A LABORATORY REPORT FILE ON

JAVA PROGRAMMING LAB

BACHELOR OF TECHNOLOGY

SUBJECT CODE: CS102491; 4TH SEMESTER

GUIDED BY

Mr. Rajeshwar Kumar Dewangan

(Assistant Professor)

Computer Science & Engineering

SUBMITTED BY

Name:- .....

University Roll No:- .....

Enrollment No:- .....

Specialization:- .....

Section:- .....

Computer Science & Engineering

SESSION: 2023-24

ज्ञानादेव तु कैवल्यम्

(Shri Gangajali Education Society)

Established 1999

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Shri Shankaracharya Technical Campus, Bhilai

An Autonomous Institute

Approved by AICTE, New Delhi, Affiliated to

Chhattisgarh Swami Vivekanand Technical University, Bhilai

Khapri, Junwani Road, Bhilai, District-Durg, Chhattisgarh Pin Code- 490020, INDIA.
Ph. No.:0788-4088888, Fax No.: 0788-2298606, E-mail : sstc@sstc.ac.in, Web : www.sstc.ac.in

CERTIFICATE

Shri Shankaracharya Technical Campus, Bhilai

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

This is certify that Mr./Ms. ..... whose University Roll No. ....., Enrollment No. ..... is a student of B.TECH. (COMPUTER SCIENCE & ENGINEERING) – 4TH SEMESTER. Under Specialization .....and Section ..... at SHRI SHANKARACHARYA TECHNICAL CAMPUS, BHILAI CG INDIA. has completed his/ her Practical work in the SUBJECT :- JAVA PROGRAMMING LAB & SUBJECT CODE :- (CS102491) is accepted & approved after proper evaluation as a creditable work required as per the norms of CHHATTISGARH SWAMI VIVEKANAND TECHNICAL UNIVERSITY, BHILAI CG INDIA. in the laboratory of this college in the SESSION 2023-24 .

(Signature of the Guide)
Mr. Rajeshwar Kumar Dewangan
Assistant Professor (CSE)
SSTC, Bhilai

(Signature of the Student)
Place:-_____
Date:- _____

(Signature of the HOD)
Dr. (Mrs.) Samta Gajbhiye
Professor & Head of Department (CSE)
SSTC, Bhilai

ज्ञानादेव तु कैवल्यम्

Student Name :- ---------- **University Roll No. :-** ----------

Branch/ Semester :- B.Tech. (CSE) - 4th SEM **Enrollment No. :-** ----------

Subject/ Code :- Java Programming Lab (CS102491) **Specialization/ Section :-** ----------

INDEX

Sr. No.	List of Programs/ Experiment Description	Page No.	Date of Performing	Date of Submission	Signature/ Remarks
1.	Write a program to find the volume of a box having its side w, h, d means width, height and depth. Its volume is v=w*h*d and also find the surface area given by the formula s=2(wh+hd+dw). use appropriate constructors for the above.	1			
2.	Develop a program to illustrate a copy constructor so that a string may be duplicated into another variable either by assignment or copying.	2			
3.	Create a base class called shape A part from constructors, it contains two methods getxyvalue() and showxyvalue() for accepting co-ordinates and to display the same. create the sub class called rectangle which contains a method to display the length, breadth of the rectangle called showxyvalue(). illustrate the concepts of overriding & constructor call sequence.	3			
4.	Write a program that creates an abstract class called dimension, create two subclasses, rectangle and triangle. include appropriate methods for both the subclass that calculate and display the area of the rectangle and triangle.	4-5			
5.	Write a program, which throws arithmetic exception. write another class (in a different file) that handles the exception.	6			
6.	Create a user defined exception class which throws exception when the user inputs the marks greater than 100 catch it and again rethrow it.	7			
7.	Write a program to illustrate various string class methods.	8			
8.	Write a program to illustrate various string buffer class methods.	9			
9.	Write a program in which a mythread class is created by extending the thread class. in another class, create objects of the mythread class and run them. in the run method print "CSVTU" 10 times. identify each thread by setting the name.	10			
10.	Write a program to illustrate various thread methods and constructors.	11-12			
11.	Write a program to implement bank account class which illustrates the concept of thread synchronization.	13-14			
12.	To write a program to create a text file using byte stream class.	15			
13.	To write a program to copy contents of one file to another.	16-17			
14.	Write a program to find number of occurrences of vowels and consonants in a file.	18			
15.	Write a program, which illustrates capturing of mouse events. use applet for this.	19-20			
16.	Write a program using swing components which simulates simple calculator.	21-24			
17.	Write a JDBC program for student mark list processing.	25-27			

Program No-01

Aim: Write a program to find the volume of a box having its side w, h, d means width, height and depth. Its volume is v=w*h*d and also find the surface area given by the formula s=2(wh+hd+dw). Use appropriate constructors for the above.

Code: -

```
class Box {
    double width;
    double height;
    double depth;
    // This is the constructor for Box.
    Box(double w, double h, double d) {
        width = w;
        height = h;
        depth = d;
    }
    // compute and return volume - Volume of box
    double volume() {
        return width * height * depth;
    }
    // compute and return volume - surface area
    double surface() {
        return (2*((width * height) + (height * depth) +
(depth * width)));
    }
}
public class boxVolumeSurface {
    public static void main(String[] args) {
        // declare, allocate, and initialize Box objects
        Box myBox1 = new Box(10, 20, 15);
        Box myBox2 = new Box(3, 6, 9);

        double vol1, sur1, vol2, sur2;
        // get volume of first box
        vol1 = myBox1.volume();
        sur1 = myBox1.surface();
        System.out.println("Volume of first box " + vol1);
        System.out.println("Surface of first box " + sur1);

        // get volume of second box
        vol2 = myBox2.volume();
        sur2 = myBox2.surface();
        System.out.println("Volume of second box " + vol2);
        System.out.println("Surface of second box " + sur2);
    }
}
```

Output of Program No-01: -

Volume of first box 3000.0
Surface of first box 1300.0
Volume of second box 162.0
Surface of second box 198.0

Program No-02

Aim: Develop a program to illustrate a copy constructor so that a string may be duplicated into another variable either by assignment or copying.

Code: -

```
public class copyConstructor {
    private double realNo, imaginaryNo;

    // A normal parametrized constructor
    public copyConstructor(double realNo, double imaginaryNo){
        this.realNo = realNo;
        this.imaginaryNo = imaginaryNo;
    }

    // copy constructor
    public copyConstructor(copyConstructor copy) {
        System.out.println("Copy constructor called");
        realNo = copy.realNo;
        imaginaryNo = copy.imaginaryNo;
    }

    public static void main(String[] args) {
        copyConstructor constructor = new copyConstructor(10,
15.34);
        System.out.println(constructor.realNo+ " + "
+constructor.imaginaryNo + "i");

        // Following involves a copy constructor call
        copyConstructor copy = new
copyConstructor(constructor);

        // Note that following doesn't involve a copy
constructor call as
        // non-primitive variables are just references.
        System.out.println(copy.realNo+ " + "
+copy.imaginaryNo + "i");
    }
}
```

Output of Program No-02: -

10.0 + 15.34i
Copy constructor called
10.0 + 15.34i

Program No-03

Aim: Create a base class called Shape. A part from Constructors, It contains two methods getxyvalue() and showxyvalue() for accepting co-ordinates and to display the same. Create the sub class called Rectangle which contains a method to display the length and breadth of the rectangle called showxyvalue(). Illustrate the concepts of Overriding and Constructor call sequence.

Code: -

```
import java.util.*;
class shape { //Base class
    int x,y;
    void getxyvalue(){
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter value of x: ");
        x=sc.nextInt();
        System.out.print("Enter value of y: ");
        y=sc.nextInt();
    }
    void showxyvalue(){
        System.out.println("Co-ordinate x: "+x);
        System.out.println("Co-ordinate y: "+y);
    }
}
class rectangle extends shape{ //Subclass
    int length, breadth;
    void getxyvalue(){ //Method overriding
        Scanner sc= new Scanner(System.in);
        System.out.print("Enter value of length: ");
        length=sc.nextInt();
        System.out.print("Enter value of breadth: ");
        breadth=sc.nextInt();
    }
    void showxyvalue(){ //Method overriding
        System.out.println("Length of Rectangle: "+length);
        System.out.println("Breadth of Rectangle: "+breadth);
    }
    public static void main(String args[]){
        shape s = new shape();
        s.getxyvalue();
        s.showxyvalue();
        shape r = new rectangle();
        r.getxyvalue();
        r.showxyvalue();
    }
}
```

Output of Program No-03: -

Enter value of x: 25
Enter value of y: 50
Co-ordinate x: 25
Co-ordinate y: 50
Enter value of length: 25
Enter value of breadth: 50
Length of Rectangle: 25
Breadth of Rectangle: 50

Program No-04

Aim: Write a program that creates an abstract class called dimension, create two subclasses, rectangle and triangle. Include appropriate methods for both the subclass that calculate and display the area of the rectangle and triangle.

Code: -

```
abstract class dimension {
    private double height;  // To hold height.
    private double width;  //To hold width or base

    public void setValues(double height, double width) { //
Set height and width
        this.height = height;
        this.width = width;
    }

    public double getHeight() { //Get height
         return height;
    }

    public double getWidth() { //Get width
         return width;
    }
    // The getArea method is abstract. It must be overridden
in a subclass.
    abstract public double getArea();
}

class RectangleSubClass extends dimension { //This class
Rectangle calculates the area of rectangle

    public double getArea() {
        return getHeight() * getWidth(); //Calculate and
return area of rectangle
    }
}

class TriangleSubClass extends dimension { //This class
Triangle calculates the area of triangle.

    public double getArea() {
        return (getHeight() * getWidth()) / 2; //Calculate and
return area of triangle
    }
}
public class abstractDisplayOfArea {
    public static void main(String[] args) {
        dimension dimension;
```

```
        RectangleSubClass rect = new RectangleSubClass(); //
assign subclass reference to subclass variable

        dimension = rect; // shape points to the object rect.
        dimension.setValues(78, 5); // Set data in Rectangle's
object
        System.out.println("Area of rectangle : " +
dimension.getArea()); //Display the area of rectangle


        TriangleSubClass tri = new TriangleSubClass(); //
assign subclass reference to subclass variable
        dimension = tri; // shape points to the object rect.
        dimension.setValues(34,3); // Set data in Triangle's
object
        System.out.println("Area of triangle : " +
dimension.getArea()); //Display the area of triangle
    }
}
```

Output of Program No-04: -

Area of rectangle : 390.0
Area of triangle : 51.0

Program No-05

Aim: Write a program, which throws Arithmetic Exception. Write another class (in a different file) that handles the Exception.

Code: -

```
public class arithmeticExceptionHandler {
    public void exceptionMassages (){
        try{

            int num1=30, num2=0;
            int output=num1/num2;
            System.out.println ("Result: "+output);
        }
        catch(ArithmeticException e){
            System.out.println ("You Shouldn't divide a number
by zero");
        }
    }
}
```

```
public class arithmeticException {
    public static void main(String[] args) {
        arithmeticExceptionHandler result = new
arithmeticExceptionHandler();
        result.exceptionMassages();

    }
}
```

Output of Program No-05: -

You Shouldn't divide a number by zero

Program No-06

Aim: Create a user defined Exception class which throws Exception when the user inputs the marks greater than 100 Catch it and again rethrow it.

Code: -

```
import java.util.*;

class myException extends Exception {
    myException(String s) {
        super(s);
    }
}

class userDefineException {
    public static void main(String argv[])throws myException
{

        System.out.print("Enter marks: ");
        Scanner sc = new Scanner(System.in);
        int marks=sc.nextInt();

        if( marks > 100 ) {
            throw new myException ("Input marks greater than
100 ");
        }
        else
            System.out.println("Marks: "+marks);
    }
}
```

Output of Program No-06: -

Enter marks: 105
Exception in thread "main" myException: Input marks greater than 100

Program No-07

Aim: Write a program to illustrate various String class methods.

Code: -

```
public class stringMethods {
    public static void main(String[] args) {
        String name = "Happy";
        System.out.println(name);
        int value = name.length();
        System.out.println(value);
        String lstring = name.toLowerCase();
        System.out.println(lstring);
        String ustring = name.toUpperCase();
        System.out.println(ustring);
        String nonTrimmedString = "     Happy     ";
        System.out.println(nonTrimmedString);
        String trimmedString = nonTrimmedString.trim();
        System.out.println(trimmedString);
        System.out.println(name.substring(1));
        System.out.println(name.substring(1,5));
        System.out.println(name.replace('p', 'r'));
        System.out.println(name.replace("p", "ier"));
        System.out.println(name.startsWith("Har"));
        System.out.println(name.endsWith("dd"));
        System.out.println(name.charAt(4));
        String modifiedName = "Harryrryrry";
        System.out.println(modifiedName.indexOf("rry"));
        System.out.println(modifiedName.indexOf("rry", 4));
        System.out.println(modifiedName.lastIndexOf("rry",
7));

        System.out.println(name.equals("Harry"));
        System.out.println(name.equalsIgnoreCase("HarRY"));
        System.out.println("I am escape sequence\tdouble
quote");
    }
}
```

Output of Program No-07: -

```
Happy                                   false
5                                       false
happy                                   y
HAPPY                                   2
    Happy                               5
Happy                                   5
appy                                    false
appy                                    false
Harry                                   I am escape sequence  double quote
Haieriery
```

Program No-08

Aim: Write a program to illustrate various String Buffer class methods.

Code: -

```
public class stringBuffer {
    public static void main(String[] args) {
        StringBuffer s1 = new StringBuffer("Deepak");
        System.out.println("String Length :" + s1.length());
        System.out.println("String Capacity :" +
s1.capacity()); // default capacity 16 + string length
        System.out.println(s1.append(" Verma"));
        System.out.println(s1.charAt(2));

        StringBuffer s2 = new StringBuffer("Deepak");
        System.out.println(s2.delete(3, 6));

        StringBuffer s3 = new StringBuffer("Deepak");
        System.out.println(s3.reverse());

        StringBuffer s4 = new StringBuffer("Deepak");
        System.out.println(s4.insert(0, "Mr "));
        System.out.println(s4.deleteCharAt(1));

        StringBuffer s5 = new StringBuffer("deepak");  //
match to return value 0
        System.out.println(s1.compareTo(s5));
    }
}
```

Output of Program No-08: -

String Length :6

String Capacity :22

Deepak Verma

e

Dee

kapeeD

Mr Deepak

M Deepak

-32

Program No-09

Aim: Write a program in which a Mythread class is created by extending the Thread class. In another class, create objects of the Mythread class and run them. In the run method print "CSVTU" 10 times. Identify each thread by setting the name.

Code: -

```
public class CSVTUmyThread extends Thread{

    public void run(){
        for (int i=1; i<=10; i++){

            //Display CSVTU 10 times from this thread
            System.out.println("CSVTU");
            try {
                Thread.sleep(1000);
            }

            catch (InterruptedException interruptedException){
                System.out.println( "First Thread is
interrupted when it is sleeping" + interruptedException);
            }
        }
    }

    public static void main(String args[]){

        CSVTUmyThread t=new CSVTUmyThread();
        System.out.println("Thread will display CSVTU 10
times!");
        t.start(); // to start thread
    }
}
```

Output of Program No-09: -

Thread will display CSVTU 10 times!
CSVTU
CSVTU
CSVTU
CSVTU
CSVTU
CSVTU
CSVTU
CSVTU
CSVTU
CSVTU

Program No-10

Aim: Write a program to illustrate various Thread methods and Constructors.

Code: -

```
class myThread1 extends Thread{
    public myThread1 (String name){ //thread constructor
        super(name);
    }
    @Override
    public void run(){
        System.out.println("My Thread first is Running");
    }
}
class myThread2 extends Thread{
    public myThread2 (Runnable target, String name){ //thread
constructor
        super(target, name);
    }
    @Override
    public void run(){
        System.out.println("My Thread second is Running");
    }
}
public class threadConstructorsMethods {
    public static void main(String[] args) {
        myThread1 t1 = new myThread1("SRK");
        myThread2 t2 = new myThread2(t1,"AJD");
        t1.start();
        t2.start();
        System.out.println("Thread t1 is true=live or
false=dead & ans is " + t1.isAlive());
        System.out.println("The ID of the thread t1 is " +
t1.getId());
        System.out.println("The Name of the thread t1 is " +
t1.getName());
        System.out.println("The Priority of the thread t1 is "
+ t1.getPriority());
        System.out.println("The ID of the thread t2 is " +
t2.getId());
        System.out.println("The Name of the thread t2 is " +
t2.getName());
        System.out.println("The Priority of the thread t2 is "
+ t2.getPriority());
        System.out.println("Thread t2 is true=live or
false=dead & ans is " + t2.isAlive());
    }
}
```

Output of Program No-10: -

My Thread first is Running
My Thread second is Running
Thread t1 is true=live or false=dead & ans is true
The ID of the thread t1 is 16
The Name of the thread t1 is SRK
The Priority of the thread t1 is 5
The ID of the thread t2 is 17
The Name of the thread t2 is AJD
The Priority of the thread t2 is 5
Thread t2 is true=live or false=dead & ans is false

Program No-11

Aim: Write a Program to implement Bank Account Class which illustrates the concept of Thread Synchronization.

Code: -

```
class bankAccount {
    int balance= 1000;
    public int getBalance(){
        return balance;
    }
    public void withdraw(int amount){
        balance= balance-amount;
    }
    public void deposit(int amount){
        balance= balance+amount;
    }
}
public class bankThreadSync implements Runnable{

    bankAccount jointAccount = new bankAccount();

    @Override
    public void run() {
        for (int i = 3; i <= 3; i++) {
            withdraw(100);
            if (jointAccount.getBalance() < 0) {
                System.out.println("account is overdrawn!");
            }
            deposit(200);
        }
    }

    private synchronized void withdraw(int amount){
        if (jointAccount.getBalance()>=amount) {
            jointAccount.withdraw(amount);

System.out.println(Thread.currentThread().getName()+" "+ "is
complete the withdraw of Rs." + amount);

System.out.println(Thread.currentThread().getName()+" after
withdrawal joint account balance Rs. " +
jointAccount.getBalance());
            try {
                Thread.sleep(100);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }else{

System.out.println(Thread.currentThread().getName()+ "
"+"doesn't have enough amount for withdraw ");
```

```
        }
    }

    private synchronized void deposit(int amount){
        if (amount>=0) {
            jointAccount.deposit(amount);

System.out.println(Thread.currentThread().getName()+" "+ "is
complete the deposited of Rs." + amount);

System.out.println(Thread.currentThread().getName()+" after
deposit joint account balance Rs. " +
jointAccount.getBalance());
            try {
                Thread.sleep(100);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }else{

System.out.println(Thread.currentThread().getName()+ " "+"you
have invalid amount for deposit");
        }
    }

    public static void main(String[] args) {
        bankThreadSync ts = new bankThreadSync();
        Thread t1 = new Thread(ts, "Naresh");
        Thread t2 = new Thread(ts, "Mahesh");
        Thread t3 = new Thread(ts, "Suresh");
        t1.start();
        t2.start();
        t3.start();
    }
}
```

Output of Program No-11: -

Suresh is complete the withdraw of Rs.100
Suresh after withdrawal joint account balance Rs. 900
Mahesh is complete the withdraw of Rs.100
Mahesh after withdrawal joint account balance Rs. 800
Mahesh is complete the deposited of Rs.200
Mahesh after deposit joint account balance Rs. 1000
Naresh is complete the withdraw of Rs.100
Naresh after withdrawal joint account balance Rs. 900
Naresh is complete the deposited of Rs.200
Naresh after deposit joint account balance Rs. 1100
Suresh is complete the deposited of Rs.200
Suresh after deposit joint account balance Rs. 1300

Program No-12

Aim: To write a program to create a text file using Byte Stream class.

Code: -

```
import java.io.*;
public class fileCreate2ByteStream {
    public static void main(String args[])
    {
        // Try block to check if any exception/s occur
        try {

            // Step 1: Creating object of the file and
            // passing local directory path of file as input
            FileOutputStream myFile  = new
FileOutputStream("myFile.txt");

            // Custom text to be written down in above file
            // Step 2: Storing text into String datatype
            String s = "Welcome to Program No. 12. This is an
example of Java program to write file using ByteStream.";
            System.out.println("File is created with writing
successfully to help of ByteStream class");

            // Step 3: Converting string into byte array
            byte b[] = s.getBytes();

            // Step 4: Write byte data to file output
            myFile.write(b);

            // Step 5: Close the file using close() method
            myFile.close();
        }

        // Catch block to handle exceptions
        catch (IOException e) {

            // Display and print the exception
            System.out.println(e);
        }
    }
}
```

Output of Program No-12: -

File is created with writing successfully to help of ByteStream class

myFile.txt: -

Welcome to Program No. 12. This is an example of Java program to write file using ByteStream.

Program No-13

Aim: To write a program to copy contents of one file to another.

Code: -

```
import java.io.*;
import java.util.*;
public class CopyFromFileaToFileb {
    public static void copyContent(File a, File b) throws
Exception {
        FileInputStream in = new FileInputStream(a);
        FileOutputStream out = new FileOutputStream(b);
        try {
            int n;
            // read() function to read the byte of data
            while ((n = in.read()) != -1) {

                // write() function to write the byte of data
                out.write(n);
            }
        }
        finally {
            if (in != null) {
                // close() function to close the stream
                in.close();
            }
            // close() function to close the stream
            if (out != null) {
                out.close();
            }
        }
        System.out.println("File Copied");
    }
    public static void main(String[] args) throws Exception
    {
        Scanner sc = new Scanner(System.in);
        // get the source file name
        System.out.println("Enter the source filename from
where you have to read/copy :");
        String a = sc.nextLine();
        // source file
        File x = new File(a);
        // get the destination file name
        System.out.println("Enter the destination filename
where you have to write/paste :");
        String b = sc.nextLine();
        // destination file
        File y = new File(b);
        // method called to copy the contents from x to y
        copyContent(x, y);
    }
}
```

Output of Program No-13: -

Enter the source filename from where you have to read/copy :
sourcefile.txt
Enter the destination filename where you have to write/paste :
destinationfile.txt
File Copied

sourcefile.txt: -
This content is copied to destination file.

destinationfile.txt: -
This content is copied to destination file.

Program No-14

Aim: Write a program to find number of occurrences of vowels and consonants in a file.

Code: -

```
import java.util.Scanner;
import java.io.*;
public class CountingVowelsConsonants {
    public static void main(String[] args) {
 //Counter variable to store the count of vowels and consonant
        int vCount = 0, cCount = 0;
        try {
            FileReader fileReading = new
FileReader("sentence.txt");
            Scanner sc = new Scanner(fileReading);
            while(sc.hasNextLine()){
                //Reading a file declared as string
                String sentence = sc.nextLine();
                System.out.println("The sentence is  :\n"
+sentence);
                for (int i=0 ; i<sentence.length(); i++){
                    char ch = sentence.charAt(i);
                    //Checks whether a character is a vowel
                    if(ch == 'a'|| ch == 'e'|| ch == 'i' ||ch
== 'o' ||ch == 'u'){
                    //Increments the vowel counter
                        vCount ++;
                    }
                    //Checks whether a character is a consonant
                    else if(ch >= 'a' && ch<='z') {
                    //Increments the consonant counter
                        cCount++;
                    }
                }
            }
            sc.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
        System.out.println("Number of vowels in the given
sentence is: "+vCount);
        System.out.println("Number of consonants in the given
sentence is: " + cCount);
    }
}
```

Output of Program No-14: -

The sentence is:
java is a high level, class based, object-oriented programming language that is designed to have as few implementation dependencies as possible.
Number of vowels in the given sentence is: 48
Number of consonants in the given sentence is: 72

Program No-15

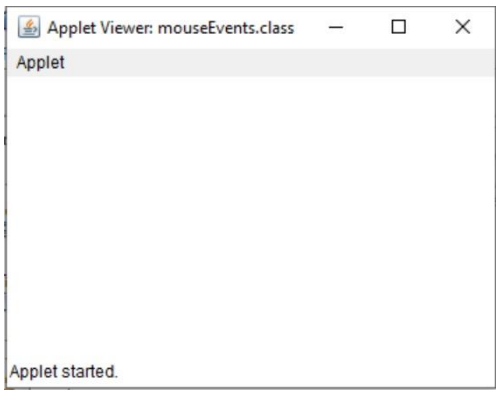
Aim: Write a program, which illustrates capturing of Mouse Events. Use Applet for this.

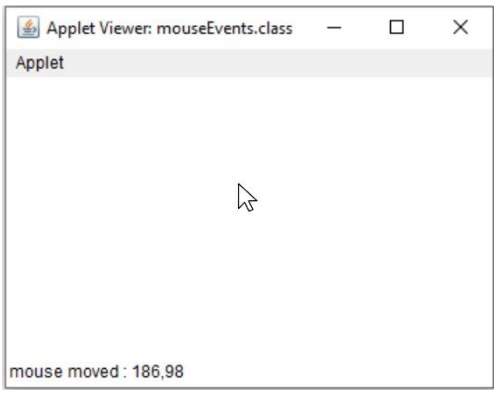
Code: -

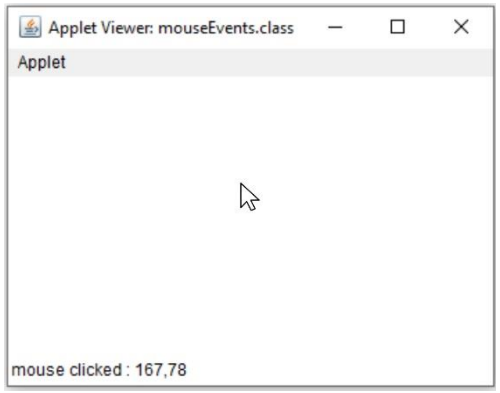
```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;

public class mouseEvents extends Applet implements
MouseListener, MouseMotionListener{
    public void init(){
        addMouseListener(this);
        addMouseMotionListener(this);
    }
    public void mouseEntered(MouseEvent e){
        showStatus("mouse entered : "+e.getX()+","+e.getY());
        for(int i=0;i<100000;i++);
    }
    public void mouseMoved(MouseEvent e){
        showStatus("mouse moved : "+e.getX()+","+e.getY());
    }
    public void mouseClicked(MouseEvent e){
        showStatus("mouse clicked : "+e.getX()+","+e.getY());
    }
    public void mousePressed(MouseEvent e){
        showStatus("mouse pressed : "+e.getX()+","+e.getY());
    }
    public void mouseReleased(MouseEvent e){
        showStatus("mouse released : "+e.getX()+","+e.getY());
    }
    public void mouseDragged(MouseEvent e){
        showStatus("mouse dragged : "+e.getX()+","+e.getY());
    }
    public void mouseExited(MouseEvent e){
        showStatus("mouse exited : "+e.getX()+","+e.getY());
    }
}
/* <body>
   <applet code = "mouseEvents.class" height = 200 width=200>
</applet>
   </body>  */
```

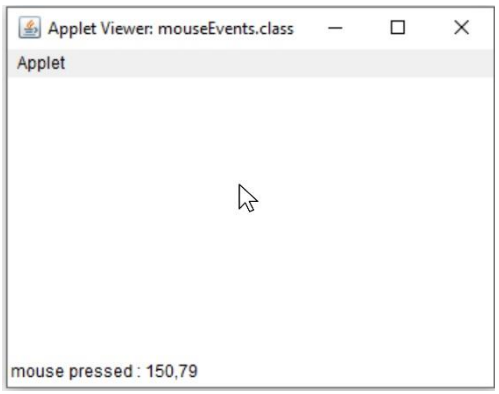
Output of Program No-15: -

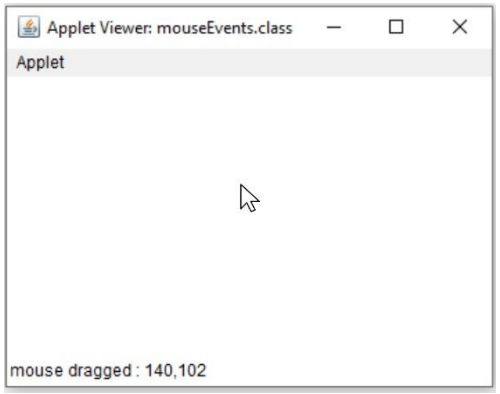
Remarks: All applet GUI runs on java jdk 8 (32bit) and older version. Because java.applet is deprecated with no replacement and marked for removal.

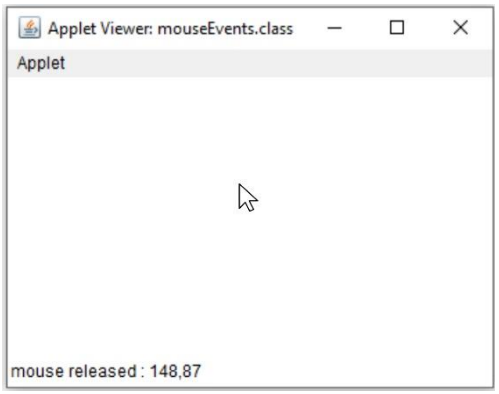


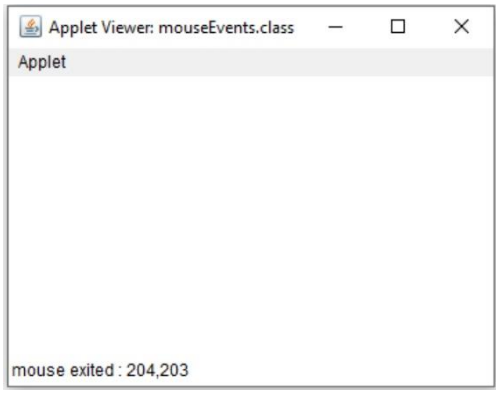












Program No-16

Aim: Write a program using swing components which simulates simple calculator.

Code: -

```
import javax.swing.*;
import java.awt.event.*;
public class SwingCalci implements ActionListener {
    JFrame frame;
    JTextField textField;
    JButton button1, button2, button3, button4, button5,
button6, button7, button8, button9, button0, buttonDot,
buttonAdd, buttonSub, buttonMul, buttonDiv, buttonEql,
buttonDel, buttonClr;
    double a, b, result;
    int operator;
    SwingCalci(){
        frame=new JFrame("Swing-Calculator");
        frame.setLayout(null);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setBounds(20, 20, 300, 350);
        frame.setResizable(false);
        textField=new JTextField();
        textField.setBounds(40,40,200,50);
        button1=new JButton("1");
        button2=new JButton("2");
        button3=new JButton("3");
        button4=new JButton("4");
        button5=new JButton("5");
        button6=new JButton("6");
        button7=new JButton("7");
        button8=new JButton("8");
        button9=new JButton("9");
        button0=new JButton("0");
        buttonDot=new JButton(".");
        buttonAdd =new JButton("+");
        buttonSub=new JButton("-");
        buttonMul=new JButton("*");
        buttonDiv=new JButton("/");
        buttonEql =new JButton("=");
        buttonDel=new JButton("Delete");
        buttonClr=new JButton("Clear");

        button7.setBounds(40,100,50,40);
        button8.setBounds(90,100,50,40);
        button9.setBounds(140,100,50,40);
        buttonDiv.setBounds(190,100,50,40);
        button4.setBounds(40,140,50,40);
        button5.setBounds(90,140,50,40);
        button6.setBounds(140,140,50,40);
        buttonMul.setBounds(190,140,50,40);
```

```
button1.setBounds(40,180,50,40);
button2.setBounds(90,180,50,40);
button3.setBounds(140,180,50,40);
buttonSub.setBounds(190,180,50,40);
buttonDot.setBounds(40,220,50,40);
button0.setBounds(90,220,50,40);
buttonEql.setBounds(140,220,50,40);
buttonAdd.setBounds(190,220,50,40);

buttonDel.setBounds(40,260,100,40);
buttonClr.setBounds(140,260,100,40);

frame.add(textField);
frame.add(button1);
frame.add(button2);
frame.add(button3);
frame.add(button4);
frame.add(button5);
frame.add(button6);
frame.add(button7);
frame.add(button8);
frame.add(button9);
frame.add(button0);
frame.add(buttonDot);
frame.add(buttonAdd);
frame.add(buttonSub);
frame.add(buttonMul);
frame.add(buttonDiv);
frame.add(buttonEql);
frame.add(buttonDel);
frame.add(buttonClr);

button1.addActionListener(this);
button2.addActionListener(this);
button3.addActionListener(this);
button4.addActionListener(this);
button5.addActionListener(this);
button6.addActionListener(this);
button7.addActionListener(this);
button8.addActionListener(this);
button9.addActionListener(this);
button0.addActionListener(this);
buttonDot.addActionListener(this);
buttonAdd.addActionListener(this);
buttonSub.addActionListener(this);
buttonMul.addActionListener(this);
buttonDiv.addActionListener(this);
buttonEql.addActionListener(this);
buttonDel.addActionListener(this);
buttonClr.addActionListener(this);
frame.setVisible(true);
```

```
}

public static void main(String[] args) {
    new SwingCalci();
}
@Override
public void actionPerformed(ActionEvent e) {
    Object source = e.getSource();
    if (source==button1){
        textField.setText(textField.getText() + "1");
    } else if (source==button2){
        textField.setText(textField.getText() + "2");
    } else if (source==button3){
        textField.setText(textField.getText() + "3");
    } else if (source==button4){
        textField.setText(textField.getText() + "4");
    } else if (source==button5){
        textField.setText(textField.getText() + "5");
    } else if (source==button6){
        textField.setText(textField.getText() + "6");
    } else if (source==button7){
        textField.setText(textField.getText() + "7");
    } else if (source==button8){
        textField.setText(textField.getText() + "8");
    } else if (source==button9){
        textField.setText(textField.getText() + "9");
    } else if (source==button0){
        textField.setText(textField.getText() + "0");
    }
    else if (source == buttonDot) {
        if (textField.getText().contains(".")) {
            return;
        } else {
            textField.setText(textField.getText() + ".");
        }
    }
    else if (source==buttonAdd) {
        a = Double.parseDouble(textField.getText());
        textField.setText("");
        operator = 1;
    } else if (source==buttonSub) {
        a = Double.parseDouble(textField.getText());
        textField.setText("");
        operator = 2;
    } else if (source==buttonMul) {
        a = Double.parseDouble(textField.getText());
        textField.setText("");
        operator = 3;
    } else if (source==buttonDiv) {
        a = Double.parseDouble(textField.getText());
        textField.setText("");
```

```
            operator = 4;
        }

        else if (source==buttonEql) {
            b = Double.parseDouble(textField.getText());
            switch (operator){
                case 1:result =a+b;
                    break;
                case 2:result =a-b;
                    break;
                case 3:result =a*b;
                    break;
                case 4:result =a/b;
                    break;
            }
            textField.setText(""+result);
        }
        else if (source==buttonClr){
            textField.setText("");
        }
        else if (source==buttonDel){
            String s=textField.getText();
            textField.setText("");
            for (int i=0; i<s.length()-1; i++)

textField.setText(textField.getText()+s.charAt(i));
        }
    }
}
```

Output of Program No-16: -

Program No-17

Aim: Write a JDBC program for Student Mark List Processing.

Code: -

```
/*In order to deal with JDBC standard 7 steps are supposed to
be followed:
    1. Import or create the database
    2. Load and register drivers
    3. Create the connection
    4. Create a statement/query
    5. Execute the statement/query
    6. Process the results
    7. Close the connection*/
// JDBC program for Student Mark Insert and Display
Processing.
import java.sql.*;
import java.util.Scanner;
public class studentsMarksJDBC {
    public static void main(String[] args) throws Exception {
        // Load and register driver (mysql-connector-java-
8.0.28.jar) file on this project
        Class.forName("com.mysql.cj.jdbc.Driver");
        // Create the connection
        String user = "root";
        String password = "password";
        String url =
"jdbc:mysql://localhost:3306/students_marks";
        Connection con = DriverManager.getConnection(url,
user, password);
        // Check the database connection
        if (con.isClosed()) {
            System.out.println("Database connection is
closed");
        } else {
            System.out.println("Database connection created
successfully...");
        }
        while (true) {
            Scanner intVal = new Scanner(System.in); // for
integer value
            Scanner strVal = new Scanner(System.in); // for
string value
            System.out.println("PRESS 1 to ADD student name
and marks");
            System.out.println("PRESS 2 to SHOW student name
and marks");
            System.out.println("PRESS 0 for EXIT program");
            int choice = intVal.nextInt();

            if (choice == 1) {
                // ADD student name and marks
```

```
                System.out.println("Enter the student name
:");
                String sname = strVal.nextLine();
                System.out.println("Enter the C++ obtained
marks :");
                int cpp = intVal.nextInt();
                System.out.println("Enter the JAVA obtained
marks :");
                int java = intVal.nextInt();
                System.out.println("Enter the PYTHON obtained
marks :");
                int python = intVal.nextInt();
                System.out.println("Enter the MySQL obtained
marks :");
                int mysql = intVal.nextInt();

                // Create a statement/query
                String query = "INSERT INTO
students_marks(sname, cpp, java, python, mysql)
VALUES(?,?,?,?,?)";
                // PreparedStatement
                PreparedStatement pstmt =
con.prepareStatement(query);
                // Set the value of parameter
                pstmt.setString(1, sname);
                pstmt.setInt(2, cpp);
                pstmt.setInt(3, java);
                pstmt.setInt(4, python);
                pstmt.setInt(5, mysql);

                // Execute the statement/query
                pstmt.executeUpdate();
                System.out.println("Data inserted
successfully....");

                // Close the database connection
                con.close();
                break; // Exit the loop

            } else if (choice == 2) {
                // SHOW student name and marks

                String query = "SELECT * FROM
students_marks;";
                // CreateStatement
                Statement pstmt = con.createStatement();
                // Set the value in ResultSet
                ResultSet set = pstmt.executeQuery(query);

                // Process the results
                while (set.next()) {
```

```
                    int sid = set.getInt(1);
                    String sname = set.getString(2);
                    int cpp = set.getInt(3);
                    int java = set.getInt(4);
                    int python = set.getInt(5);
                    int mysql = set.getInt(6);

                    System.out.println("Student ID :" + sid);
                    System.out.println("Student Name :" +
sname);
                    System.out.println("C++ Marks :" + cpp);
                    System.out.println("JAVA Marks :" + java);
                    System.out.println("PYTHON Marks :" +
python);
                    System.out.println("MySQL Marks :" +
mysql);
System.out.println("========================================");
                }
                // Close the database connection
                con.close();
                break; // Exit the loop
            } else if (choice == 0) {
                // EXIT program
                break;
            }
        }
    }
}
```

Output of Program No-17: -