CORE JAVA PRACTICALS

NAME- SIDDHESHWAR SARANG

CLASS - SYIT

ROLL NO - 2041041

DATE - 08-04-2021



VIDYA PRASARAK MANDAL’ S

COLLEGE OF ARTS, SCIENCE AND COMMERCE

MITHAGAR ROAD, MULUND (E), MUMBAI -400 081

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**& COMPUTER SCIENCE**

CERTIFICATE

This is to certify that MR**.SARANG SIDDHESHWAR OMPRAKASH Seat** No.**2041041** has completed the practicals in the subject of Core Java in partial fulfillment of **B.Sc (IT)  SEM - IV.**

It is further certified that he/she has completed all required  practicals of the subject successfully.

**SIGNATURE OF PRACTICAL INCHARGE SIGNATURE OF H.O.D/**

**CO- ORDINATOR**

|  |  |
| --- | --- |
| **List of Practicals** | |
| **1.** | **Java Basics** |
| a. | Write a Java program that takes a number as input and prints its multiplication  table upto 10. |
| b. | Write a Java program to display the following pattern.  \*\*\*\*\*  \*\*\*\*   \*\*\*   \*\*   \* |
| c. | Write a Java program to print the area and perimeter of a circle. |
|  |  |
| **2.** | **Use of Operators** |
| a. | Write a Java program to add two binary numbers. |
| b. | Write a Java program to convert a decimal number to binary number and vice  versa. |
| c. | Write a Java program to reverse a string. |
|  |  |
| **3.** | **Java Data Types** |
| a. | Write a Java program to count the letters, spaces, numbers and other characters of  an input string. |
| b. | Implement a Java function that calculates the sum of digits for a given char array  consisting of the digits '0' to '9'. The function should return the digit sum as a long  value. |
| c. | Find the smallest and largest element from the array |
|  |  |
| **4.** | **Methods and Constructors** |
| a. | Designed a class SortData that contains the method asec() and desc(). |
| b. | Designed a class that demonstrates the use of constructor and destructor. |
| c. | Write a java program to demonstrate the implementation of abstract class. |
|  |  |

|  |  |
| --- | --- |
| **5.** | **Inheritance** |
| a. | Write a java program to implement single level inheritance. |
| b. | Write a java program to implement method overriding |
| c. | Write a java program to implement multiple inheritance. |
|  |  |
| **6.** | **Packages and Arrays** |
| a. | Create a package, Add the necessary classes and import the package in java class. |
| b. | Write a java program to add two matrices and print the resultant matrix. |
| c. | Write a java program for multiplying two matrices and print the product for the  same. |
|  |  |
| **7.** | **Vectors and Multithreading** |
| a. | Write a java program to implement the vectors. |
| b. | Write a java program to implement thread life cycle. |
| c. | Write a java program to implement multithreading. |
|  |  |
| **8.** | **File Handling** |
| a. | Write a java program to open a file and display the contents in the console  window. |
| b. | Write a java program to copy the contents from one file to other file. |
| c. | Write a java program to read the student data from user and store it in the file. |
|  |  |
| **9.** | **GUI and Exception Handling** |
| a. | Design a AWT program to print the factorial for an input value. |
| b. | Design an AWT programto perform various string operations like reverse string,  string concatenation etc. |
| c. | Write a java program to implement exception handling. |
|  |  |
| **10.** | **GUI Programming.** |
| a. | Design an AWT application that contains the interface to add student information  and display the same. |
| b. | Design a calculator based on AWT application. |
| c. | Design an AWT application to generate result marks sheet. |
|  |  |

**PRACTICAL NO 1**

**PRACTICAL NO 1 (A)**

**CODE-**

import java.util.Scanner;

public class PR1A

{

public static void main(String[] args)

{

Scanner s = new Scanner(System.in);

System.out.print("enter number: ");

int n=s.nextInt();

for(int i=1; i<=10; i++)

{

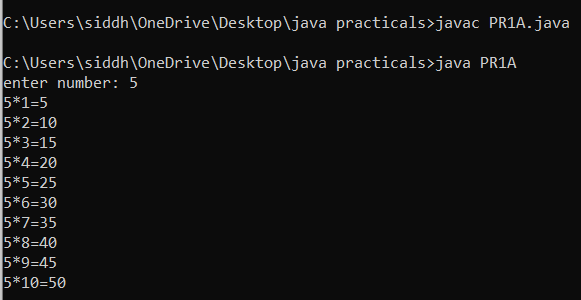
System.out.Println (n+"\*"+i+"="+n\*i);

}

}

}

**OUTPUT-**



**PRACTICAL NO 1 (B)**

**CODE-**

import java.util.Scanner;

public class PR1B

{

public static void main (String[] args)

{

for (int i=5; i>=1; i--)

{

for (int j=1; j<=i; j++)

{

System.out.print("\*");

}

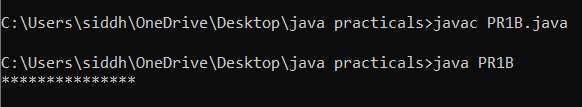
}

System.out.println();

}

}

**OUTPUT-**



**PRACTICAL NO 1 (C)**

**CODE-**

import java.util.Scanner;

public class PR1C

{

public static void main(String[] args)

{

double r=1.2f;

double perimeter=2\*3.14\*r;

double area=3.14\*r\*r;

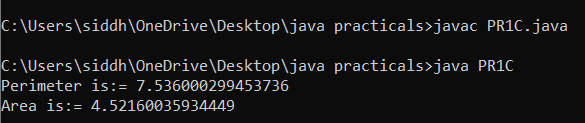
System.out.println("Perimeter is:= "+perimeter);

System.out.println("Area is:= "+area);

}

}

**OUTPUT-**



**PRACTICAL NO 2:**

**PRACTICAL NO 2 (A)**

**CODE-**

import java.util.Scanner;

public class PR2A{

public static void main (String[] args)

{

long binary1, binary2;

int i = 0, remainder = 0;

int[] sum = new int [20];

Scanner in = new Scanner (System.in);

System.out.print("Input first binary number:");

binary1 = in.nextLong();

System.out.print("Input second binary number:");

binary2 = in.nextLong();

while (binary1 != 0 || binary2 !=0)

{

sum[i++] = (int)((binary1 % 10 + binary2 % 10 + remainder)% 2);

remainder = (int)((binary1 % 10 + binary2 % 10 + remainder)/ 2);

binary1 = binary1 / 10;

binary2 = binary2 / 10;

}

if (remainder !=0){

sum[i++] = remainder;

}

--i;

System.out.print("sum of two binary numbers:");

while(i >= 0){

System.out.print(sum[i--]);

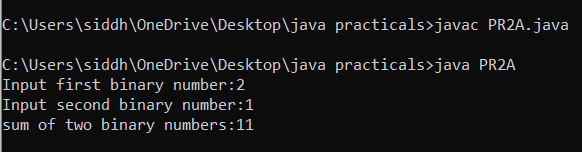
}

System.out.print("\n");

}

}

**OUTPUT-**



**PRACTICAL NO 2 (B)(1)**

**CODE-**

import java.io.\*;

class PR2B1

{

//function to convert decimal to binary

static void decToBinary(int n)

{

//array to store binary number

int[]binaryNum = new int[1000];

//counter to binary array

int i = 0;

while (n>0)

{

//storing remainder in binary array

binaryNum[i] = n % 2;

n = n / 2;

i++;

}

//printing binary array in reverse order

for (int j = i-1; j>=0; j--)

System.out.print(binaryNum[j]);

}

public static void main(String[]args)

{

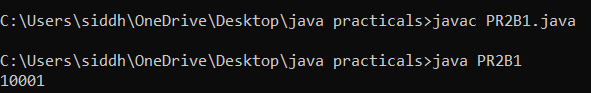
int n = 17;

decToBinary(n);

}

}

**OUTPUT-**



**PRACTICAL NO 2(B)(2)**

**CODE-**

import java.io.\*;

class PR2B2{

static int binaryToDecimal(int n)

{

int num = n;

int dec\_value = 0;

//initializing base

//value to 1, i.e 2^0

int base = 1;

int temp = num;

while(temp > 0){

int last\_digit = temp % 10;

temp = temp / 10;

dec\_value += last\_digit \* base ;

base = base \* 2;

}

return dec\_value;

}

public static void main(String[]args)

{

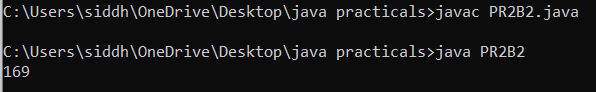
int num = 10101001;

System.out.println(binaryToDecimal(num));

}

}

**OUTPUT-**



**PRACTICAL NO 2 (C)**

**CODE-**

import java.util.Scanner;

public class PR2C{

public static void main (String[] args)

{

//two variables to hold two input binary numbers

long b1, b2;

int i = 0, carry = 0;

//this is to hold the output binary number

int[] sum = new int[10];

//to read the input binary numbers entered by user

Scanner scanner = new Scanner (System.in);

//getting first binary number from user

System.out.print ("Enter first binary number:");

b1 = scanner.nextLong();

//getiinf second binary number from user

System.out.print("Enter second binary number:");

b2 = scanner.nextLong();

//closing scanner after use to avoid memory leak

scanner.close();

while (b1 != 0 || b2!= 0)

{

sum[i++] = (int) ((b1 % 10 + b2 % 10 + carry) % 2);

carry = (int) ((b1 % 10 + b2 % 10 + carry) / 2);

b1 = b1 / 10;

b2 = b2 / 10;

}

if (carry != 0){

sum[i++] = carry;

}

--i;

System.out.print("output:");

while (i >= 0){

System.out.print(sum[i--]);

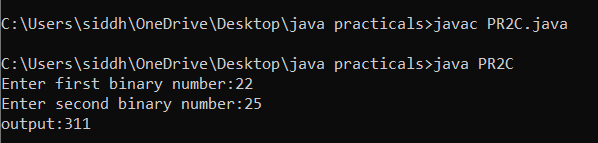
}

System.out.print("\n");

}

}

**OUTPUT-**



**PRACTICAL NO 3**

**PRACTICAL NO 3(A)**

**CODE-**

import java.util.Scanner;

public class PR3A

{

public static void main(String[] args){

Scanner sc= new Scanner (System.in);

System.out.println("Enter a string:");

String str= sc.nextLine();

count(str);

}

public static void count(String x)

{

char[] ch = x.toCharArray();

int letter = 0;

int space = 0;

int num = 0;

int other = 0;

for(int i = 0;i<x.length();i++)

{

if(Character.isLetter(ch[i]))

{

letter ++;

}

else if (Character.isDigit(ch[i]))

{

num ++;

}

else if(Character.isSpaceChar(ch[i]))

{

space ++;

}

else

{

other ++;

}

}

System.out.println("the no of letters,spaces, digits in string is:");

System.out.println("letter:"+letter);

System.out.println("space:"+space);

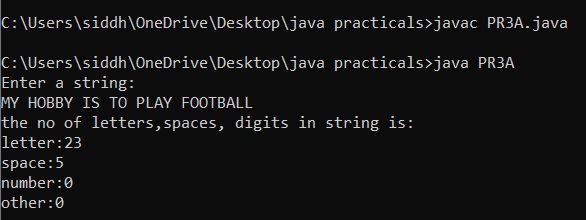
System.out.println("number:"+num);

System.out.println("other:"+other);

}

}

**OUTPUT-**



**PRACTICAL NO 3 (B)**

**CODE-**

import java.util.\*;

class PR3B

{

public static void main (String args[])

{

Scanner ob=new Scanner(System.in);

System.out.print("Enter the any string:");

String s=ob.nextLine ();

count(s);

}

public static void count(String str)

{

int sum=0;

int d=0;

char ch[]=str.toCharArray();

for(int i=0;i<str.length();i++)

{

if(Character.isDigit(ch[i]))

{

sum+=Character.getNumericValue(ch[i]);

}

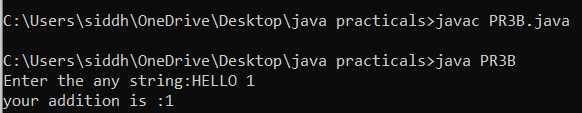
}

System.out.println("your addition is :"+sum);

}

}

**OUTPUT-**



**PRACTICAL NO 3 (C)**

**CODE-**

public class PR3C{

public static void main(String[]args){

//number array

int numbers [] = new int []

{55,32,45,98,82,11,9,39,50};

//assign first element of an array to largest and smallest

int smallest = numbers [0];

int largest = numbers [0];

for (int i = 1;i < numbers.length; i++){

if(numbers[i] > largest)

largest = numbers[i];

else if (numbers[i] < smallest)

smallest = numbers[i];

}

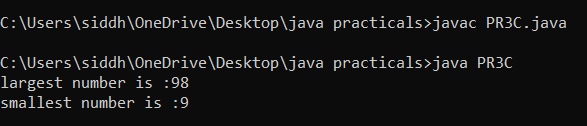
System.out.println("largest number is :" + largest);

System.out.println("smallest number is :" + smallest);

}

}

**OUTPUT-**



**PRACTICAL NO 4**

**PRACTICAL NO 4 (A)**

**CODE-**

import java.util.\*;

class PR4A

{

Scanner input=new Scanner(System.in);

int num,i;

int arr[];

int temp=0;

public void getdata()

{

System.out.print("enter the size of array:");

num=input.nextInt();

arr=new int[num];

System.out.print("enter the number:");

for(i=0;i<num;i++)

{

arr[i]=input.nextInt();

}

}

void putdata()

{

System.out.print("given numbers are:");

for(i=0;i<num;i++)

{

System.out.println(arr[i]);

}

}

void asce()

{

for(i=0;i<num;i++)

{

for(int j=i+1;j<num;j++)

{

if(arr[i]>arr[j])

{

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

System.out.print("ascending order of number are:");

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

{

System.out.println(arr[i]);

}

}

void desc()

{

for(i=0;i<num;i++)

{

for(int j=i+1;j<num;j++)

{

if(arr[i]<arr[j])

{

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

System.out.print("descending order of number are:");

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

{

System.out.println(arr[i]);

}

}

public static void main(String args[])

{

PR4A ob=new PR4A();

ob.getdata();

ob.putdata();

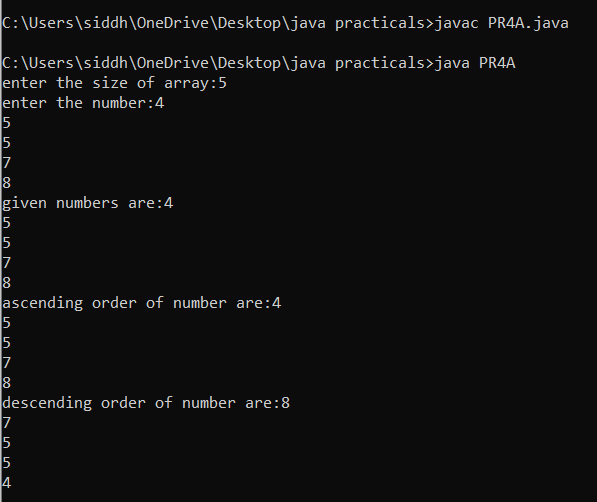
ob.asce();

ob.desc();

}

}

**OUTPUT-**



**PRACTICAL NO 4 (B)**

**CODE-**

class xyz

{

xyz()

{

System.out.println("constrictor method.........");

}

protected void finalize()

{

System.out.print("Garbage collected.......");

}

}

class PR4B

{

public static void main(String args[])

{

xyz ob=new xyz();

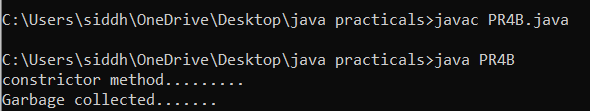
ob=null;

System.gc();

}

}

**OUTPUT-**



**PRACTICAL NO 4 (C)**

**CODE-**

import java.util.Scanner;

abstract class test

{

abstract void get();

}

class test1 extends test

{

void get()

{

int a,b;

Scanner ob=new Scanner(System.in);

System.out.print("enter 1st number:");

a=ob.nextInt();

System.out.println("enter 2nd number:");

b=ob.nextInt();

System.out.println("Addition is:"+(a+b));

}

}

class PR4C

{

public static void main(String args[])

{

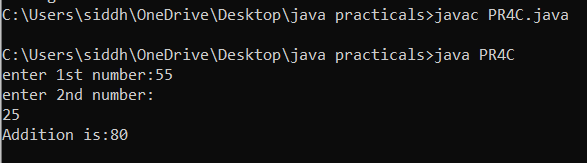
test1 obj=new test1();

obj.get();

}

}

**OUTPUT-**



**PRACTICAL NO 5**

**PRACTICAL NO 5 (A)**

**CODE-**

class shape

{

int length;

int breadth;

}

class Rectangle extends shape

{

int area;

public void calculateArea()

{

area=length\*breadth;

}

}

class PR5A{

public static void main(String args[])

{

Rectangle r = new Rectangle();

//assigning values to shape class attributes

r.length = 10;

r.breadth = 20;

//calculate the area

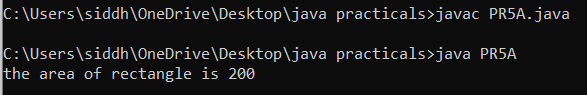
r.calculateArea();

System.out.println("the area of rectangle is "+ r.area);

}

}

**OUTPUT-**



**PRACTICAL NO 5 (B)**

**CODE-**

class shape

{

int r,b;

double area=0.0;

public void print()

{

System.out.println("area="+area);

}

}

class circle extends shape

{

//to initialize value of radius

circle(int x)

{

r=x;

}

//method overriding

public void print()

{

area=Math.PI\*r\*r; //3.14\*2\*2

//super is used to call method of base class shape

super.print();

}

}

class rect extends shape

{

rect(int x,int y)

{

r=x;b=y;

}

//method overriding

public void print()

{

area=r\*b; //area=2\*3

super.print();

}

}

class PR5B

{

public static void main(String args[])

{

shape s1=new shape();

circle c1=new circle(2);

rect r1=new rect(2,3);

s1.print();

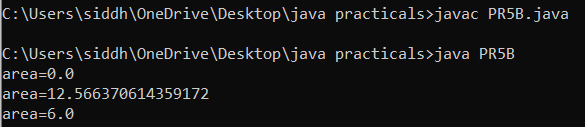
c1.print();

r1.print();

}

}

**OUTPUT-**



**PRACTICAL NO 5 (C)**

**CODE-**

interface MotorBike

{

int speed=50;

public void totalDistance(); //method declaration

}

interface Cycle

{

int Distance=150;

public void speed();

}

public class PR5C implements MotorBike,Cycle

{

int totalDistance;

int avgSpeed;

public void totalDistance()

{

totalDistance=speed\*Distance;

System.out.println("Total distance travelled:"+totalDistance);

}

public void speed()

{

int avgSpeed=totalDistance/speed;

System.out.println("average speed maintained: "+avgSpeed);

}

public static void main(String args[])

{

PR5C t1=new PR5C();

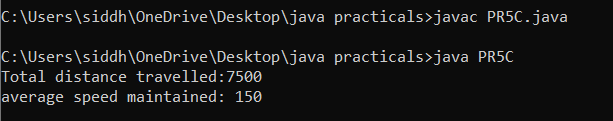
t1.totalDistance();

t1.speed();

}

}

**OUTPUT-**



**PRACTICAL NO 6**

**PRACTICAL NO 6 (A)**

**CODE-**

package mypack;

public class PR6A

{

public static void main(String args[])

{

System.out.println("welcome to package");

}

}

**OUTPUT-**



**PRACTICAL NO 6 (B)**

**CODE-**

public class PR6B

{

public static void main(String args[])

{

//creating two matrices

int a[][]={{1,3,4},{2,4,3},{3,4,5}};

int b[][]={{1,3,4},{2,4,3},{1,2,4}};

//creating another matrix to store the sum of two matrices

int c[][]=new int[3][3]; //3 rows and 3 columns

//adding and printing addition of 2 matrices

for (int i=0;i<3;i++)

{

for (int j=0;j<3;j++)

{

c[i][j]=a[i][j]+b[i][j];

System.out.print(c[i][j]+"");

}

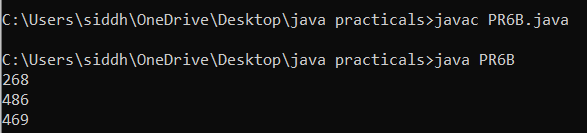
System.out.println(); //new row//line

}

}

}

**OUTPUT-**



**PRACTICAL NO 6 (C)**

**CODE-**

public class PR6C

{

public static void main(String args[])

{

//creating two matrices

int a[][]={{1,1,1},{2,2,2},{3,3,3}};

int b[][]={{1,1,1},{2,2,2},{3,3,3}};

//creating another matrix to store the ,ultiplication of two matrices

int c[][]=new int[3][3];//3 rows and 3 columns

//multiplying and printing multiplication of 2 matrices

for(int i=0;i<3;i++)

{

for(int j=0;j<3;j++)

{

c[i][j]=0;

for(int k=0;k<3;k++)

{

c[i][j]+=a[i][j]\*b[k][j];

}//end of k loop

System.out.print(c[i][j]+" ");//printing matrix element

}//end of j loop

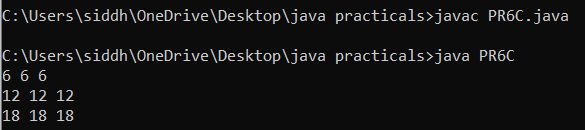
System.out.println();//new line

}

}

}

**OUTPUT-**



**PRACTICAL NO 7**

**PRACTICAL NO 7 (A)**

**CODE-**

import java.util.Vector;

class PR7A

{

public static void main(String args[])

{

Vector<String>v=new

Vector<String>();

v.add("Red");

v.add("Green");

v.add("Blue");

System.out.println("Vector Elements are:-"+v);

v.add(2,"Yellow");

System.out.println("After Adding Elements at second position:-"+v);

System.out.println("Elements at third position:-"+v.get(3));

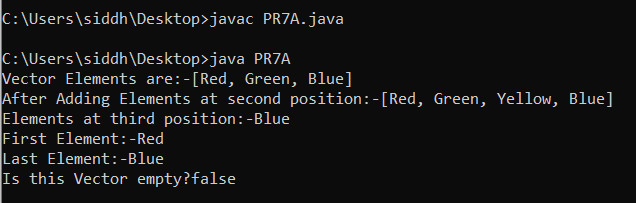
System.out.println("First Element:-"+v.firstElement());

System.out.println("Last Element:-"+v.lastElement());

System.out.println("Is this Vector empty?"+v.isEmpty());

}

}

**OUTPUT-**

**PRACTICAL NO 7 (B)**

**CODE-**

import java.util.Vector;

class PR7B

{

public static void main(String args[])

{

System.out.println(Thread.currentThread().getName());

for(int i=0;i<10;i++)

{

new Thread(""+i)

{

public void run()

{

System.out.println("Thread:"+getName()+"running");

}

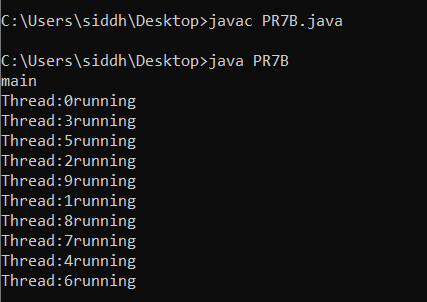
}.start();

}

}

}

**OUTPUT-**



**PRACTICAL NO 7 (C)**

**CODE-**

import java.io.\*;

class A extends Thread

{

public void run()

{

System.out.println("Start A");

for(int i=1;i<=5;i++)

{

System.out.println("Thread A i:"+i);

}

System.out.println("Exit A");

}

}

class B extends Thread

{

public void run()

{

System.out.println("Start B");

for(int j=1;j<=5;j++)

{

System.out.println("Thread B j:"+j);

}

System.out.println("Exit B");

}

}

class C extends Thread

{

public void run()

{

System.out.println("Start C");

for(int k=1;k<=5;k++)

{

System.out.println("Thread C k:"+k);

}

System.out.println("Exit C");

}

}

class D extends Thread

{

public void run()

{

System.out.println("Start D");

for(int m=1;m<=5;m++)

{

System.out.println("Thread D m:"+m);

}

System.out.println("Exit D");

}

}

class PR7C

{

public static void main(String args[])

{

new A().start();

new B().start();

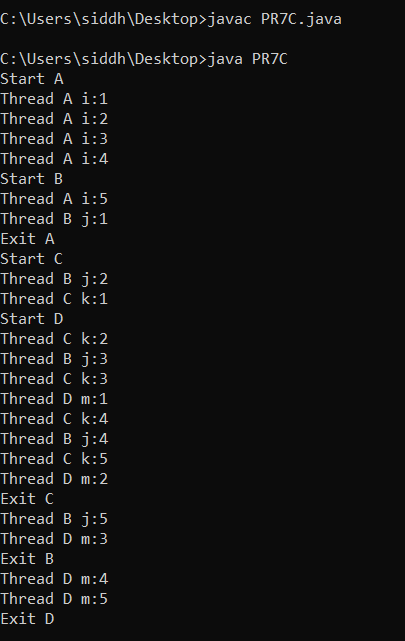
new C().start();

new D().start();

}

}

OUTPUT-



**PRACTICAL NO 8**

**PRACTICAL NO 8 (A)**

**CODE-**

import java.util.Scanner;

import java.io.\*;

public class PR8A

{

public static void main(String[] args)

{

String fname;

Scanner scan = new

Scanner(System.in);

//enter filename with extension to open and read its content

System.out.print("Enter file name to open(with extensions like file.txt):");

fname=scan.nextLine();

//this will reference only one line at a time

String line = null;

try

{

//filereader reads text files in the default encoding

FileReader fileReader = new

FileReader(fname);

//always wrap the filereader in bufferedreader

BufferedReader bufferedReader = new

BufferedReader(fileReader);

while((line = bufferedReader.readLine()) != null)

{

System.out.println(line);

}

//always close the file after use

bufferedReader.close();

}

catch(IOException ex)

{

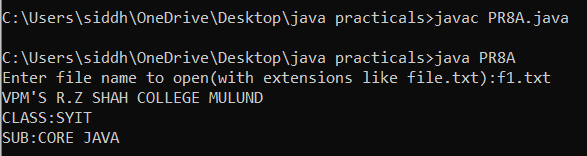
System.out.println("Error reading file named '" + fname + "'");

}

}

}

**OUTPUT-**



**PRACTICAL NO 8 (B)**

**CODE-**

import java.io.\*;

class PR8B

{

public static void main(String args[])

throws IOException

{

FileInputStream Fread = new

FileInputStream("Hello.txt");

FileOutputStream Fwrite=new

FileOutputStream("Hello1.txt");

int c;

while((c=Fread.read())!=-1)

Fwrite.write((char)c);

System.out.println("File is copied");

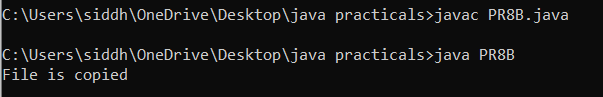
Fread.close();

Fwrite.close();

}

}

**OUTPUT-**



**PRACTICAL NO 8(C)**

**CODE-**

import java.io.\*;

public class PR8C

{

public static void main(String args[])

throws IOException

{

int roll;

String name,city;

DataInputStream dis=new

DataInputStream(System.in);

File file=new File("Student.txt");

file.createNewFile();

FileWriter writer = new FileWriter(file);

System.out.println("Enter students roll no:");

roll=Integer.parseInt(dis.readLine());

System.out.println("Enter students name:");

name=dis.readLine();

System.out.println("Enter students city:");

city=dis.readLine();

//writing data

writer.write("Roll no is:"+roll);

writer.write("Name is:"+name);

writer.write("City is:"+city);

writer.flush();

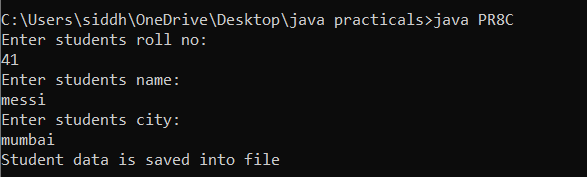
writer.close();

System.out.println("Student data is saved into file");

}

}

**OUTPUT-**



**PRACTICAL NO 9**

**PRATICAL NO 9 (A)**

**CODE-**

import java.awt.\*;

import java.awt.event.\*;

class PR9A extends Frame implements ActionListener

{

Label l1,l2;

TextField t1,t2;

Button b1;

public PR9A()

{

setLayout(new FlowLayout());

b1=new Button("Find");

l1=new Label("Enter the number:");

l2=new Label("The factorial is:");

t1=new TextField();

t2=new TextField();

b1=new Button("Calculate");

add(l1);

add(t1);

add(l2);

add(t2);

add(b1);

b1.addActionListener(this);

}

public void actionPerformed(ActionEvent ae)

{

int n,i,f=1;

n=Integer.parseInt(t1.getText());

for(i=1;i<=n;i++)

{

f=f\*i;

}

t2.setText(""+f);

}

public static void main(String arg[])

{

PR9A ob=new PR9A();

ob.setSize(300,30);

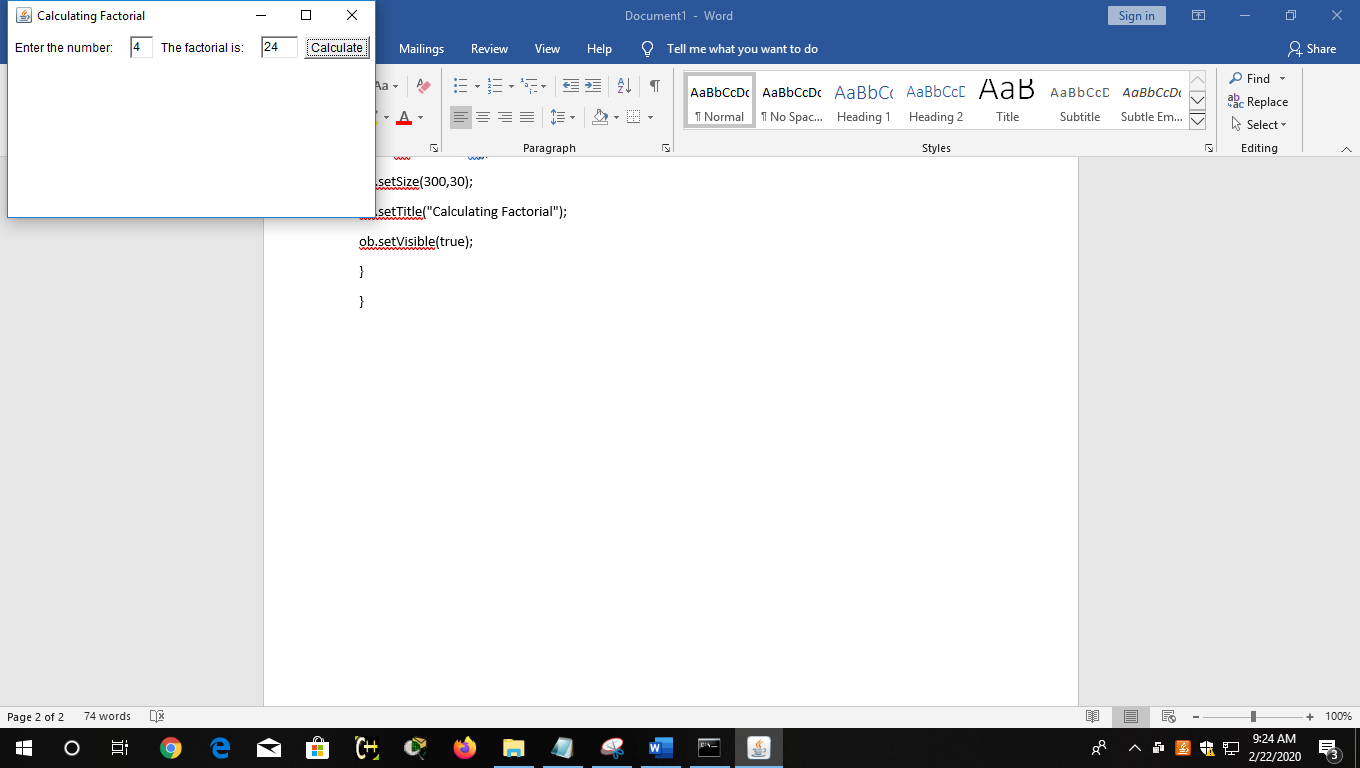
ob.setTitle("Calculating Factorial");

ob.setVisible(true);

}

}

**OUTPUT:**



**PRACTICAL NO 9 (B)**

**CODE-**

import java.awt.\*;

import java.awt.event.\*;

class PR9B extends Frame implements ActionListener

{

Label lstring1, lstring2;

TextField tfstring1, tfstring2;

Button submit;

TextArea display;

PR9B()

{

lstring1 = new Label("String 1 (str1)");

lstring2 = new Label("String 2 (str2)");

tfstring1 = new TextField();

tfstring2 = new TextField();

submit = new Button("Perform Operations");

display = new TextArea("", 2 , 100 , TextArea.SCROLLBARS\_NONE);

lstring1.setBounds(10, 40, 100, 20);

tfstring1.setBounds(10, 65, 100, 20);

lstring2.setBounds(120, 40, 100, 20);

tfstring2.setBounds(120, 65, 100, 20);

submit.setBounds(10, 90, 210, 30);

display.setBounds(10, 130, 210, 100);

display.setEditable(false);

add(lstring1);

add(lstring2);

add(tfstring1);

add(tfstring2);

add(submit);

add(display);

submit.addActionListener(this);

setTitle("String Operations");

setSize(230,240);

setLayout(null);

setVisible(true);

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

dispose();

}

});

}

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==submit)

{

String str1 = tfstring1.getText();

String str2 = tfstring2.getText();

StringBuilder sb = new StringBuilder(str2);

sb.reverse();

String oper = " str1.concat(str2) = " + str1.concat(str2) + "n str1.toLowerCase() = " + str1.toLowerCase() + "n str2.toUpperCase() = " + str2.toUpperCase() + "n str2.trim() = " + str2.trim() + "n str2.replace(' ', '#') = " + str2.replace(' ', '#') + "n str2 reverse : " + sb;

display.setText(oper);

}

}

public static void main(String[] args)

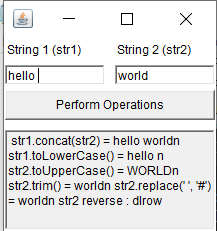
{

new PR9B();

}

}

**OUTPUT-**



**PRACTICAL NO 9 (C)**

**CODE-**

class PR9C

{

public static void main(String args[])

{

int d=0;

int n=20;

try

{

int fraction=n/d;

System.out.println("This line will not be Executed");

}

catch(ArithmeticException e)

{

System.out.println("In the catch Block due to Exception="+e);

}

System.out.println("End of Main");

}

}

**OUTPUT-**



**PRACTICAL NO 10**

**PRACTICAL NO 10 (A)**

**CODE-**

import java.io.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.applet.\*;

class Frame1 extends Frame implements ActionListener

{

String msg="";

Button btnNew,btnSubmit,btnView;

Label lblName,lblAge,lblAddr,lblGender,lblQua;

TextField txtName,txtAge;

TextArea txtAddr,txtAns;

CheckboxGroup ChkGrp;

Checkbox chkMale,chkFemale;

Checkbox chkMca,chkBca,chkBba,chkMba;

Frame1(String name)

{

super(name);

setLayout(new GridLayout(3,2));

lblName = new Label("Name: ");

lblAge = new Label("Age: ");

lblAddr = new Label("Address : ");

lblGender = new Label("Gender: ");

lblQua = new Label("Qualification: ");

txtName = new TextField(20);

txtAge = new TextField(20);

txtAddr = new TextArea();

ChkGrp = new CheckboxGroup();

chkMale = new Checkbox("Male",ChkGrp,false);

chkFemale = new Checkbox("Female",ChkGrp,false);

chkMca = new Checkbox("MCA");

chkBca = new Checkbox("BCA");

chkMba = new Checkbox("MBA");

chkBba = new Checkbox("BBA");

btnNew = new Button("NEW");

btnSubmit = new Button("SUBMIT");

btnView = new Button("VIEW");

btnNew.addActionListener(this);

btnSubmit.addActionListener(this);

btnView.addActionListener(this);

add(lblName);

add(txtName);

add(lblAge);

add(txtAge);

add(lblAddr);

add(txtAddr);

add(lblGender);

add(chkMale);

add(chkFemale);

add(lblQua);

add(chkBca);

add(chkBba);

add(chkMca);

add(chkMba);

add(btnNew);

add(btnSubmit);

add(btnView);

txtAns = new TextArea();

add(txtAns);

}

public void actionPerformed(ActionEvent ae)

{

String s="";

boolean b;

FileInputStream Fin;

DataInputStream dis;

FileOutputStream Fout;

DataOutputStream dos;

try

{

Fout = new FileOutputStream("Biodata.txt",true);

dos = new DataOutputStream(Fout);

String str = ae.getActionCommand();

if(str.equals("SUBMIT"))

{

s=txtName.getText().trim();

dos.writeUTF(s);

dos.writeInt(Integer.parseInt(txtAge.getText()));

s=txtAddr.getText();

dos.writeUTF(s);

if(chkMale.getState())

dos.writeUTF("Male ");

if(chkFemale.getState())

dos.writeUTF("Female ");

s="";

if(chkMca.getState())

s="MCA ";

if(chkBca.getState())

s+="BCA ";

if(chkBba.getState())

s+="BBA ";

if(chkMba.getState())

s+="MBA ";

s+="!";

dos.writeUTF(s);

Fout.close();

}

if(str.equals("VIEW"))

{

String tmp,name,addr,gender,qual;

int age;

Fin = new FileInputStream("Biodata.txt");

dis = new DataInputStream(Fin);

int i=0,j;

while(Fin.available()>0)

{

name = dis.readUTF();

age = dis.readInt();

addr = dis.readUTF();

gender = dis.readUTF();

qual = dis.readUTF();

if(name.equals(txtName.getText().trim()))

{

txtAge.setText(age+"");

txtAddr.setText(addr);

if(gender.equals("Male "))

chkMale.setState(true);

else

chkFemale.setState(true);

while(qual.charAt(i)!='!')

{

j=qual.indexOf(' ');

tmp = qual.substring(i,j);

if(tmp.equals("MCA"))

chkMca.setState(true);

if(tmp.equals("BCA"))

chkBca.setState(true);

if(tmp.equals("BBA"))

chkBba.setState(true);

if(tmp.equals("MBA"))

chkMba.setState(true);

i=j+1;

}

break;

}

}

Fin.close();

}

if(str.equals("NEW"))

{

txtName.setText("");

txtAge.setText("");

txtAddr.setText("");

chkMale.setState(false);

chkFemale.setState(false);

chkMca.setState(false);

chkBca.setState(false);

chkBba.setState(false);

chkMba.setState(false);

}

}

catch(Exception e)

{

System.out.println("The Exception Is : " +e);

}

}

}

class Bio2

{

public static void main(String args[])

{

try{

Frame1 F = new Frame1("Biodata");

F.setSize(400,400);

F.show();

}catch(Exception e)

{

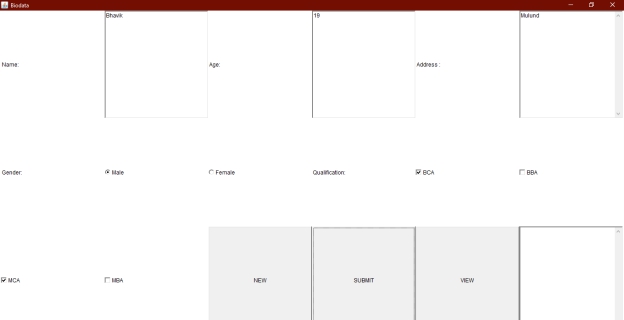
System.out.println(e);

}

}

}

**OUTPUT:**





**PRACTICAL NO 10 (B)**

**CODE-**

import java.awt.\*;

import java.awt.event.\*;

class Calculator implements ActionListener

{

Frame f=new Frame();

Label l1=new Label("First number");

Label l2=new Label("Second number");

Label l3=new Label("Third number");

TextField t1=new TextField();

TextField t2=new TextField();

TextField t3=new TextField();

Button b1=new Button("Add");

Button b2=new Button("Sub");

Button b3=new Button("Mul");

Button b4=new Button("Div");

Button b5=new Button("Cancel");

Calculator()

{

//Giving coordinates

t1.setBounds(50,100,100,20);

t2.setBounds(50,140,100,20);

t3.setBounds(50,180,100,20);

t1.setBounds(200,100,100,20);

t2.setBounds(200,140,100,20);

t3.setBounds(200,180,100,20);

b1.setBounds(50,250,50,20);

b2.setBounds(110,250,50,20);

b3.setBounds(170,250,50,20);

b4.setBounds(50,250,50,20);

b5.setBounds(290,250,50,20);

//Adding components to the frame

f.add(l1);

f.add(l2);

f.add(l3);

f.add(t1);

f.add(t2);

f.add(t3);

f.add(b1);

f.add(b2);

f.add(b3);

f.add(b4);

f.add(b5);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

b5.addActionListener(this);

f.setLayout(null);

f.setVisible(true);

f.setSize(400,350);

}

public void actionPerformed(ActionEvent e)

{

int n1=Integer.parseInt(t1.getText());

int n2=Integer.parseInt(t2.getText());

if(e.getSource()==b1)

{

t3.setText(String.valueOf(n1+n2));

}

if(e.getSource()==b2)

{

t3.setText(String.valueOf(n1-n2));

}

if(e.getSource()==b3)

{

t3.setText(String.valueOf(n1\*n2));

}

if(e.getSource()==b4)

{

t3.setText(String.valueOf(n1/n2));

}

if(e.getSource()==b5)

{

System.exit(0);

}

}

public static void main(String args[])

{

new Calculator();

}

}

**OUTPUT:**

