Program:

import java.util.Scanner;

class EB{

public static void main(String args[]){

double amt;

Scanner sc=new Scanner(System.in);

System.out.println("Enter your card number:");

int cno=sc.nextInt();

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

String cname=sc.next();

System.out.println("Enter your previous reading:");

int preading=sc.nextInt();

System.out.println("Enter your current reading:");

int creading=sc.nextInt();

System.out.println("Enter your connection type 1.Domestic,2.commercial");

int type=sc.nextInt();

double units=creading-preading;

double dunits;

dunits=units-100;

switch(type)

{

case 1:

if(units<=100)

amt=units\*1;

else if(dunits <=200 && dunits>=101)

amt=100+dunits\*2.50;

else if(dunits <=500 && dunits>=201)

amt=100+dunits\*4;

else

amt=100+dunits\*6;

System.out.println("customer no:"+cno+ "\ncustomer name:" +cname+ "\nunits" +units+ "\n Bill Amt:" +amt);

break;

case 2:

if(units<=100)

amt=units\*2;

else if(dunits <=200 && dunits>=101)

amt=100+dunits\*4.50;

else if(dunits <=500 && dunits>=201)

amt=100+dunits\*6;

else

amt=100+dunits\*7;

System.out.println("customer no:"+cno+"\ncustomer name:"+cname+"\nunits"+units+"\n Bill Amt:"+amt);

default:

}

sc.close();

}

}

Output:

Enter your card number:

10247

Enter your name:

Nandhu

Enter your previous reading:

700

Enter your current reading:

1000

Enter your connection type 1.Domestic,2.commercial

1

customer no:10247

customer name:Nandhu

units300.0

Bill Amt:600.0

Program:

Currencyconversion:

package currencyconversion;

import java.util.\*;

public class currency

{

double inr,usd;

double euro,yen;

Scanner in=new Scanner(System.in);

public void dollartorupee()

{

System.out.println("Enter dollars to convert into Rupees:");

usd=in.nextInt();

inr=usd\*67;

System.out.println("Dollar ="+usd+"equal to INR="+inr);

}

public void rupeetodollar()

{

System.out.println("Enter Rupee to convert into Dollars:");

inr=in.nextInt();

usd=inr/67;

System.out.println("Rupee ="+inr+"equal to Dollars="+usd);

}

public void eurotorupee()

{

System.out.println("Enter euro to convert into Rupees:");

euro=in.nextInt();

inr=euro\*79.50;

System.out.println("Euro ="+euro +"equal to INR="+inr);

}

public void rupeetoeuro()

{

System.out.println("Enter Rupees to convert into Euro:");

inr=in.nextInt();

euro=(inr/79.50);

System.out.println("Rupee ="+inr +"equal to Euro="+euro);

}

public void yentorupee()

{

System.out.println("Enter yen to convert into Rupees:");

yen=in.nextInt();

inr=yen\*0.61;

System.out.println("YEN="+yen +"equal to INR="+inr);

}

public void rupeetoyen()

{

System.out.println("Enter Rupees to convert into Yen:");

inr=in.nextInt();

yen=(inr/0.61);

System.out.println("INR="+inr +"equal to YEN"+yen);

}

}

Distanceconversion:

package distanceconversion;

import java.util.\*;

public class distance

{

double km,m,miles;

Scanner sc = new Scanner(System.in);

public void kmtom()

{

System.out.print("Enter in km ");

km=sc.nextDouble();

m=(km\*1000);

System.out.println(km+"km" +"equal to"+m+"metres");

}

public void mtokm()

{

System.out.print("Enter in meter ");

m=sc.nextDouble();

km=(m/1000);

System.out.println(m+"m" +"equal to"+km+"kilometres");

}

public void milestokm()

{

System.out.print("Enter in miles");

miles=sc.nextDouble();

km=(miles\*1.60934);

System.out.println(miles+"miles" +"equal to"+km+"kilometres");

}

public void kmtomiles()

{

System.out.print("Enter in km");

km=sc.nextDouble();

miles=(km\*0.621371);

System.out.println(km+"km" +"equal to"+miles+"miles");

}

}

Timeconversion:

package timeconversion;

import java.util.\*;

public class timer

{

int hours,seconds,minutes;

int input;

Scanner sc = new Scanner(System.in);

public void secondstohours()

{

System.out.print("Enter the number of seconds: ");

input = sc.nextInt();

hours = input / 3600;

minutes = (input % 3600) / 60;

seconds = (input % 3600) % 60;

System.out.println("Hours: " + hours);

System.out.println("Minutes: " + minutes);

System.out.println("Seconds: " + seconds);

}

public void minutestohours()

{

System.out.print("Enter the number of minutes: ");

minutes=sc.nextInt();

hours=minutes/60;

minutes=minutes%60;

System.out.println("Hours: " + hours);

System.out.println("Minutes: " + minutes);

}

public void hourstominutes()

{

System.out.println("enter the no of hours");

hours=sc.nextInt();

minutes=(hours\*60);

System.out.println("Minutes: " + minutes);

}

public void hourstoseconds()

{

System.out.println("enter the no of hours");

hours=sc.nextInt();

seconds=(hours\*3600);

System.out.println("Minutes: " + seconds);

}

}

Converter:

import java.util.\*;

import java.io.\*;

import currencyconversion.\*;

import distanceconversion.\*;

import timeconversion.\*;

class converter

{

public static void main(String args[])

{

Scanner s=new Scanner(System.in);

int choice,ch;

currency c=new currency();

distance d=new distance();

timer t=new timer();

do

{

System.out.println("1.dollar to rupee ");

System.out.println("2.rupee to dollar ");

System.out.println("3.Euro to rupee ");

System.out.println("4..rupee to Euro ");

System.out.println("5.Yen to rupee ");

System.out.println("6.Rupee to Yen ");

System.out.println("7.Meter to kilometer ");

System.out.println("8.kilometer to meter ");

System.out.println("9.Miles to kilometer ");

System.out.println("10.kilometer to miles");

System.out.println("11.Hours to Minutes");

System.out.println("12.Hours to Seconds");

System.out.println("13.Seconds to Hours");

System.out.println("14.Minutes to Hours");

System.out.println("Enter ur choice");

choice=s.nextInt();

switch(choice)

{

case 1:

{

c.dollartorupee();

break;

}

case 2:

{

c.rupeetodollar();

break;

}

case 3:

{

c.eurotorupee();

break;

}

case 4:

{

c.rupeetoeuro();

break;

}

case 5:

{c.yentorupee();

break;}

case 6 :

{

c.rupeetoyen();

break;

}

case 7 :

{

d.mtokm();

break;

}

case 8 :

{

d.kmtom();

break;

}

case 9 :

{

d.milestokm();

break;

}

case 10 :

{

d.kmtomiles();

break;

}

case 11 :

{

t.hourstominutes();

break;

}

case 12 :

{

t.hourstoseconds();

break;

}

case 13 :

{

t.secondstohours();

break;

}

case 14 :

{

t.minutestohours();

break;

}}

System.out.println("Enter 0 to quit and 1 to continue ");

ch=s.nextInt();

}while(ch==1);

}

}

Output:

1.dollar to rupee

2.rupee to dollar

3.Euro to rupee

4..rupee to Euro

5.Yen to rupee

6.Rupee to Yen

7.Meter to kilometer

8.kilometer to meter

9.Miles to kilometer

10.kilometer to miles

11.Hours to Minutes

12.Hours to Seconds

13.Seconds to Hours

14.Minutes to Hours

Enter ur choice

7

Enter in meter 500

500.0mequal to0.5kilometres

Enter 0 to quit and 1 to continue

1

1.dollar to rupee

2.rupee to dollar

3.Euro to rupee

4..rupee to Euro

5.Yen to rupee

6.Rupee to Yen

7.Meter to kilometer

8.kilometer to meter

9.Miles to kilometer

10.kilometer to miles

11.Hours to Minutes

12.Hours to Seconds

13.Seconds to Hours

14.Minutes to Hours

Enter ur choice

1

Enter dollars to convert into Rupees:

5

Dollar =5.0equal to INR=335.0

Enter 0 to quit and 1 to continue

0

Program:

import java.util.Scanner;

public class QuadraticEquation {

public static void main(String[] Strings) {

Scanner input = new Scanner(System.in);

System.out.print("Enter the value of a: ");

double a = input.nextDouble();

System.out.print("Enter the value of b: ");

double b = input.nextDouble();

System.out.print("Enter the value of c: ");

double c = input.nextDouble();

double d= b \* b - 4.0 \* a \* c;

System.out.print(d);

if (d> 0.0) {

double r1 = (-b + Math.pow(d, 0.5)) / (2.0 \* a);

double r2 = (-b - Math.pow(d, 0.5)) / (2.0 \* a);

System.out.println("The roots are " + r1 + " and " + r2);

}

else if (d == 0.0) {

double r1 = -b / (2.0 \* a);

System.out.println("The root is " + r1);

}

else {

System.out.println("Roots are not real.");

}

} }

Output:

Enter the value of a: 1

Enter the value of b: 6

Enter the value of c: 8

4.0The roots are -2.0 and -4.0

Program:

import java.util.Scanner ;

public class FibonacciSeries

{

public static void main(String [] array)

{

Scanner s = new Scanner(System.in) ;

System.out.print("Fibonacci Series\nEnter no.of items to be printed :\t") ;

int n = s.nextInt() ;

int a = 0, b = 1, c = 0 ;

System.out.print("\n0\t1\t") ;

for(int i = 0 ; i <= n-2 ; i++)

{

c = a + b ;

a = b ;

b= c ;

System.out.print(c+"\t") ;

}

}

}

Output:

Fibonacci Series

Enter no.of items to be printed : 6

0 1 1 2 3 5 8

Program:

import java.util.Scanner;

public class StringSorting

{

public static void main(String[] args)

{

int count;

String temp;

Scanner scan = new Scanner(System.in);

System.out.print("Enter number of strings you would like to enter:");

count = scan.nextInt();

String str[] = new String[count];

Scanner scan2 = new Scanner(System.in);

System.out.println("Enter the Strings one by one:");

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

{

str[i] = scan2.nextLine();

}

scan.close();

scan2.close();

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

{

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

if (str[i].compareTo(str[j])>0)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

}

}

}

System.out.print("Strings in Sorted Order:");

for (int i = 0; i <= count - 1; i++)

{

System.out.print(str[i] + ", ");

}

}

}

Output:

Enter number of strings you would like to enter:4

Enter the Strings one by one:

icecream

xerox

ball

high

Strings in Sorted Order:ball, high, icecream, xerox

Program:

import java.io.\*;

import java.util.\*;

public class Primenumbers

{

public static void main(String[] args){

{

System.out.println("Enter number:");

}

int num , count;

Scanner s=new Scanner(System.in);

num=s.nextInt();

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

{

count = 0;

for (int j = 2; j <= i / 2; j++)

{

if (i % j == 0)

{

count++;

break;

}

}

if (count == 0) {

System.out.println(i);

}

}

}

}

Output:

Enter number:

30

1

2

3

5

7

11

13

17

19

23

29

Program:

import java.io.\*;

import java.util.Scanner;

public class mulmatrix{

public static void main(String args[]){

int r1,r2,c1,c2,i,j,k,sum;

Scanner s=new Scanner(System.in);

System.out.println("Enter the number of rows of matrix1: ");

r1=s.nextInt();

System.out.println("Enter the number of columns of matrix1 :");

c1=s.nextInt();

System.out.println("Enter the number of rows of matrix2: ");

r2=s.nextInt();

System.out.println("Enter the number of columns of matrix2: ");

c2=s.nextInt();

if(c1!=r2){

System.out.println("matrix multipication is not possible");

System.exit(0);

}

int A[][]=new int[r1][c1];

int B[][]=new int[r2][c2];

int C[][]=new int[c1][r2];

System.out.println("\nEnter First matrix elements:\n");

for(i=0;i<r1;i++){

for(j=0;j<c1;j++){

A[i][j]=s.nextInt();

}

}

System.out.println("\nEnter second matrix elements:\n");

for(i=0;i<r2;i++)

{

for(j=0;j<c2;j++)

{

B[i][j]=s.nextInt();

}

}

for(i=0;i<r1;i++)

{

for(j=0;j<c2;j++)

{

for(k=0;k<1;k++)

{

C[i][j]+=A[i][j]\*B[i][j];

}

}

}

System.out.println("Resultant");

for(int row[]:C){

for(int column:row){

System.out.print(column+"\t");

}

System.out.println();

}

}

}

Output:

Enter the number of rows of matrix1:

2

Enter the number of columns of matrix1:

2

Enter the number of rows of matrix2:

2

Enter the number of columns of matrix2:

2

Enter First matrix elements:

1

2

3

4

Enter second matrix elements:

5

6

7

8

Resultant

5 12

21 32

Program:

import java.util.Scanner;

class palindrome

{

public static void main(String args[])

{

String str,rev="";

Scanner sc=new Scanner(System.in);

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

str=sc.nextLine();

int length=str.length();

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

rev =rev+str.charAt(i);

if(str.equals(rev))

System.out.println(str+"is a palindrome");

else

System.out.println(str+"is not a palindrome");

}

}

Output:

Enter a string:

malayalam

malayalam is a palindrome

Enter a string:

london

london is not a palindrome

Program:

import java.io.\*;

import java.util.\*;

import java.lang.\*;

class employee

{

String name;

int id;

String address;

String mail;

String mobile;

float da,hra,pf,scf,gross,net;

float b;

void getdata()

{

Scanner sc=new Scanner(System.in);

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

name=sc.next();

System.out.println("Enter id:");

id=sc.nextInt();

System.out.println("Enter the address:");

address=sc.next();

System.out.println("Enter mail:");

mail=sc.next();

System.out.println("Enter mobile:");

mobile=sc.next();

}

void calc(float basic)

{

b=basic;

da=(float)(basic\*97/100);

hra=(float)(basic\*10/100);

pf=(float)(basic\*12/100);

scf=(float)(basic\*0.1/100);

gross=basic+da+hra+pf+scf;

net=gross-pf;

}

void display()

{

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Employee details\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Employee Name="+name);

System.out.println("Employee ID="+id);

System.out.println("Employee address="+address);

System.out.println("Employee mobile number="+mobile);

System.out.println("Employee MailID="+mail);

System.out.println("Basic pay="+b);

System.out.println("DA="+da);

System.out.println("HRA="+hra);

System.out.println("PF="+pf);

System.out.println("Staff Club Fund="+scf);

System.out.println("Gross Salary="+gross);

System.out.println("Net Salary="+net);

}

}

class programmer extends employee

{

float bp;

programmer()

{

bp=2500;

}

}

class ap extends employee

{

float bp;

ap()

{

bp=5000;

}

}

class asso extends employee

{

float bp;

asso()

{

bp=7500;

}

}

class prof extends employee

{

float bp;

prof()

{

bp=10000;

}

}

class emp

{

public static void main(String args[])

{

int choice;

Scanner sc=new Scanner(System.in);

System.out.println("1.PROGRAMMER\n2.ASSISTANT PROFESSOR\n3.ASSOCIATE PROFESSOR\n4.PROFESSOR");

System.out.println("Enter the choice:");

choice=sc.nextInt();

switch(choice)

{

case 1:

System.out.println("Enter the programmer details:");

programmer ob1=new programmer();

ob1.getdata();

ob1.calc(ob1.bp);

ob1.display();

break;

case 2:

System.out.println("Enter the Assistant Professor details:");

ap ob2=new ap();

ob2.getdata();

ob2.calc(ob2.bp);

ob2.display();

break;

case 3:

System.out.println("Enter the Associate Professor details:");

asso ob3=new asso();

ob3.getdata();

ob3.calc(ob3.bp);

ob3.display();

break;

case 4:

System.out.println("Enter the Professor deatails:");

prof ob4=new prof();

ob4.getdata();

ob4.calc(ob4.bp);

ob4.display();

break;

}

}

}

Output:

1.PROGRAMMER

2.ASSISTANT PROFESSOR

3.ASSOCIATE PROFESSOR

4.PROFESSOR

Enter the choice:

1

Enter the programmer details:

Enter name:

Nandhu

Enter id:

337

Enter the address:

Chennai

Enter mail:

programmer@gmail.com

Enter mobile:

9273485091

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Employee details\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Employee Name=Nandhu

Employee ID=337

Employee address=Chennai

Employee mobile number=9273485091

Employee MailID=programmer@gmail.com

Basic pay=2500.0

DA=2425.0

HRA=250.0

PF=300.0

Staff Club Fund=2.5

Gross Salary=5477.5

Net Salary=5177.5

Program:

import java.io.\*;

interface Mystack

{

public void pop();

public void push();

public void display();

}

class Stack\_array implements Mystack

{

final static int n=5;

int stack[]=new int[n];

int top=-1;

public void push()

{

try

{

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

if(top==(n-1))

{

System.out.println(" Stack Overflow");

return;

}

else

{

System.out.println("Enter the element");

int ele=Integer.parseInt(br.readLine());

stack[++top]=ele;

}

}

catch(IOException e)

{

System.out.println("e");

}

}

public void pop()

{

if(top<0)

{

System.out.println("Stack underflow");

return;

}

else

{

int popper=stack[top];

top--;

System.out.println("Popped element:" +popper);

}

}

public void display()

{

if(top<0)

{

System. out. println("Stack is empty");

return;

}

else

{

String str="";

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

str=str+""+stack[i]+"<--";

System.out.println("Elements are:"+str);

}

}

}

class StackADT

{

public static void main(String arg[])throws IOException

{

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

System.out.println("Implementation of Stack using Array");

Stack\_array stk=new Stack\_array();

int ch=0;

do

{

System.out.println("1.Push 2.Pop 3.Display 4.Exit ");

System.out.println("Enter your choice:");

ch=Integer.parseInt(br.readLine());

switch(ch)

{

case 1:

stk.push();

break;

case 2:

stk.pop();

break;

case 3:

stk.display();

break;

case 4:

System.exit(0);

}

}

while(ch<5);

}

}

Output:

Implementation of Stack using Array

1.Push 2.Pop 3.Display 4.Exit

Enter your choice:

1

Enter the element

76

1.Push 2.Pop 3.Display 4.Exit

Enter your choice:

1

Enter the element

45

1.Push 2.Pop 3.Display 4.Exit

Enter your choice:

1

Enter the element

87

1.Push 2.Pop 3.Display 4.Exit

Enter your choice:

1

Enter the element

28

1.Push 2.Pop 3.Display 4.Exit

Enter your choice:

3

Elements are:76<--45<--87<--28<--

1.Push 2.Pop 3.Display 4.Exit

Enter your choice:

4

Program:

import java.util.\*;

abstract class shape

{

int a,b;

abstract public void printarea();

}

class rectangle extends shape

{

public int area\_rect;

public void printarea()

{

Scanner s=new Scanner(System.in);

System.out.println("enter the length and breadth of rectangle");

a=s.nextInt();

b=s.nextInt();

area\_rect=a\*b;

System.out.println("Length of rectangle "+a +"breadth of rectangle "+b);

System.out.println("The area ofrectangle is:"+area\_rect);

}

}

class triangle extends shape

{

double area\_tri;

public void printarea()

{

Scanner s=new Scanner(System.in);

System.out.println("enter the base and height of triangle");

a=s.nextInt();

b=s.nextInt();

System.out.println("Base of triangle "+a +"height of triangle "+b);

area\_tri=(0.5\*a\*b);

System.out.println("The area of triangle is:"+area\_tri);

}

}

class circle extends shape

{

double area\_circle;

public void printarea()

{

Scanner s=new Scanner(System.in);

System.out.println("enter the radius of circle");

a=s.nextInt();

area\_circle=(3.14\*a\*a);

System.out.println("Radius of circle"+a);

System.out.println("The area of circle is:"+area\_circle);

}

}

public class shapeclass

{

public static void main(String[] args)

{

rectangle r=new rectangle();

r.printarea();

triangle t=new triangle();

t.printarea();

circle r1=new circle();

r1.printarea();

}

}

Output:

enter the length and breadth of rectangle

6

7

Length of rectangle 6breadth of rectangle 7

The area ofrectangle is:42

enter the base and height of triangle

2

5

Base of triangle 2height of triangle 5

The area of triangle is:5.0

enter the radius of circle

3

Radius of circle3

The area of circle is:28.259999999999998

Program:

import java.io.\*;

import java.util.\*;

public class OperationsOnStringList {

public static void main(String[] args) throws IOException {

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

Scanner sc = new Scanner(System.in);

ArrayList <String> al = new ArrayList<String>();

boolean flag = true;

while(flag) {

System.out.println("1. Insert");

System.out.println("2. Search");

System.out.println("3. Delete");

System.out.println("4. Display");

System.out.println("5. Exit");

int option = sc.nextInt();

if(option >= 1 && option <=5) {

switch(option) {

case 1:

System.out.println("Enter element to add: ");

al.add(br.readLine());

System.out.println("Added !");

break;

case 2:

System.out.println("Enter element to search: ");

String se = br.readLine();

if(al.contains(se)) {

System.out.println("Element is at Index: "+al.indexOf(se));

}

else {

System.out.println("No Such Element in the List !");

}

break;

case 3:

System.out.println("Enter which element to delete: ");

String re = br.readLine();

if(al.contains(re)) {

al.remove(re);

System.out.println("Removed !");

}

else {

System.out.println("No such element in the List !");

}

break;

case 4:

if(al.size()==0) {

System.out.println("Empty List !");

}

else {

System.out.println(al);

}

break;

default:

flag = false;

}

}

else {

System.out.println("Invalid option");

}

}

sc.close();

}

}

Output:

1. Insert

2. Search

3. Delete

4. Display

5. Exit

1

Enter element to add:

67

Added !

1. Insert

2. Search

3. Delete

4. Display

5. Exit

1

Enter element to add:

34

Added !

1. Insert

2. Search

3. Delete

4. Display

5. Exit

1

Enter element to add:

92

Added !

1. Insert

2. Search

3. Delete

4. Display

5. Exit

4

[67, 34, 92]

1. Insert

2. Search

3. Delete

4. Display

5. Exit

5

Program:

#predefined:

import java.io.\*;

class Predefined {

public static void main (String[] args){

int a=5;

int b=0;

try{

System.out.println(a/b);

}

catch(ArithmeticException e){

e.printStackTrace();

}

}

}

#userdefined:

class MyException extends Exception {

}

public class Userdefined {

public static void main(String args[])

{

try {

throw new MyException();

}

catch (MyException ex) {

System.out.println("Hi this is my catch block");

System.out.println(ex.getMessage());

}

}

}

Output:

#predefined:

java.lang.ArithmeticException: / by zero

#userdefined:

Hi this is my catch block

Null

Program:

import java.util.Scanner;

import java.io.File;

class Filedemo

{

public static void main(String[]args)

{

Scanner input=new Scanner(System.in);

String s=input.nextLine();

File f1=new File(s);

System.out.println("File Name:"+f1.getName());

System.out.println("Path:"+f1.getPath());

System.out.println("Abs Path:"+f1.getAbsolutePath());

System.out.println("Parent:"+f1.getParent());

System.out.println("This file is:"+(f1.exists()?"Exists":"Does not exists"));

System.out.println("Is file:"+f1.isFile());

System.out.println("Is Directory:"+f1.isDirectory());

System.out.println("Is Readable:"+f1.canRead());

System.out.println("Is Writable:"+f1.canWrite());

System.out.println("Is Absolute:"+f1.isAbsolute());

System.out.println("File Last Modified:"+f1.lastModified());

System.out.println("File Size:"+f1.length()+"bytes");

System.out.println("Is Hidden:"+f1.isHidden());

}}

Output:

Doc 1

File Name:Doc 1

Path:Doc 1

Abs Path:Z:\337\Doc 1

Parent:null

This file is:Does not exists

Is file:false

Is Directory:false

Is Readable:false

Is Writable:false

Is Absolute:false

File Last Modified:0

File Size:0bytes

Is Hidden:false

Program:

import java.util.\*;

class even implements Runnable

{

public int x;

public even(int x)

{

this.x=x;

}

public void run()

{

System.out.println("New thread"+x+"is even and square of"+x+"is:"+x\*x);

}

}

class odd implements Runnable

{

public int x;

public odd(int x)

{

this.x=x;

}

public void run()

{

System.out.println("New thread"+x+"is odd and cube of"+x+"is:"+x\*x\*x);

}

}

class A extends Thread

{

public void run()

{

int num=0;

Random r=new Random();

try

{

for(int i=0;i<5;i++)

{

num=r.nextInt(100);

System.out.println("Main thread and Generated Number is"+num);

if(num%2==0)

{

Thread t1=new Thread(new even(num));

t1.start();

}

else{

Thread t2=new Thread(new odd(num));

t2.start();

}

Thread.sleep(1000);

System.out.println("-----------------------------------------");

}

}

catch(Exception ex)

{

System.out.println(ex.getMessage());

}

}

}

public class multithread

{

public static void main(String args[])

{

A a=new A();

a.start();

}}

Output:

Main thread and Generated Number is13

New thread13is odd and cube of13is:2197

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

Main thread and Generated Number is93

New thread93is odd and cube of93is:804357

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

Main thread and Generated Number is82

New thread82is even and square of82is:6724

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

Main thread and Generated Number is1

New thread1is odd and cube of1is:1

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

Main thread and Generated Number is57

New thread57is odd and cube of57is:185193

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

Program:

import java.util.LinkedList;

public class Threadcommunication {

public static void main(String[] args)

throws InterruptedException

{

final PC pc = new PC();

Thread t1 = new Thread(new Runnable() {

@Override

public void run()

{

try {

pc.produce();

}

catch (InterruptedException e) {

e.printStackTrace();

}

}

});

Thread t2 = new Thread(new Runnable() {

@Override

public void run()

{

try {

pc.consume();

}

catch (InterruptedException e) {

e.printStackTrace();

}

}

});

t1.start();

t2.start();

t1.join();

t2.join();

}

public static class PC {

LinkedList<Integer> list = new LinkedList<>();

int capacity = 2;

public void produce() throws InterruptedException

{

int value = 0;

while (true) {

synchronized (this)

{

while (list.size() == capacity)

wait();

System.out.println("Producer produced-"+ value);

list.add(value++);

notify();

Thread.sleep(1000);

}

}

}

public void consume() throws InterruptedException

{

while (true) {

synchronized (this)

{

while (list.size() == 0)

wait();

int val = list.removeFirst();

System.out.println("Consumer consumed-"+ val);

notify();

Thread.sleep(1000);

}

}

}

}

}

Output:

Producer produced-0

Producer produced-1

Consumer consumed-0

Consumer consumed-1

Producer produced-2

Producer produced-3

Consumer consumed-2

Consumer consumed-3

Producer produced-4

Producer produced-5

Consumer consumed-4

Consumer consumed-5

Producer produced-6

Producer produced-7

Consumer consumed-6

Consumer consumed-7

Producer produced-8

Producer produced-9

Consumer consumed-8

Consumer consumed-9

Producer produced-10

Producer produced-11

Consumer consumed-10

Consumer consumed-11

……………………..

……………………..

Program:

class MyClass<T extends Comparable<T>>

{

T[] vals;

MyClass(T[] o)

{

vals = o;

}

public T min()

{

T v = vals[0];

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

if(vals[i].compareTo(v) < 0)

v = vals[i];

return v;

}

public T max()

{

T v = vals[0];

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

if(vals[i].compareTo(v) > 0)

v = vals[i];

return v;

}

}

class genericfunction

{

public static void main(String args[])

{

int i;

Integer values[]={10,2,5,4,6,1};

Character characters[]={'v','p','s','a','n','h'};

Double d[]={20.2,45.4,71.6,88.3,54.6,10.4};

MyClass<Integer> iob = new MyClass<Integer>(values);

MyClass<Character> cob = new MyClass<Character>(characters);

MyClass<Double>dob = new MyClass<Double>(d);

System.out.println("Max value in values: " + iob.max());

System.out.println("Min value in values: " + iob.min());

System.out.println("Max value in characters: " + cob.max());

System.out.println("Min value in characters: " + cob.min());

System.out.println("Max value in characters: " + dob.max());

System.out.println("Min value in characters: " + dob.min());

}

}

Output:

Max value in values: 10

Min value in values: 1

Max value in characters: v

Min value in characters: a

Max value in characters: 88.3

Min value in characters: 10.4

Program:

import javax.swing.\*;

import javax.swing.event.\*;

import java.awt.\*;

import java.awt.event.\*;

class TrafficLightSimulator extends JFrame implements ItemListener {

JLabel lbl1, lbl2;

JPanel nPanel, cPanel;

CheckboxGroup cbg;

public TrafficLightSimulator() {

setTitle("Traffic Light Simulator");

setSize(600,400);

setLayout(new GridLayout(2, 1));

nPanel = new JPanel(new FlowLayout());

cPanel = new JPanel(new FlowLayout());

lbl1 = new JLabel();

Font font = new Font("Verdana", Font.BOLD, 70);

lbl1.setFont(font);

nPanel.add(lbl1);

add(nPanel);

Font fontR = new Font("Verdana", Font.BOLD, 20);

lbl2 = new JLabel("Select Lights");

lbl2.setFont(fontR);

cPanel.add(lbl2);

cbg = new CheckboxGroup();

Checkbox rbn1 = new Checkbox("Red Light", cbg, false);

rbn1.setBackground(Color.RED);

rbn1.setFont(fontR);

cPanel.add(rbn1);

rbn1.addItemListener(this);

Checkbox rbn2 = new Checkbox("Orange Light", cbg, false);

rbn2.setBackground(Color.ORANGE);

rbn2.setFont(fontR);

cPanel.add(rbn2);

rbn2.addItemListener(this);

Checkbox rbn3 = new Checkbox("Green Light", cbg, false);

rbn3.setBackground(Color.GREEN);

rbn3.setFont(fontR);

cPanel.add(rbn3);

rbn3.addItemListener(this);

add(cPanel);

setVisible(true);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}

public void itemStateChanged(ItemEvent i) {

Checkbox chk = cbg.getSelectedCheckbox();

String str=chk.getLabel();

char choice=str.charAt(0);

switch (choice) {

case 'R':lbl1.setText("STOP");

lbl1.setForeground(Color.RED);

break;

case 'O':lbl1.setText("READY");

lbl1.setForeground(Color.ORANGE);

break;

case 'G':lbl1.setText("GO");

lbl1.setForeground(Color.GREEN);

break;

}

}

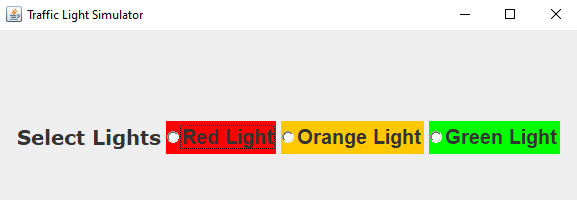
public static void main(String[] args) {

new TrafficLightSimulator();

}

}

Output:









Program:

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import javax.swing.event.\*;

class Calculator extends JFrame

{

private final Font BIGGER\_FONT = new Font("monspaced",Font.PLAIN, 20);

private JTextField textfield;

private boolean number = true;

private String equalOp = "=";

private CalculatorOp op = new CalculatorOp();

public Calculator() {

textfield = new JTextField("", 12);

textfield.setHorizontalAlignment(JTextField.RIGHT);

textfield.setFont(BIGGER\_FONT);

ActionListener numberListener = new NumberListener();

String buttonOrder = "1234567890 ";

JPanel buttonPanel = new JPanel();

buttonPanel.setLayout(new GridLayout(4, 4, 4, 4));

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

String key = buttonOrder.substring(i, i+1);

if (key.equals(" ")) {

buttonPanel.add(new JLabel(""));

} else {

JButton button = new JButton(key);

button.addActionListener(numberListener);

button.setFont(BIGGER\_FONT);

buttonPanel.add(button);

}

}

ActionListener operatorListener = new OperatorListener();

JPanel panel = new JPanel();

panel.setLayout(new GridLayout(4, 4, 4, 4));

String[] opOrder = {"+", "-", "\*", "/","=","C","sin","cos","log"};

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

JButton button = new JButton(opOrder[i]);

button.addActionListener(operatorListener);

button.setFont(BIGGER\_FONT);

panel.add(button);

}

JPanel pan = new JPanel();

pan.setLayout(new BorderLayout(4, 4));

pan.add(textfield, BorderLayout.NORTH );

pan.add(buttonPanel , BorderLayout.CENTER);

pan.add(panel , BorderLayout.EAST);

this.setContentPane(pan);

this.pack();

this.setTitle("Calculator");

this.setResizable(false);

}

private void action()

{

number = true;

textfield.setText("");

equalOp = "=";

op.setTotal("");

}

class OperatorListener implements ActionListener

{

public void actionPerformed(ActionEvent e)

{

String displayText = textfield.getText();

if (e.getActionCommand().equals("sin"))

{

textfield.setText("" + Math.sin(Double.valueOf(displayText).doubleValue()));

}else

if (e.getActionCommand().equals("cos"))

{

textfield.setText("" + Math.cos(Double.valueOf(displayText).doubleValue()));

}

else

if (e.getActionCommand().equals("log"))

{

textfield.setText("" + Math.log(Double.valueOf(displayText).doubleValue()));

}

else if (e.getActionCommand().equals("C"))

{

textfield.setText("");

}

else

{

if (number)

{

action();

textfield.setText("");

}

else

{

number = true;

if (equalOp.equals("="))

{

op.setTotal(displayText);

}else

if (equalOp.equals("+"))

{

op.add(displayText);

}

else if (equalOp.equals("-"))

{

op.subtract(displayText);

}

else if (equalOp.equals("\*"))

{

op.multiply(displayText);

}

else if (equalOp.equals("/"))

{

op.divide(displayText);

}

textfield.setText("" + op.getTotalString());

equalOp = e.getActionCommand();

}}

}

}

class NumberListener implements ActionListener {

public void actionPerformed(ActionEvent event) {

String digit = event.getActionCommand();

if (number) {

textfield.setText(digit);

number = false;

} else {

textfield.setText(textfield.getText() + digit);

}

}}

public class CalculatorOp {

private int total;

public CalculatorOp() {

total = 0;

}

public String getTotalString() {

return ""+total;

}

public void setTotal(String n) {

total = convertToNumber(n);

}

public void add(String n) {

total += convertToNumber(n);

}

public void subtract(String n) {

total -= convertToNumber(n);

}

public void multiply(String n) {

total \*= convertToNumber(n);

}

public void divide(String n) {

total /= convertToNumber(n);

}

private int convertToNumber(String n) {

return Integer.parseInt(n);

}

}}

class SwingCalculator {

public static void main(String[] args) {

JFrame frame = new Calculator();

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setVisible(true);

}

}

Output:

