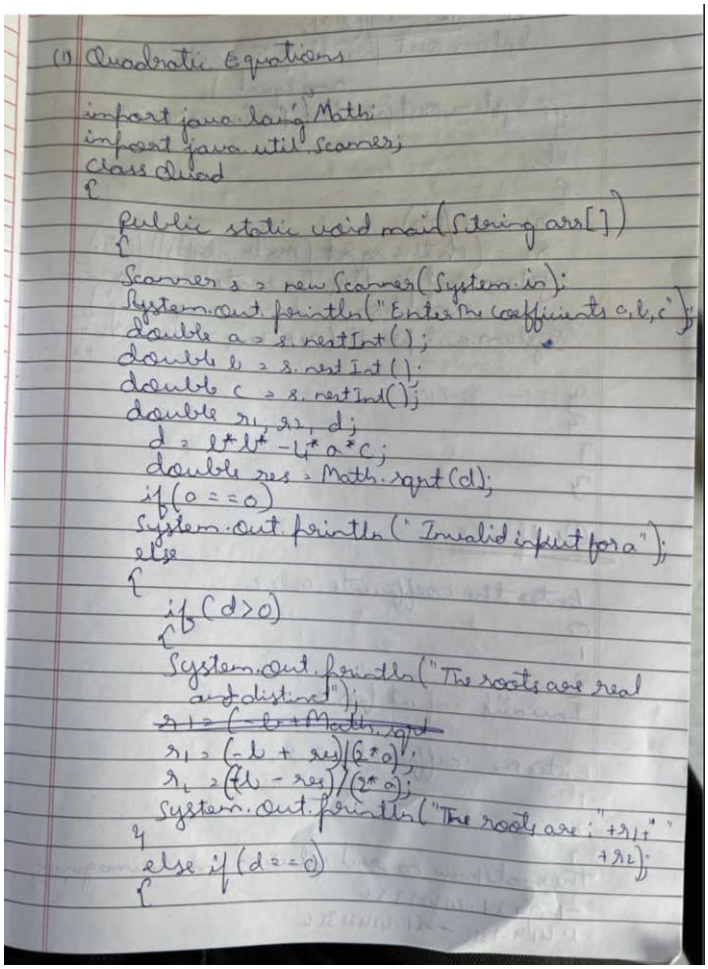
LAB PROGRAM-1

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.

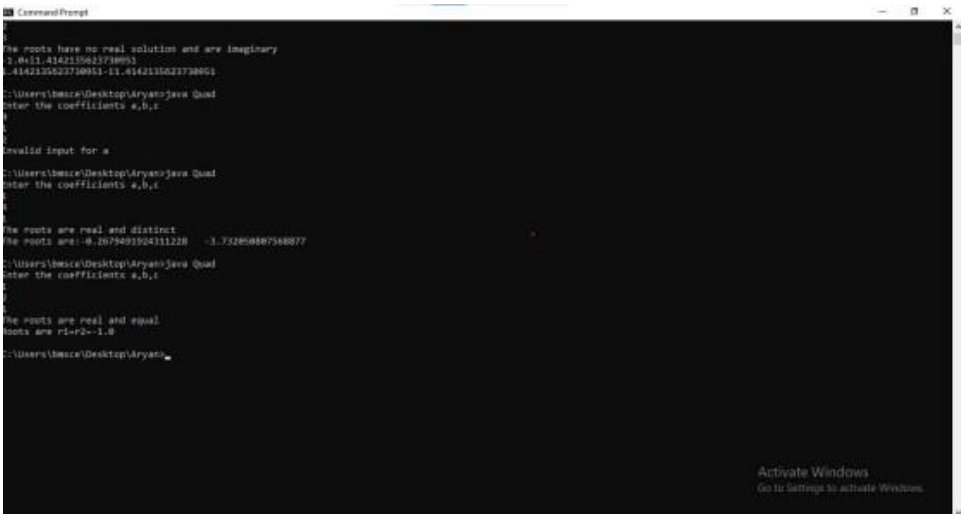
PROGRAM



```
(1) Quadratic Equations

import java.lang.Math;
import java.util.Scanner;
class Quad
{
  public static void main(String args[])
  {
  Scanner s = new Scanner(System.in);
  System.out.println("Enter the coefficients a,b,c");
  double a = s.nextInt();
  double b = s.nextInt();
  double c = s.nextInt();
  double r1, r2, d;
    d = b*b - 4*a*c;
  double res = Math.sqrt(d);
  if(a==0)
  System.out.println("Invalid input for a");
  else
  {
    if(d>0)
    {
    System.out.println("The roots are real
    and distinct");
    r1 = (-b + res)/(2*a);
    r2 = (-b - res)/(2*a);
    System.out.println("The roots are: "+r1+" "
                                          +r2);
  }
  else if(d==0)
  {
```

Enter the coefficients a, b c
1
2
1
The roots are real and equal
Roots are r1 = r2 = -1.0

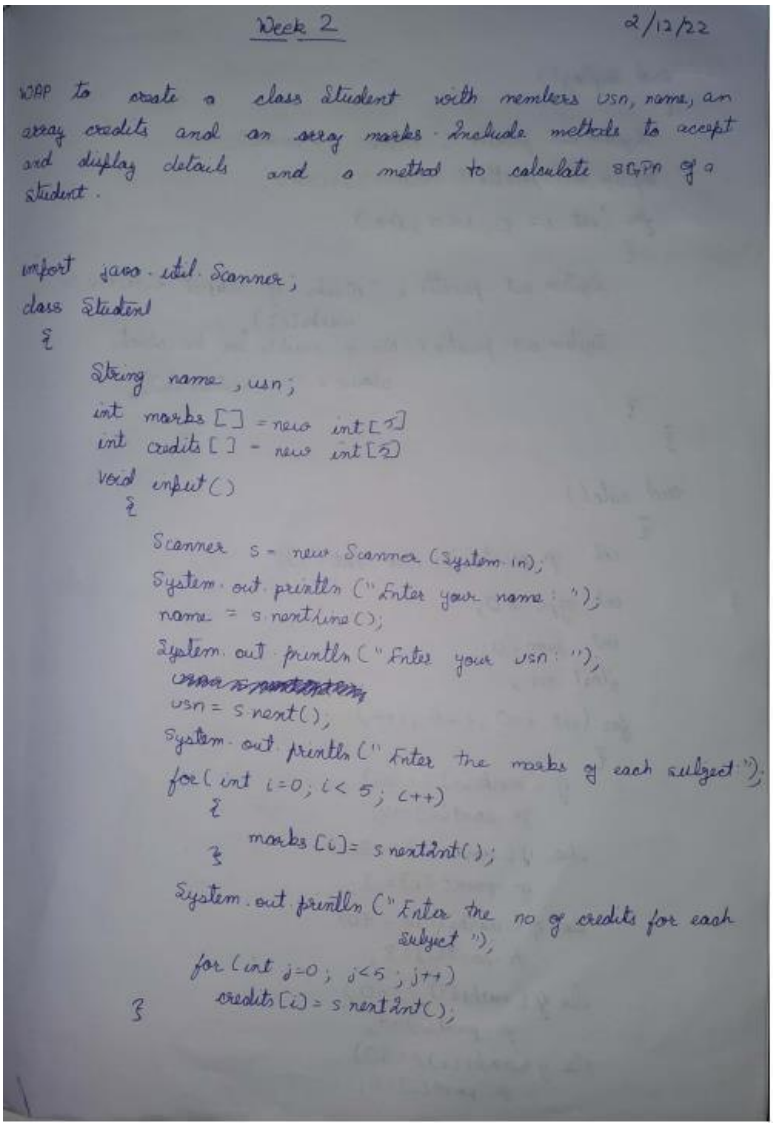
Enter the coefficients a, b, c
1
4
1
The roots are real and distinct
The roots are: 0.2679491924 -3.7320508075



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 SGPA of a student.

PROGRAM

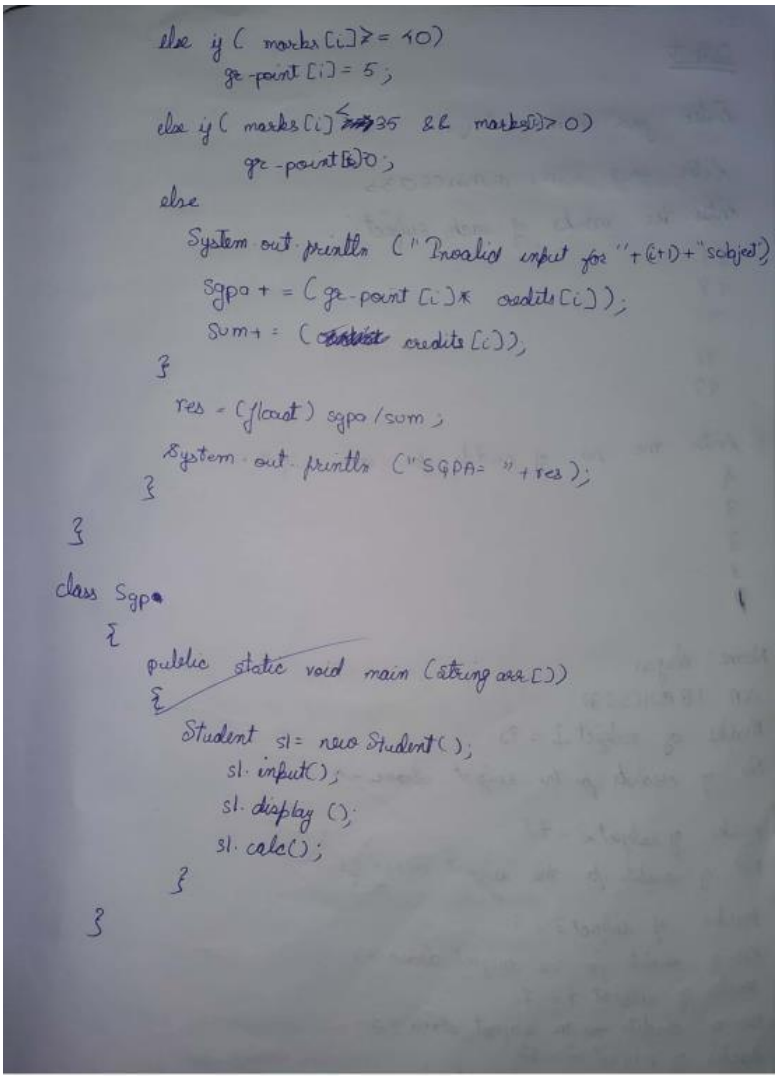


Week 2 2/12/22

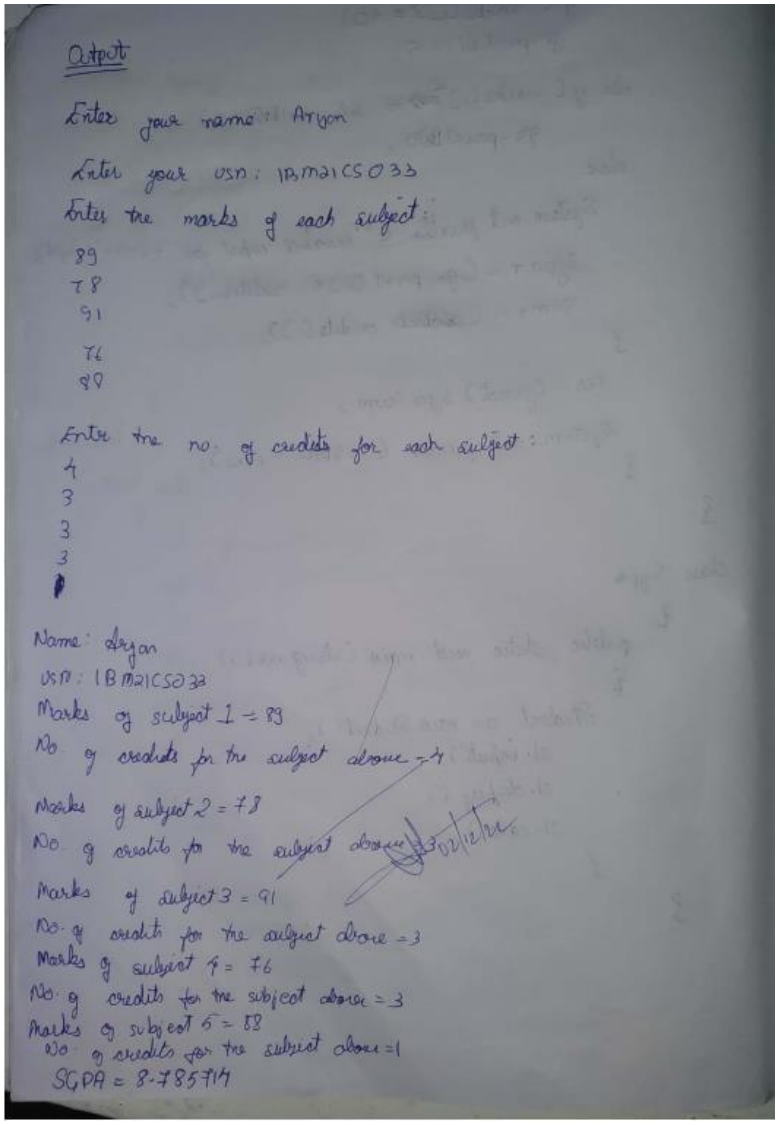
WAP to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

```
import java.util.Scanner;
class Student
{
    String name, usn;
    int marks[] = new int[5];
    int credits[] = new int[5];
    void input()
    {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter your name: ");
        name = s.nextLine();
        System.out.println("Enter your usn: ");
        usn = s.next();
        System.out.println("Enter the marks of each subject");
        for(int i=0; i<5; i++)
        {
            marks[i] = s.nextInt();
        }
        System.out.println("Enter the no of credits for each subject");
        for(int j=0; j<5; j++)
            credits[i] = s.nextInt();
    }
```

```
void display()
{
  System.out.println("Name : "+name);
  System.out.println("USN : ")+usn);
  for (int i= 0; i<5 ; i++)
   {
     System.out.println("Marks of subject "+(i+1)+" = "+
                        marks[i]);
     System.out.println("No. of credits for the subject
                        above = "+credits[i]);
   }
}

void calc()
  {
    int gr_point[] = new int[5];
    int sgpa = 0;
    int sum = 0;
    float res;
    for (int i=0 ; i<5 ; i++)
     {
       if (marks[i] >= 90)
          gr_point[i] =10;
       else if (marks[i]>=80)
          gr_point[i] = 9;
       else if (marks[i]>=70)
          gr_point[i] = 8;
       else if (marks[i] >=60)
          gr_point[i] = 7;
       else if (marks[i]>=50)
          gr_point[i] = 6;
```



```
        else if ( marks[i] >= 40)
            ge-point[i] = 5;
        else if ( marks[i] < 35 && marks[i] > 0)
            ge-point[i] = 0;
        else
          System.out.println ("Invalid input for "+(i+1)+"subject");
          Sgpa += (ge-point[i] * credits[i]);
          sum += (credits[i]);
        }
         res = (float) sgpa / sum;
        System.out.println ("SGPA= "+res);
      }
}

class Sgpa
   {
      public static void main (String args[])
      {
        Student s1 = new Student();
          s1.input();
          s1.display();
          s1.calc();
        }
}
```


Output

Enter your name: Aryan

Enter your usn: 1BM21CS033

Enter the marks of each subject:

89
78
91
76
88

Enter the no. of credits for each subject:

4
3
3
3
1

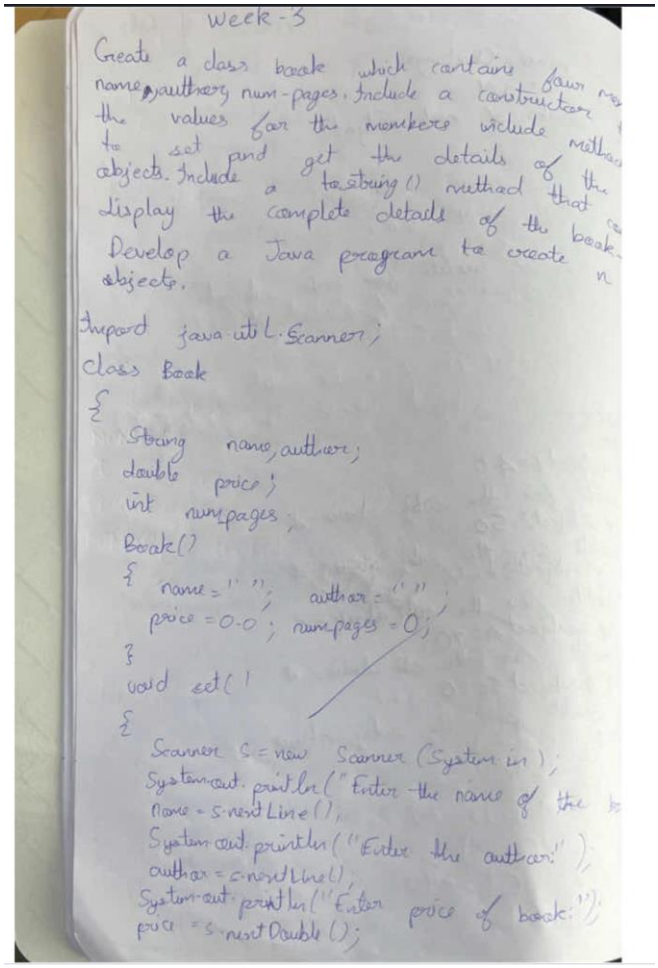
Name: Aryan
USN: 1BM21CS033
Marks of subject 1 = 89
No of credits for the subject above = 4
Marks of subject 2 = 78
No of credits for the subject above = 3
Marks of subject 3 = 91
No. of credits for the subject above = 3
Marks of subject 4 = 76
No. of credits for the subject above = 3
Marks of subject 5 = 88
No. of credits for the subject above = 1
SGPA = 8.785714

OUTPUT

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.

PROGRAM



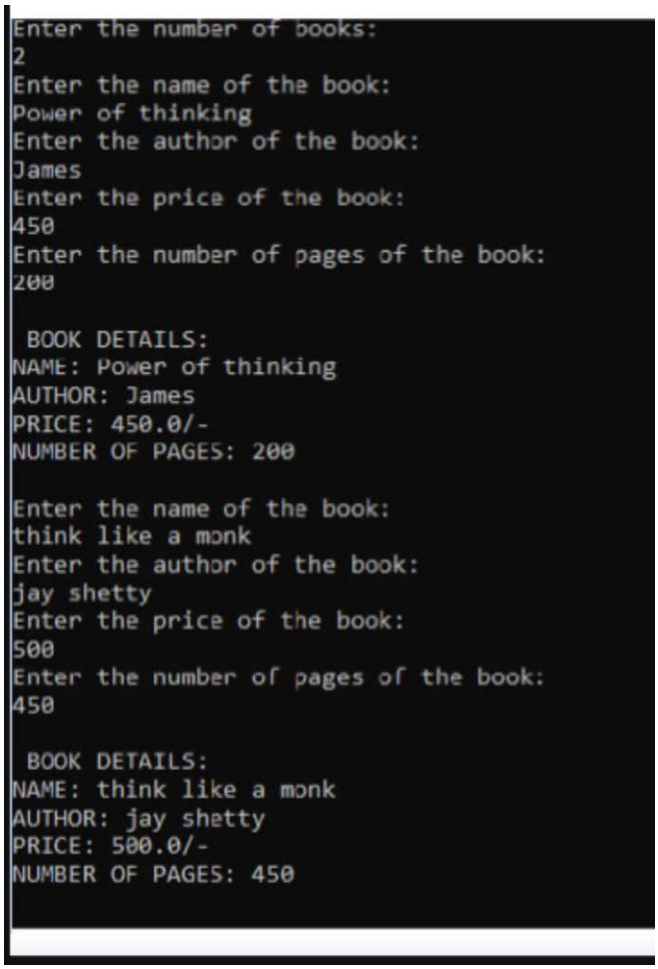
```
week-5
Create a class book which contains four
name, author, num-pages. Include a construction
the values for the members include
to set and get the details of the
objects. Include a toString() method that
display the complete details of the book
Develop a Java program to create
objects.

import java.util.Scanner;
class Book
{
  String name, author;
  double price;
  int numpages;
  Book()
  { name = " "; author = " ";
    price = 0.0; numpages = 0;
  }
  void set()
  {
    Scanner S = new Scanner (System.in);
    System.out.println("Enter the name of the
    name = S.nextLine();
    System.out.println("Enter the author:");
    author = S.nextLine();
    System.out.println("Enter price of book:");
    price = S.nextDouble();
```

```
System.out.println("Enter number of pages:");
num_pages = s.nextInt();
}
public String toString()
{
    return ("Name: " + name + "\nAuthor: " + author + "\nPrice
    + price + "\nNumber of pages: " + num_pages + "\n");
}
}
class Book_main
{
public static void main (String args[])
{
    Scanner s = new Scanner (System.in);
    System.out.println("Enter no. of books");
    int n = s.nextInt();
    Book Books[] = new Book[n];
    for (int i=0; i<n; i++)
    {
        books[i] = new Book();
        books[i].set();
    System.out.println("Book details:\n");
    System.out.println(books[i].get());
    }
}
}
```

Output

```
Enter the number of books : 2
Enter the name of the book: Gulmohar
Enter the author of the book: Mohan
Enter the price of the book: 350.0
Enter the number of pages : 200
Book Details:
Name : Gulmohar
Author : Mohan
Price : 350
Number of pages: 200
Enter the name of the book : Treasure Trove
Enter the author of the book : Roald Dahl
Enter the price of the book : 200.0
Enter the number of pages: 99
Book Details :
Name : Treasure Trove
Author: Roald Dahl
Price: 200.0
Number of pages: 99
```

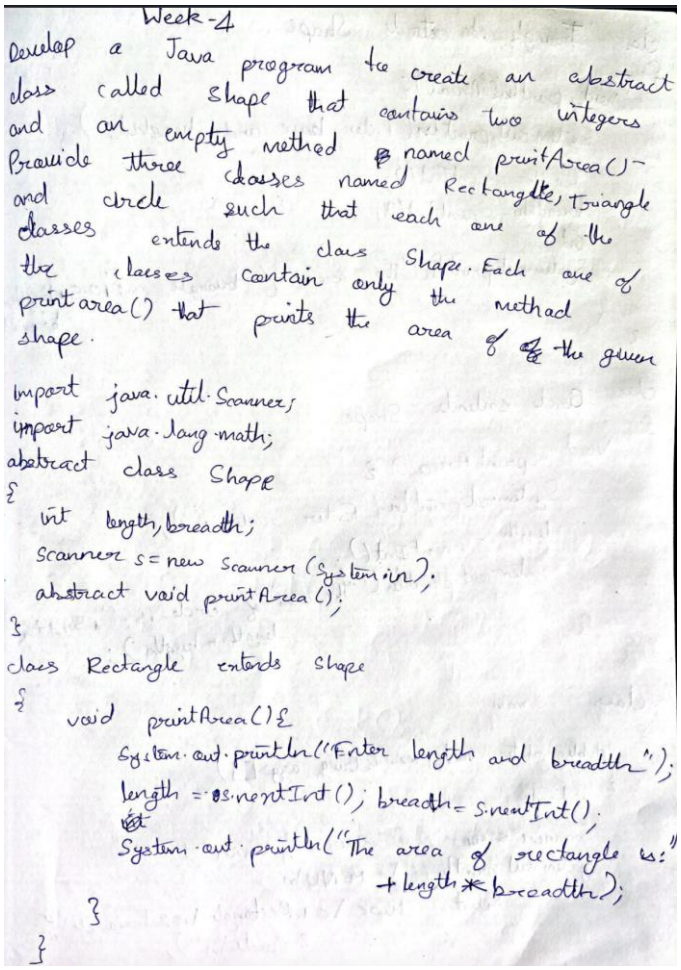
OUTPUT


Enter the number of books:
2
Enter the name of the book:
Power of thinking
Enter the author of the book:
James
Enter the price of the book:
450
Enter the number of pages of the book:
200
BOOK DETAILS:
NAME: Power of thinking
AUTHOR: James
PRICE: 450.0/-
NUMBER OF PAGES: 200
Enter the name of the book:
think like a monk
Enter the author of the book:
jay shetty
Enter the price of the book:
500
Enter the number of pages of the book:
450
BOOK DETAILS:
NAME: think like a monk
AUTHOR: jay shetty
PRICE: 500.0/-
NUMBER OF PAGES: 450

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.

PROGRAM



Week-4

Develop a Java program to create an abstract class called 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;
import java.lang.math;
abstract class Shape
{
  int length, breadth;
  Scanner s = new Scanner(System.in);
  abstract void printArea();
}
class Rectangle extends Shape
{
    void printArea(){
        System.out.println("Enter length and breadth");
        length = s.nextInt(); breadth = s.nextInt();
        System.out.println("The area of rectangle is:"
                           + length * breadth);
    }
}
```

```
class Triangle extends Shape
{
    void printArea(){
        System.out.println("Enter base and height:");
        length = s.nextInt();
        breadth = s.nextInt();
        System.out.println("The area of triangle is :"+0.5*length*
                                                        breadth);
    }
}

class Circle extends Shape
{
    void printArea() {
        System.out.println("Enter radius:");
        length = s.nextInt();
        System.out.println("The area of circle is :"+3.14*
                                length*length);
    }
}

class S_main
{
    public static void main(String args[])
    {
        int ch;
        Scanner ss = new Scanner(System.in);
        System.out.println("\n MENU\n
                Select Shape \n 1. Rectangle \n 2. Triangle \n
                        3. Circle \n");
```

```
ch = sc.nextInt();
switch(ch)
{
 case 1: Rectangle r = new Rectangle();
         r.printArea();
         break;
 case 2: Triangle t = new Triangle();
         t.printArea();
         break;
case 3: Circle C = new Circle();
        C.printArea();
        break;
 default: System.out.println("Invalid input");
  }
 }
}
```

② Select Shape
1. Rectangle
2. Triangle
3. Circle
2
Enter base length and height
20 80
Area of triangle =
2000

③ Select Shape
1. Rectangle
2. Triangle
3. Circle
3.
Enter the radius:
10
Area of the circle =
314

09/10/24

OUTPUT

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.

PROGRAM

Lab program 5

Develop a java program to create a class Bank that maintains two kinds of accounts for customers Savings and ~~checking~~ current account. The savings acc provides compound interest and withdrawal facilities but no cheque book facilities. The current account provides cheque book facility but no interest. Current account holder should 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, acc no, acc type. 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

a) Accept the deposit from customer and update the balance.

b) display the balance

c) Compute and deposit interest

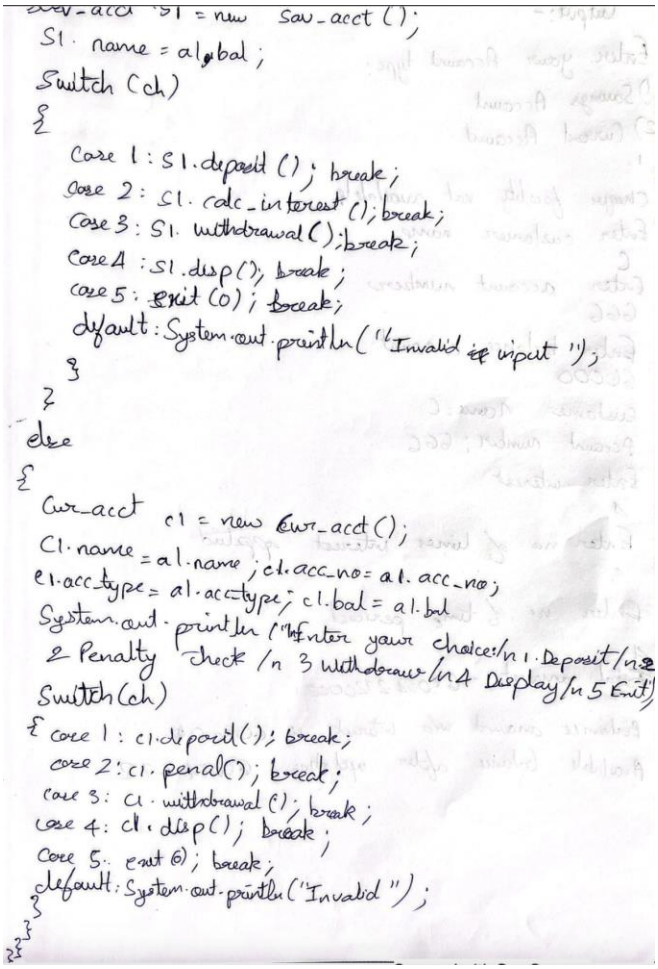
d) Permit withdrawal and update balance

```
Import java.util.Scanner;
Import java.lang.Math;
Class ~~Account~~ Account
{
   String name, acc_type;
   int acc_no
```

```
double bal, dep;
Scanner ss = new Scanner(System.in);
void setd()
{
    System.out.println("Enter your name, account number,
                        account type, Balance:-");
    name = ss.next();
    acc_no = ss.nextInt();
    acc_type = ss.next();
    bal = ss.nextDouble();
}
void dep() {
    System.out.println("Name: " + name);
    System.out.println("Account Number:" + acc_no);
    System.out.println("Account type: " + acc_type);
    System.out.println("Current balance: " + bal);
}
void deposit(){
    System.out.println("Enter the amount to be
    deposited:");
    dep = ss.nextDouble();
    bal += dep;
    System.out.println("Balance amount=" + bal);
}
```

```
boolean acc(String acc_type)
{ if(acc_type == "Savings")
    return true;
  else if (acc_type == "Current")
    return false;
  else
    return true;
  }
}
class Cur_acct extends Account
{
  int penal()
  { double min,pen;
  System.out.println("Enter minimum balance & penalty amount if not followed:");
  min = ss.nextDouble(); pen = ss.nextDouble();
  if (bal < min)
  { bal -= pen;
  System.out.println("Penalty imposed for having insufficient funds");
  else
  return 1;
}
```

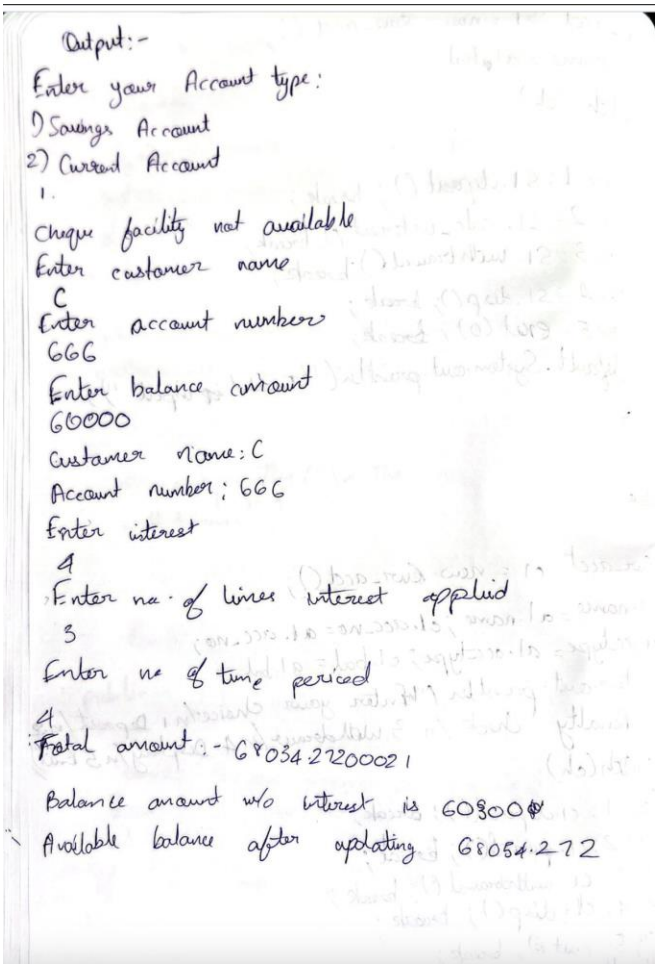
```
void withdrawal()
{ double amt;
  System.out.println("Enter amount to be
  withdrawn:");
amt = ss.nextDouble(); int a = penal();
if (a==1)
{ if (bal >=amt)
    { bal -= amt;
      System.out.println("Account balance after
        withdrawal is:"+bal);}
  }
  else
    System.out.println("The amount cant be
    withdrawn");
  }
}
class sav_acct extends Account
{
   void calc_interest()
   { System.out.println("Enter time and rate
     of interest");
     double t=ss.nextDouble();
     double r = ss.nextDouble();
     double CI= bal*Math.pow((1+r/
```



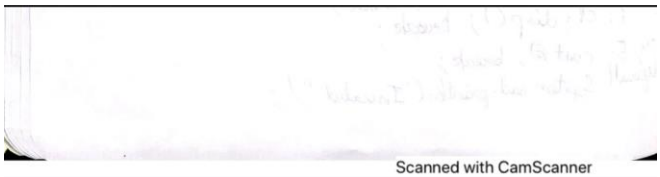
```
Sav_acct s1 = new Sav_acct();
s1.name = a1.bal;
Switch (ch)
{
   Case 1: s1.deposit(); break;
   Case 2: s1.calc_interest(); break;
   Case 3: s1.withdrawal(); break;
   Case 4: s1.disp(); break;
   case 5: exit(0); break;
   default: System.out.println("Invalid input");
   }
}
else
{
  Cur_acct c1 = new Cur_acct();
  C1.name = a1.name; c1.acc_no = a1.acc_no;
  e1.acctype = a1.acctype; c1.bal = a1.bal
  System.out.println("\nEnter your choice:\n1.Deposit/n2
  & Penalty check /n 3 Withdraw/n4 Display/n5 Exit");
  Switch(ch)
 { case 1: c1.deposit(); break;
   case 2: c1.penal(); break;
   case 3: c1.withdrawal(); break;
  case 4: c1.disp(); break;
  Case 5: exit(0); break;
  default: System.out.println("Invalid");
  }
}}
```

```
int ch = cs.nextInt();
switch(ch)
{
   case 1 : s1.deposit(); break;
   case 2 : s1.cal_interest(); break;
   case 3 : s1.withdraw(); break;
   case 4 : s1.disp(); break;
   case 5 : exit(0); break;
   default : System.out.println("Invalid Input");
}
}
else
{

Cur_acct c1 = new Cur_acct();
c1.name = a1.name; c1.acno = a1.acc_no;
c1.acc_type = a1.acc_type; c1.bal = a1.bal;
System.out.println("\n Enter your choice :\n 1.Deposit \n 2.Display
  check \n 3.Withdraw \n 4.Display \n 5.Exit");
int ch = cs.nextInt();
switch(ch)
      {
       case 1 : c1.deposit(); break;
       case 2 : c1.print(); break;
       case 3 : c1.withdraw(); break;
       case 4 : exit(0); break;
       default : System.out.println("Invalid Input");
      }
}/
```

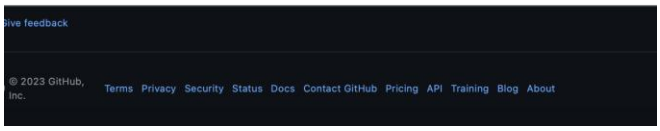
Enter your account type (1. Savings or 2. Current)
2
Current
Enter your name
John
Enter your account number
2222
Enter the Bank Balance
50000
Enter your choice
1. Deposit
2. Calculate Interest
3. Withdraw
4. Display
5. Exit
1
Enter the amount to be deposited
5000
Account Balance after deposit is 55000
Enter your choice
1. Deposit
2. Calculate interest
3. Withdraw
4. Display
5. Exit
3
Enter amount to be withdrawn:
5000
Account Balance after withdrawal is 50000


Output:-
Enter your Account type:
1) Savings Account
2) Current Account
1.
Cheque facility not available
Enter customer name
C
Enter account number
666
Enter balance amount
60000
Customer name: C
Account number: 666
Enter interest
4
Enter no. of times interest applied
3
Enter no of time period
4
Total amount:- 68034.2220002 1
Balance amount w/o interest is 60800
Available balance after updating 68054.272

OUTPUT




```
Enter your account type:
1. Savings account
2. Current account
1
Cheque Facility not available
Enter customer name
rrr
Enter rrr's account number
5555
Enter balance amount
60000
Customer Name:rrr
Your account number:5555
Your Account Balance:60000.0
Press 1 to deposit
1
Enter amount to be deposited
500
Enter rate of interest
4
Enter number of times interest applied per time period
3
Enter number of time periods
4
Interest amount=68054.27200000001
Balance amount without interest is60500.0
Available balance after updating is68054.27200000001
Press 1 to withdraw ammount
```

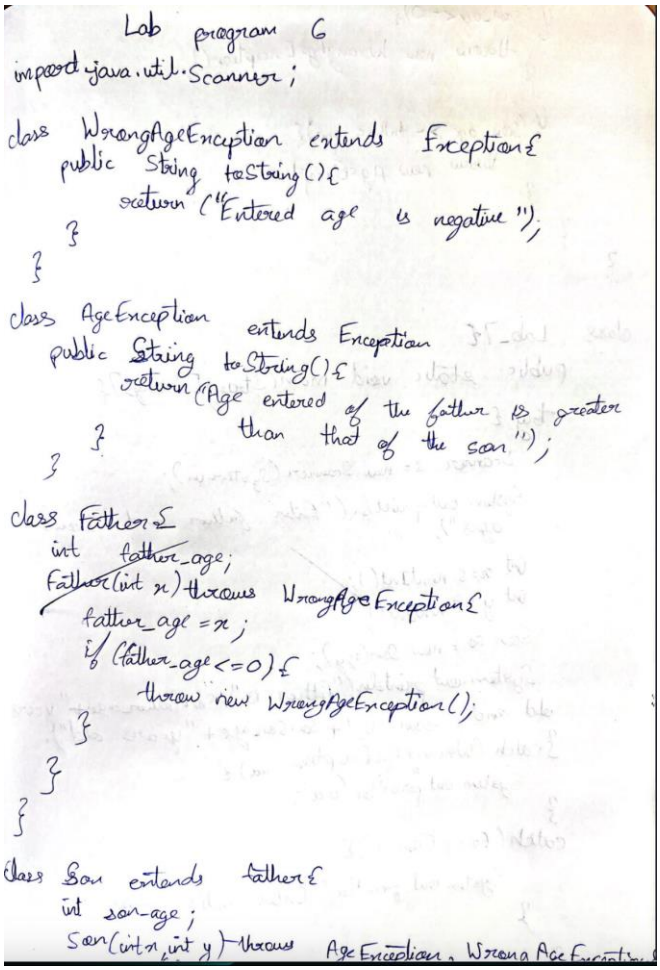




ter your account type:
Savings account
Current account
ease facility not available
ter customer name
a
ter sa's account number
2
ter balance amount
0000
stomer Name:sa
ur account number:22
ur Account Balance:10000.0
ess 1 To Deposit
nvalid Input
ter rate of interest
ter number of times interest applied per time period
ter number of time periods
terest amount=11575.250000000001
lance amount without interest is10000.0
ailable balance after updating is11575.250000000001
ess 1 to withdraw amount
ter the amount to be withdrawn
00
ailable Balance:6500.0
\Users\bmsce\Desktop>java Lab5

LAB PROGRAM-6

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.

PROGRAM



```
Lab program 6
import java.util.Scanner;

class WrongAgeException extends Exception{
    public String toString(){
        return ("Entered age is negative");
    }
}

class AgeException extends Exception
    public String toString(){
        return ("Age entered of the father is greater
                than that of the son");
    }
}

class Father{
    int father_age;
    Father(int x) throws WrongAgeException{
        father_age = x;
        if (father_age<=0){
            throw new WrongAgeException();
        }
    }
}

class Son extends father{
    int son_age;
    Son(int x, int y) throws AgeException, WrongAge
```

```
class Son extends father{
    int son_age;
    Son(int x, int y) throws AgeException, WrongAgeException{
        super(x);
        son_age = y;
```

Scanned with CamScanner

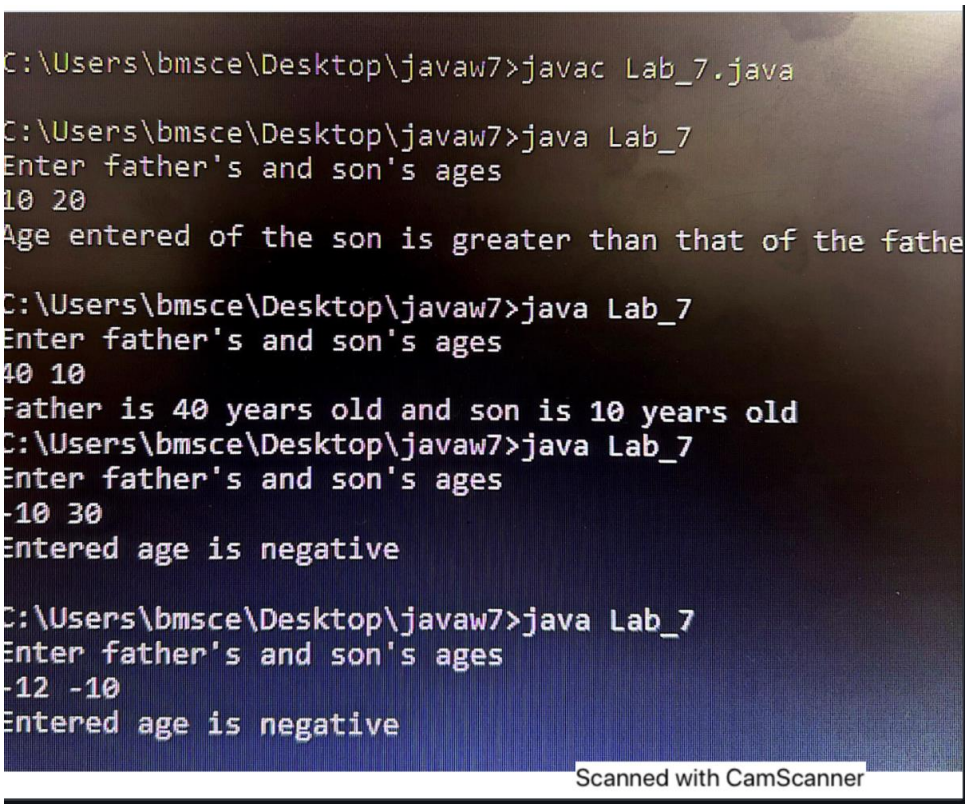
```
        if (son_age<=0){
            throw new WrongAgeException();
        }
        if (son_age >= father_age){
            throw new AgeException();
        }
    }
}

class Lab_7{
    public static void main(String[] args){
        try {
            Scanner s = new Scanner(System.in);
            System.out.println("Enter father and son
            ages");
            int x=s.nextInt();
```

Output

```
Enter father's and son's ages
56
22
Father is 56 years old and son is 22 years old

Enter father's and son's ages
31
45
Age entered of the son is greater than that of the father

Enter father's and son's ages
-76
44
Entered age is negative.
```

30/12/2022

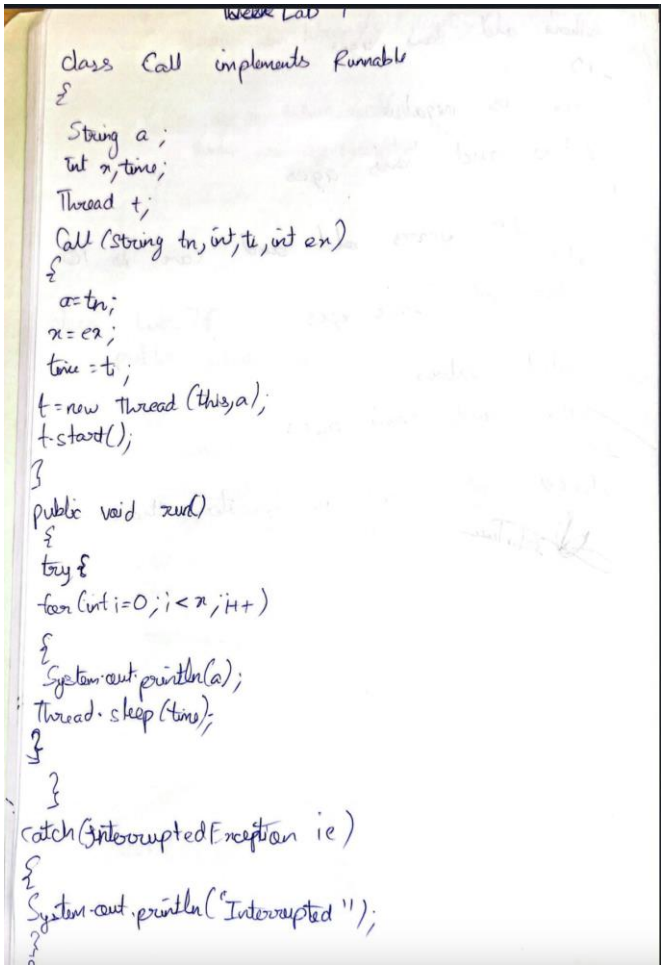
OUTPUT


C:\Users\bmsce\Desktop\javaw7>javac Lab_7.java
C:\Users\bmsce\Desktop\javaw7>java Lab_7
Enter father's and son's ages
10 20
Age entered of the son is greater than that of the fathe
C:\Users\bmsce\Desktop\javaw7>java Lab_7
Enter father's and son's ages
40 10
Father is 40 years old and son is 10 years old
C:\Users\bmsce\Desktop\javaw7>java Lab_7
Enter father's and son's ages
-10 30
Entered age is negative
C:\Users\bmsce\Desktop\javaw7>java Lab_7
Enter father's and son's ages
-12 -10
Entered age is negative
Scanned with CamScanner

LAB PROGRAM-7

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 Call implements Runnable
{
 String a;
 int n, time;
 Thread t;
 Call (String tn, int ti, int ex)
 {
  a=tn;
  n=ex;
  time=ti;
 t=new Thread (this,a);
 t.start();
 }
 public void run()
  {
  try {
  for (int i=0;i<n;i++)
  {
  System.out.println(a);
 Thread.sleep(time);
 }
  }
catch(InterruptedException ie)
{
System.out.println("Interrupted");
```

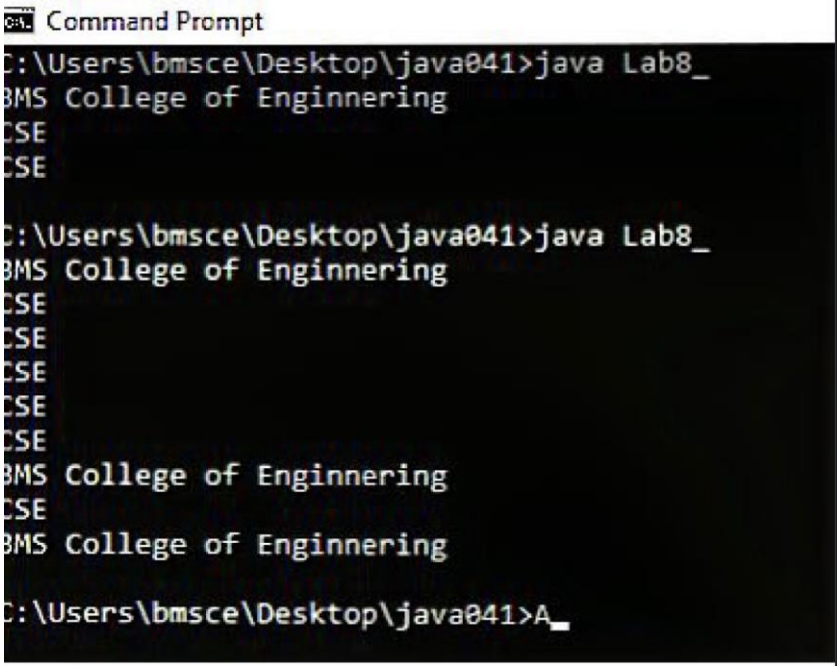
```
class Lab8_
{
 public static void main (String xx[])
 {
  new Call ("BMS college of engineering", 10000, 3);
  new Call ("CSE", 2000, 6);
 }
}
```

Output

```
BMS college of Engineering
CSE
CSE
CSE
CSE
CSE
BMS college of Engineering
CSE
BMS college of Engineering
```

06/01/2023

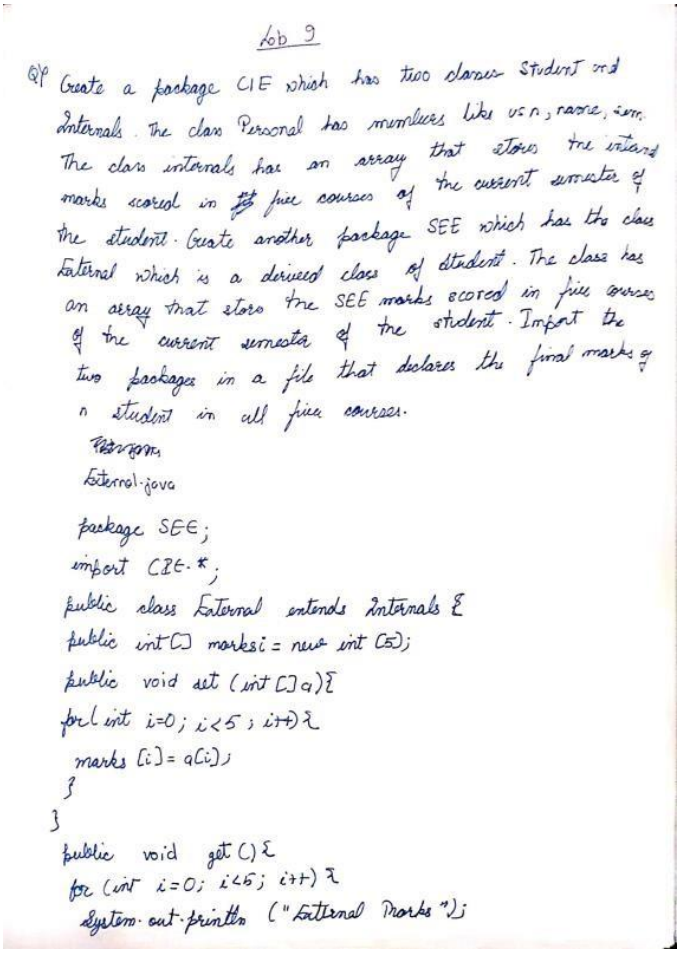
OUTPUT



LAB PROGRAM 8

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.
Develop a Generic Class with Two Type Parameters.

PROGRAM



Lab 9

Q) 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. The 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 student in all five courses.

External.java

```
package SEE;
import CIE.*;
public class External extends Internals {
public int[] marksi = new int[5];
public void set(int[] a){
for(int i=0; i<5; i++){
 marks[i] = a[i];
}
}
public void get(){
for(int i=0; i<5; i++)
 System.out.println("External Marks");
```

```
System.out.println ("Subject " + i + ": " + marks[i]);
}
}
}
```

external.java

```
package CIE;
public class Externals extends Student {
public int[] marks = new int[5];
public void setint (int[] a) {
for (int i=0; i<5; i++) {
marks[i] = a[i];
}
}
public void getint() {
System.out.println ("External marks");
for(int i=0; i<5; i++){
System.out.println ("Subject " + i + ": " + marks[i]);
}
}
}
```

Student.java

```
package CIE;

public class Student {
public int USN;
public String name;
public int sem;

public void setdets (int u, String s , int q) {
  USN=u;
  name =s;
   Sem = q;
}
public void getdets () {
 System.out.println(" Student details ");
System.out.println ( "USN:"+USN + "\n NAME:" + name + "\nSEMESTER:"+
                                                    sem);
 }
}
```

Output

Enter number of students.

1

Enter student details

1 a 2

Enter external marks.

12 12 12 12 12

Enter internal marks.

12 12 12 12 12

Student Details

USN: 1

Name: a

SEM: 2

Internal Marks

Subject 1: 12

Subject 2: 12

Subject 3: 12

Subject 4: 12

Subject 5: 12

External Marks

Subject 1: 12

Subject 2: 12

Subject 3: 12

Subject 4: 12

Subject 5: 12

Final Marks.

Subject 1: 18

Subject 2: 18

Subject 3: 18

Subject 4: 18

Subject 5: 18

```
Enter number of students
1
Enter student details
1 aqw 2
Enter internal marks
12 12 12 12 12
Enter external marks
12 12 12 12 12
Student details
USN:1
NAME:aqw
SEMESTER:2
Internal marks
Subject 0: 12
Subject 1: 12
Subject 2: 12
Subject 3: 12
Subject 4: 12
External marks
Subject 0: 12
External marks
Subject 1: 12
External marks
Subject 2: 12
External marks
Subject 3: 12
External marks
Subject 4: 12
Final marks
Subject 0: 18
Subject 1: 18
Subject 2: 18
Subject 3: 18
Subject 4: 18
```