OBSERVATION:

Program 1

1] 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.

```
import java.util.Scanner;

class Quad {
    public static void main (String xx[]){
        int a,b,c;
        Scanner s = new Scanner (System.in);
        System.out.println("Enter the coefficients a,b,c");
        a = s.nextInt();
        b = s.nextInt();
        c = s.nextInt();
        double des = (b*b)-(4*a*c);
        double root1, root2;
        if (a==0){
        System.out.println("The equation is not quadratic");
        }
        else if (des>0){
                root1 = -b+Math.sqrt(des);
                root2 = -b-Math.sqrt(des);
                System.out.println("The roots are real
                and distinct\n Root 1:  "+root1+"\n root
                2 : "+root2);
        }
        else if (des==0){
                root1 = root2 = -b/(2*a);
                System.out.println("The roots are
                real and equal\n Root 1 : "+root1+"
                \n root 2 : "+root2);
        }
```

```
else {
        root1 = -b/(2*a);
        root2 = Math.sqrt(Math.abs(dis));
        System.out.printf("The roots are imaginary\n
        Root 1 : " + root1 + "+i" + root2 + "\nRoot 2: "
        + root1 + "-i" + root2);
        S.close();
    }
  }
}
```

OUTPUT:

```
C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main\OOJ-1BM21cs019--main>javac lab1_java.java

C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main\OOJ-1BM21cs019--main>java Quad
Enter the coefficients a,b,c
0 0 0
The equation is not quadratic


C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main\OOJ-1BM21cs019--main>java Quad
Enter the coefficients a,b,c
1 2 1
The roots are real and equal
Root 1:  -1.0
root 2:   -1.0

C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main\OOJ-1BM21cs019--main>java Quad
Enter the coefficients a,b,c
1 4 1
The roots are real and distinct
root 1:    -0.5358983848622456
root 2:   -7.464101615137754

C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main\OOJ-1BM21cs019--main>java Quad
Enter the coefficients a,b,c
1 1 1
The roots are imaginary
Root1 :   0.0+i1.7320508075688772
Root 2:   0.0-i1.7320508075688772

C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main\OOJ-1BM21cs019--main>
```

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.

Lab-program 2

```
import java.util.Scanner;

class Student {
    String usn;
    String name;
    int credits[];
    int marks[];
    double sgpa=0;
    int cie=0;
    int totcred=0;
    void accept(String usn, String name,
                int credits[], int marks[]){
        this.usn = usn;
        this.name = name;
        this.credits = credits;
        this.marks = marks;
    }
    Student(int credsize){
        credits = new int[credsize];
        marks = new int[credsize];
    }
    void display(){
        System.out.println("usn : " + usn + "\n" +
        "name:" + name + "\n" + "sgpa:" + sgpa);
    }
    void sgpacalc(){
        for (int i=0; i<marks.length; i++){
        if (marks[i] >= 40)
            cie = 0
        else    cie = marks[i]/10;
        sgpa = (sgpa/totcred)*10;
        sgpa += cie * credits[i];
        totcred += credits[i] * 10;
    }
        sgpa = (sgpa/totcred)*10;
    }
}
```

```
class lab_3{
public static void main(String[] args) {
        Scanner s= new Scanner(System.in);
        System.out.println("Enter the no of courses:");
        int n = s.nextInt();
        int credits[]= new int[n];
        int marks[]= new int[n];
        Students st = new Students(n);
        System.out.println("Enter the credits of
                the courses:");
        for (int i=0;i<n;i++)
        {
        credits[i]=s.nextInt();
        }
        System.out.println("Enter your usn,name");
        String usn = s.next();
        String name = s.nextLine(); // for spaced input.
        String skipline = s.nextLine(); \\consumes \n
        System.out.println("Enter your marks in
                each subject");
        for (int i=0;i<n;i++){
            System.out.println("enter the marks
                    obtained in "+i+" course(credits"
                    +credits[i]+"):");
            marks[i]=s.nextInt();
        }
        st.accept(usn, name, credits, marks);
        st.spacial();
        st.display();
        s.close();
    }
}
```

OUTPUT:

```
C:\Users\BMSCECSEIL74\Desktop>javac LAB2.java

C:\Users\BMSCECSEIL74\Desktop>java lab_2
Enter the number of courses
9
Enter the creadits of the courses:
4
1
1
1
1
3
3
3
3
Enter your  usn,name
1BM21CS019
AMSHU G M
Enter your marks in each subject
enter the marks obtained in 0 course (credits= 4) :90
enter the marks obtained in 1 course (credits= 1) :98
enter the marks obtained in 2 course (credits= 1) :98
enter the marks obtained in 3 course (credits= 1) :84
enter the marks obtained in 4 course (credits= 1) :86
enter the marks obtained in 5 course (credits= 3) :56
enter the marks obtained in 6 course (credits= 3) :87
enter the marks obtained in 7 course (credits= 3) :87
enter the marks obtained in 8 course (credits= 3) :87

--------------
usn:1BM21CS019
name:AMSHU G M
sgpa:8.85
```

Create a class Book which contains 4 members: name, author, price, num_pages. Include a constructor to set values for members. Include methods to set and get details of objects. Include a toString() method that could display details of the book.

```
import java.util.Scanner;

class Book{

    int num_pages;
    double price;
    String name;
    String author;
    Book(){
            num_pages=0
            price = 0.0;
            name = "some_book";
            author = "anshu";
        }

    Book(int num_pages, double price, String name, String author)
        {
        this.num_pages = num_pages;
        this. price = price;
        this. name = name;
        this.author = author; }
    void set_data(int num_pages, double price, String
            name, String author) {
            this. num_pages = num_pages;
            this. price = price;
            this. name = name;
            this. author = author; }
    void get_data(){
        System.out.println("Book details\n name : "+
name +"\n author : "+author+"\n no of pages : "+num_pages
```

```
"\n price : " + price );
System.out.println("\n ---------- \n");
public String toString(){
return ("Book details \n name: "+name + "\n author
+ author + "\n no of pages: " + num_pages + "\n price
: " + price + "\n ---------- \n");
}
}
class lab_3 {
public static void main (String[] args){
Scanner s=new Scanner(System.in);
System.out.println("enter the number of books");
int n = s.nextInt();
Book bs[] = new Book[n];
for (int i=0; i<n; i++){
b[i] = new Book();
System.out.print ("enter the name of book: ");
String name = s.nextLine();
System.out.print ("enter the no of pages: ");
String author = s.nextLine();
System.out.println ("enter the no of pages");
int num_pages = s.nextInt();
System.out.print ("enter the price of book");
double price = s.nextDouble();
System.out.println();
bs[i].setdata (num_pages, price, name, author);}
for (int i=0 ; i<n; i++){
bs[i].getdata(); }
Book b2 = new Book(2987.65, "compilation
point ", "dan brown");
b2.getdata();
System.out.println(b2);
s.close(); } }
```

OUTPUT:

```
C:\Users\amshu\OneDrive\Desktop>javac lab3.java

C:\Users\amshu\OneDrive\Desktop>java lab_3
enter the number of books
2
eneter the name of the book: The fault in our stars
eneter the author's name: John Green
eneter the number of pages in the book: 345
eneter the price of the book: 34

eneter the name of the book: Da vinci code
eneter the author's name: Dan brown
eneter the number of pages in the book: 600
eneter the price of the book: 567

Book details
name: The fault in our stars
author: John Green
nmber of pages: 345
price: 34.0

------------------------

Book details
name: Da vinci code
author: Dan brown
nmber of pages: 600
price: 567.0

------------------------

Book details
name: deception point
author: dan brown
nmber of pages: 20
price: 87.65

------------------------
```

```
Book details
name: The fault in our stars
author: John Green
nmber of pages: 345
price: 34.0

------------------------

Book details
name: Da vinci code
author: Dan brown
nmber of pages: 600
price: 567.0

------------------------

Book details
name: deception point
author: dan brown
nmber of pages: 20
price: 87.65

------------------------

Book details
name: deception point
author: dan brown
nmber of pages: 20
price: 87.65
--------------------------------


C:\Users\amshu\OneDrive\Desktop>
```

4] Develop a java program to create a class named shape. This class contains two integers and an empty method named printArea(). Provide 3 classes, Rectangle and Triangle such that each extends shape. Each one should contain only one method printArea() and should print the area of given shape.

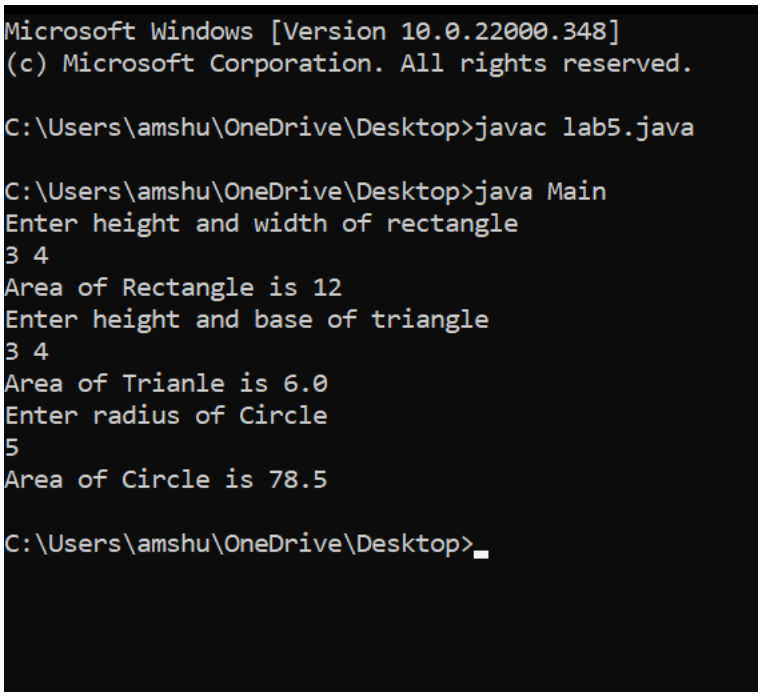
Lab-Program 4

```
import java.util.Scanner;

abstract class shape
{
    shape(){}
    int h,b;
    abstract void printArea();
}
class Rectangle extends shape
{
    Scanner s = new Scanner(System.in);
    void printArea()
    {
        System.out.println("Enter height and width");
        int h = s.nextInt();
        int b = s.nextInt();
        System.out.println("Area of rectangle= "+h*b);
    }
    Rectangle(){} s.close();
}
class Triangle extends shape
{
    Scanner s = new Scanner(System.in);
    void printArea()
    {
        System.out.println("Enter height and base");
        h = s.nextInt();
        b = s.nextInt();
        System.out.println("Area of triangle = "+0.5*b*h);
    }
    Triangle(){} s.close();
}
```

```
class lab_3{
public static void main(String[] args) {
        Scanner s= new Scanner(System.in);
        System.out.println("Enter the no of courses:");
        int n = s.nextInt();
        int credits[]= new int[n];
        int marks[]= new int[n];
        Students st = new Students(n);
        System.out.println("Enter the credits of
                the courses:");
        for (int i=0;i<n;i++)
        {
        credits[i]=s.nextInt();
        }
        System.out.println("Enter your usn,name");
        String usn = s.next();
        String name = s.nextLine(); // for spaced input.
        String skipline = s.nextLine(); \\consumes \n
        System.out.println("Enter your marks in
                each subject");
        for (int i=0;i<n;i++){
            System.out.println("enter the marks
                    obtained in "+i+" course(credits"
                    +credits[i]+"):");
            marks[i]=s.nextInt();
        }
        st.accept(usn, name, credits, marks);
        st.spacial();
        st.display();
        s.close();
    }
}
```

OUTPUT:


Microsoft Windows [Version 10.0.22000.348]
(c) Microsoft Corporation. All rights reserved.
C:\Users\amshu\OneDrive\Desktop>javac lab5.java
C:\Users\amshu\OneDrive\Desktop>java Main
Enter height and width of rectangle
3 4
Area of Rectangle is 12
Enter height and base of triangle
3 4
Area of Trianle is 6.0
Enter radius of Circle
5
Area of Circle is 78.5
C:\Users\amshu\OneDrive\Desktop>

LAB_5

QUESTION:

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.

Complete the observation and execution of both the above programs tomorrow.

LAB-5

5. Develop a java program to create a class Bank that maintains two kinds of accounts 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.

Compute the observer

```
import java.util.Scanner;

class Account
{
    String name;
    int type;
    long acno;
    double balance;
    void getA()
    {
        Scanner s = new Scanner(System.in);
        System.out.println("enter customer name");
        name = s.nextLine();
        System.out.println("enter account no");
        acno = s.nextLong();
        System.out.println("enter bank balance");
        balance = s.nextDouble();
    }
    void display()
    {
        System.out.println("customer name is :" +name);
        if(type==1){
            System.out.println("customer account
                type is :savings");
        }
        else{
            System.out.println("customer account
                type is :current");}
        System.out.println("customer acco
            number is :" +acno);
        System.out.println("current balan
            :" +balance);}

    void deposit()
    {   System.out.println("Enter the amo
                to be deposited:
```

```
Scanner x = new Scanner(System.in);
    double amt = x.nextDouble();
    balance += amt;
}

class sav_acct extends Account
{
    double interest;
    Scanner s = new Scanner(System.in);
    sav_acct() {
        type = 1; }
    void cinterest()
    {
        int time;
        float rate;
        System.out.println("compound Interest
            details:");
        System.out.println("enter time in years:");
        time = s.nextInt();
        System.out.println("enter rate of interest:");
        rate = s.nextFloat();
        System.out.println("Interest will be
            compounded 5 times a year");
        interest = balance * (Math.pow((1+rate/5),
            (5*time))-1);
        balance += interest;
    }
void withdraw()
    {
        System.out.println("enter the amount
            to be withdrawn =");
        double amt = s.nextDouble();
        if (balance > amt)
            {balance -= amt;}
```

```
else
{ System.out.println("Amount to be withdrawn
        greater than balance!!!"); }
    }
}

class Curr_acct extends Account
{
    double check_amt;
    Curr_acct(){
        type=2;
    }
    void cheque()
    {
        System.out.println("Enter the cheque
        amount:");
        Scanner s=new Scanner(System.in);
        check_amt=s.nextDouble();
        if(check_amt>balance-5000)
        {
            System.out.println("Rs.500
            penalty imposed.... IS it OK to
            proceed? Enter y for yes and
            n for no");
            String option.equals("y"))
            { balance=balance-check_amt;
            else {System.out.println("no
                check debited");}
        }
        else
        {
            System.out.println("Rupees"
            +check_amt+"debited");
            balance-=check_amt; }
    }
```

```
void withdraw()
{
    System.out.println("Enter the amount to
    be withdrawn:");
    Scanner s = new Scanner(System.in);
    double amt = s.nextDouble();
    if(balance>amt) { balance -= amt; }
    else
    { System.out.println("Amount to be withdrawn
      greater than balance!!!"); }
}

class Bank {
    public static void main(String ss[]){
        String opt, opt2;
        Scanner s = new Scanner(System.in);
        System.out.println("1. Savings ac
        2. current ?");
        int a;
        a = s.nextInt();
        if(a==1) {
            sav_acct st = new sav_acct();
            while(true) {
        System.out.println("Enter the choice
1. Set the values for savings acc\n 2.
display\n 3. deposit\n 4. interest\n 5. Withdraw\n
6. exit\n");
        opt = s.next();
        switch(opt)
        {
        case "1": st.set();
                  break;
        case "2": st.display(); break;
        case "3": st.deposit(); break;
```

```
case "4" : s1.(method2)(); break;
case "5" : s1.withdraw(); break;
case "6" : System.exit(0);
}
}
}
else if (a==2){
Curr_acct c1 = new Curr_acct();
while (true){
System.out.println("enter the choice:
\n 1. set the value for current account\n 2.
display\n 3. deposit\n 4. transfer check\n 5.
withdraw\n 6. exit\n");
op2 = s.nextInt();
switch(op2)
{
case 1: c1.setA();
break;
case 2 : c1.display(); break;
case 3 : c1.deposit();break;
case 4 : c1.cheque(); break;
case 5 : c1.withdraw(); break;
case 6 : System.exit(0); }}
}
```

0:

1] Savings or 2. Current ?

Enter the choice :

1] Set the values for Savings ac

2. display

3. deposit

4. Interest

5. Withdraw

6. Exit.

1

Enter customer name : ramshu

Enter bank account no: 123

Enter bank balance : 777777

3

Enter the amount to be deposited : 12000

4

Enter time in years : 1

Enter rate of interest : 7

2

customer name is : ramshu

Customer account type is : Savings

Customer account number is : 123

Current balance is : 6.36767E7

5.

Enter the amount to be withdrawn : 17

6.

1. Savings or 2. current ?

2.

Enter the choice :

1. Set the values for current account
2. display
3. deposit
4. transfercheck
5. withdraw
6. exit.

4.

Enter cheque amount : 12345678.

Rs. 500 penalty imposed ... IS it OK to proceed ?

Enter y for yes and n for no

y

2.

Customer name is : anshu

Customer account type is : current

Customer account number : 1234

Current balance is : 12111E7

3.

Enter the amount to be deposited : 12345678

5.

Enter the amount to be withdrawn :

Balance = 12345678.

6.

16/12/22

OUTPUT:

```
C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main (2)\OOJ-1BM21cs019--main>java Bank
Enter the choice:
1a.Set the values for savings acc
1b. display
1c. deposit
1d. Interest
1e. Withdraw
1f. exit
1a
Enter customer name: amshu
Enter type of account: current
Enter account number: 12
Enter bank balance: 12000
Enter the choice:
2a.Set the values for current account
2b. display
2c. deposit
2d. minBalance
2e. Withdraw
2f. exit
2c
Enter the amount to be deposited: 12000
Enter the choice:
1a.Set the values for savings acc
1b. display
1c. deposit
1d. Interest
1e. Withdraw
1f. exit
1a
Enter customer name: amshu
Enter type of account: savings
Enter account number: 123
Enter bank balance: 12899
Enter the choice:
2a.Set the values for current account
2b. display
```

```
Enter customer name: amshu
Enter type of account: savings
Enter account number: 123
Enter bank balance: 12899
Enter the choice:
2a.Set the values for current account
2b. display
2c. deposit
2d. minBalance
2e. Withdraw
2f. exit
2b
Customer name is: null
Customer account type is: null
Customer account number is: 0
Current balance is: 12000.0
Enter the choice:
1a.Set the values for savings acc
1b. display
1c. deposit
1d. Interest
1e. Withdraw
1f. exit
1b
Customer name is: amshu
Customer account type is: savings
Customer account number is: 123
Current balance is: 12899.0
Enter the choice:
2a.Set the values for current account
2b. display
2c. deposit
2d. minBalance
2e. Withdraw
2f. exit
2d
No penalty imposed
Enter the choice:
1a.Set the values for savings acc
1b. display
```

```
2b. display
2c. deposit
2d. minBalance
2e. Withdraw
2f. exit
2d
No penalty imposed
Enter the choice:
1a.Set the values for savings acc
1b. display
1c. deposit
1d. Interest
1e. Withdraw
1f. exit
1f
```

WEEK_6

QUESTION:

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.

7] Write a program that demonstrate 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.

```
class Son extends Father{
    try
import java.util.Scanner;
class WrongAge Exception extends Exception{
        public String toString(){
            return ("Negative age can't be accepted");
        }
    }
class AgeException extends Exception{
        public String toString(){
            return ("Son can't be older than
                father");
        }
    }
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,
    wrongAgeException{
        super(x);
        son_age = y;
        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's and
                son's ages");
            int x = s.nextInt();
            int y = s.nextInt();
            Son so = new Son(x,y);
            System.out.printf("Father is %d years
            old and son is %d years old", so.father
            age, so.son_age);
        }
        catch (AgeException a){
            System.out.println(a);
        }
        catch (wrongAgeException wa){
            System.out.println(wa);
        }
        catch (Exception e){
```

```
System.out.println("Enter valid values");
    }
  }
}
```

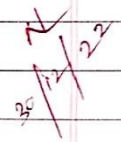
Output:

Enter father's and son's ages
12 13
Son can't be older than father

Enter father's and son's age
-12 13
Negative age can't be accepted.

Enter father's and son's age
12 -13
Negative age can't be accepted.

Enter father's and son's age
45 25
Father is 45 years old and son is 25 years old.



OUTPUT:

```
enterfather's and sons ages
12 13
son cannot be older than father
enterfather's and sons ages
-12 13
Negative age cannotbe accepted
enterfather's and sons ages
45 25
father is 45 years old and son is 25 years old
```

Week_7

QUESTION:

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.

Once the above program is completed, practice all programs done in threads.

Lab-7

Write a program to create two threads, one thread displaying "BMS College of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

```
class thread1 implements Runnable {
    Thread t;
    thread1 (String s) {
        t = new Thread(this, "new thread");
    }
    public void run() {
        try {
            for (int i=5; i>0; i--) {
                System.out.println("BMSCE");
                Thread.sleep(10000);
            }
        }
        catch (InterruptedException e) {
            System.out.println("Thread
                Interrupted.");
        }
    }
}
class thread2 implements Runnable {
    Thread t;
    thread2 (String s) {
        t = new Thread(this, "BMS");
    }
    public void run() {
        try {
        for (int i =25; i>0; i--) {
```

```
        System.out.println("CSE");
        Thread.sleep(2000);
      }
    }
    catch (InterruptedException e){
        System.out.println("Thread interrupted");
    }
  }
}
class threadmain {
    public static void main(String args[]){
        thread t1 = new thread1("thread");
        thread t2 = new thread2("thread");
        t1.start();
        t2.start();
    }
}
```

Output:

BMSCE
CSE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE

CSE

CSE

CSE

BMSCE

CSE

CSE

CSE

CSE

CSE

BMSCE

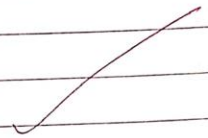
CSE

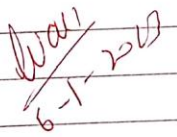
CSE

CSE

CSE

CSE





OUTPUT:

```
C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main (2)\OOJ-1BM21cs019--main>javac lab_7_java.java

C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main (2)\OOJ-1BM21cs019--main>java threadmain
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
CSE
CSE
CSE
CSE
CSE
```

WEEEK_8

QUESTION:

Develop
a Generic Class with Two Type Parameters.

Lab-8

Develop a Generic class with two parameters

```
import java.util.*;
import java.lang.*;

class Gen<T,V>{

    T obj;
    V obj;

    Gen(T x, V y){
        obj = x;
        ob2 = y; }

    T getObj(){
        return obj; }
    T getOb2(){
        return ob2; }

    void showType(){
        System.out.println("value of T = " + obj.getClass().
            getName());
        System.out.println(" value of V = " + ob2.getClass()
            getName());
    }

}
class generics{
    public static void main(String args[]){
        Gen<Integer, String> w;
        w = new Gen<Integer, String>(10,
            "Anshu");
        int x = w.getObj();
```

```
        System.out.println("The value of x = "+x);
        String y = w.getOb2();
        System.out.println("The value of y = "+y);
        w.showtype();
    }
}
```

Output:

The value of x = 001
The value of y = Amshu
value of r = lang.Integer
value of v = lang.String

OUTPUT:

```
C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main (2)\OOJ-1BM21cs019--main>javac lab8_java.java

C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main (2)\OOJ-1BM21cs019--main>java generics
The value for x =1
The value for y =Amshu
value of T =java.lang.Integer
value of V =java.lang.String
```

WEEK_9

QUESTION:

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.

Lab 9

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 SEE marks scored in five courses of the current semester. Create another package SEE which has the class External which is a derived class of student. This class has an array that stores SEE marks scored in five courses of student. Import the two packages in a file that declares the final marks of a student in all five courses.

```
package CIE;

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

public void getStudDetails(int n, String s, int q){
    USN=n;
    name=s;
    sem=q; }

public void displayStudDetails(){
    System.out.println("Student details");
    System.out.println("USN:"+USN+"\nName:"
        +name+"\nSEMESTER:"+sem);
  }
}
```

```
import CIE.*;
import SEE.*;
import java.util.Scanner;

class Test5
{
public static void main (String args[])
{
Scanner s = new Scanner (System.in);
System.out.println("Enter the number of students");
int n = s.nextInt();
External[] stop = new External[n];
for (int i=0; i<n; i++)
{
stop[i] = new External();
System.out.println("Enter external marks");
stop[i].setExternal(s.nextInt());
System.out.println("Enter internal marks");
int[] iat = new int[5];
for (int j=0; j<5; j++)
{
iat[j] = s.nextInt();
}
String name = s.next();
int usn = s.nextInt();
stop[i].setDetails(usn, name);
Internal istop = new Internal();
istop.setMarks(iat);
}
for (int i=0; i<n; i++)
{
stop[i].display();
}
}
}
```

```
for(int i=0; i<n;i++){
System.out.println("Subject "+i+" : "+(double)
marks[i]/2 + sss[i], marks[i]));
}}
}.
```

Output:

Enter USN, name, current semester:

1BM21CS019

Amshu

3

Enter CIE marks of subject 1

47

Enter CIE marks of subject 2

46

Enter CIE marks of subject 3.

45

Enter CIE marks of subject 4

43

Enter CIE marks of subject 5

44

Enter SEE marks of subject 1

48

Enter SEE marks of subject 2

47

Enter SEE marks of subject 3

44

Enter SEE marks of Subject 4

49

Enter SEE marks of subject 5.

50

Student Details

Name : Anshu

USN : 1BM21CS019

Semester : 3

Total marks in subject 1 : 95

Total marks in subject 2 : 93

Total marks in subject 3 : 89

Total marks in subject 4 : 92

Total marks in subject 5 : 94.

OUTPUT:

```
C:\Users\amshu\OneDrive\Desktop\Downloads\OOJ-1BM21cs019--main (2)\OOJ-1BM21cs019--main>java Test
Enter number of students
1
Enter student details
1
amshu
3
Enter internal marks
50
50
48
48
49
Enter external marks
48
48
47
50
50
Student details
USN:1
NAME:amshu
SEMESTER:3
Internal marks
Subject 0: 50
Subject 1: 50
Subject 2: 48
Subject 3: 48
Subject 4: 49
External marks
Subject 0: 48
External marks
Subject 1: 48
External marks
Subject 2: 47
External marks
Subject 3: 50
External marks
Subject 4: 50
```

```
Student details
USN:1
NAME:amshu
SEMESTER:3
Internal marks
Subject 0: 50
Subject 1: 50
Subject 2: 48
Subject 3: 48
Subject 4: 49
External marks
Subject 0: 48
External marks
Subject 1: 48
External marks
Subject 2: 47
External marks
Subject 3: 50
External marks
Subject 4: 50
Final marks
Subject 0: 74
Subject 1: 74
Subject 2: 71
Subject 3: 73
Subject 4: 74
```