
Subject Name: Project based learning in Java lab
Subject Code: 20CSP-321

1. Aim

Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.

2. Software Requirements: Eclipse IDE

3. Code

```
package exp.three;

import java.util.Scanner;

public class InterestCalculator {

	public static void main(String[] args) {

		Scanner sc = new Scanner(System.in);
		System.out.println("SELECT THE OPTIONS " + "\n1." + " Interest Calculator-SB" + " \n2."
				+ " Interest Calculator-FD" + "\n3." + " InterestCalculator-RD" + "\n4 " + "
Exit");
		System.out.print("Enter your choice: ");
		int choice = sc.nextInt();
		System.out.println();

		switch (choice) {
		case 1:
			SBaccount sb = new SBaccount();
```

```
                try {
                        System.out.print("Enter the Average SB amount: ");
                        double amount = sc.nextDouble();
                        System.out.println("Interest gained is : Rs " + sb.calculateInterest(amount));

                } catch (InvalidAmountException e) {
                        System.out.println("Exception : Invalid amount");
                }
                break;

        case 2:
                try {
                        FDaccount fd = new FDaccount();
                        System.out.print("Enter the FD Amount: ");
                        double fAmount = sc.nextDouble();
                        System.out.println("Interest gained is: Rs " + fd.calculateInterest(fAmount));
                } catch (InvalidAgeException e) {
                        System.out.println("Invalid Age Entered...");
                } catch (InvalidAmountException e) {
                        System.out.println("Invalid Amount Entered...");

                } catch (InvalidDaysException e) {
                        System.out.println("Invalid Days Entered...");

                }
                break;
        case 3:
                try {
                        RDaccount rd = new RDaccount();
                        System.out.print("Enter the RD amount: ");
                        double Ramount = sc.nextDouble();
                        System.out.println("Interest gained is: Rs " + rd.calculateInterest(Ramount));
                } catch (InvalidAgeException e) {
                        System.out.println("Invalid Age Entered...");
                } catch (InvalidAmountException e) {
                        System.out.println("Invalid Amount Entered...");

                } catch (InvalidMonthsException e) {
                        System.out.println("Invalid Days Entered...");
                }

                break;

        case 4:
                System.out.println("DO YOU WANT TO CALCULATE AGAIN ????" + " " + "RUN
THE PROGRAM AGAIN...");
```

```
                        break;
                default:
                        System.out.println("Wrong choice...");

                }
                sc.close();
        }
}

abstract class Account {
        double interestRate;
        double amount;

        abstract double calculateInterest(double amount)
                        throws InvalidMonthsException, InvalidAgeException, InvalidAmountException,
InvalidDaysException;
}

class FDaccount extends Account {

        double FDinterestRate;
        double FDAmount;
        int noOfDays;
        int ageOfACHolder;
        double General, SCitizen;
        Scanner FDScanner = new Scanner(System.in);

        double calculateInterest(double amount) throws InvalidAgeException, InvalidAmountException,
InvalidDaysException {
                this.FDAmount = amount;

                System.out.print("Enter FD days: ");
                noOfDays = FDScanner.nextInt();
                System.out.print("Enter FD age holder: ");
                ageOfACHolder = FDScanner.nextInt();
                if (amount < 0) {
                        throw new InvalidAmountException();
                }
                if (noOfDays < 0) {
                        throw new InvalidDaysException();
                }
                if (ageOfACHolder < 0) {
                        throw new InvalidAgeException();
                }
                if (amount < 10000000) {
                        if (noOfDays >= 7 && noOfDays <= 14) {
```

```
                        General = 0.0450;
                        SCitizen = 0.0500;
                    } else if (noOfDays >= 15 && noOfDays <= 29) {
                        General = 0.0470;
                        SCitizen = 0.0525;
                    } else if (noOfDays >= 30 && noOfDays <= 45) {
                        General = 0.0550;
                        SCitizen = 0.0600;
                    } else if (noOfDays >= 45 && noOfDays <= 60) {
                        General = 0.0700;
                        SCitizen = 0.0750;
                    } else if (noOfDays >= 61 && noOfDays <= 184) {
                        General = 0.0750;
                        SCitizen = 0.0800;
                    } else if (noOfDays >= 185 && noOfDays <= 365) {
                        General = 0.0800;
                        SCitizen = 0.0850;
                    }
                    FDinterestRate = (ageOfACHolder < 50) ? General : SCitizen;
                } else {
                    if (noOfDays >= 7 && noOfDays <= 14) {
                        interestRate = 0.065;
                    } else if (noOfDays >= 15 && noOfDays <= 29) {
                        interestRate = 0.0675;
                    } else if (noOfDays >= 30 && noOfDays <= 45) {
                        interestRate = 0.00675;
                    } else if (noOfDays >= 45 && noOfDays <= 60) {
                        interestRate = 0.080;
                    } else if (noOfDays >= 61 && noOfDays <= 184) {
                        interestRate = 0.0850;
                    } else if (noOfDays >= 185 && noOfDays <= 365) {
                        interestRate = 0.10;
                    }
                }

                return FDAmount * FDinterestRate;
        }
}

class InvalidAgeException extends Exception {
}

class InvalidAmountException extends Exception {
}

class InvalidDaysException extends Exception {
```

```
}

class InvalidMonthsException extends Exception {
}

class RDaccount extends Account {

	double RDInterestRate;
	double RDamount;
	int noOfMonths;
	double monthlyAmount;
	double General, SCitizen;
	Scanner RDScanner = new Scanner(System.in);

	double calculateInterest(double Ramount)
			throws InvalidMonthsException, InvalidAmountException, InvalidAgeException {
		this.RDamount = Ramount;
		System.out.print("Enter RD months: ");
		noOfMonths = RDScanner.nextInt();
		System.out.print("Enter RD holder age: ");
		int age = RDScanner.nextInt();
		if (RDamount < 0) {
			throw new InvalidAmountException();
		}
		if (noOfMonths < 0) {
			throw new InvalidMonthsException();
		}
		if (age < 0) {
			throw new InvalidAgeException();
		}
		if (noOfMonths >= 0 && noOfMonths <= 6) {
			General = .0750;
			SCitizen = 0.080;
		} else if (noOfMonths >= 7 && noOfMonths <= 9) {
			General = .0775;
			SCitizen = 0.0825;
		} else if (noOfMonths >= 10 && noOfMonths <= 12) {
			General = .0800;
			SCitizen = 0.0850;
		} else if (noOfMonths >= 13 && noOfMonths <= 15) {
			General = .0825;
			SCitizen = 0.0875;
		} else if (noOfMonths >= 16 && noOfMonths <= 18) {
			General = .0850;
			SCitizen = 0.0900;
		} else if (noOfMonths >= 22) {
```

```
                    General = .0875;
                    SCitizen = 0.0925;
                }
                RDInterestRate = (age < 50) ? General : SCitizen;
                return RDamount * RDInterestRate;

        }

}

class SBaccount extends Account {
        double SBamount, SbInterestRate, interest;
        Scanner SBScanner = new Scanner(System.in);

        double calculateInterest(double amount) throws InvalidAmountException {
                this.SBamount = amount;
                if (SBamount < 0) {
                        throw new InvalidAmountException();
                }
                System.out.println("\nSelect account type \n1. NRI \n2. Normal ");
                System.out.print("Enter your choice: ");
                int accountChoice = SBScanner.nextInt();
                switch (accountChoice) {
                case 1:
                        SbInterestRate = .06;
                        break;
                case 2:
                        SbInterestRate = .04;
                        break;
                default:
                        System.out.println("Please choose right account again...");

                }
                return amount * SbInterestRate;
        }
}
```

4. Output

Console Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:45:26 pm – 5:45:47 pm)

```
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4  Exit
Enter your choice: 1

Enter the Average SB amount: 6500000

Select account type
1. NRI
2. Normal
Enter your choice: 1
Interest gained is : Rs 390000.0
```

Writable | Smart Insert | 7 : 45 : 129

Console Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:47:05 pm – 5:47:17 pm)

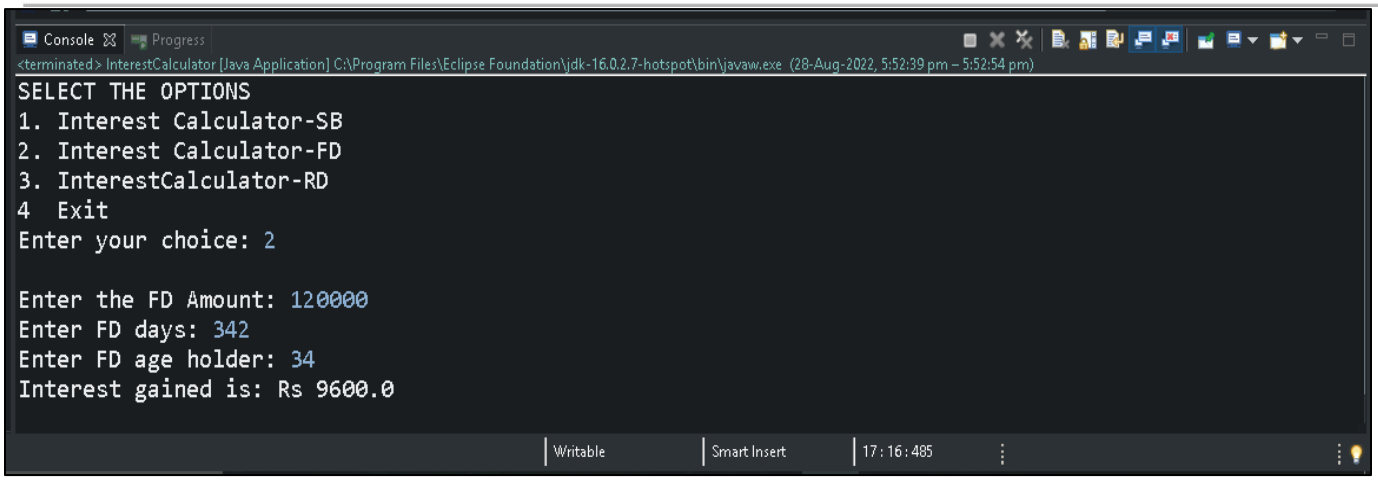
```
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4  Exit
Enter your choice: 1

Enter the Average SB amount: 340000

Select account type
1. NRI
2. Normal
Enter your choice: 2
Interest gained is : Rs 13600.0
```

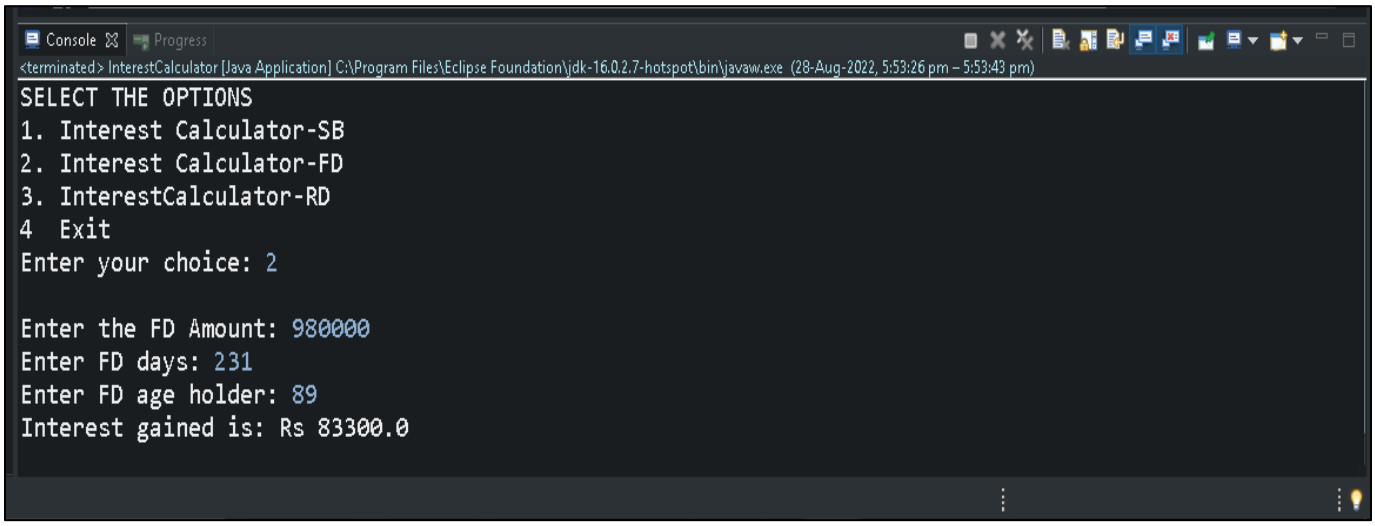
Writable | Smart Insert | 8 : 1 : 131

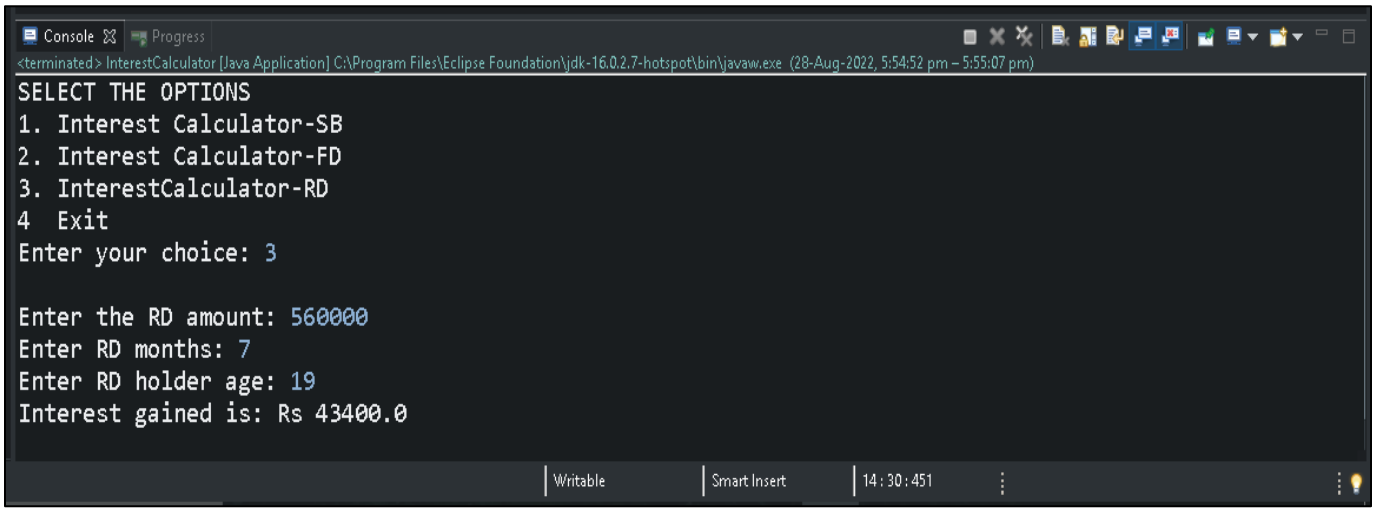
Console Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:47:57 pm – 5:48:11 pm)

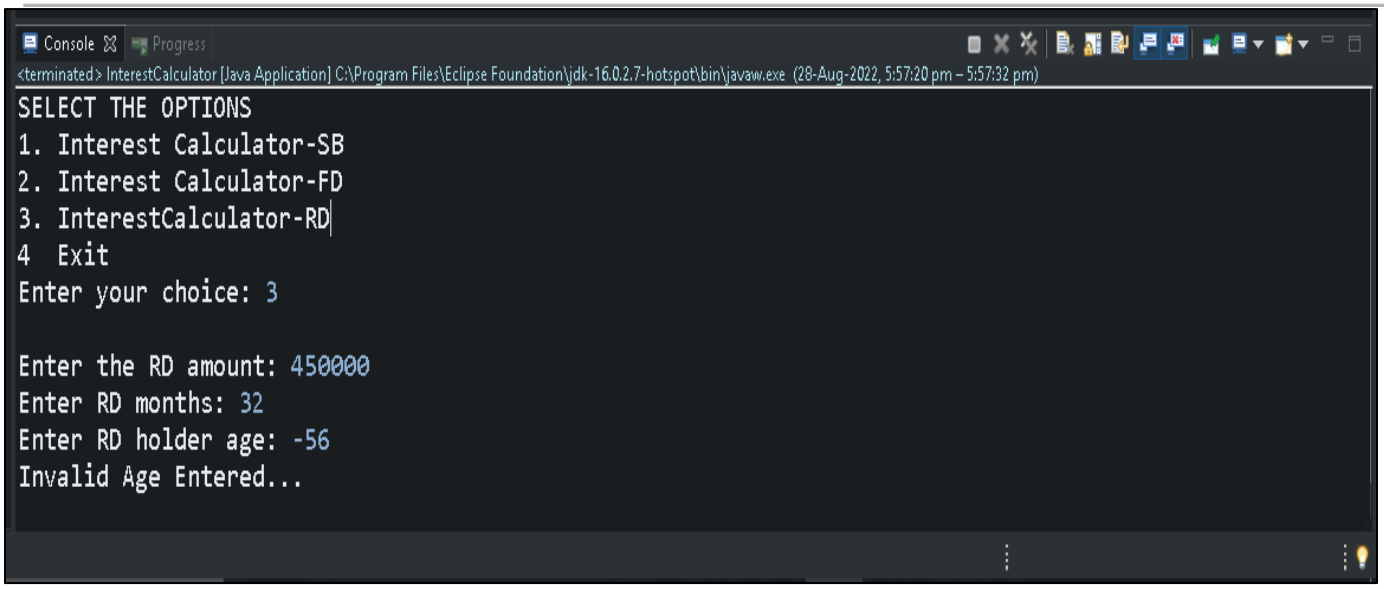
```
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4  Exit
Enter your choice: 1

Enter the Average SB amount: 23000

Select account type
1. NRI
2. Normal
Enter your choice: 4
Please choose right account again
Interest gained is : Rs 0.0
```

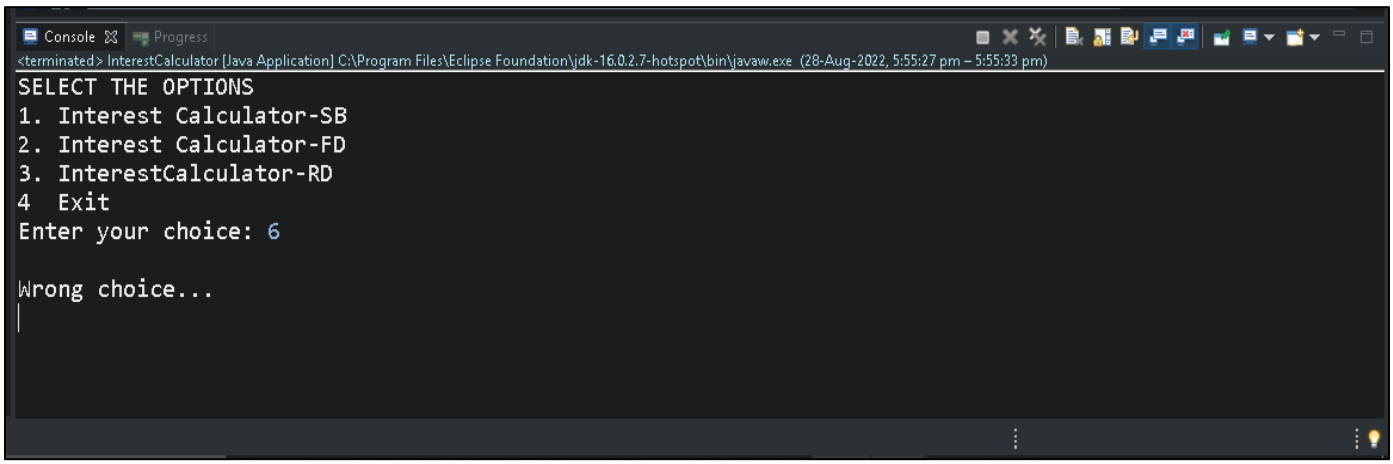
Writable | Smart Insert | 36 : 59 : 1141

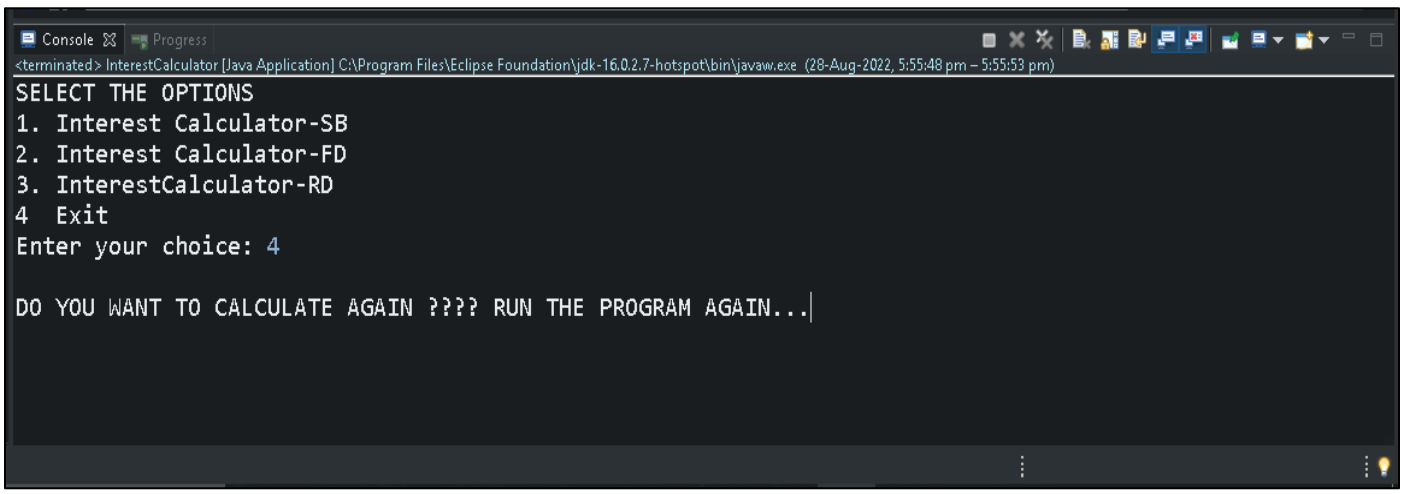

Console
Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:52:39 pm – 5:52:54 pm)
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4 Exit
Enter your choice: 2
Enter the FD Amount: 120000
Enter FD days: 342
Enter FD age holder: 34
Interest gained is: Rs 9600.0
Writable
Smart Insert
17 : 16 : 485


Console
Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:53:26 pm – 5:53:43 pm)
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4 Exit
Enter your choice: 2
Enter the FD Amount: 980000
Enter FD days: 231
Enter FD age holder: 89
Interest gained is: Rs 83300.0


Console
Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:54:52 pm – 5:55:07 pm)
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4 Exit
Enter your choice: 3
Enter the RD amount: 560000
Enter RD months: 7
Enter RD holder age: 19
Interest gained is: Rs 43400.0
Writable
Smart Insert
14 : 30 : 451


Console
Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:57:20 pm – 5:57:32 pm)
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4 Exit
Enter your choice: 3
Enter the RD amount: 450000
Enter RD months: 32
Enter RD holder age: -56
Invalid Age Entered...


Console
Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:55:27 pm – 5:55:33 pm)
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4 Exit
Enter your choice: 6
Wrong choice...


Console
Progress
<terminated> InterestCalculator [Java Application] C:\Program Files\Eclipse Foundation\jdk-16.0.2.7-hotspot\bin\javaw.exe (28-Aug-2022, 5:55:48 pm – 5:55:53 pm)
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4 Exit
Enter your choice: 4
DO YOU WANT TO CALCULATE AGAIN ???? RUN THE PROGRAM AGAIN...