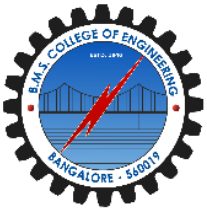
B.M.S. COLLEGE OF ENGINEERING
Basavanagudi, Bengaluru- 560019
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



LAB REPORT

On

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted By:
AYUSH GIRISH GAONKAR
1BM22CS063

In partial fulfilment of
BACHELOR OF ENGINEERING
In
COMPUTER SCIENCE AND ENGINEERING
2023-24

Faculty-In-Charge
Shravya A R
Assistant Professor
Department of Computer Science and Engineering

Table of Contents:

LAB-1

/*Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions. */

```
import java.util.Scanner;

class Quadratic {
   int a, b, c;
   double d, r1, r2;

   void QuadCalc(int a, int b, int c) {
      d = b * b - (4 * a * c);
      if (d < 0) {
         System.out.println("There are no real solutions");
      } else if (d > 0) {
         r1 = (-b + Math.sqrt(d)) / (2 * a);
         r2 = (-b - Math.sqrt(d)) / (2 * a);
         System.out.println("Roots are " + r1 + " and " + r2);
      } else if (d == 0) {
         r1 = r2 = -b / (2.0 * a);
         System.out.println("Equal roots: " + r1);
      }
   }
}

public class Lab1 {
   public static void main(String[] args) {
```

```
        System.out.println("ayush");
        System.out.println("1BM22CS063");
        Scanner s = new Scanner(System.in);
        Quadratic q = new Quadratic();
        System.out.println("Enter coefficients a, b, c:");
        int a = s.nextInt();
        int b = s.nextInt();
        int c = s.nextInt();
        q.QuadCalc(a, b, c);
    }
}
```

Output:

```
C:\Users\Admin\OneDrive\Desktop\AYUSH_G_G>javac Lab1.java

C:\Users\Admin\OneDrive\Desktop\AYUSH_G_G>java Lab1
ayush
1BM22CS063
Enter coefficients a, b, c:
4
5
6
There are no real solutions

C:\Users\Admin\OneDrive\Desktop\AYUSH_G_G>java Lab1
ayush
1BM22CS063
Enter coefficients a, b, c:
2
1
4
There are no real solutions

C:\Users\Admin\OneDrive\Desktop\AYUSH_G_G>java Lab1
ayush
1BM22CS063
Enter coefficients a, b, c:
1
-5
6
Roots are 3.0 and 2.0

C:\Users\Admin\OneDrive\Desktop\AYUSH_G_G>java Lab1
ayush
1BM22CS063
Enter coefficients a, b, c:
1
-4
4
Equal roots: 2.0
```

LAB-2

/*Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student. */

```
import java.util.Scanner;
class Student {
    String usn, name;
    int[] credits;
    int[] marks;
    int size;

    void accept() {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter name:");
        name = s.nextLine();
        System.out.println("Enter usn:");
        usn = s.nextLine();
        System.out.println("Enter number of subjects:");
        size = s.nextInt();
        credits = new int[size];
        marks = new int[size];
        for (int i = 0; i < size; i++) {
            System.out.println("Enter Subject " + (i + 1) + " marks:");
            marks[i] = s.nextInt();
            System.out.println("Enter credits for Subject " + (i + 1) + ":");
            credits[i] = s.nextInt();
        }
    }
    float calculate() {
```

```
        float sgpa = 0;
        int totalCredits = 0;
        for (int i = 0; i < size; i++) {
            sgpa += (marks[i] / 10) * credits[i];
            totalCredits += credits[i];
        }
        sgpa /= totalCredits;
        return sgpa;
    }

    void display() {
        System.out.println("Name: " + name);
        System.out.println("USN: " + usn);
        for (int i = 0; i < size; i++) {
            System.out.println("Subject " + (i + 1) + " Marks: " + marks[i]);
        }
        System.out.println("SGPA: " + calculate());
    }
}

public class Lab2 {
    public static void main(String[] args) {
        System.out.println("AYUSH");
        System.out.println("1BM22CS063");
        Student s = new Student();
        s.accept();
        s.display();
    }
}
```

Output:

```
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Lab2
AYUSH
1BM22CS063
Enter name:
Ayush
Enter usn:
1BM22CS063
Enter number of subjects:
6
Enter Subject 1 marks:
43
Enter credits for Subject 1:
4
Enter Subject 2 marks:
43
Enter credits for Subject 2:
4
Enter Subject 3 marks:
45
Enter credits for Subject 3:
4
Enter Subject 4 marks:
46
Enter credits for Subject 4:
3
Enter Subject 5 marks:
46
Enter credits for Subject 5:
3
Enter Subject 6 marks:
42
Enter credits for Subject 6:
2
Name: Ayush
USN: 1BM22CS063
Subject 1 Marks: 43
Subject 2 Marks: 43
Subject 3 Marks: 45
Subject 4 Marks: 46
Subject 5 Marks: 46
Subject 6 Marks: 42
SGPA: 4.0
```

LAB-3

/*Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.*/

```
import java.util.Scanner;

class Book {
   String bookname;
   String author;
   int price, pages;

   Book(String bookname, String author, int price, int pages) {
      this.bookname = bookname;
      this.author = author;
      this.price = price;
      this.pages = pages;
   }

   public void setBookname(String bookname) {
      this.bookname = bookname;
   }

   public void setBookauthor(String author) {
      this.author = author;
   }

   public void setPrice(int price) {
      this.price = price;
```

```
    }

    public void setPages(int pages) {
        this.pages = pages;
    }

    public String getBookname() {
        return bookname;
    }

    public String getBookauthor() {
        return author;
    }

    public int getPrice() {
        return price;
    }

    public int getPages() {
        return pages;
    }

    public String toString() {
        return "Book name: " + bookname + "\nAuthor: " + author + "\nPrice: " + price +
"\nPages: " + pages;
    }
}

class Lab3 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
```

```
        System.out.println("Enter number of books");
        int size = s.nextInt();
        s.nextLine();
        Book[] b = new Book[size];
        for (int i = 0; i < size; i++) {
            System.out.println("Enter name of Book " + (i + 1));
            String bookname = s.nextLine();
            System.out.println("Enter author of Book " + (i + 1));
            String author = s.nextLine();
            System.out.println("Enter price of Book " + (i + 1));
            int price = s.nextInt();
            System.out.println("Enter pages of Book " + (i + 1));
            int pages = s.nextInt();
            s.nextLine();
            b[i] = new Book(bookname, author, price, pages);
        }

        for (int i = 0; i < b.length; i++) {
            System.out.println("Book " + (i + 1) + ":\n" + b[i].toString());
        }
    }
}
```

Output:

```
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> javac Lab3.java
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Lab3
Enter number of books
2
Enter name of Book 1
maths
Enter author of Book 1
john
Enter price of Book 1
100
Enter pages of Book 1
100
Enter name of Book 2
physics
Enter author of Book 2
albert
Enter price of Book 2
200
Enter pages of Book 2
150
Book 1:
Book name: maths
Author: john
Price: 100
Pages: 100
Book 2:
Book name: physics
Author: albert
Price: 200
Pages: 150
```

LAB-4

/*Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape*/

```
import java.util.Scanner;
abstract class Shape {
   public int a, b;

   public abstract void printArea();
}
class Rectangle extends Shape {
   Rectangle(int a, int b) {
      this.a = a;
      this.b = b;
   }
  public void printArea() {
      System.out.println("Rectangle area is: " + (a * b));
   }
}
class Triangle extends Shape {
   Triangle(int a, int b) {
      this.a = a;
      this.b = b;
   }
 public void printArea() {
      System.out.println("Triangle area is: " + (0.5 * a * b));
   }
}
class Circle extends Shape {
   Circle(int a) {
```

```
        this.a = a;
    }

    public void printArea() {
        System.out.println("Circle area is: " + (Math.PI * a * a));
    }
}
public class Lab4 {
    public static void main(String[] args) {
        System.out.println("ayush");
        System.out.println("1BM22CS063");
        Scanner s = new Scanner(System.in);
        System.out.println("Enter length and breadth of rectangle: ");
        int len = s.nextInt();
        int bre = s.nextInt();
        Shape r = new Rectangle(len, bre);
        System.out.println("Enter base and height of Triangle: ");
        int ba = s.nextInt();
        int h = s.nextInt();
        Shape t = new Triangle(ba, h);
        System.out.println("Enter radius of circle ");
        int rad = s.nextInt();
        Shape c = new Circle(rad);
        r.printArea();
        t.printArea();
        c.printArea();
    }
}
```

Output:

```
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> javac Lab4.java
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Lab4
ayush
1BM22CS063
Enter length and breadth of rectangle:
4
5
Enter base and height of Triangle:
3
5
Enter radius of circle
5
Rectangle area is: 20
Triangle area is: 7.5
Circle area is: 78.53981633974483
```

LAB-5

/*Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed.

Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

a) Accept deposit from customer and update the balance.

b) Display the balance.

c) Compute and deposit interest

d) Permit withdrawal and update the balance

Check for the minimum balance, impose penalty if necessary and update the balance. */

```
import java.util.Scanner;

class Account {
   String customerName;
   long accountNumber;
   String accountType;
   double balance;

   Account(String customerName, long accountNumber, String accountType, double balance)
{
      this.customerName = customerName;
      this.accountNumber = accountNumber;
      this.accountType = accountType;
      this.balance = balance;
```

```
    }

    void deposit(double amount) {
        balance += amount;
        System.out.println("\nDeposit of " + amount + " was successful. Balance: " + balance);
    }

    void displayBalance() {
        System.out.println("\nAccount Number: " + accountNumber + "\nCustomer Name: " +
customerName + "\nAccount Type: " + accountType + "\nBalance: " + balance);
    }

    void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
            System.out.println("\nWithdrawal of " + amount + " successful. Updated balance: " +
balance);
        } else {
            System.out.println("\nInsufficient funds. Withdrawal failed.\n");
        }
    }
}

class SavingsAccount extends Account {
    SavingsAccount(String customerName, long accountNumber, double balance) {
        super(customerName, accountNumber, "Savings", balance);
    }

    void interest(double rate) {
        double interest = balance * rate / 100;
        balance += interest;
```

```
        System.out.println("\nInterest computed and deposited. Updated balance: " + balance);
    }
}

class CurrentAccount extends Account {
    double minimumBalance = 1000;
    double serviceCharge = 50;

    CurrentAccount(String customerName, long accountNumber, double balance) {
        super(customerName, accountNumber, "Current", balance);
    }

    void withdraw(double amount) {
        if (balance - amount >= minimumBalance) {
            balance -= amount;
            System.out.println("\nWithdrawal of " + amount + " successful. Updated balance: " +
balance);
        } else {
            System.out.println("\nInsufficient funds. Withdrawal failed.\n");
        }
    }

    void checkMinimumBalance() {
        if (balance < minimumBalance) {
            balance -= serviceCharge;
            System.out.println("\nMinimum balance not maintained. Service charge imposed.
Updated balance: " + balance);
        } else {
            System.out.println("\nMinimum balance maintained. Service charge not imposed.
Updated balance: " + balance);
        }
```

```
    }
}

public class Bank {
    public static void main(String[] args) {


        Scanner s = new Scanner(System.in);

        System.out.println("Savings Account: ");
        System.out.print("Enter customer name: ");
        String name = s.nextLine();
        System.out.print("Enter account number: ");
        long number = s.nextLong();
        System.out.print("Enter initial balance: ");
        double balance = s.nextDouble();
        SavingsAccount savingsAccount = new SavingsAccount(name, number, balance);

        System.out.println("Current Account: ");
        System.out.print("Enter customer name: ");
        name = s.next();
        System.out.print("Enter account number: ");
        number = s.nextLong();
        System.out.print("Enter balance: ");
        balance = s.nextDouble();
        CurrentAccount currentAccount = new CurrentAccount(name, number, balance);
        int choice;
        do {
            System.out.println("\nSelect an option:");
            System.out.println("1. Deposit to Savings Account");
```

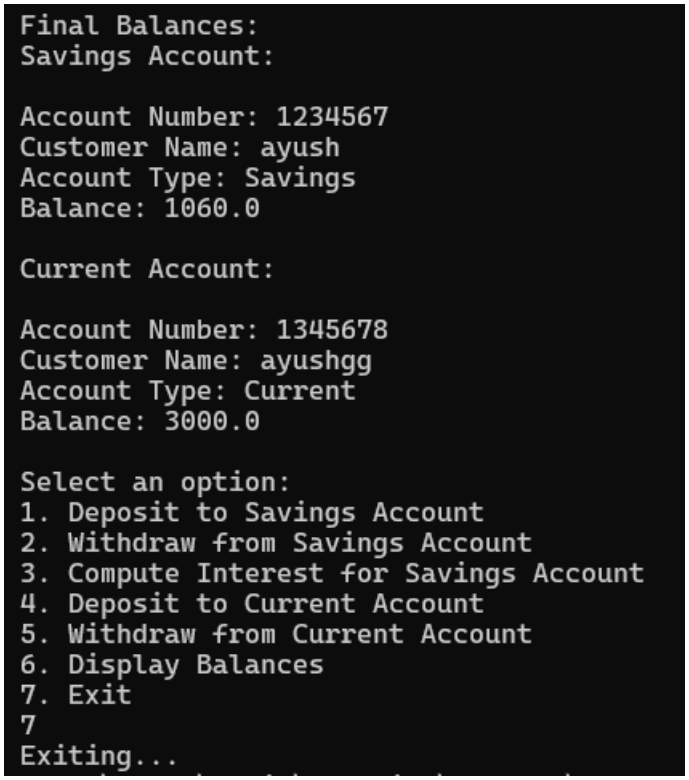
```
System.out.println("2. Withdraw from Savings Account");
System.out.println("3. Compute Interest for Savings Account");
System.out.println("4. Deposit to Current Account");
System.out.println("5. Withdraw from Current Account");
System.out.println("6. Display Balances");
System.out.println("7. Exit");

choice = s.nextInt();

switch (choice) {
   case 1:
      System.out.print("Enter deposit amount for Savings Account: ");
      double depositAmount = s.nextDouble();
      savingsAccount.deposit(depositAmount);
      break;
   case 2:
      System.out.print("Enter withdrawal amount for Savings Account: ");
      double withdrawalAmount = s.nextDouble();
      savingsAccount.withdraw(withdrawalAmount);
      break;
   case 3:
      System.out.print("Enter interest rate for Savings Account: ");
      double interestRate = s.nextDouble();
      savingsAccount.interest(interestRate);
      break;
   case 4:
      System.out.print("Enter deposit amount for Current Account: ");
      double depositCurrent = s.nextDouble();
      currentAccount.deposit(depositCurrent);
      break;
```

```
            case 5:
                System.out.print("Enter withdrawal amount for Current Account: ");
                double withdrawalCurrent = s.nextDouble();
                currentAccount.withdraw(withdrawalCurrent);
                break;
            case 6:
                System.out.println("\nFinal Balances:");
                System.out.println("Savings Account:");
                savingsAccount.displayBalance();
                System.out.println("\nCurrent Account:");
                currentAccount.displayBalance();
                break;
            case 7:
                System.out.println("Exiting...");
                break;
            default:
                System.out.println("Invalid choice. Please try again.");
        }
    } while (choice != 7);
  }
}
```

Output:

```
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Bank
Savings Account:
Enter customer name: ayush
Enter account number: 1234567
Enter initial balance: 2000
Current Account:
Enter customer name: ayushgg
Enter account number: 1345678
Enter balance: 2000

Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
1
Enter deposit amount for Savings Account: 2000

Deposit of 2000.0 was successful. Balance: 4000.0

Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
2
Enter withdrawal amount for Savings Account: 3000

Withdrawal of 3000.0 successful. Updated balance: 1000.0

Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
3
Enter interest rate for Savings Account: 6

Interest computed and deposited. Updated balance: 1060.0
```

```
Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
4
Enter deposit amount for Current Account: 2000

Deposit of 2000.0 was successful. Balance: 4000.0

Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
1000
Invalid choice. Please try again.

Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
5
Enter withdrawal amount for Current Account: 1000

Withdrawal of 1000.0 successful. Updated balance: 3000.0

Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
6
```


Final Balances:
Savings Account:
Account Number: 1234567
Customer Name: ayush
Account Type: Savings
Balance: 1060.0
Current Account:
Account Number: 1345678
Customer Name: ayushgg
Account Type: Current
Balance: 3000.0
Select an option:
1. Deposit to Savings Account
2. Withdraw from Savings Account
3. Compute Interest for Savings Account
4. Deposit to Current Account
5. Withdraw from Current Account
6. Display Balances
7. Exit
7
Exiting...

LAB-6

/*Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.*/

//CIE PACKAGE

//Student.java

```
package CIE;

public class Student {
    public String usn, name;
    public int sem;
    public Student(String usn, String name, int sem) {
        this.usn = usn;
        this.name = name;
        this.sem = sem;
    }
}
```

//Internals.java

```
package CIE;

public class Internals extends Student {
    public double[] internalMarks = new double[5];
    public Internals(String usn, String name, int sem, double[] internalMarks) {
        super(usn, name, sem);
        this.internalMarks = internalMarks;
    }
}
```

//SEE PACKAGE

```
//External.java
package SEE;

import CIE.Student;
public class External extends Student {
    public double[] seeMarks = new double[5];
    public External(String usn, String name, int sem, double[] seeMarks) {
        super(usn, name, sem);
        this.seeMarks = seeMarks;
    }
}
//FinalMarks.java
import CIE.Internals;
import SEE.External;
import java.util.Scanner;
public class FinalMarks {
    public static void main(String[] args) {
            System.out.println("ayush");
        System.out.println("1BM22CS063");
        Scanner input = new Scanner(System.in);

        System.out.println("Enter the number of students: ");
        int n = input.nextInt();
        input.nextLine();

        Internals[] internals = new Internals[n];
        External[] externals = new External[n];


        for (int i = 0; i < n; i++) {
```

```
            System.out.println("Enter details of Student " + (i + 1));
            System.out.println("Enter USN: ");
            String usn = input.nextLine();
            System.out.println("Enter Name: ");
            String name = input.nextLine();
            System.out.println("Enter Semester: ");
            int sem = input.nextInt();
            input.nextLine();
            double[] internalMarks = new double[5];
            System.out.println("Enter Internal Marks for 5 courses: ");
            for (int j = 0; j < 5; j++) {
                    System.out.println("Enter Internal Marks for Course " + (j + 1) + ": ");
               internalMarks[j] = input.nextDouble();
            }
            input.nextLine();

            internals[i] = new Internals(usn, name, sem, internalMarks);
        }
        for (int i = 0; i < n; i++) {
            System.out.println("Enter SEE Marks for Student " + (i + 1));
            double[] seeMarks = new double[5];
            for (int j = 0; j < 5; j++) {
               System.out.println("Enter SEE Marks for Course " + (j + 1) + ": ");
               seeMarks[j] = input.nextDouble();
            }
            input.nextLine();

            externals[i] = new External(internals[i].usn, internals[i].name, internals[i].sem,
seeMarks);
        }
        System.out.println("\nFinal Marks of Students:");
```

```
        for (int i = 0; i < n; i++) {
            System.out.println("Student " + (i + 1) + " : USN: " + internals[i].usn + "\nName: " +
                    internals[i].name + "\nSemester: " + internals[i].sem);

            for (int j = 0; j < 5; j++) {
                System.out.println("Subject " + (j + 1) + ": " +
                        ((internals[i].internalMarks[j]) + (externals[i].seeMarks[j] / 2)) + "\n");
            }
            System.out.println();
        }
    }
}
```

Output:

```
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> cd Lab6
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G\Lab6> javac CIE/Student.java
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G\Lab6> javac CIE/Internals.java
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G\Lab6> javac SEE/External.java
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G\Lab6> java FinalMarks.java
ayush
1BM22CS063
Enter the number of students:
3
Enter details of Student 1
Enter USN:
1BM22CS063
Enter Name:
Ayush
Enter Semester:
3
Enter Internal Marks for 5 courses:
Enter Internal Marks for Course 1:
45
Enter Internal Marks for Course 2:
43
Enter Internal Marks for Course 3:
43
Enter Internal Marks for Course 4:
42
Enter Internal Marks for Course 5:
47
Enter details of Student 2
Enter USN:
1BM22CS999
Enter Name:
kishan
Enter Semester:
3
Enter Internal Marks for 5 courses:
Enter Internal Marks for Course 1:
49
Enter Internal Marks for Course 2:
49
Enter Internal Marks for Course 3:
48
Enter Internal Marks for Course 4:
47
Enter Internal Marks for Course 5:
46
```

```
Enter details of Student 3
Enter USN:
1BM22CS888
Enter Name:
abc
Enter Semester:
3
Enter Internal Marks for 5 courses:
Enter Internal Marks for Course 1:
45
Enter Internal Marks for Course 2:
47
Enter Internal Marks for Course 3:
48
Enter Internal Marks for Course 4:
49
Enter Internal Marks for Course 5:
47
Enter SEE Marks for Student 1
Enter SEE Marks for Course 1:
98
Enter SEE Marks for Course 2:
97
Enter SEE Marks for Course 3:
97
Enter SEE Marks for Course 4:
96
Enter SEE Marks for Course 5:
95
Enter SEE Marks for Student 2
Enter SEE Marks for Course 1:
94
Enter SEE Marks for Course 2:
93
Enter SEE Marks for Course 3:
93
Enter SEE Marks for Course 4:
89
Enter SEE Marks for Course 5:
90
Enter SEE Marks for Student 3
Enter SEE Marks for Course 1:
94
Enter SEE Marks for Course 2:
93
Enter SEE Marks for Course 3:
93
Enter SEE Marks for Course 4:
95
Enter SEE Marks for Course 5:
96

Final Marks of Students:
Student 1 : USN: 1BM22CS063
Name: Ayush
Semester: 3
Subject 1: 94.0

Subject 2: 91.5

Subject 3: 91.5

Subject 4: 90.0

Subject 5: 94.5


Student 2 : USN: 1BM22CS999
Name: kishan
Semester: 3
Subject 1: 96.0

Subject 2: 95.5

Subject 3: 94.5

Subject 4: 91.5

Subject 5: 91.0


Student 3 : USN: 1BM22CS888
Name: abc
Semester: 3
Subject 1: 92.0

Subject 2: 93.5

Subject 3: 94.5

Subject 4: 96.5

Subject 5: 95.0
```

LAB-7

/*Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.*/

```
import java.util.Scanner;

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

class Father {
   Father(int fage) throws WrongAge {
      if (fage < 0) {
         throw new WrongAge("Age cannot be negative");
      } else {
         System.out.println("Father's age is " + fage);
      }
   }
}

class Son extends Father {
   Son(int fage, int sage) throws WrongAge {
      super(fage);
      if (sage >= fage) {
```

```
            throw new WrongAge("Son's age cannot be greater than father's age");
        } else {
            System.out.println("Son's age is " + sage);
        }
    }
}

public class Lab7 {
    public static void main(String[] args) {
        System.out.println("ayush");
        System.out.println("1BM22CS063");
        Scanner scanner = new Scanner(System.in);

        try {
            System.out.println("Enter father's age:");
            int fage = scanner.nextInt();
            System.out.println("Enter son's age:");
            int sage = scanner.nextInt();

            Son son = new Son(fage, sage);

        } catch (WrongAge e) {
            System.out.println("Exception caught: " + e.getMessage());
        }
        scanner.close();
    }
}
```

Output:

```
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> javac Lab7.java
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Lab7
ayush
1BM22CS063
Enter father's age:
43
Enter son's age:
45
Father's age is 43
Exception caught: Son's age cannot be greater than father's age
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Lab7
ayush
1BM22CS063
Enter father's age:
42
Enter son's age:
25
Father's age is 42
Son's age is 25
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Lab7
ayush
1BM22CS063
Enter father's age:
-40
Enter son's age:
19
Exception caught: Age cannot be negative
```

LAB-8

/*Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.*/

```
class BMS extends Thread {
   public void run() {
     for(int i=0;i<3;i++) {
         System.out.println("BMS College of Engineering");
         try {
            Thread.sleep(10000);
         } catch (InterruptedException e) {
            e.printStackTrace();
         }
      }
   }
}

class CSE extends Thread {
   public void run() {
     for(int i=0;i<10;i++) {
         System.out.println("CSE");
         try {
            Thread.sleep(2000);
         } catch (InterruptedException e) {
            e.printStackTrace();
         }
      }
   }
}
```

```
public class Lab8 {
    public static void main(String[] args) {
      System.out.println("ayush");
        System.out.println("1BM22CS063");
        BMS b = new BMS();
        b.start();

        CSE c = new CSE();
        c.start();
    }
}
```

Output:

```
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> javac Lab8.java
PS C:\Users\Admin\Onedrive\Desktop\AYUSH_G_G> java Lab8
ayush
1BM22CS063
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
```