B.M.S. College of Engineering

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)
Bull Temple Road, Basavanagudi, Bengaluru – 560019

Department of
Computer Science & Engineering (CSE)

Lab Programs Observation

Course Title: Object Oriented Java Programming

Course Code: 23CS3PCOOJ

BY
E Bhavya Hanwitha

B.M.S. College of Engineering

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)
Bull Temple Road, Basavanagudi, Bengaluru – 560019

Department of
Computer Science & Engineering (CSE)

CERTIFICATE

This is to certify that the report on "**Java Lab Programs**" has been carried out by **Bhavya Hanwitha** bearing USN **1BM22CS095** as a part of AAT for the course **Object Oriented Java Programming** with course code **23CS3PCOOJ**, Computer Science and Engineering from Visvesvaraya Technological University, Belgaum during the year 2023–24. It is certified that all corrections/suggestions indicated for Internal Assessments have been incorporated in the report**.**

Bhavya hanwitha **Shravya AR**

1BM22CS095 Assistant Professor

Department of CSE

BMSCE, Bengaluru-19

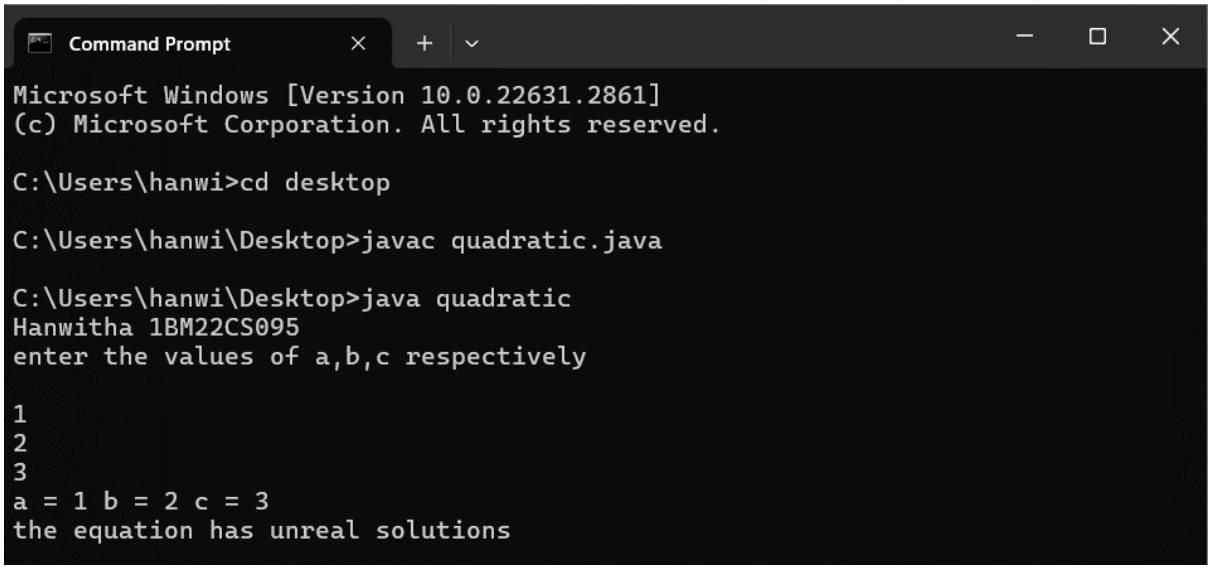
Table of contents

LAB PROGRAM 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.

CODE

```
import java.util.Scanner;
import java.lang.Math;
class quadratic
{public static void main(String XX[])

{ System.out.println("Bhavya Hanwitha 1BM22CS095");
int a,b,c;
System.out.println("enter the values of a,b,c respectively\n"); Scanner s1= new
Scanner(System.in);

a = s1.nextInt();
b = s1.nextInt();
c = s1.nextInt();
double d= b*b - 4*a*c ;
System.out.println("a = " + a +" b = " + b +" c = " + c); if(a==0) {System.out.println("not a
quadratic equation");} else if( d>0)

{
System.out.println("the equation has two real and different solutions"); double r1=(-b +
Math.sqrt(d))/(2*a);
double r2=(-b - Math.sqrt(d))/(2*a);
System.out.println("r1 = " + r1);
System.out.println("r2 = " + r2);

}

else if(d==0) {

System.out.println("the equation has real and equal solutions"); double r1= -b/(2*a);
double r2= -b/(2*a);
System.out.println("r1 = " + r1);

System.out.println("r2 = " + r2); }

else if(d<0) {

System.out.println("the equation has unreal solutions");

}

 }

}
```


Command Prompt
Microsoft Windows [Version 10.0.22631.2861]
(c) Microsoft Corporation. All rights reserved.
C:\Users\hanwi>cd desktop
C:\Users\hanwi\Desktop>javac quadratic.java
C:\Users\hanwi\Desktop>java quadratic
Hanwitha 1BM22CS095
enter the values of a,b,c respectively
1
2
3
a = 1 b = 2 c = 3
the equation has unreal solutions

Lab program 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 percentage of a student.

CODE

```
import java.util.Scanner; class Student{

String usn;

String name;
int marks[]= new int[6];

void Details()

{
Scanner s=new Scanner(System.in); System.out.println("Enter USN");
usn=s.next();
System.out.println("Enter Name"); name=s.next();
System.out.println("Enter marks for 6 subjects:"); for(int i = 0; i < 6; i++)
{
System.out.print("Subject " + (i + 1) + ": "); marks[i]=s.nextInt();
}

}
double percentage() {

int total=0;
for(int i=0;i<6;i++) {
total+=marks[i];
}
double p=total/6; return p;

}
void display() {

System.out.println("\nStudent Details:"); System.out.println("USN: " + usn);
System.out.println("Name: " + name); System.out.println("Marks:");

for(int i = 0; i < 6; i++)

{
System.out.println("Subject " + (i + 1) + ": " + marks[i]);

}

System.out.println("Percentage: "+ percentage() + "%"); }

}
class studpercent
```

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

System.out.println("Hanwitha 1BM22CS095");

System.out.print("Enter the number of students: "); int n = s.nextInt();
Student[] students = new Student[n];
for (int i = 0; i < n; i++)

{
students[i] = new Student();
System.out.println("\nEnter details for Student " + (i + 1) + ":"); students[i].Details();
}
for (Student student : students)
{
student.display();
}
}
}
```

```
C:\Users\hanwi\Desktop>javac spercen.java

C:\Users\hanwi\Desktop>java spercen
Hanwitha 1BM22CS095
Enter the number of students: 1

Enter details for Student 1:
Enter USN
095
Enter Name
han
Enter marks for 6 subjects:
Subject 1: 100
Subject 2: 90
Subject 3: 100
Subject 4: 80
Subject 5: 70
Subject 6: 100

Student Details:
USN: 095
Name: han
Marks:
Subject 1: 100
Subject 2: 90
Subject 3: 100
Subject 4: 80
Subject 5: 70
Subject 6: 100
Percentage: 90.0%
```

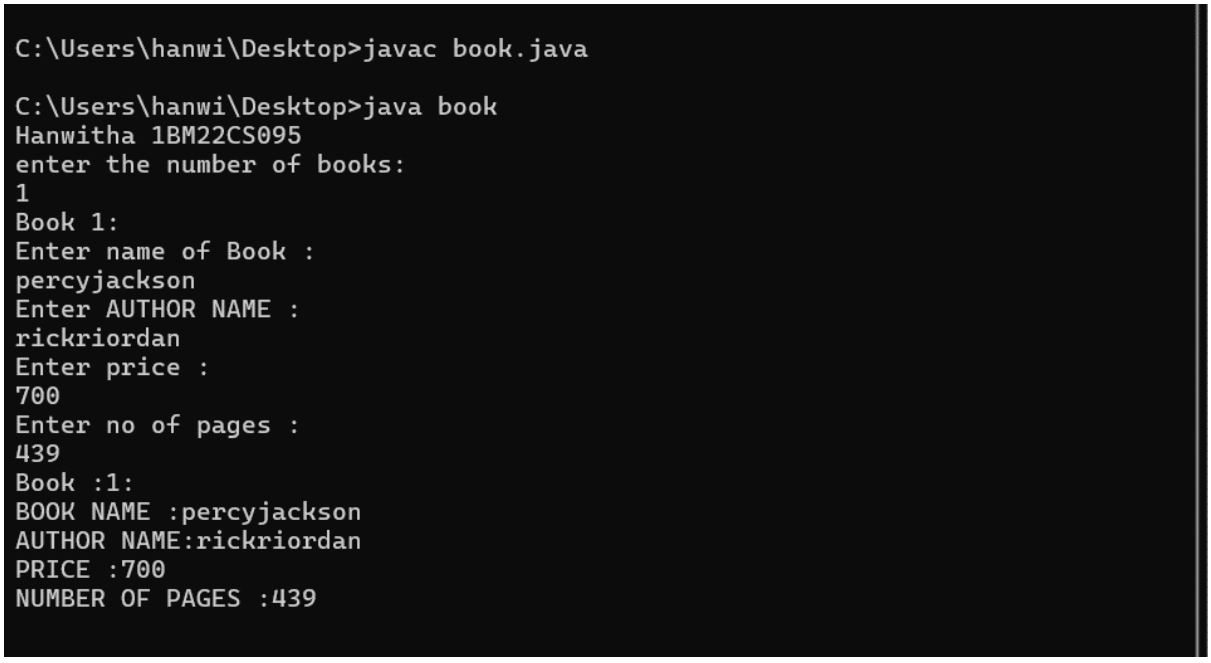
Lab program 3

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

CODE

```
import java.util.Scanner;

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

int numpages;

Books(){}
Books(String name,String author,int price,int numpages)
{
this.name=name;
this.author=author;
this.price=price;
this.numpages=numpages;
}
public String toString()
{
String name,author;
String price,numpages;
name="BOOK NAME :"+this.name+"\n"; author="AUTHOR NAME:"+this.author+"\n";
price="PRICE :"+this.price+"\n";
numpages="NUMBER OF PAGES :"+this.numpages+"\n"; return
name+author+price+numpages;
}
}
class bookdb
{
public static void main(String args[])
{
System.out.println("Hanwitha 1BM22CS095”); Scanner s=new Scanner(System.in);
int n;
String name;
String author;

int price;
int numpages;
System.out.println("enter the number of books:"); n=s.nextInt();
Books b[];
b= new Books[n];
```

```
for (int i=0;i<n;i++)
{
System.out.println("Book "+(i+1)+":"); System.out.println("Enter name of Book :"); name
=s.next();
System.out.println("Enter AUTHOR NAME :"); author =s.next();
System.out.println("Enter price :");
price =s.nextInt();
System.out.println("Enter no of pages :"); numpages =s.nextInt();
b[i]=new Books(name,author,price,numpages);
}
for(int i=0;i<n;i++)
System.out.println("Book :"+(i+1)+":\n"+b[i]);
}
}
```



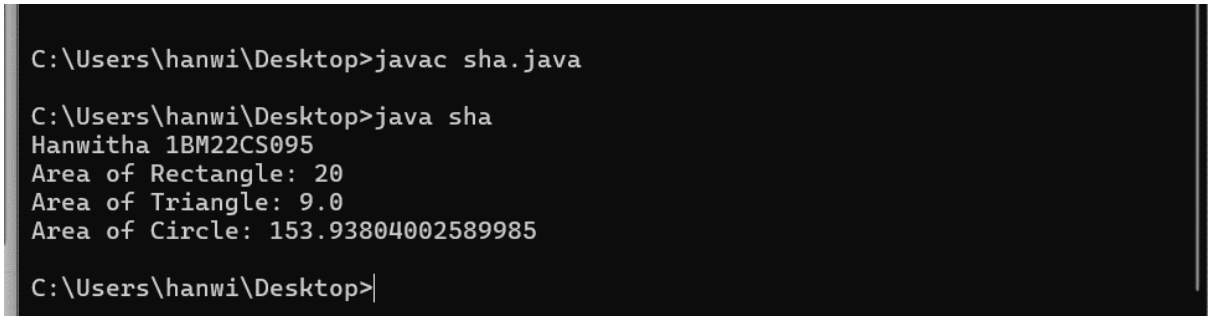
Lab program 4

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

CODE

```
abstract class Shape {
protected int dimension1;
protected int dimension2;
public Shape(int dimension1, int dimension2) { this.dimension1 = dimension1;
this.dimension2 = dimension2;

}
public abstract void printArea();
}
class Rectangle extends Shape {
public Rectangle(int length, int width) { super(length, width);
}
public void printArea() {
int area = dimension1 * dimension2; System.out.println("Area of Rectangle: " + area);
}
}
class Triangle extends Shape {
public Triangle(int base, int height) {
super(base, height);
}
public void printArea() {
double area = 0.5 * dimension1 * dimension2; System.out.println("Area of Triangle: "+ area);
}
}
class Circle extends Shape {
public Circle(int radius) {
super(radius, 0);
}
public void printArea() {
double area = Math.PI * dimension1 * dimension1; System.out.println("Area of Circle: " +
area);
```

```
}
}

public class shapes {
public static void main(String[] args) { System.out.println("Hanwitha 1BM22CS095");
Rectangle rectangle = new Rectangle(4, 5); rectangle.printArea();
Triangle triangle = new Triangle(3, 6); triangle.printArea();
Circle circle = new Circle(7);
circle.printArea();
}
}
```



Lab program 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.

CODE

```
class Bank {
public static void main(String[] args) {
System.out.println("Hanwitha 1BM22CS095");
SavingsAccount savingsAccount = new SavingsAccount("hanwitha", "SA1001");
CurrentAccount currentAccount = new CurrentAccount("Diksha", "CA2002");

savingsAccount.deposit(5000); savingsAccount.displayBalance();
savingsAccount.computeInterest(); savingsAccount.displayBalance();
savingsAccount.withdraw(2000); savingsAccount.displayBalance();

currentAccount.deposit(8000); currentAccount.displayBalance();

currentAccount.withdraw(5000); currentAccount.displayBalance(); }
}

class Account {
protected String customerName;
protected String accountNumber;
protected double balance;
public Account(String customerName, String accountNumber) { this.customerName =
```

```
customerName;
this.accountNumber = accountNumber;
this.balance = 0;
}
public void deposit(double amount) {

balance += amount;
System.out.println("Deposit of " + amount + " successful.");
}
public void displayBalance() {
System.out.println("Account Number:"+ accountNumber + "\nBalance: " + balance); }
}
class SavingsAccount extends Account {
public SavingsAccount(String customerName, String accountNumber) {
super(customerName, accountNumber);
}
public void computeInterest() {
double interestRate = 0.05; // Assuming a 5% interest rate
double interest = balance * interestRate;
balance += interest;
System.out.println("Interest of " + interest + "computed and added to the balance."); }
public void withdraw(double amount) {
if (balance >= amount) {
balance -= amount;
System.out.println("Withdrawal of " + amount + "successful.");
} else {
System.out.println("Insufficient funds for withdrawal.");

}
}
}
class CurrentAccount extends Account {
private double minimumBalance = 1000; // Assuming a minimum balance requirement public
CurrentAccount(String customerName, String accountNumber) { super(customerName,
accountNumber);

}
public void withdraw(double amount) {
if (balance - amount >= minimumBalance) {
balance -= amount;
System.out.println("Withdrawal of $" + amount + "successful.");
} else {
System.out.println("Insufficient funds. Service charge applied."); imposeServiceCharge();
}
}
private void imposeServiceCharge() {
double serviceCharge = 20; // Assuming a service charge of $20 balance -= serviceCharge;

System.out.println("Service charge of $" + serviceCharge + " imposed."); }
}
```

```
C:\Users\hanwi\Desktop>javac Bank.java

C:\Users\hanwi\Desktop>java Bank
Hanwitha 1BM22CS095
Deposit of 5000.0 successful.
Account Number:SA1001
Balance: 5000.0
Interest of 250.0computed and added to the balance.
Account Number:SA1001
Balance: 5250.0
Withdrawal of 2000.0successful.
Account Number:SA1001
Balance: 3250.0
Deposit of 8000.0 successful.
Account Number:CA2002
Balance: 8000.0
Withdrawal of $5000.0successful.
Account Number:CA2002
Balance: 3000.0
```

Lab program 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.

CODE

Internals.java
```
package cie;
public class Internals extend} s Student{
public int[] marks=new int[5];
```

External.java
```
package see;
import cie.Student;
public class External extends Student{ public int[] seemarks=new int[5];

}
```

Student.java package cie;
```
public class Student{ public String name; public String usn; public int sem;
}
```

Main.java

```
import cie.Internals; import see.External; import java.util.Scanner;

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

System.out.println("Hanwitha  1BM22CS095”); System.out.println("Enter the number of students"); Scanner input = new Scanner(System.in);

int n = input.nextInt();

Internals[] s1 = new Internals[n]; External[] s2 = new External[n]; int[] finalcie = new int[n]; int[] finalsee = new int[n];

for (int i = 0; i < n; i++) {
s1[i] = new Internals();
System.out.println("Enter the name");
s1[i].name = input.next();
System.out.println("Enter the usn");
s1[i].usn = input.next();
System.out.println("Enter the sem");
s1[i].sem = input.nextInt();
System.out.println("Enter the internal (cie) marks of 5 subjects"); for (int j = 0; j < 5; j++) {
```

```
s1[i].marks[j] = input.nextInt();

finalcie[i] += s1[i].marks[j]; }

}

for (int i = 0; i < n; i++) {
s2[i] = new External();
System.out.println("Enter the name");
s2[i].name = input.next();
System.out.println("Enter the usn");
s2[i].usn = input.next();
System.out.println("Enter the sem");
s2[i].sem = input.nextInt();
System.out.println("Enter the external(see) marks of 5 subjects"); for (int j = 0; j < 5; j++) {

s2[i].seemarks[j] = input.nextInt();

finalsee[i] += s2[i].seemarks[j]; }

}

System.out.println("Final marks:"); for (int i = 0; i < n; i++) {

System.out.println("Name: " + s1[i].name + " USN: " + s1[i].usn + " Sem: " + s1[i].sem);
System.out.println("Internal marks: " + finalcie[i]);
System.out.println("External marks: " + finalsee[i]);
System.out.println("Total marks: " + (finalcie[i] + finalsee[i]));

}
```

```
Enter the number of students
1
Enter the name
priya
Enter the usn
123
Enter the sem
3
Enter the internal (cie) marks of 5 subjects
40
46
38
48
42
Enter the name
priya
Enter the usn
123
Enter the sem
3
Enter the external(see) marks of 5 subjects
40
50
45
43
39
Final marks:
Name: priya USN: 123 Sem: 3
Internal marks: 214
External marks: 217
Total marks: 431
```

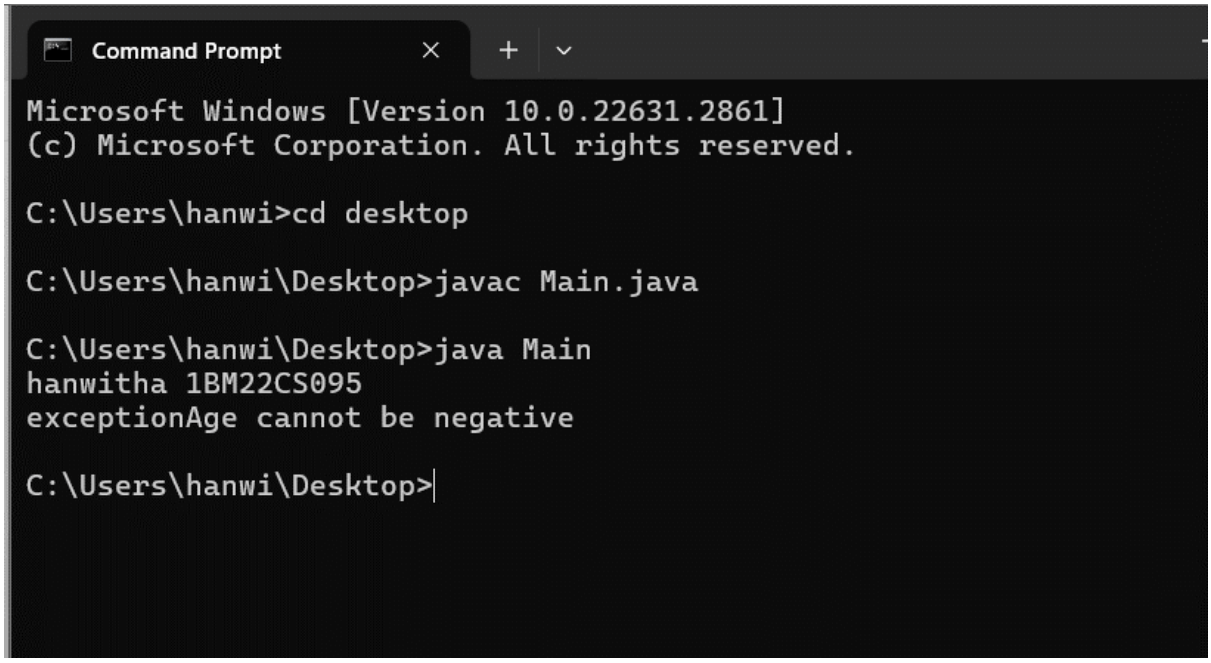
Lab program 7

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.

CODE

```
class WrongAge extends Exception { public WrongAge(String message) {

super(message); }

}
class Father {

int age;

public Father(int age) throws WrongAge { if (age < 0) {

throw new WrongAge("Age cannot be negative"); }

this.age = age; }

public int getAge() { return age;

} }

class Son extends Father { int sonAge;

public Son(int fatherAge, int sonAge) throws WrongAge {

super(fatherAge);
if (sonAge >= fatherAge) {

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

this.sonAge = sonAge; }

public int getSonAge() { return sonAge;
```

```
} }

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

System.out.println("hanwitha 1BM22CS095"); try {

Father father = new Father(-1);

 System.out.println("Father's age: " + father.getAge());

Son son = new Son(-1, 20);

System.out.println("Son's age: " + son.getSonAge()); } catch (WrongAge e) {

System.out.println("Exception caught: " + e.getMessage()); }

} }
```



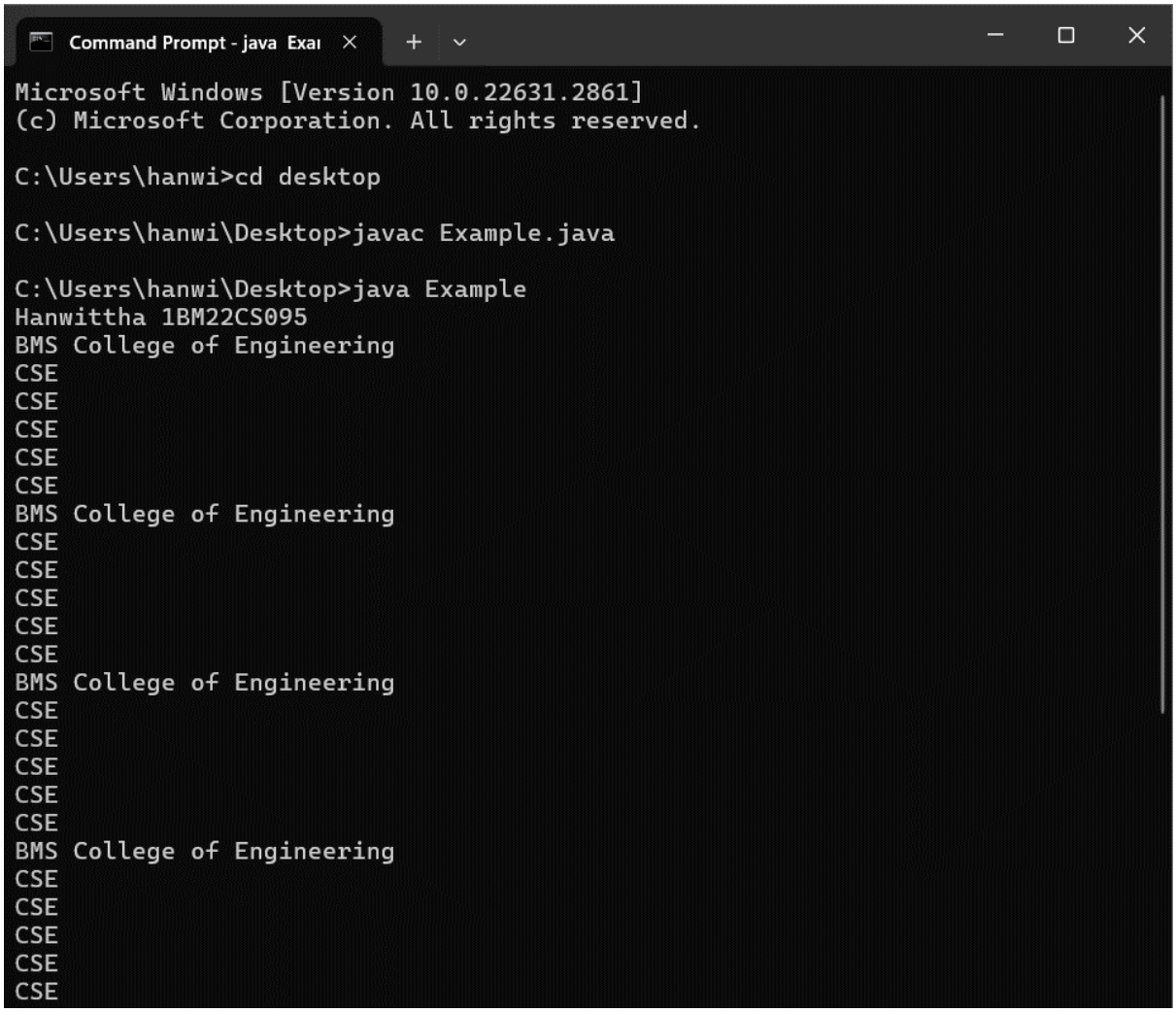
Lab program 8

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.

CODE

```
class DispMessage extends Thread { String message;

int interval;

public DispMessage(String message, int interval) { this.message = message;
this.interval = interval;

}

public void run() { while (true) {

System.out.println(message); try {

Thread.sleep(interval); // Sleep for the specified interval } catch (InterruptedException e) {

e.printStackTrace(); }

} }
```

```
}
public class Example{

public static void main(String[] args) {
System.out.println("Hanwitha 1BM22CS095");
DispMessage bmsThread = new DispMessage("BMS College of Engineering", 10000);
DispMessage cseThread = new DispMessage("CSE", 2000);

bmsThread.start();

cseThread.start(); }

}
```



THANK YOU