B.M.S COLLEGE OF ENGINEERING BENGALURU
Autonomous Institute, Affiliated to VTU

LAB REPORT

OBJECT ORIENTED JAVA PROGRAMMING
23CS3PCOOJ

PROGRAMS WITH OUTPUTS

Submitted by:

CHARAN G - 1BM22CS078

III SEMESTER 3 'B'

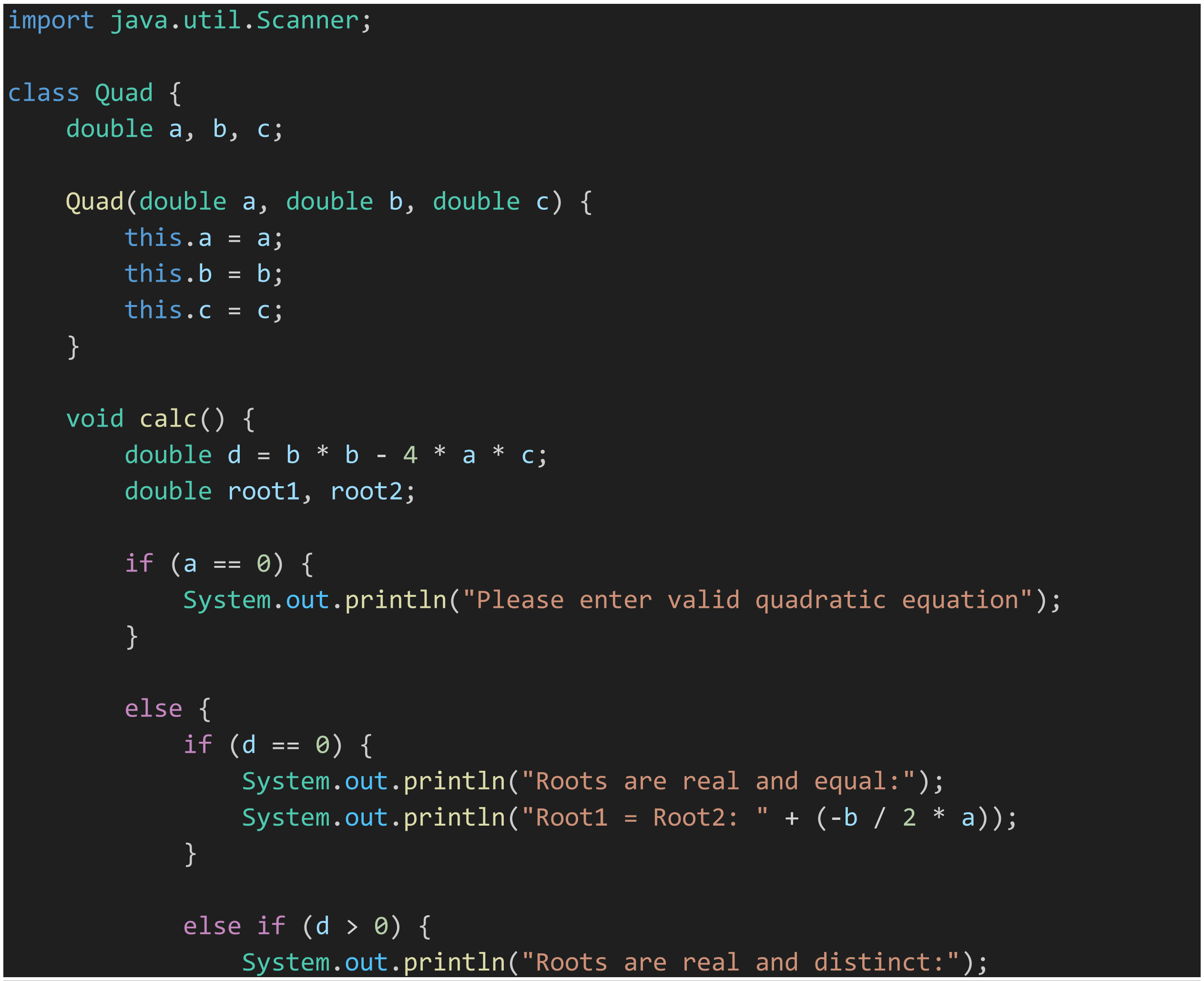
Department of Computer Science and Engineering
B.M.S College of Engineering
Bull Temple Road, Basavanagudi, Bangalore 560 019
2023-2024

LAB PROGRAMS

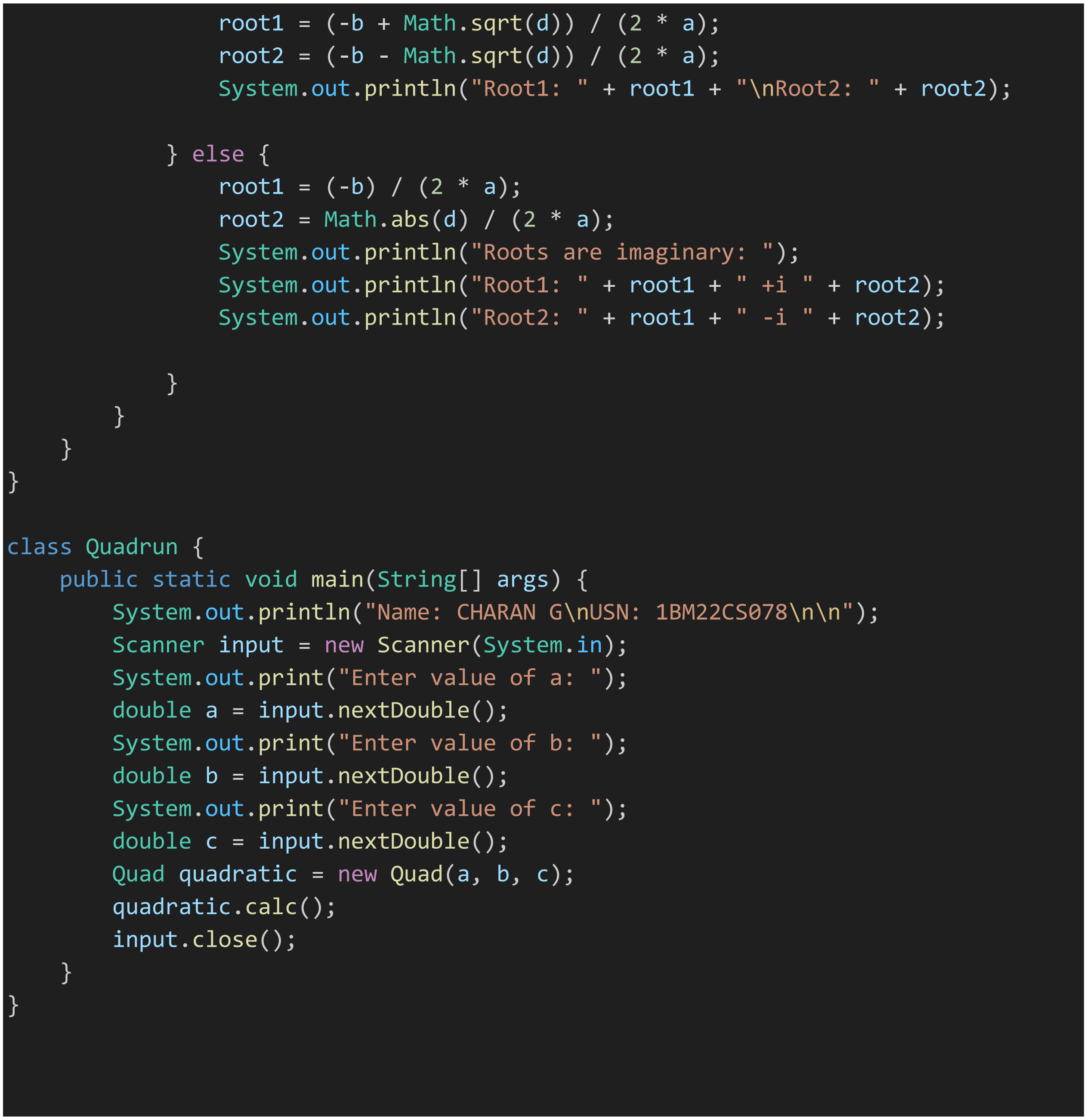
LAB PROGRAM – 1 – QUADRATIC EQUATION

Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a, b, c and use the quadratic formula. If the discriminate $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

PROGRAM:

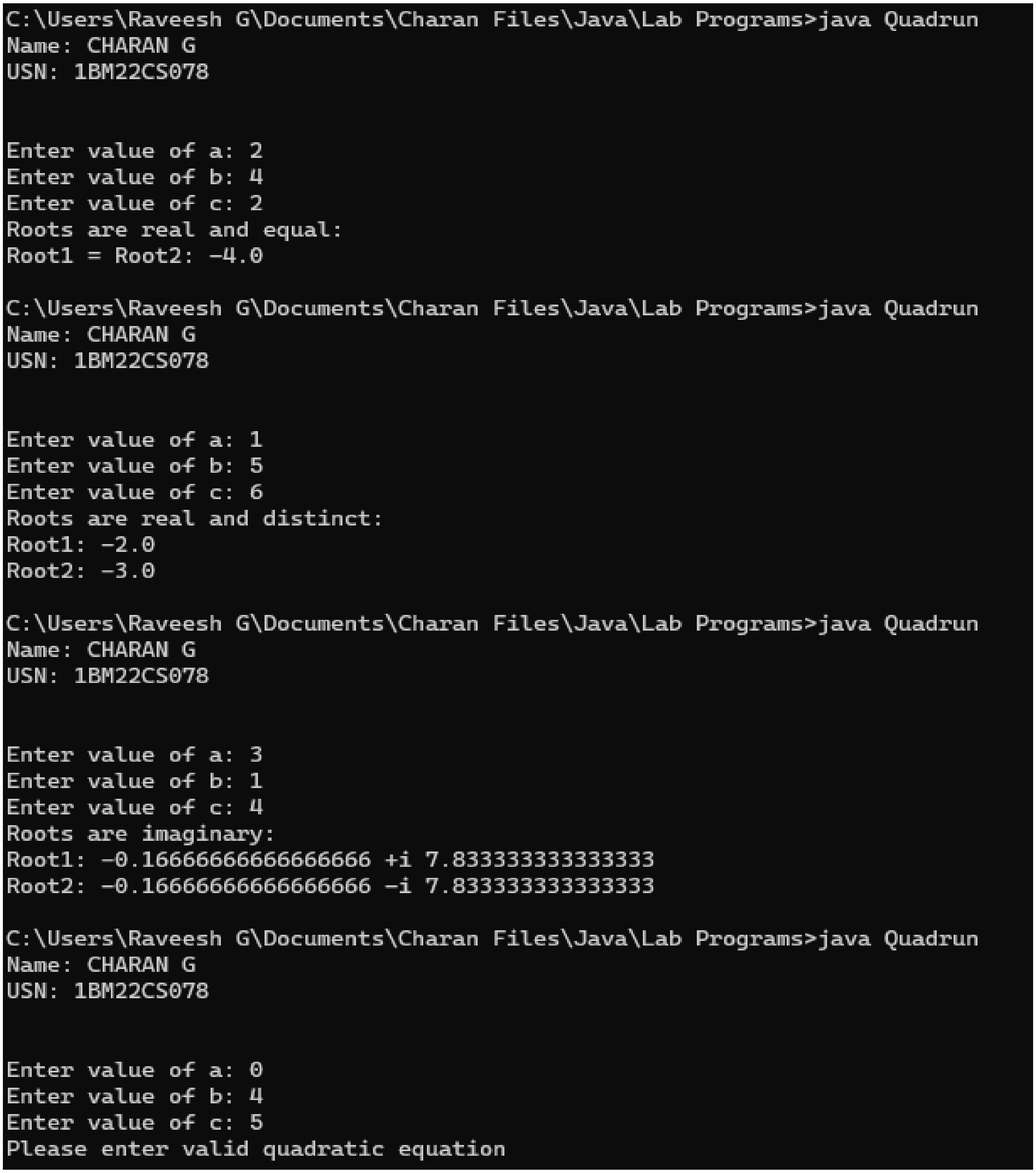


```java
import java.util.Scanner;

class Quad {
    double a, b, c;

    Quad(double a, double b, double c) {
        this.a = a;
        this.b = b;
        this.c = c;
    }

    void calc() {
        double d = b * b - 4 * a * c;
        double root1, root2;

        if (a == 0) {
            System.out.println("Please enter valid quadratic equation");
        }

        else {
            if (d == 0) {
                System.out.println("Roots are real and equal:");
                System.out.println("Root1 = Root2: " + (-b / 2 * a));
            }

            else if (d > 0) {
                System.out.println("Roots are real and distinct:");
```



```
                root1 = (-b + Math.sqrt(d)) / (2 * a);
                root2 = (-b - Math.sqrt(d)) / (2 * a);
                System.out.println("Root1: " + root1 + "\nRoot2: " + root2);

            } else {
                root1 = (-b) / (2 * a);
                root2 = Math.abs(d) / (2 * a);
                System.out.println("Roots are imaginary: ");
                System.out.println("Root1: " + root1 + " +i " + root2);
                System.out.println("Root2: " + root1 + " -i " + root2);

            }
        }
    }
}

class Quadrun {
    public static void main(String[] args) {
        System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");
        Scanner input = new Scanner(System.in);
        System.out.print("Enter value of a: ");
        double a = input.nextDouble();
        System.out.print("Enter value of b: ");
        double b = input.nextDouble();
        System.out.print("Enter value of c: ");
        double c = input.nextDouble();
        Quad quadratic = new Quad(a, b, c);
        quadratic.calc();
        input.close();
    }
}
```

OUTPUT:

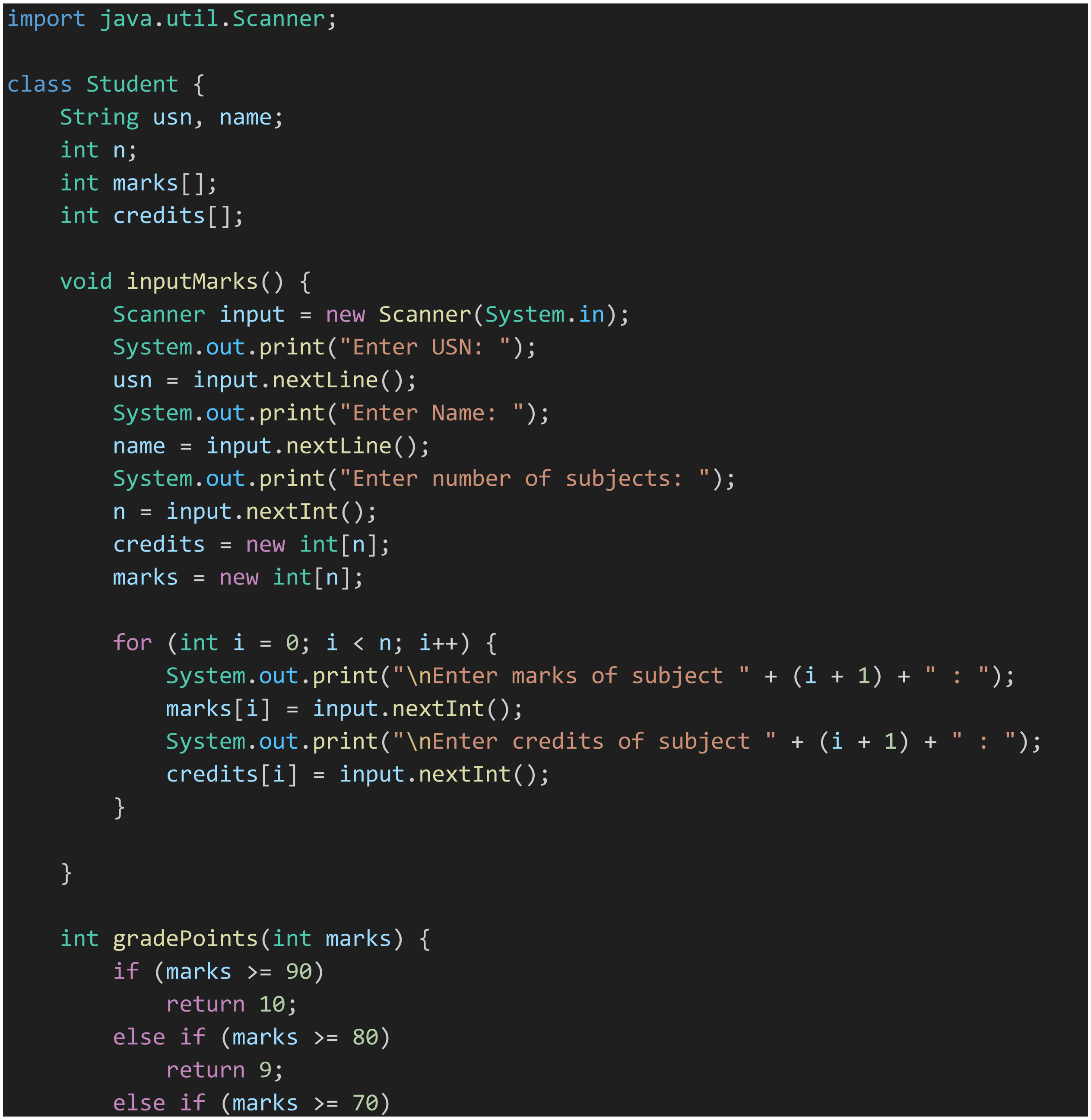


```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java Quadrun
Name: CHARAN G
USN: 1BM22CS078


Enter value of a: 2
Enter value of b: 4
Enter value of c: 2
Roots are real and equal:
Root1 = Root2: -4.0

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java Quadrun
Name: CHARAN G
USN: 1BM22CS078


Enter value of a: 1
Enter value of b: 5
Enter value of c: 6
Roots are real and distinct:
Root1: -2.0
Root2: -3.0

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java Quadrun
Name: CHARAN G
USN: 1BM22CS078


Enter value of a: 3
Enter value of b: 1
Enter value of c: 4
Roots are imaginary:
Root1: -0.16666666666666666 +i 7.833333333333333
Root2: -0.16666666666666666 -i 7.833333333333333

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java Quadrun
Name: CHARAN G
USN: 1BM22CS078


Enter value of a: 0
Enter value of b: 4
Enter value of c: 5
Please enter valid quadratic equation
```

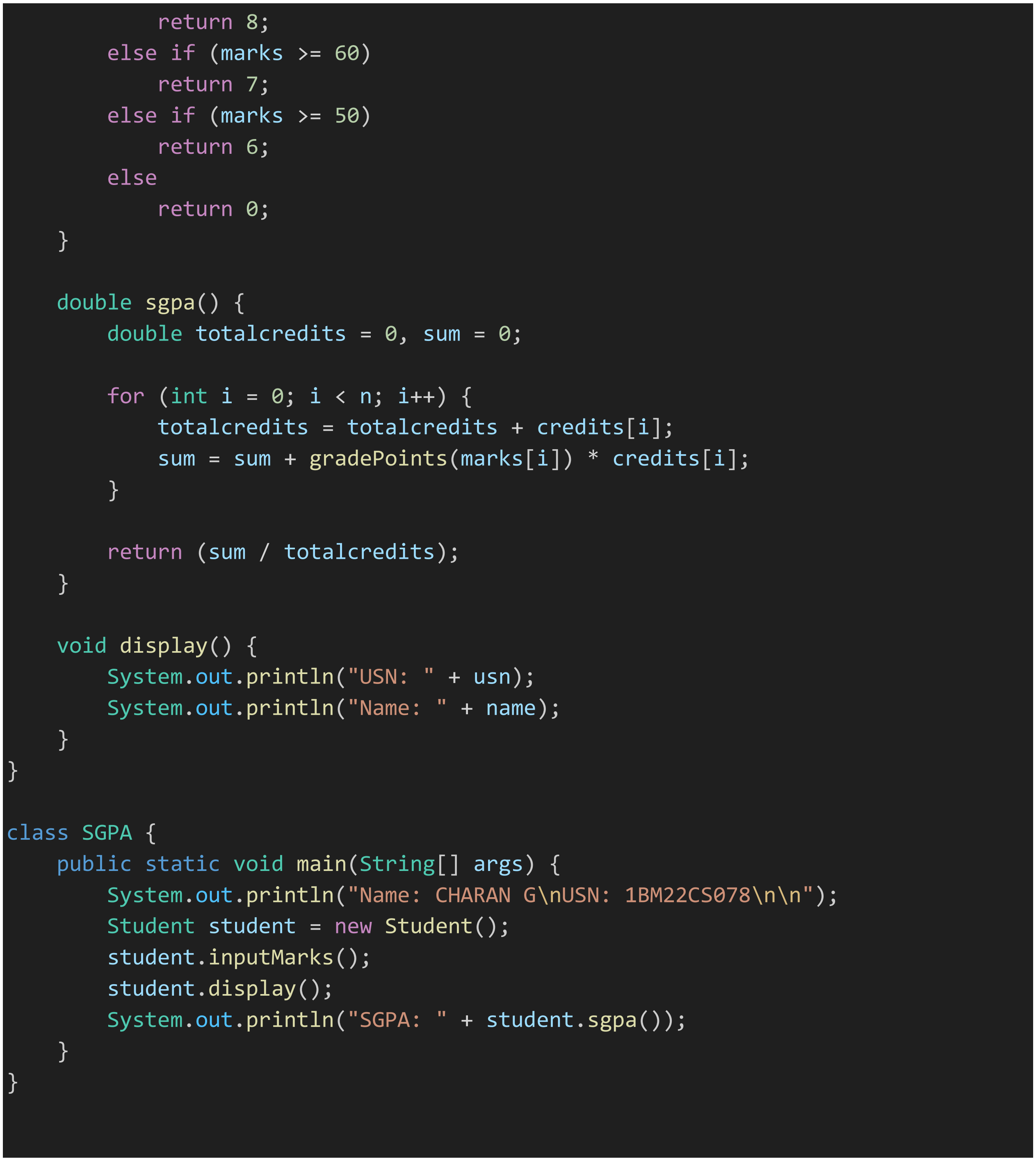
LAB PROGRAM – 2 – SGPA OF A STUDENT

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.

PROGRAM:

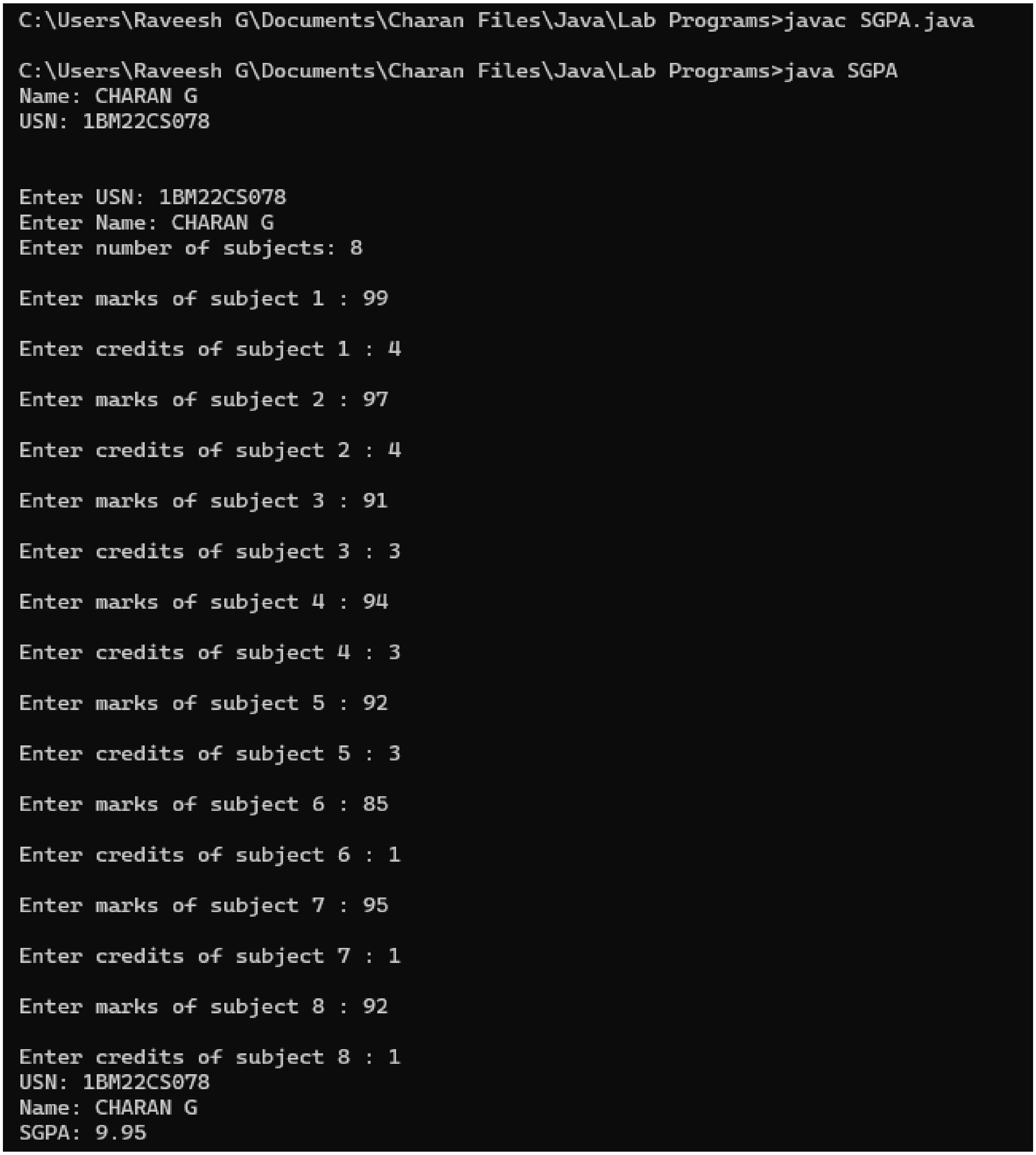


```java
import java.util.Scanner;

class Student {
    String usn, name;
    int n;
    int marks[];
    int credits[];

    void inputMarks() {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter USN: ");
        usn = input.nextLine();
        System.out.print("Enter Name: ");
        name = input.nextLine();
        System.out.print("Enter number of subjects: ");
        n = input.nextInt();
        credits = new int[n];
        marks = new int[n];

        for (int i = 0; i < n; i++) {
            System.out.print("\nEnter marks of subject " + (i + 1) + " : ");
            marks[i] = input.nextInt();
            System.out.print("\nEnter credits of subject " + (i + 1) + " : ");
            credits[i] = input.nextInt();
        }

    }

    int gradePoints(int marks) {
        if (marks >= 90)
            return 10;
        else if (marks >= 80)
            return 9;
        else if (marks >= 70)
```



```java
            return 8;
        else if (marks >= 60)
            return 7;
        else if (marks >= 50)
            return 6;
        else
            return 0;
    }

    double sgpa() {
        double totalcredits = 0, sum = 0;

        for (int i = 0; i < n; i++) {
            totalcredits = totalcredits + credits[i];
            sum = sum + gradePoints(marks[i]) * credits[i];
        }

        return (sum / totalcredits);
    }

    void display() {
        System.out.println("USN: " + usn);
        System.out.println("Name: " + name);
    }
}

class SGPA {
    public static void main(String[] args) {
        System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");
        Student student = new Student();
        student.inputMarks();
        student.display();
        System.out.println("SGPA: " + student.sgpa());
    }
}
```

OUTPUT:

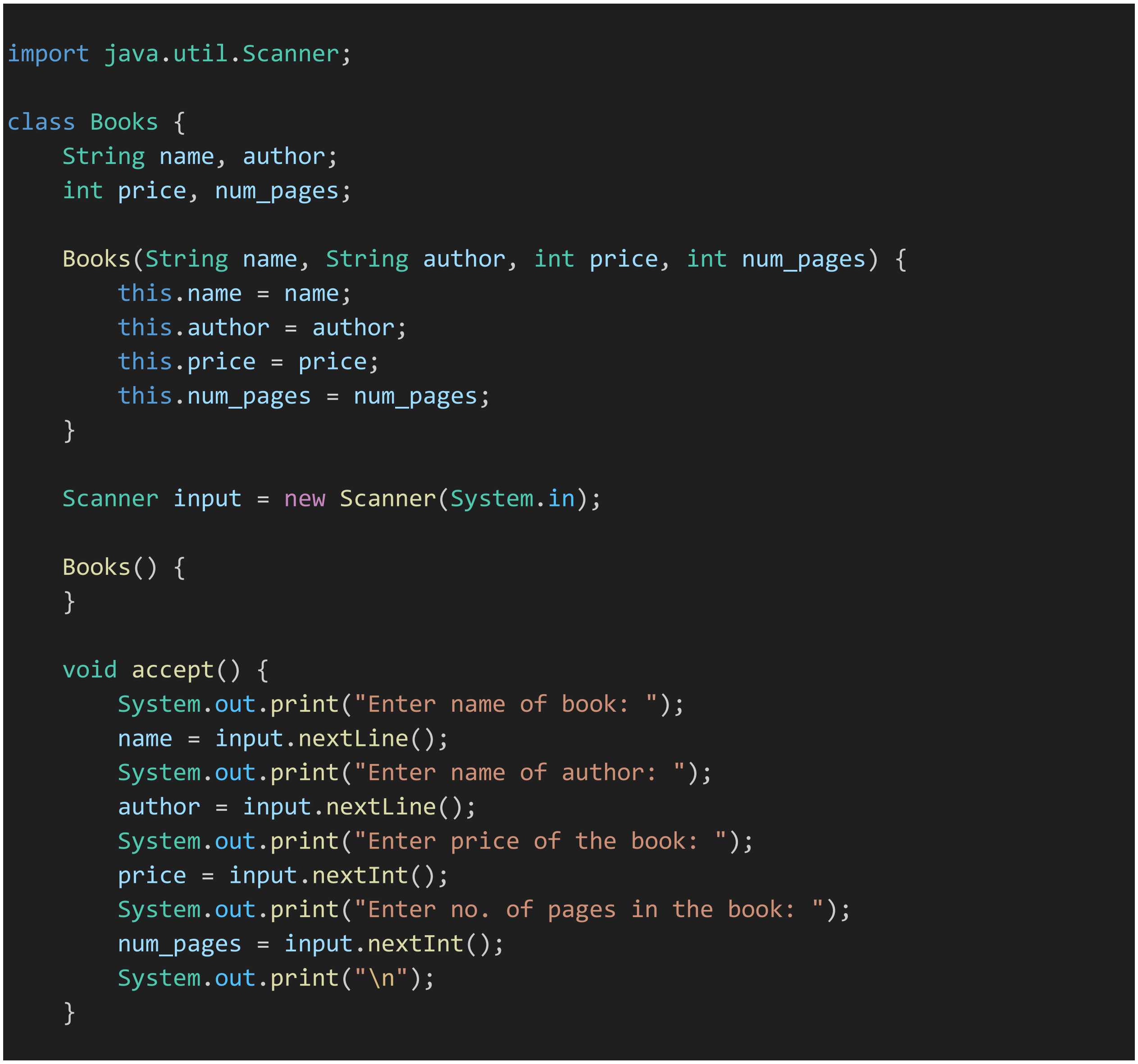


```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>javac SGPA.java

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java SGPA
Name: CHARAN G
USN: 1BM22CS078


Enter USN: 1BM22CS078
Enter Name: CHARAN G
Enter number of subjects: 8

Enter marks of subject 1 : 99

Enter credits of subject 1 : 4

Enter marks of subject 2 : 97

Enter credits of subject 2 : 4

Enter marks of subject 3 : 91

Enter credits of subject 3 : 3

Enter marks of subject 4 : 94

Enter credits of subject 4 : 3

Enter marks of subject 5 : 92

Enter credits of subject 5 : 3

Enter marks of subject 6 : 85

Enter credits of subject 6 : 1

Enter marks of subject 7 : 95

Enter credits of subject 7 : 1

Enter marks of subject 8 : 92

Enter credits of subject 8 : 1
USN: 1BM22CS078
Name: CHARAN G
SGPA: 9.95
```

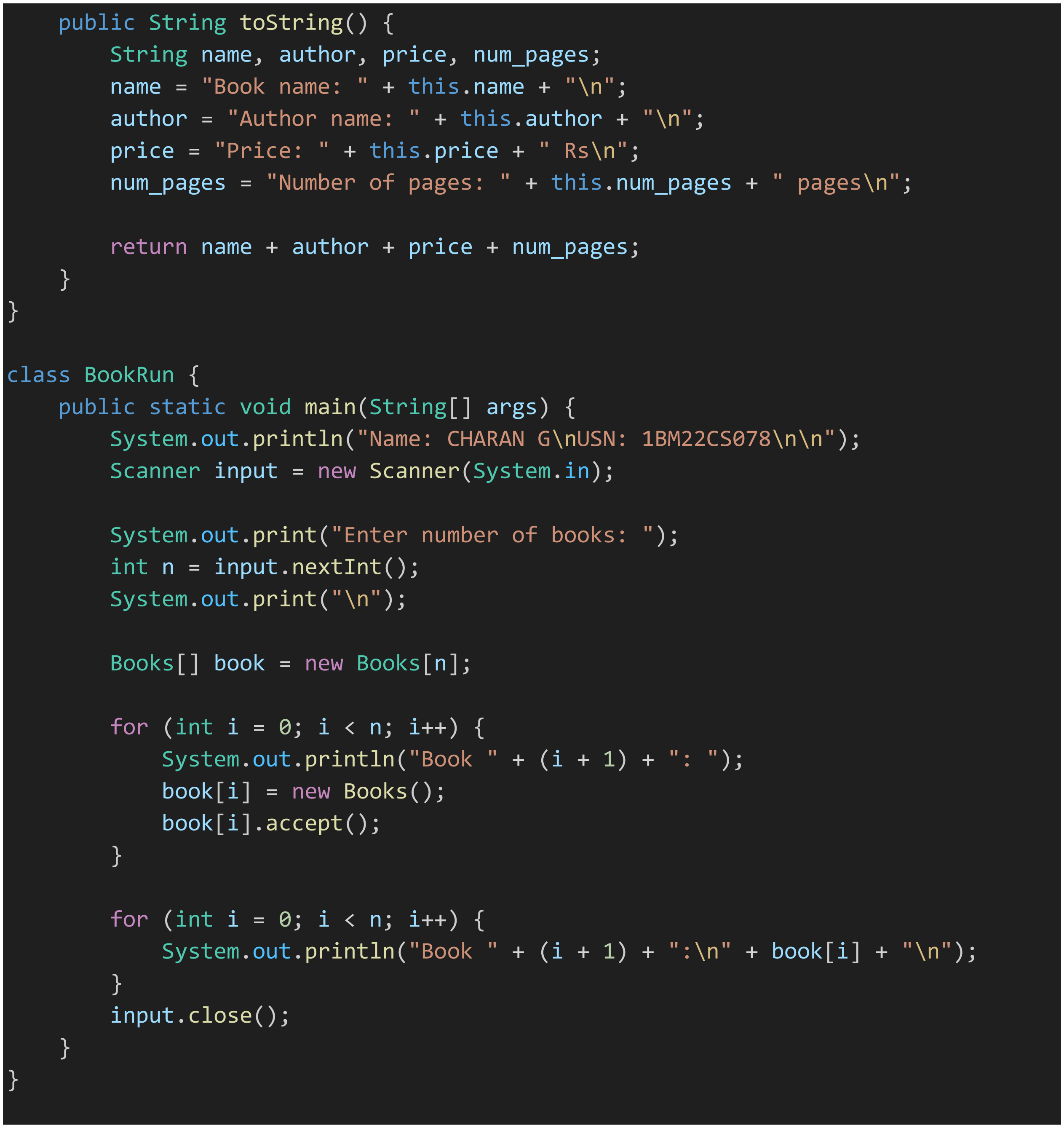
LAB PROGRAM – 3 – BOOK DATA WITH toString()

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.

PROGRAM:

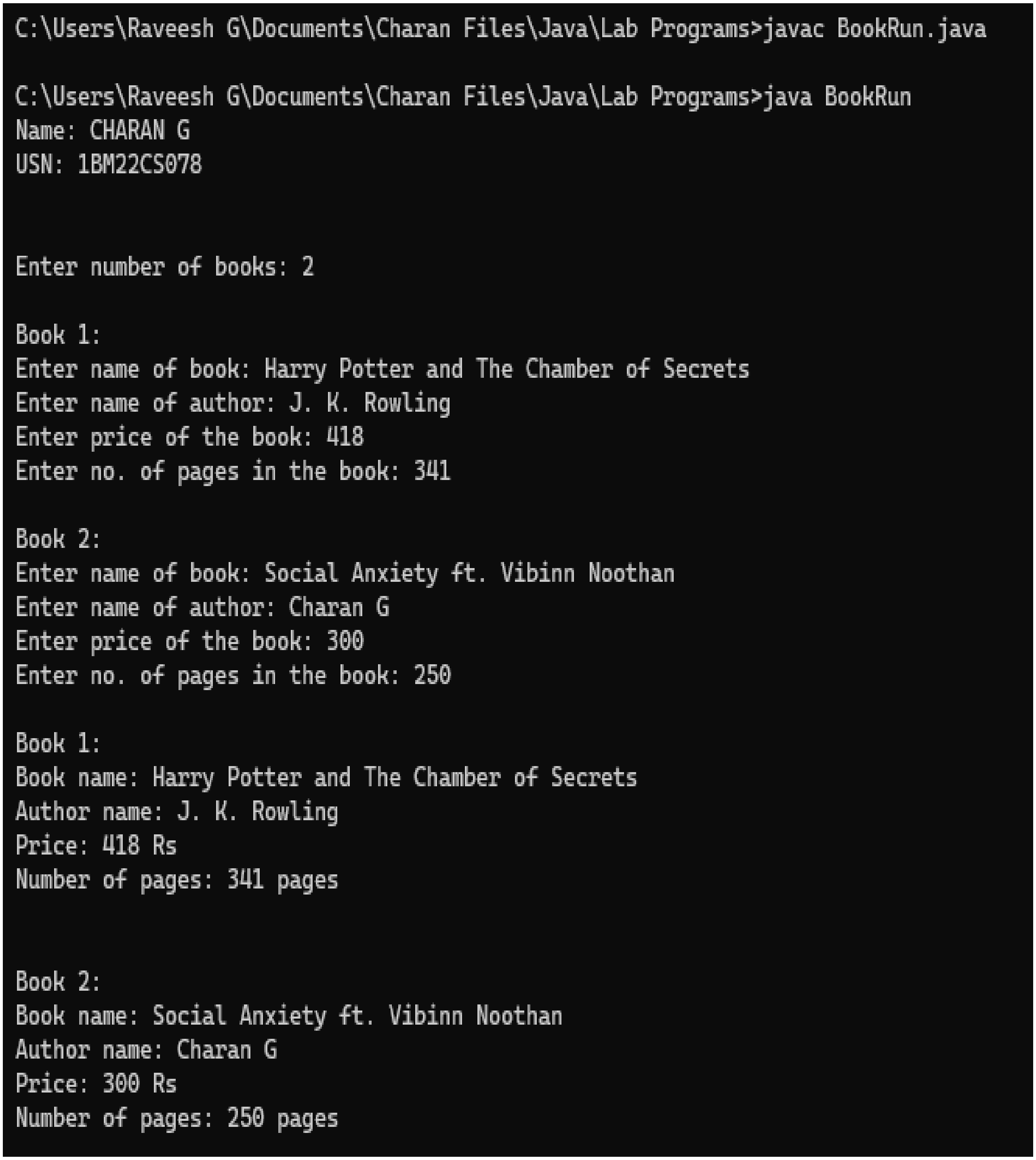


```
import java.util.Scanner;

class Books {
    String name, author;
    int price, num_pages;

    Books(String name, String author, int price, int num_pages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }

    Scanner input = new Scanner(System.in);

    Books() {
    }

    void accept() {
        System.out.print("Enter name of book: ");
        name = input.nextLine();
        System.out.print("Enter name of author: ");
        author = input.nextLine();
        System.out.print("Enter price of the book: ");
        price = input.nextInt();
        System.out.print("Enter no. of pages in the book: ");
        num_pages = input.nextInt();
        System.out.print("\n");
    }
```



```
    public String toString() {
        String name, author, price, num_pages;
        name = "Book name: " + this.name + "\n";
        author = "Author name: " + this.author + "\n";
        price = "Price: " + this.price + " Rs\n";
        num_pages = "Number of pages: " + this.num_pages + " pages\n";

        return name + author + price + num_pages;
    }
}

class BookRun {
    public static void main(String[] args) {
        System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");
        Scanner input = new Scanner(System.in);

        System.out.print("Enter number of books: ");
        int n = input.nextInt();
        System.out.print("\n");

        Books[] book = new Books[n];

        for (int i = 0; i < n; i++) {
            System.out.println("Book " + (i + 1) + ": ");
            book[i] = new Books();
            book[i].accept();
        }

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

OUTPUT:

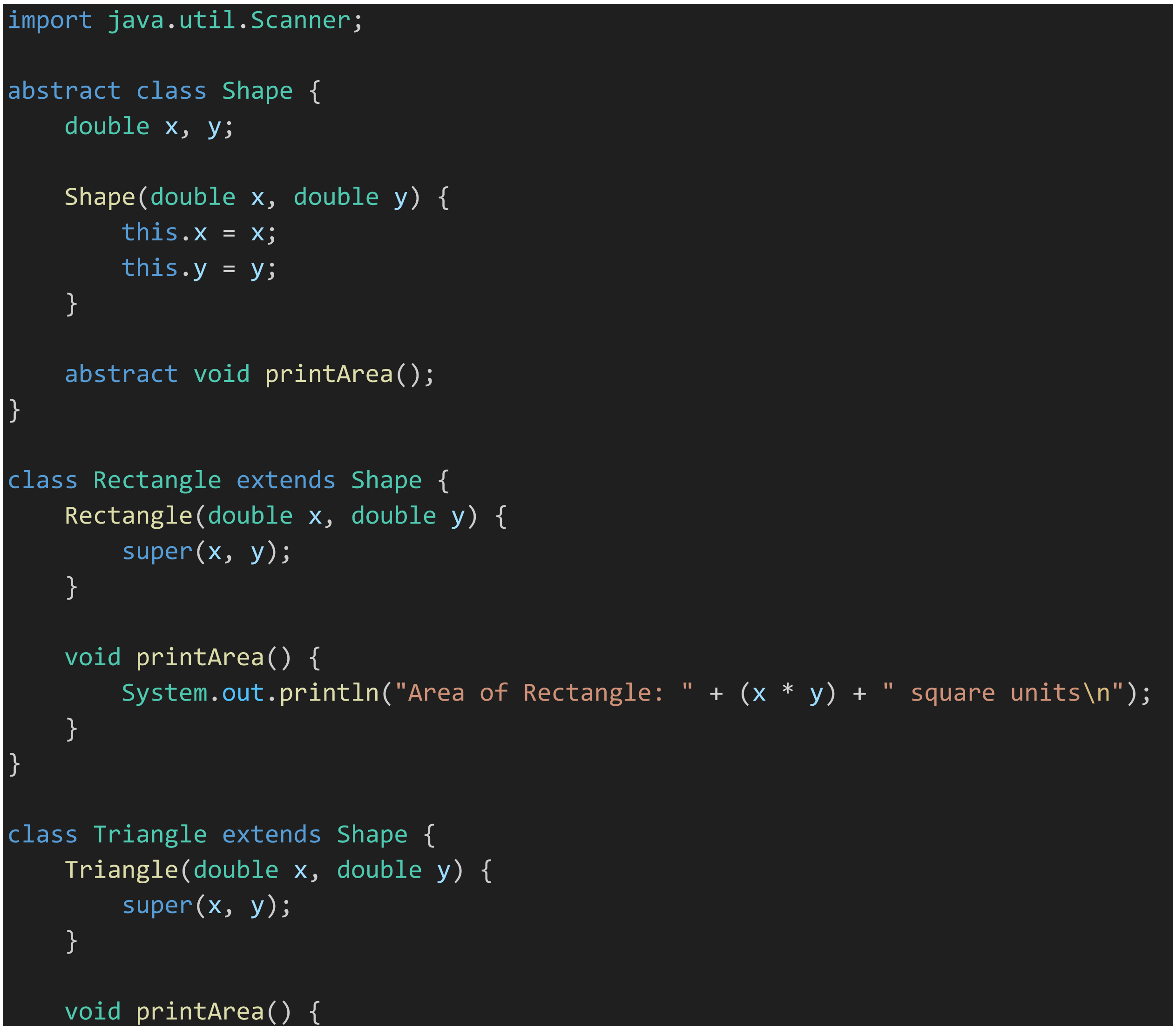


```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>javac BookRun.java

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java BookRun
Name: CHARAN G
USN: 1BM22CS078


Enter number of books: 2

Book 1:
Enter name of book: Harry Potter and The Chamber of Secrets
Enter name of author: J. K. Rowling
Enter price of the book: 418
Enter no. of pages in the book: 341

Book 2:
Enter name of book: Social Anxiety ft. Vibinn Noothan
Enter name of author: Charan G
Enter price of the book: 300
Enter no. of pages in the book: 250

Book 1:
Book name: Harry Potter and The Chamber of Secrets
Author name: J. K. Rowling
Price: 418 Rs
Number of pages: 341 pages


Book 2:
Book name: Social Anxiety ft. Vibinn Noothan
Author name: Charan G
Price: 300 Rs
Number of pages: 250 pages
```

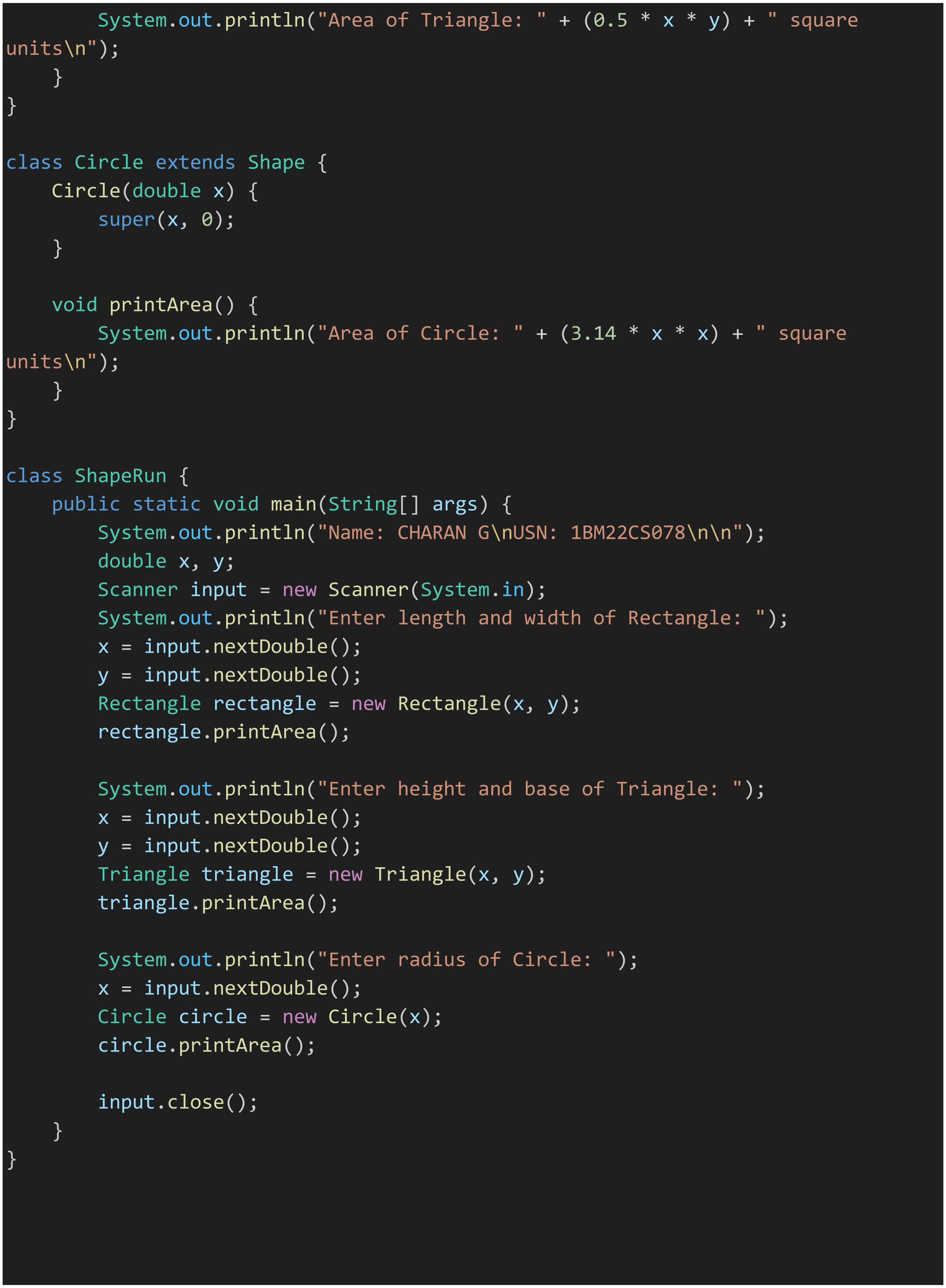
PROGRAM – 4 – ABSTRACT CLASS – SHAPE - AREA

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.

PROGRAM:

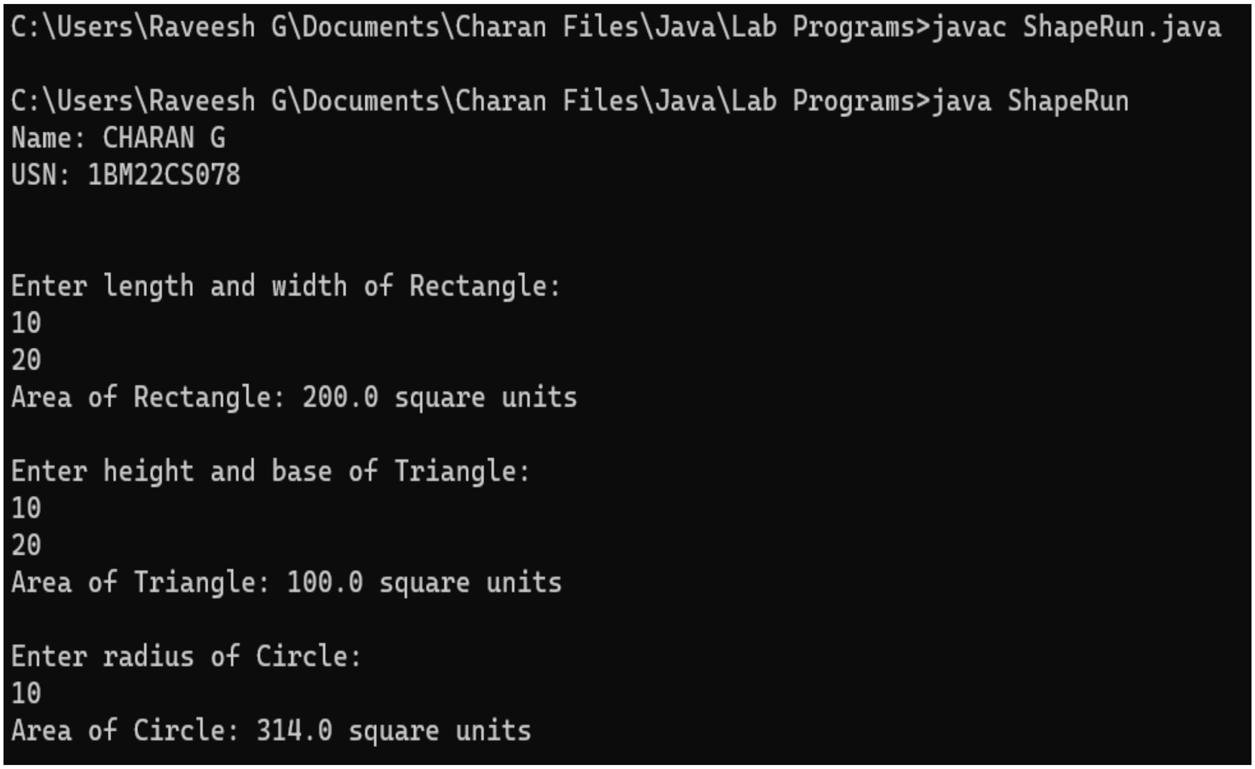


```
import java.util.Scanner;

abstract class Shape {
    double x, y;

    Shape(double x, double y) {
        this.x = x;
        this.y = y;
    }

    abstract void printArea();
}

class Rectangle extends Shape {
    Rectangle(double x, double y) {
        super(x, y);
    }

    void printArea() {
        System.out.println("Area of Rectangle: " + (x * y) + " square units\n");
    }
}

class Triangle extends Shape {
    Triangle(double x, double y) {
        super(x, y);
    }

    void printArea() {
```



```
        System.out.println("Area of Triangle: " + (0.5 * x * y) + " square
units\n");
    }
}

class Circle extends Shape {
    Circle(double x) {
        super(x, 0);
    }

    void printArea() {
        System.out.println("Area of Circle: " + (3.14 * x * x) + " square
units\n");
    }
}

class ShapeRun {
    public static void main(String[] args) {
        System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");
        double x, y;
        Scanner input = new Scanner(System.in);
        System.out.println("Enter length and width of Rectangle: ");
        x = input.nextDouble();
        y = input.nextDouble();
        Rectangle rectangle = new Rectangle(x, y);
        rectangle.printArea();

        System.out.println("Enter height and base of Triangle: ");
        x = input.nextDouble();
        y = input.nextDouble();
        Triangle triangle = new Triangle(x, y);
        triangle.printArea();

        System.out.println("Enter radius of Circle: ");
        x = input.nextDouble();
        Circle circle = new Circle(x);
        circle.printArea();

        input.close();
    }
}
```

OUTPUT:



```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>javac ShapeRun.java

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java ShapeRun
Name: CHARAN G
USN: 1BM22CS078


Enter length and width of Rectangle:
10
20
Area of Rectangle: 200.0 square units

Enter height and base of Triangle:
10
20
Area of Triangle: 100.0 square units

Enter radius of Circle:
10
Area of Circle: 314.0 square units
```

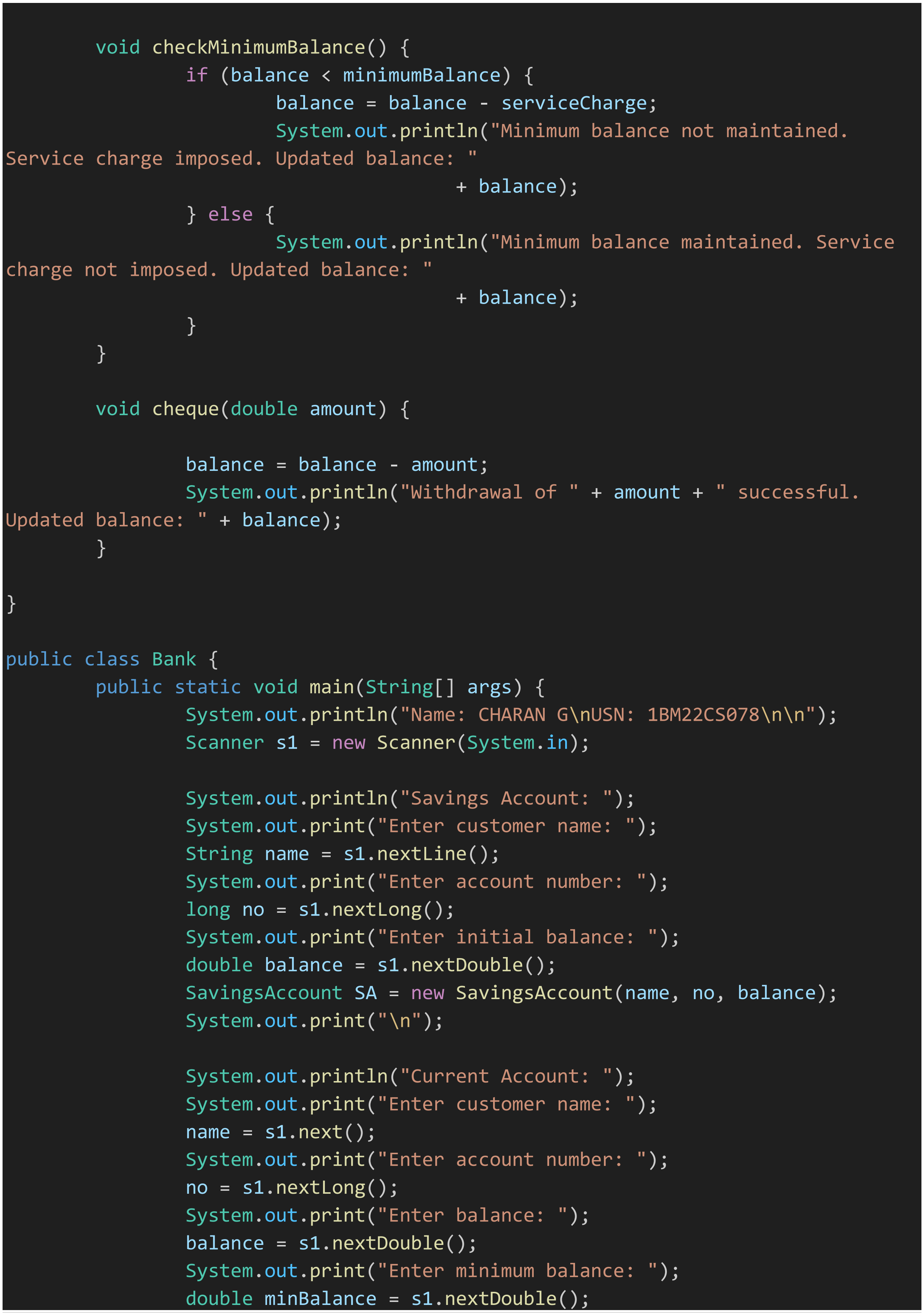
PROGRAM – 5 – BANK – SAVINGS AND CURRENT ACCOUNT

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.

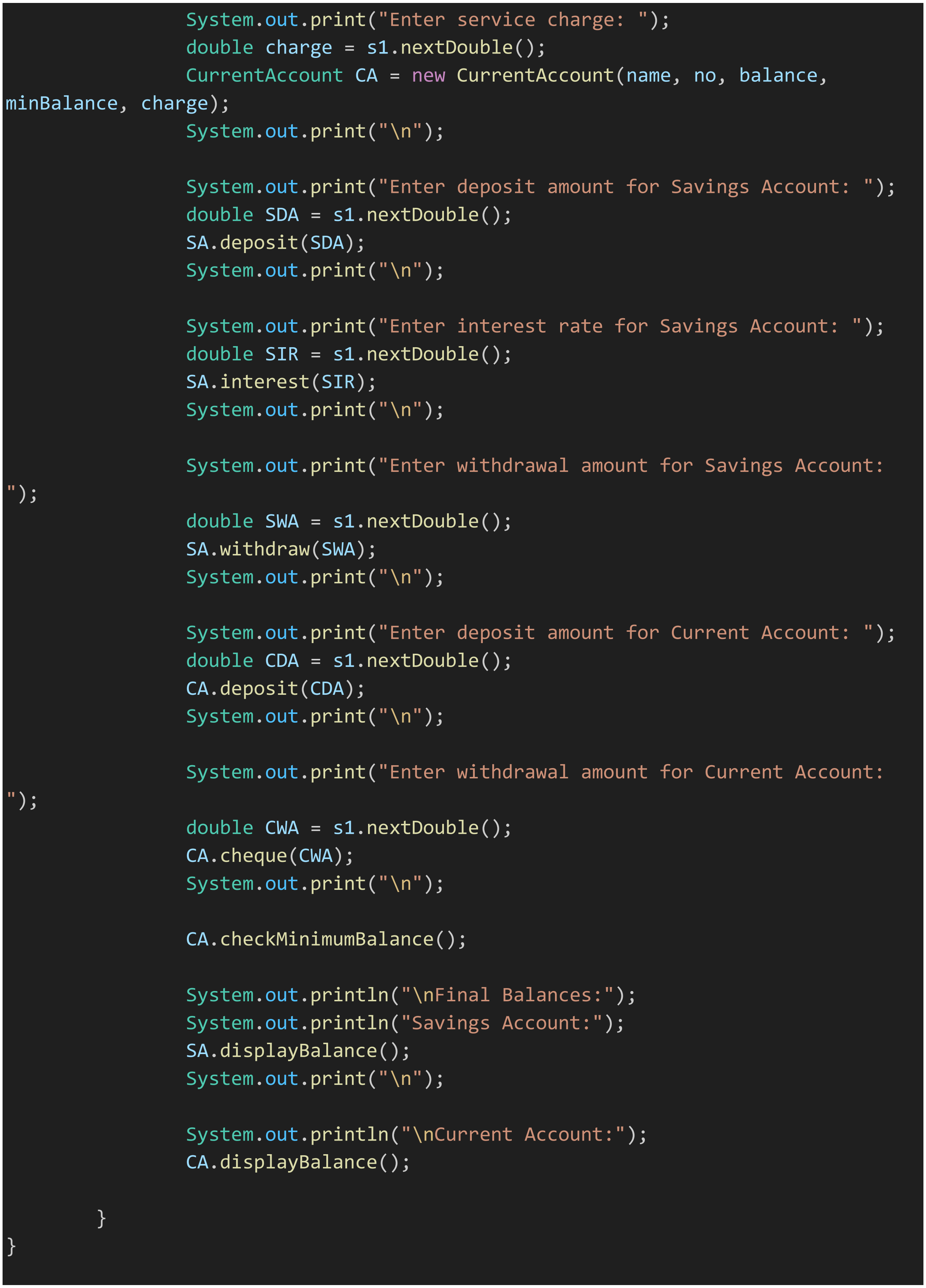
PROGRAM:

```java
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 = balance + amount;
                System.out.println("Deposit of " + amount + " was successful.
Balance: " + balance);
        }

        void displayBalance() {
                System.out.println("\nAccount Number: " + accountNumber +
"\nCustomer Name: " + customerName
```

```
                                + "\nAccount Type: " + accountType + "\nBalance: 
" + balance);
        }

}

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 = balance + interest;
                System.out.println("Interest computed and deposited. Updated 
balance: " + balance);
        }

        void withdraw(double amount) {

                if (amount <= balance) {
                        balance = balance - amount;
                        System.out.println("Withdrawal of " + amount + " 
successful. Updated balance: " + balance);
                }

                else {
                        System.out.println("Insufficient funds. Withdrawal 
failed.");
                }

        }

}

class CurrentAccount extends Account {
        double minimumBalance;
        double serviceCharge;

        CurrentAccount(String customerName, long accountNumber, double balance, 
double minimumBalance,
                        double serviceCharge) {
                super(customerName, accountNumber, "Current", balance);
                this.minimumBalance = minimumBalance;
                this.serviceCharge = serviceCharge;
        }
```

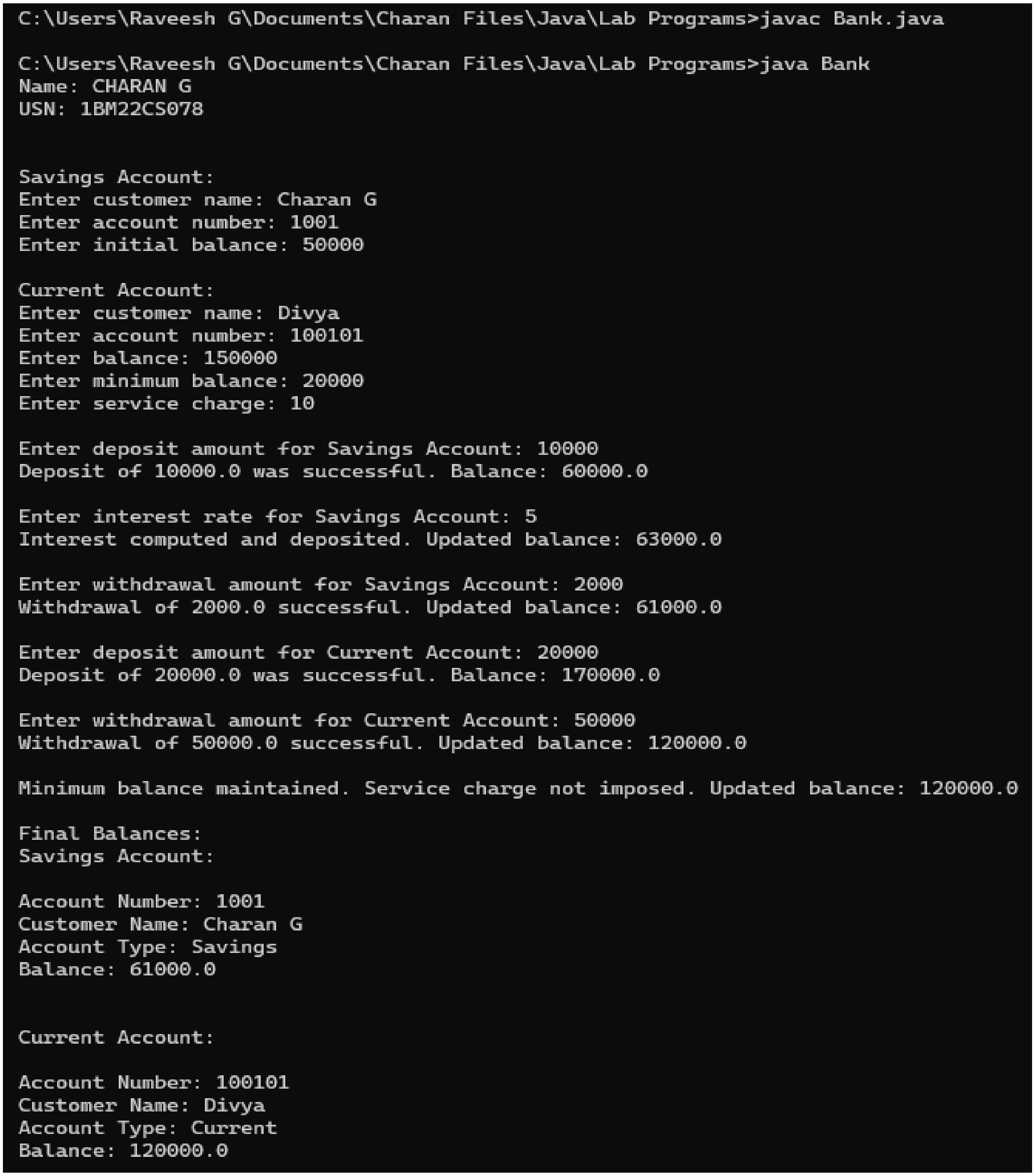


```
        void checkMinimumBalance() {
                if (balance < minimumBalance) {
                        balance = balance - serviceCharge;
                        System.out.println("Minimum balance not maintained.
Service charge imposed. Updated balance: "
                                 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

        void cheque(double amount) {

                balance = balance - amount;
                System.out.println("Withdrawal of " + amount + " successful.
Updated balance: " + balance);
        }

}

public class Bank {
        public static void main(String[] args) {
                System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");
                Scanner s1 = new Scanner(System.in);

                System.out.println("Savings Account: ");
                System.out.print("Enter customer name: ");
                String name = s1.nextLine();
                System.out.print("Enter account number: ");
                long no = s1.nextLong();
                System.out.print("Enter initial balance: ");
                double balance = s1.nextDouble();
                SavingsAccount SA = new SavingsAccount(name, no, balance);
                System.out.print("\n");

                System.out.println("Current Account: ");
                System.out.print("Enter customer name: ");
                name = s1.next();
                System.out.print("Enter account number: ");
                no = s1.nextLong();
                System.out.print("Enter balance: ");
                balance = s1.nextDouble();
                System.out.print("Enter minimum balance: ");
                double minBalance = s1.nextDouble();
```



```
                System.out.print("Enter service charge: ");
                double charge = s1.nextDouble();
                CurrentAccount CA = new CurrentAccount(name, no, balance, 
minBalance, charge);
                System.out.print("\n");

                System.out.print("Enter deposit amount for Savings Account: ");
                double SDA = s1.nextDouble();
                SA.deposit(SDA);
                System.out.print("\n");

                System.out.print("Enter interest rate for Savings Account: ");
                double SIR = s1.nextDouble();
                SA.interest(SIR);
                System.out.print("\n");

                System.out.print("Enter withdrawal amount for Savings Account: 
");
                double SWA = s1.nextDouble();
                SA.withdraw(SWA);
                System.out.print("\n");

                System.out.print("Enter deposit amount for Current Account: ");
                double CDA = s1.nextDouble();
                CA.deposit(CDA);
                System.out.print("\n");

                System.out.print("Enter withdrawal amount for Current Account: 
");
                double CWA = s1.nextDouble();
                CA.cheque(CWA);
                System.out.print("\n");

                CA.checkMinimumBalance();

                System.out.println("\nFinal Balances:");
                System.out.println("Savings Account:");
                SA.displayBalance();
                System.out.print("\n");

                System.out.println("\nCurrent Account:");
                CA.displayBalance();

        }
}
```

OUTPUT:



```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>javac Bank.java

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java Bank
Name: CHARAN G
USN: 1BM22CS078


Savings Account:
Enter customer name: Charan G
Enter account number: 1001
Enter initial balance: 50000

Current Account:
Enter customer name: Divya
Enter account number: 100101
Enter balance: 150000
Enter minimum balance: 20000
Enter service charge: 10

Enter deposit amount for Savings Account: 10000
Deposit of 10000.0 was successful. Balance: 60000.0

Enter interest rate for Savings Account: 5
Interest computed and deposited. Updated balance: 63000.0

Enter withdrawal amount for Savings Account: 2000
Withdrawal of 2000.0 successful. Updated balance: 61000.0

Enter deposit amount for Current Account: 20000
Deposit of 20000.0 was successful. Balance: 170000.0

Enter withdrawal amount for Current Account: 50000
Withdrawal of 50000.0 successful. Updated balance: 120000.0

Minimum balance maintained. Service charge not imposed. Updated balance: 120000.0

Final Balances:
Savings Account:

Account Number: 1001
Customer Name: Charan G
Account Type: Savings
Balance: 61000.0


Current Account:

Account Number: 100101
Customer Name: Divya
Account Type: Current
Balance: 120000.0
```

PROGRAM – 6 – CIE AND SEE PACKAGES OF ‘N’ STUDENTS

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.

PROGRAM:

In CIE folder:

```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;
    }
}
```

```java
package CIE;
import CIE.Student;

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;
    }
}
```

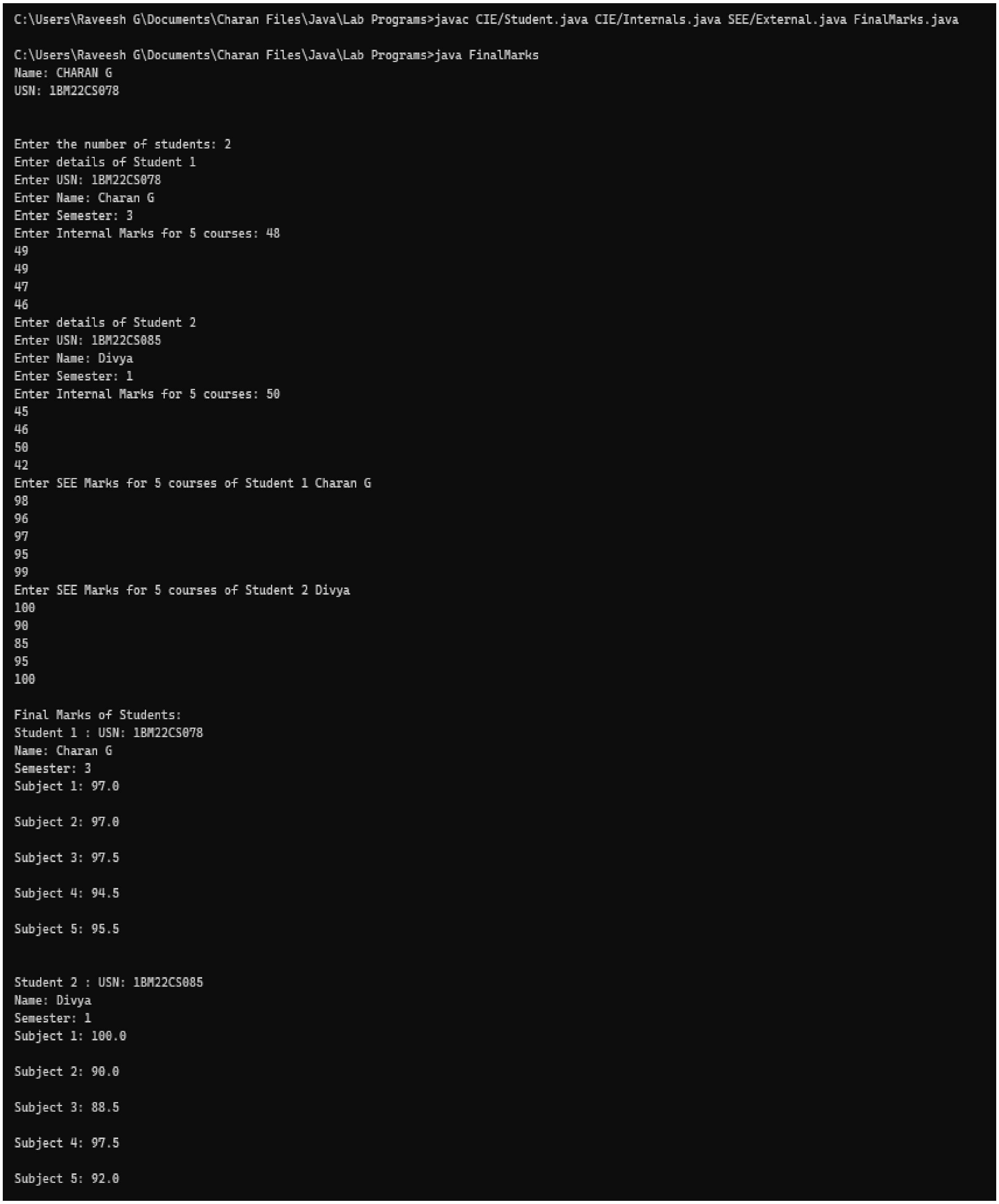
In SEE folder:

```
package SEE;

import CIE.*;

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;
    }
}
```

Outside both folders:

```
import CIE.Internals;
import SEE.External;
import java.util.Scanner;

public class FinalMarks {
    public static void main(String[] args) {
        System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");
        Scanner input = new Scanner(System.in);

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

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

        // Input Internal Marks
        for (int i = 0; i < n; i++) {
            System.out.println("Enter details of Student " + (i + 1));
            System.out.print("Enter USN: ");
            String usn = input.nextLine();
            System.out.print("Enter Name: ");
            String name = input.nextLine();
//input.nextLine();

            System.out.print("Enter Semester: ");
            int sem = input.nextInt();
            input.nextLine();
            double[] internalMarks = new double[5];
            System.out.print("Enter Internal Marks for 5 courses: ");
            for (int j = 0; j < 5; j++) {
                internalMarks[j] = input.nextDouble();
            }
input.nextLine();

            internals[i] = new Internals(usn, name, sem, internalMarks);
        }

        // Input SEE Marks
        for (int i = 0; i < n; i++) {
            System.out.println("Enter SEE Marks for 5 courses of Student " +
(i+1) + " " + internals[i].name);
            String usn = internals[i].usn;
            String name = internals[i].name;
            int sem = internals[i].sem;
            double[] seeMarks = new double[5];

            for (int j = 0; j < 5; j++) {
                seeMarks[j] = input.nextDouble();
            }

            externals[i] = new External(usn, name, sem, seeMarks);
        }

        // Display Final Marks
        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:

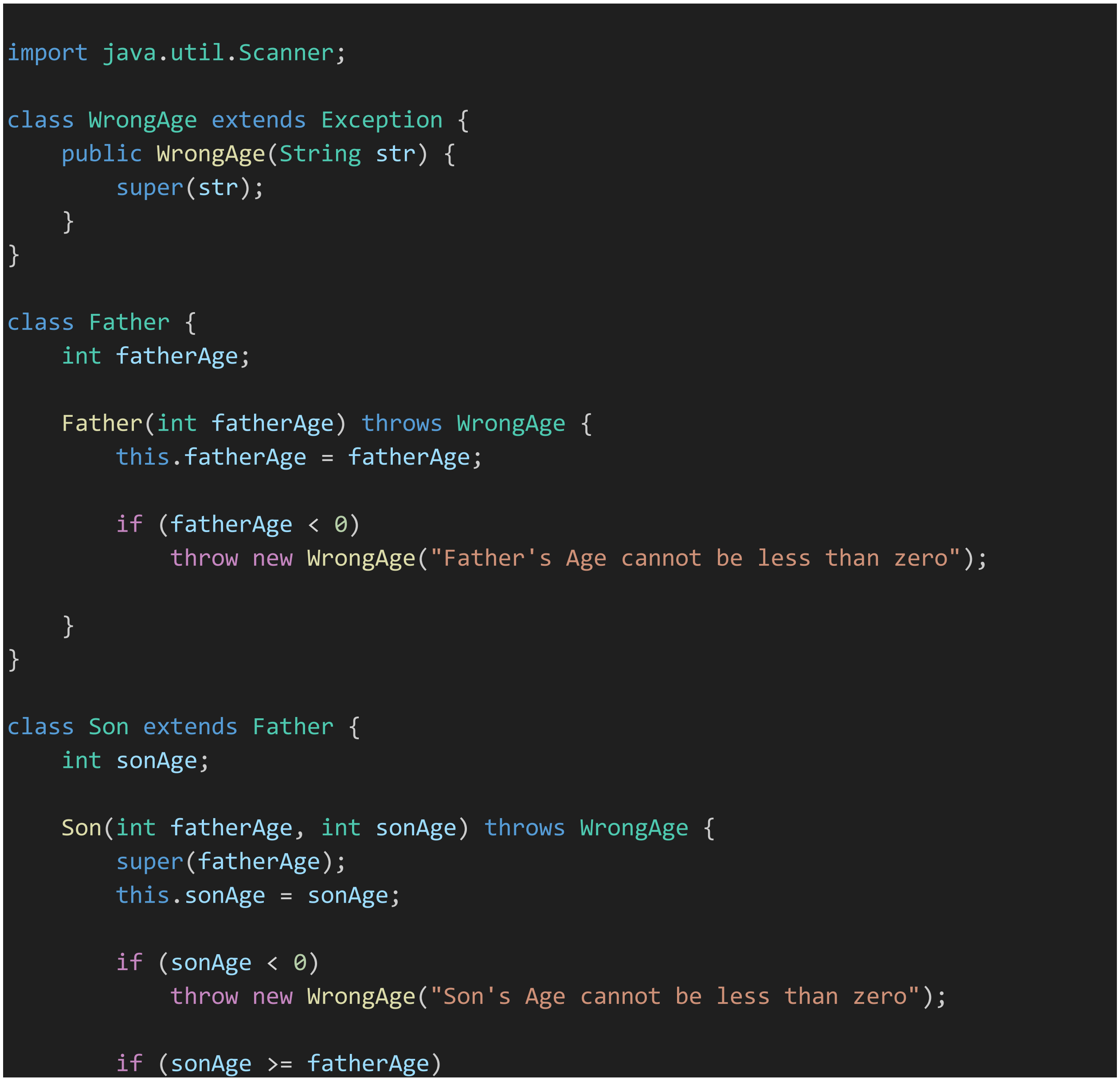


```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>javac CIE/Student.java CIE/Internals.java SEE/External.java FinalMarks.java

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java FinalMarks
Name: CHARAN G
USN: 1BM22CS078


Enter the number of students: 2
Enter details of Student 1
Enter USN: 1BM22CS078
Enter Name: Charan G
Enter Semester: 3
Enter Internal Marks for 5 courses: 48
49
49
47
46
Enter details of Student 2
Enter USN: 1BM22CS085
Enter Name: Divya
Enter Semester: 1
Enter Internal Marks for 5 courses: 50
45
46
50
42
Enter SEE Marks for 5 courses of Student 1 Charan G
98
96
97
95
99
Enter SEE Marks for 5 courses of Student 2 Divya
100
90
85
95
100

Final Marks of Students:
Student 1 : USN: 1BM22CS078
Name: Charan G
Semester: 3
Subject 1: 97.0

Subject 2: 97.0

Subject 3: 97.5

Subject 4: 94.5

Subject 5: 95.5


Student 2 : USN: 1BM22CS085
Name: Divya
Semester: 1
Subject 1: 100.0

Subject 2: 90.0

Subject 3: 88.5

Subject 4: 97.5

Subject 5: 92.0
```

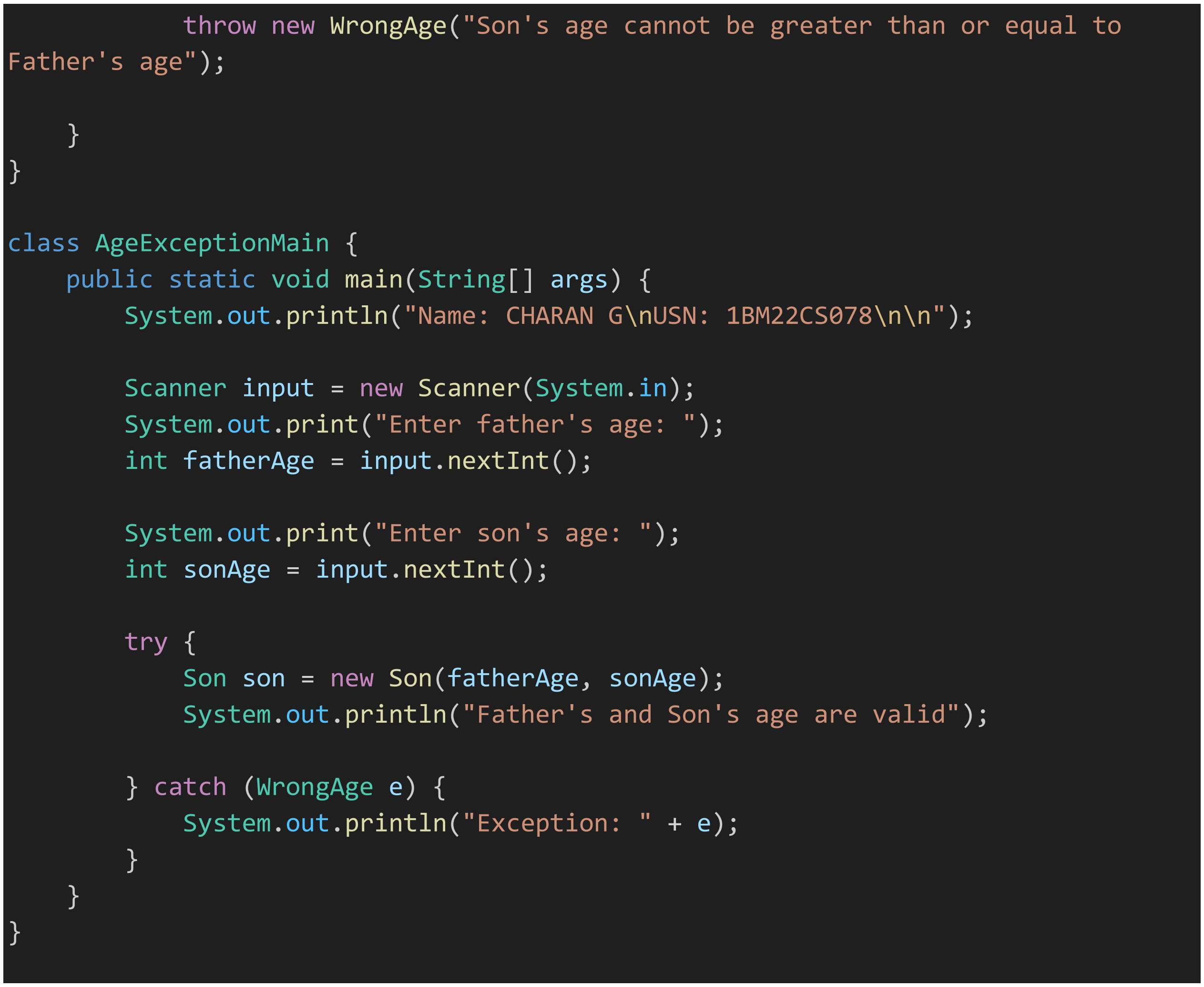
PROGRAM – 7 – EXCEPTION IN INHERITANCE TREE

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=father’s age.

PROGRAM:

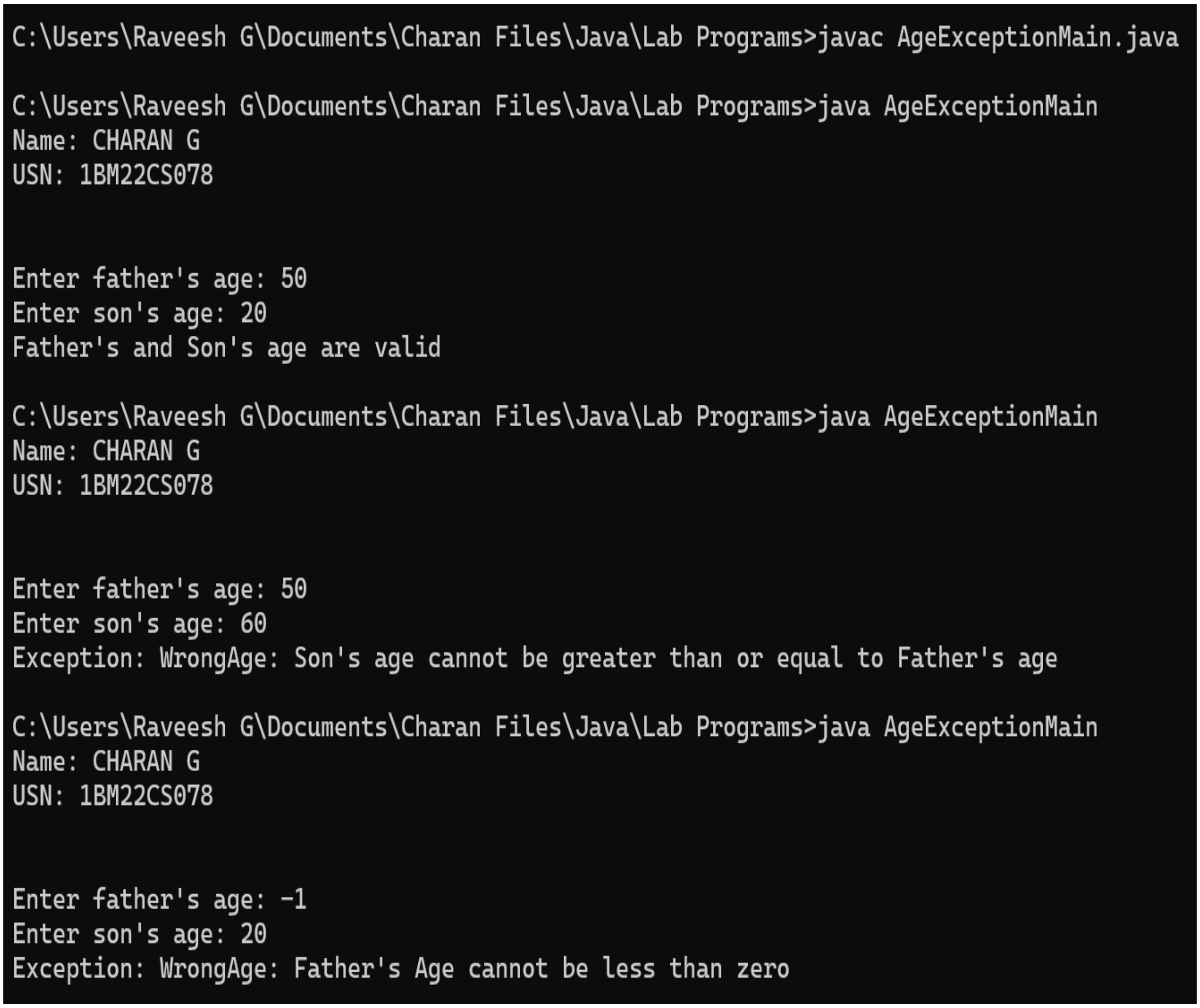


```
import java.util.Scanner;

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

class Father {
    int fatherAge;

    Father(int fatherAge) throws WrongAge {
        this.fatherAge = fatherAge;

        if (fatherAge < 0)
            throw new WrongAge("Father's Age cannot be less than zero");

    }
}

class Son extends Father {
    int sonAge;

    Son(int fatherAge, int sonAge) throws WrongAge {
        super(fatherAge);
        this.sonAge = sonAge;

        if (sonAge < 0)
            throw new WrongAge("Son's Age cannot be less than zero");

        if (sonAge >= fatherAge)
```



```
            throw new WrongAge("Son's age cannot be greater than or equal to 
Father's age");

    }
}

class AgeExceptionMain {
    public static void main(String[] args) {
        System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");

        Scanner input = new Scanner(System.in);
        System.out.print("Enter father's age: ");
        int fatherAge = input.nextInt();

        System.out.print("Enter son's age: ");
        int sonAge = input.nextInt();

        try {
            Son son = new Son(fatherAge, sonAge);
            System.out.println("Father's and Son's age are valid");

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

OUTPUT:

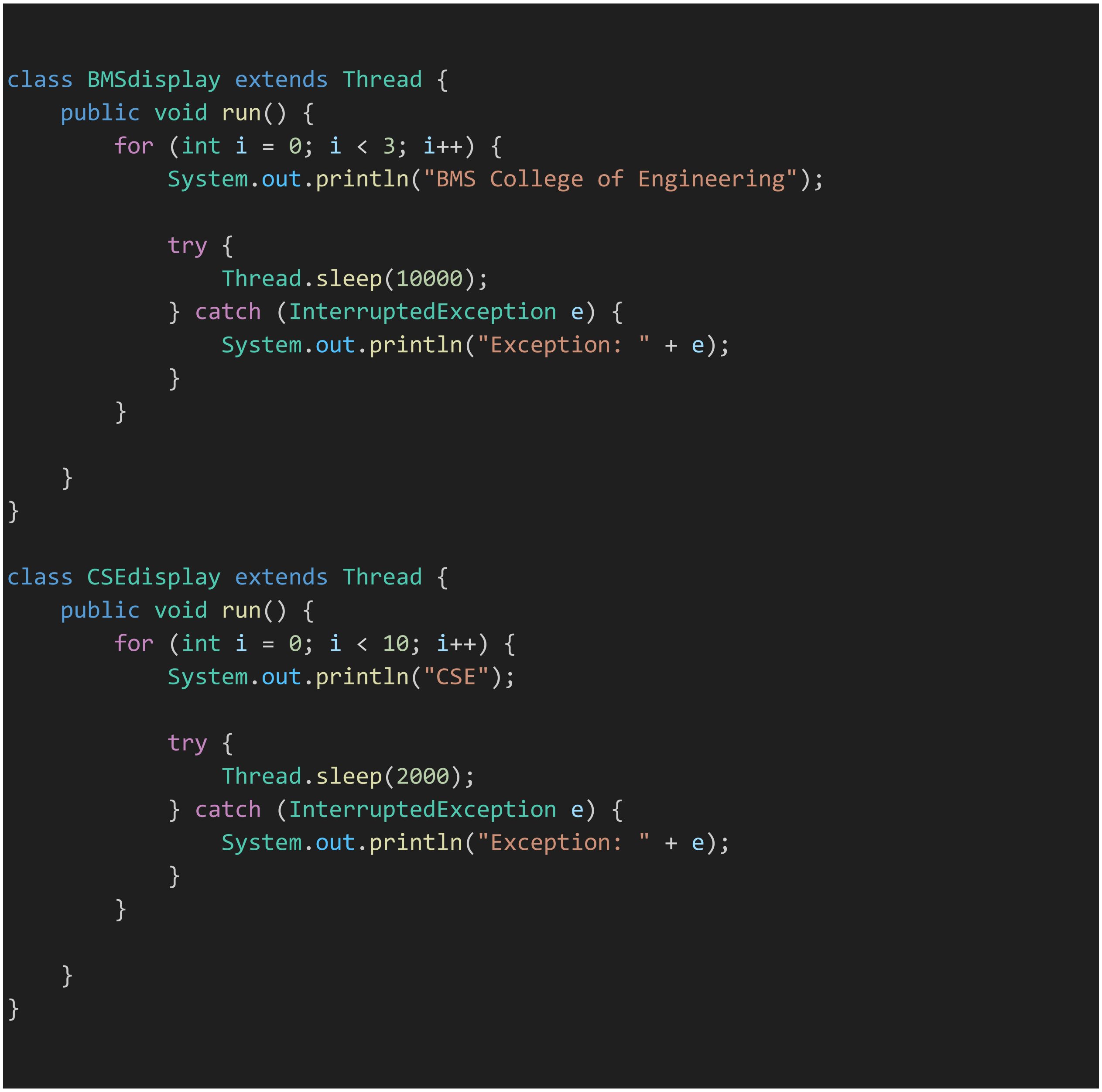


```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>javac AgeExceptionMain.java

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java AgeExceptionMain
Name: CHARAN G
USN: 1BM22CS078


Enter father's age: 50
Enter son's age: 20
Father's and Son's age are valid

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java AgeExceptionMain
Name: CHARAN G
USN: 1BM22CS078


Enter father's age: 50
Enter son's age: 60
Exception: WrongAge: Son's age cannot be greater than or equal to Father's age

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java AgeExceptionMain
Name: CHARAN G
USN: 1BM22CS078


Enter father's age: -1
Enter son's age: 20
Exception: WrongAge: Father's Age cannot be less than zero
```

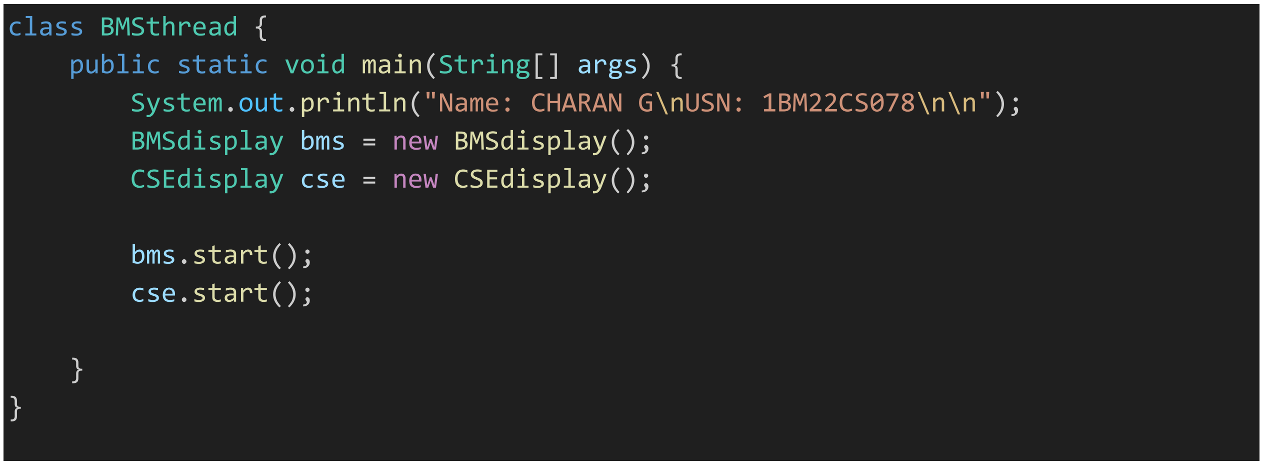
PROGRAM – 8 - MULTITHREADING

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.

PROGRAM:

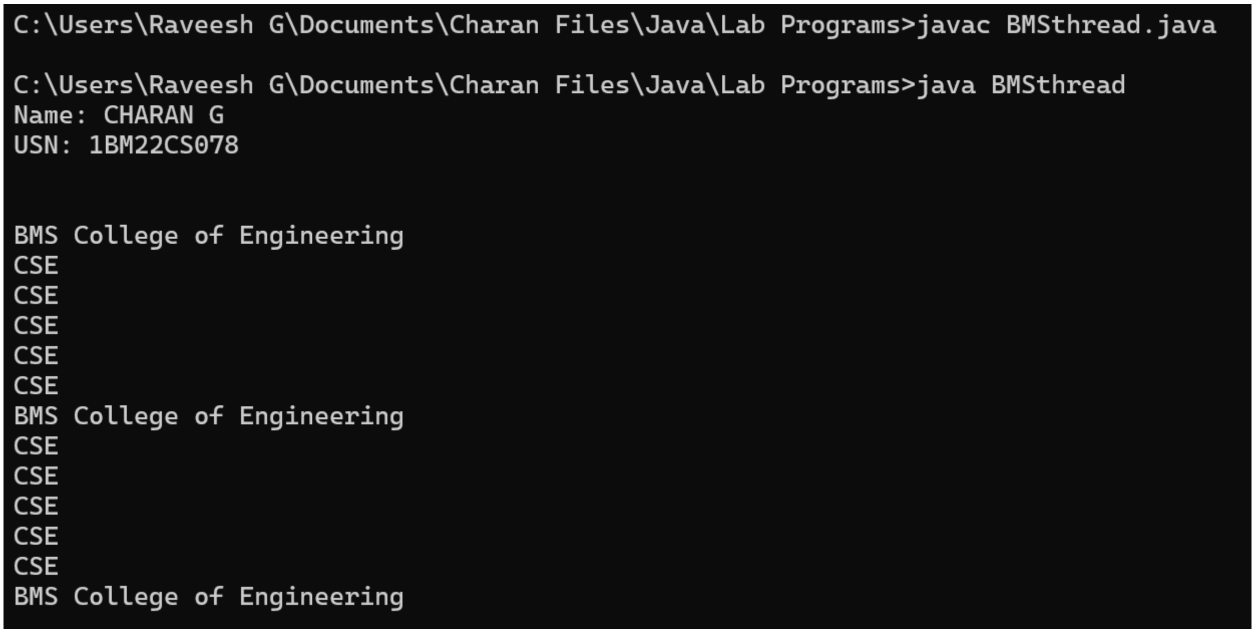


```
class BMSdisplay extends Thread {
    public void run() {
        for (int i = 0; i < 3; i++) {
            System.out.println("BMS College of Engineering");

            try {
                Thread.sleep(10000);
            } catch (InterruptedException e) {
                System.out.println("Exception: " + e);
            }
        }

    }
}

class CSEdisplay extends Thread {
    public void run() {
        for (int i = 0; i < 10; i++) {
            System.out.println("CSE");

            try {
                Thread.sleep(2000);
            } catch (InterruptedException e) {
                System.out.println("Exception: " + e);
            }
        }

    }
}
```



```java
class BMSthread {
    public static void main(String[] args) {
        System.out.println("Name: CHARAN G\nUSN: 1BM22CS078\n\n");
        BMSdisplay bms = new BMSdisplay();
        CSEdisplay cse = new CSEdisplay();

        bms.start();
        cse.start();

    }
}
```

OUTPUT:



```
C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>javac BMSthread.java

C:\Users\Raveesh G\Documents\Charan Files\Java\Lab Programs>java BMSthread
Name: CHARAN G
USN: 1BM22CS078


BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
```