**Program : 1**

**Aim :** Define a class ‘Product’ with data members pcode , pname , price . Create 3 objects of the class and find the product having the lowest price.

**Source Code:**

import java.util.\*;

class Product{

int pcode;

String name;

int price;

Product(){

pcode=123;

name="Pen";

price=10;

}

Product(int pcode,String name,int price){

this.pcode=pcode;

this.name=name;

this.price=price;

}

void display(){

System.out.println("Pcode = "+pcode+"\nName = "+name+"\nPrice = "+price);

}

}

class ProductCompare{

public static void main(String args[]){

Product p1=new Product();

Product p2=new Product(234,"Book",48);

System.out.println("First Product");

p1.display();

System.out.println("Second Product");

p2.display();

Scanner s=new Scanner(System.in);

System.out.println("Enter the third product's productcode,product price ,product name");

int pcode=s.nextInt();

String name = s.next();

name+=s.nextLine();

int price=s.nextInt();

System.out.println("Third Product");

Product p3=new Product(pcode,name,price);

p3.display();

if(p1.price<p2.price && p1.price<p3.price){

System.out.println("\n\nProduct with Lowest price = "+p1.name+"-"+p1.price);

}

else if(p2.price<p1.price && p2.price<p3.price){

System.out.println("\n\nProduct with Lowest price = "+p2.name+"-"+p2.price);

}

else{

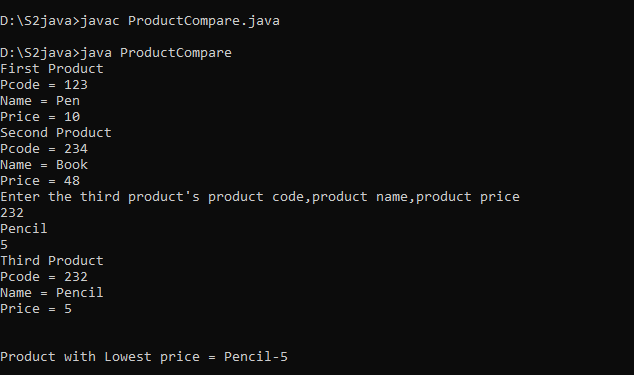
System.out.println("\n\nProduct with Lowest price = "+p3.name+"-"+p3.price);

}

}

}

**Output**



**Program : 2**

**Aim**: Read two matrix from the console & perform matrix addition.

**Source Code:**

import java.util.\*;

class Matrix{

int m,n,p,q,i,j;

int a[][]=new int[20][20];

int b[][]=new int[20][20];

int c[][]=new int[20][20];

Scanner s=new Scanner(System.in);

void add(){

System.out.println("\nEnter the limit of first matrix : ");

m=s.nextInt();

n=s.nextInt();

System.out.println("\nEnter the limit of second matrix : ");

p=s.nextInt();

q=s.nextInt();

if(m==p &&n==q){

System.out.println("\nEnter First array elements ");

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

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

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

}

}

System.out.println("\nEnter Second array elements ");

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

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

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

}

}

System.out.println("\nFirst matrix ");

for (int i = 0; i < m; i++){

for (int j = 0; j < n; j++){

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

}

System.out.println("");

}

System.out.println("\nSecond matrix ");

for (int i = 0; i < p; i++){

for (int j = 0; j < q; j++){

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

}

System.out.println("");

}

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

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

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

}

}

System.out.println("Matrix after addition:");

for (int i = 0; i < p; i++){

for (int j = 0; j < q; j++){

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

}

System.out.println("");

}

}

else{

System.out.println("\nAddition cannot be performed");

}

}

}

class MatrixAddition{

public static void main(String args[]){

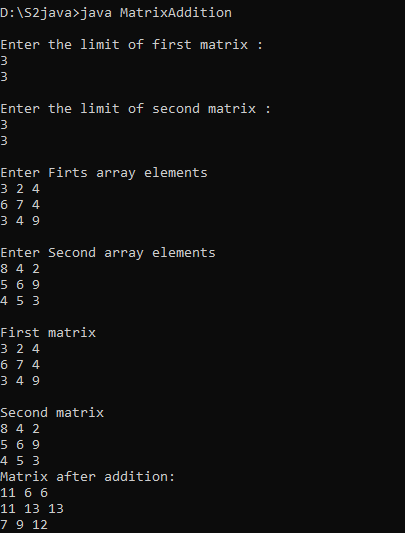
Matrix mobj=new Matrix();

mobj.add();

}

}

**Output**



**Program : 3**

**Aim**:Add two complex numbers.

**Source Code**

import java.util.\*;

public class ComplexNumber{

double real;

double img;

ComplexNumber(double r,double i){

this.real=r;

this.img=i;

}

ComplexNumber(){}

ComplexNumber sum(ComplexNumber c1,ComplexNumber c2){

ComplexNumber temp=new ComplexNumber();

temp.real=c1.real+c2.real;

temp.img=c1.img+c2.img;

return temp;

}

void display(){

System.out.println(real+" +i"+img);

}

public static void main(String args[]){

Double r1,r2,i1,i2;

Scanner s=new Scanner(System.in);

System.out.println("Enter real and imaginary part of first complex number : ");

r1=s.nextDouble();

i1=s.nextDouble();

System.out.println("Enter real and imaginary part of second complex number : ");

r2=s.nextDouble();

i2=s.nextDouble();

ComplexNumber c1=new ComplexNumber(r1,i1);

ComplexNumber c2=new ComplexNumber(r2,i2);

System.out.println("First Complex Number : ");

c1.display();

System.out.println("Second Complex Number : ");

c2.display();

ComplexNumber temp1=new ComplexNumber();

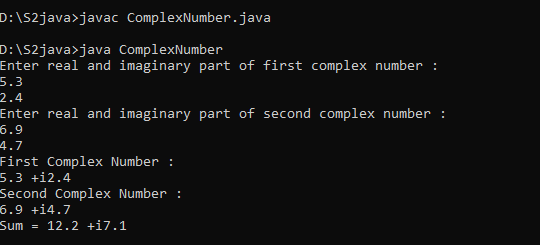
temp1=temp1.sum(c1,c2);

System.out.println("Sum = "+temp1.real+" +i"+temp1.img);

}

}

**Output**



**Program : 4**

**Aim:** Create CPU with attribute price. Create inner class Processor (no. of cores, manufacturer) and static nested class RAM (memory, manufacturer). Create an object of CPU and print information of Processor and RAM.

**Source Code**

package s2java;

import java.util.\*;

import java.io.\*;

class Cpu{

double price;

class Processor{

double cores;

String manufacturer;

double getCache(){

return 5.6;

}

}

static class Ram{

double memory;

String manufacturer;

double getClockSpeed(){

return 4.9;

}

}

}

public class CpuMain {

public static void main(String[] args) {

Scanner s= new Scanner(System.*in*);

Cpu objcpu=new Cpu();

Cpu.Processor objproc=objcpu.new Processor();

Cpu.Ram objram=new Cpu.Ram();

System.*out*.println("Enter the price of CPU");

double p=s.nextDouble();

System.*out*.println("Enter no of cores");

double c=s.nextDouble();

System.*out*.println("Enter the manufacturer of processor");

String m=s.next();

System.*out*.println("Enter the memory of RAM");

double d=s.nextDouble();

System.*out*.println("Enter the manufacturer of ram");

String n=s.next();

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

objcpu.price=p;

objproc.cores=c;

objproc.manufacturer=m;

objram.memory=d;

objram.manufacturer=n;

System.*out*.println();

System.*out*.println("Price of the cpu: "+objcpu.price);

System.*out*.println("No of core in processor: "+objproc.cores);

System.*out*.println("Name of the manufacturer: "+objproc.manufacturer);

System.*out*.println("Processor cache: "+objproc.getCache());

System.*out*.println("Memory capacity: "+objram.memory+" GB");

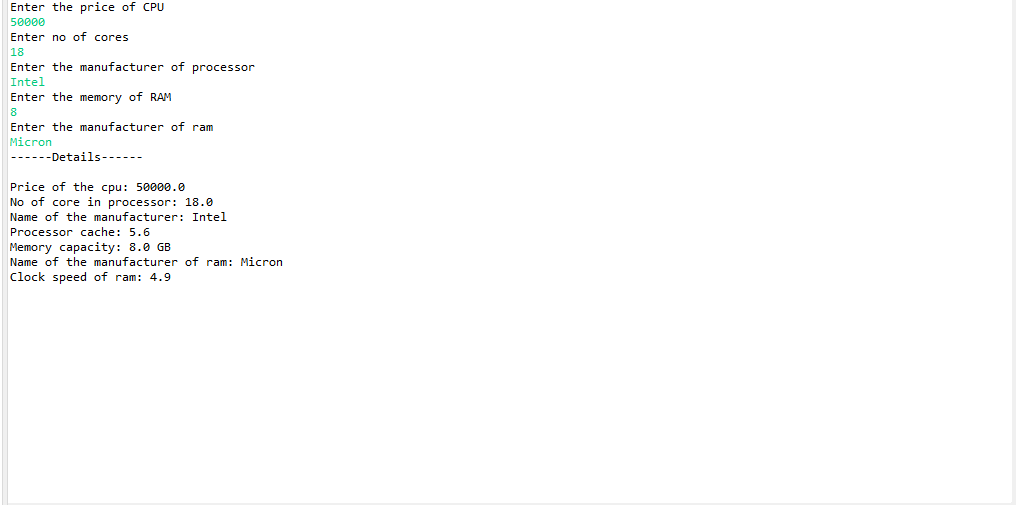
System.*out*.println("Name of the manufacturer of ram: "+objram.manufacturer);

System.*out*.println("Clock speed of ram: "+objram.getClockSpeed());

}

}

**Output**



**Program 5**

**Aim :** Program to Sort strings.

**Source Code**

package s2java;

import java.util.Arrays;

import java.util.Scanner;

public class Sort {

public static void main(String[] args) {

int ch,n,i;

Scanner s=new Scanner(System.*in*);

System.*out*.println("Enter the limit");

n=s.nextInt();

String arr[]=new String[n];

System.*out*.println("------Menu------\n1-Read\n2-Sort using inbuilt sort functions\n3-Sort without using inbuilt sort function\n4-Exit");

while(true)

{

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

ch=s.nextInt();

switch(ch) {

case 1:System.*out*.println("Enter the values into array");

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

arr[i]=s.next();

}

break;

case 2:Arrays.*sort*(arr);

System.*out*.println("Sorted String : "+Arrays.*toString*(arr));

break;

case 3:for(i=0;i<n;i++) {

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

if(arr[i].compareTo(arr[j])>0){

String temp=new String();

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

System.*out*.println("Sorted String :");

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

System.*out*.println(arr[i]+" ");

}

break;

case 4: System.*out*.println("Exiting .......");

System.*exit*(0);

default:System.*out*.println("Invalid choice");

break;

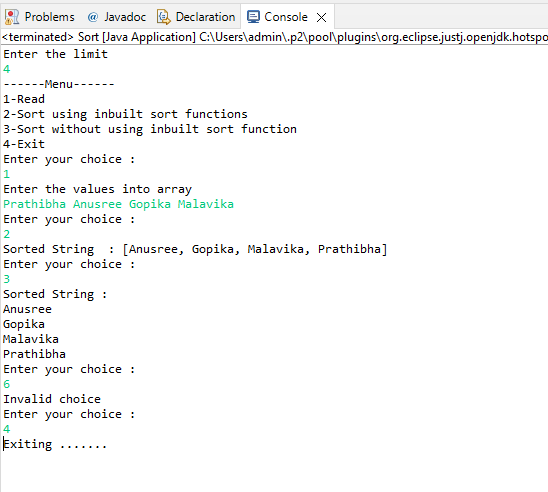
}

}

}

}

**Output**



**Program 6**

**Aim :** Perform string manipulations.

**Source Code**

package s2java;

import java.util.\*;

public class StringManipulation {

public static void main(String[] args) {

int ch;

Scanner sc=new Scanner(System.*in*);

String s1=new String();

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

s1=sc.next();

s1+=sc.nextLine();

System.*out*.println("----String Operations---------\n1-String length\n2-String Concatenation\n3-Character Extraction\n4-String Comparison\n5-Searching Substrings\n6-Modifying a string\n7-Exit");

while(true) {

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

ch=sc.nextInt();

switch(ch) {

case 1: System.*out*.println("Length of string : "+s1.length());

break;

case 2:System.*out*.println("Enter the string to be concatenated");

String s2=new String();

s2=sc.next();

System.*out*.println("String after concatenation : "+s1.concat(s2));

break;

case 3:System.*out*.println("Enter the position of the character");

int p;

p=sc.nextInt();

System.*out*.println("Character at the position "+p+" : "+s1.charAt(p));

break;

case 4:System.*out*.println("Enter the string to be compared ");

String s3=new String();

s3=sc.next();

s3+=sc.nextLine();

Boolean result=s1.equals(s3);

System.*out*.println("Is the two strings "+s1+" and "+s3+" are equal?"+result);

break;

case 5:System.*out*.println("Enter the substring to be searched : ");

String sub=new String();

sub=sc.next();

int last\_index=s1.lastIndexOf(sub);

int first\_index=s1.indexOf(sub,2);

System.*out*.println("Last index = "+last\_index+"\nFirst index = "+first\_index);

break;

case 6:System.*out*.println("Enter the string to be replaced");

String s4=new String();

s4=sc.next();

System.*out*.println("Enter string for replacing ");

String s5=new String();

s5=sc.next();

String replace\_string=s1.replace(s4,s5);

System.*out*.println("The string after replace : "+replace\_string);

break;

case 7:System.*out*.println("Exiting");

System.*exit*(0);

default:System.*out*.println("Invalid Input");

break;

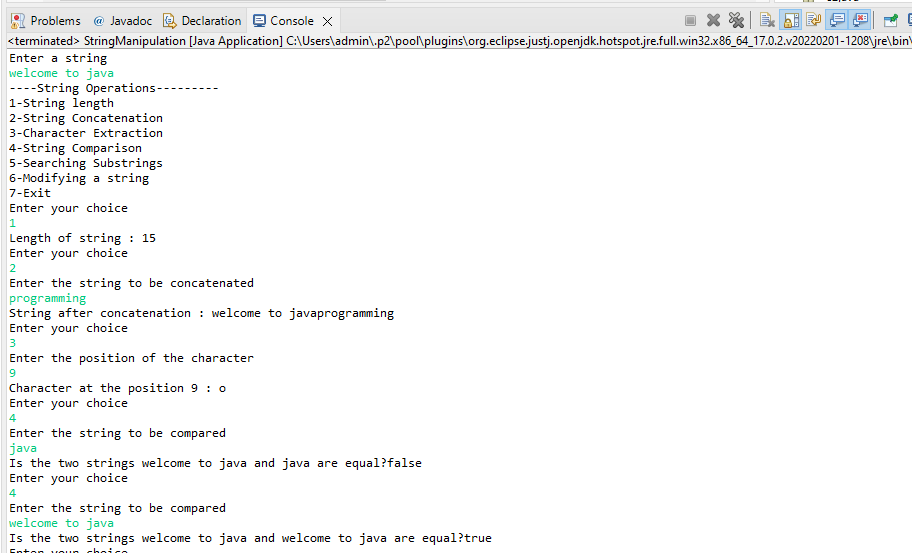
}

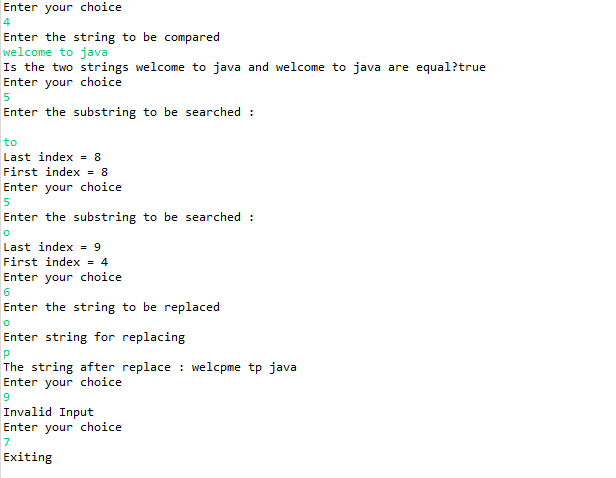
}

}

}

**Output**





**Program : 7**

**Aim:** Program to create a class for Employee having attributes eNo, eName eSalary. Read n

employ information and Search for an employee given eNo, using the concept of Array of objects.

**Source Code**

package s2java;

import java.util.\*;

import java.io.\*;

class Employeeinfo{

int eNo;

String eName;

float eSalary;

public void input(){

Scanner sc = new Scanner(System.*in*);

System.*out*.println("Enter Employee Number:");

this.eNo= sc.nextInt();

System.*out*.println("Enter Employee Name:");

this.eName = sc.next();

eName+=sc.nextLine();

System.*out*.println("Enter the salary:");

this.eSalary = sc.nextFloat();

}

void display(){

System.*out*.println("Employee Number: "+eNo);

System.*out*.println("Employee Name: "+eName);

System.*out*.println("Employee Salary: "+eSalary);

}

}

public class Employee {

public static void main(String[] args) {

// TODO Auto-generated method stub

int f=0;

Scanner sc=new Scanner(System.*in*);

System.*out*.println("Enter the number of employees:");

int n=sc.nextInt();

Employeeinfo e[]=new Employeeinfo[n];

for(int i=0;i<n;i++) {

e[i]=new Employeeinfo();

e[i].input();

System.*out*.println("\n");

}

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

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

{

e[i].display();

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

}

System.*out*.println("Enter the employee number for details:");

int empno=sc.nextInt();

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

{

if(e[i].eNo==empno)

{

e[i].display();

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

f=1;

}

}

if(f!=1){

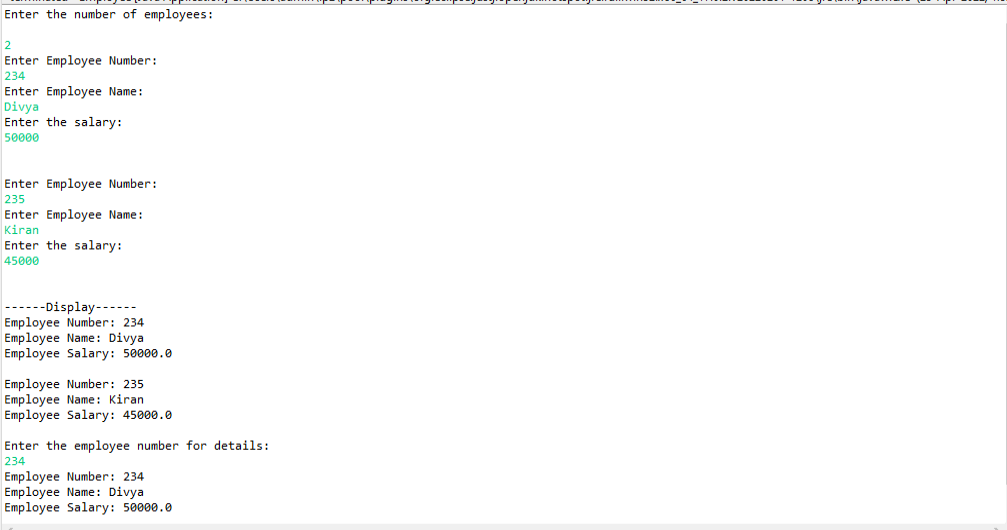
System.*out*.println("Employee doesnot exists");

}

}

}

**Output**



**Program : 8**

**Aim :** Area of different shapes using overloaded functions.

**Source Code**

package s2java;

import java.util.\*;

class Overloading{

void area(float a) {

System.*out*.println("Area of square = "+Math.*pow*(a,2));

}

int area(int a,int b) {

System.*out*.println("Area of Rectangle = "+a\*b);

return 0;

}

float area(double a) {

double r=3.14\*a\*a;

System.*out*.println("Area of circle = "+r);

return 0;

}

double area(float h,int b) {

System.*out*.println("Area of traingle = "+0.5\*b\*h);

return 0;

}

}

public class AreaOverloading {

public static void main(String[] args) {

int l,b,tb,ch;

double radius;

float th,s;

Overloading obj=new Overloading();

Scanner sc=new Scanner(System.*in*);

System.*out*.println("\n1-Square\n2-Circle\n3-Rectangle\n4-Triangle\n5-Exit");

while(true) {

System.*out*.println("Enter your choice\n");

ch=sc.nextInt();

switch(ch) {

case 1 :System.*out*.println("Enter the side of square : ");

s=sc.nextFloat();

obj.area(s);

break;

case 2:System.*out*.println("Enter the radius of circle :");

radius=sc.nextDouble();

obj.area(radius);

break;

case 3:System.*out*.println("Enter the length and breadth of rectangle :");

l=sc.nextInt();

b=sc.nextInt();

obj.area(l,b);

break;

case 4:System.*out*.println("Enter height and breadth of triangle : ");

th=sc.nextFloat();

tb=sc.nextInt();

obj.area(th,tb);

break;

case 5:System.*out*.println("Exiting");

System.*exit*(0);

default:System.*out*.println("Invalid Choice");

break;

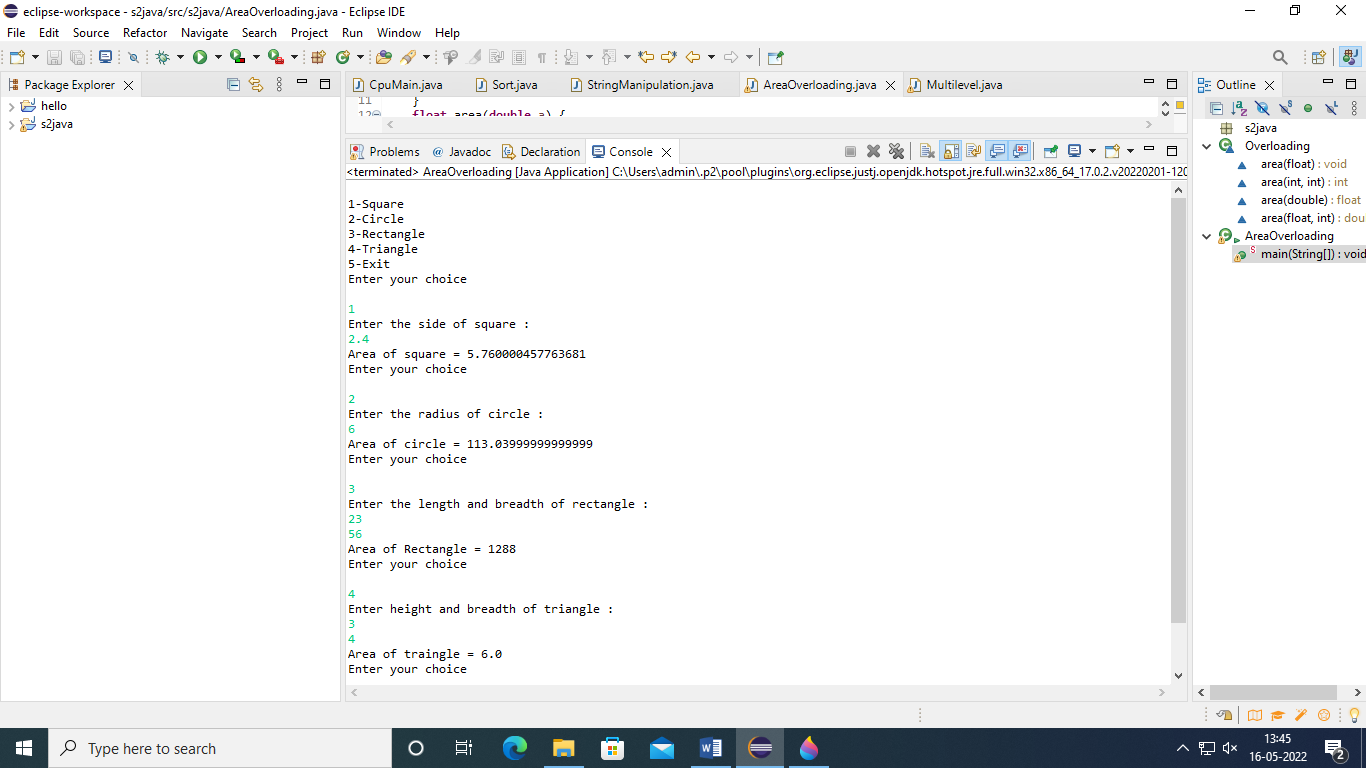
}

}

}

}

**Output**



**Program 9**

**Aim :** Create a class ‘Person’ with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class ‘Employee’ that inherits the properties of class Person and also contains its own data members like Empid, Company\_name, Qualification, Salary and its own constructor. Create another class ‘Teacher’ that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and contain constructors and methods to display the data members. Use array of objects to display details of N teachers.

**Source Code**

package s2java;

import java.util.\*;

class Person{

String name,gender,address;

int age;

Person(String name,String gender,String address,int age){

this.name=name;

this.gender=gender;

this.address=address;

this.age=age;

}

}

class Emp extends Person{

int empid;

String company\_name,qualification;

float salary;

Emp(String name,String gender,String address,int age,int empid,String company\_name,String qualification,float salary){

super(name,gender,address,age);

this.empid=empid;

this.company\_name=company\_name;

this.qualification=qualification;

this.salary=salary;

}

}

class Teacher extends Emp{

String subject,department;

int teacher\_id;

Teacher(String name,String gender,String address,int age,int empid,String company\_name,String qualification,float salary,String subject,String department,int tid){

super(name,gender,address,age,empid,company\_name,qualification,salary);

this.subject=subject;

this.department=department;

this.teacher\_id=tid;

}

void display() {

System.*out*.println("Name :"+name+"\nGender : "+gender+"\nAddress : "+address+"\nAge : "+age+"\nEmployee id : "+empid+"\nCompany Name : "+company\_name+"\nQualification : "+qualification+"\nSalary : "+salary+"\nSubject : "+subject+"\nDepartment : "+department+"\nTeacher id : "+teacher\_id);

}

}

public class Multilevel {

public static void main(String[] args) {

String n,g,add,cname,quali,sub,dept;

int age,eid,tid;

float s;

Scanner sc=new Scanner(System.*in*);

System.*out*.println("Enter the number of teachers");

int ch=sc.nextInt();

Teacher t[]=new Teacher[ch];

for(int i=0;i<ch;i++) {

System.*out*.println("\nEnter the name ");

n=sc.next();

n+=sc.nextLine();

System.*out*.println("Enter gender ");

g=sc.next();

g+=sc.nextLine();

System.*out*.println("Enter address ");

add=sc.next();

add+=sc.nextLine();

System.*out*.println("Enter age ");

age=sc.nextInt();

System.*out*.println("Enter Employee id ");

eid=sc.nextInt();

System.*out*.println("Enter Company name ");

cname=sc.next();

cname+=sc.nextLine();

System.*out*.println("Enter Qualification ");

quali=sc.next();

quali+=sc.nextLine();

System.*out*.println("Enter Salary ");

s=sc.nextFloat();

System.*out*.println("Enter Subject ");

sub=sc.next();

sub+=sc.nextLine();

System.*out*.println("Enter Department");

dept=sc.next();

dept+=sc.nextLine();

System.*out*.println("Enter teacher id");

tid=sc.nextInt();

t[i]=new Teacher(n,g,add,age,eid,cname,quali,s,sub,dept,tid);

}

for(int i=0;i<ch;i++) {

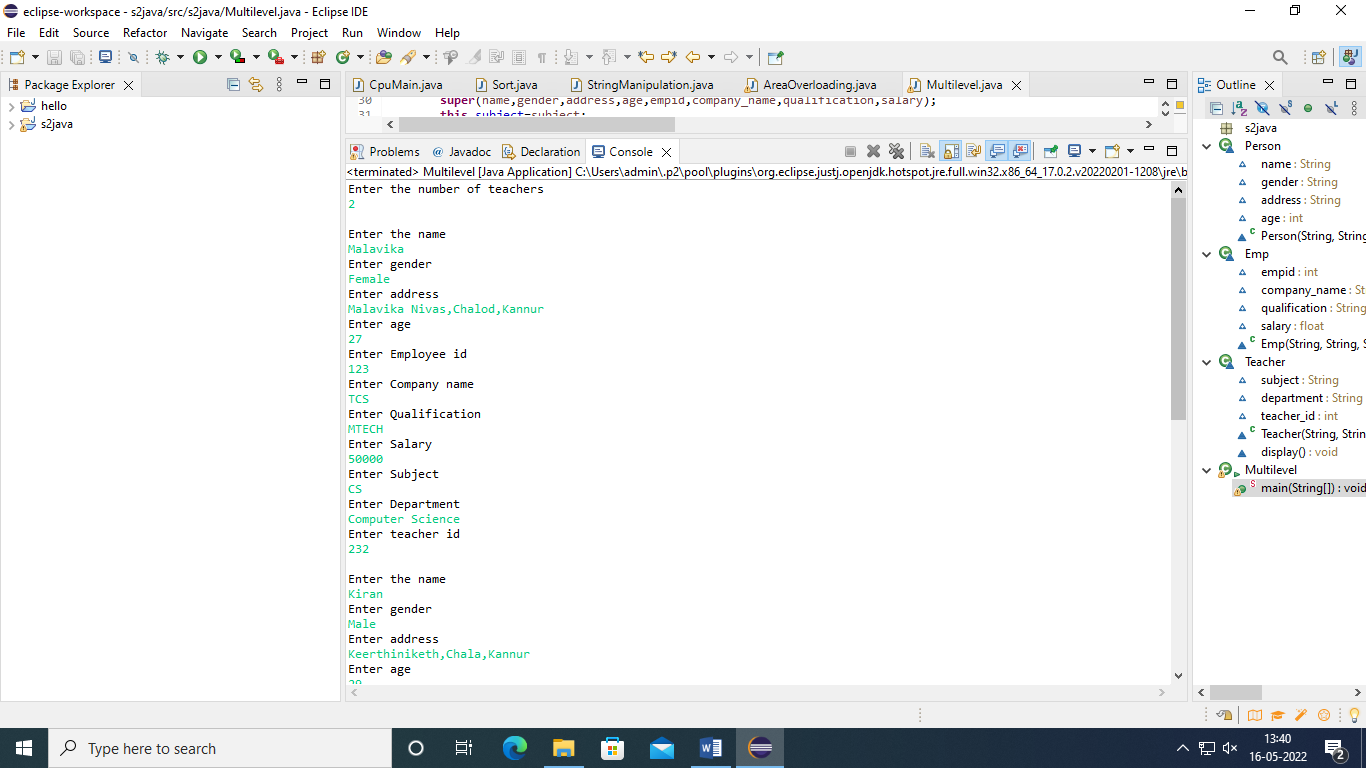
t[i].display();

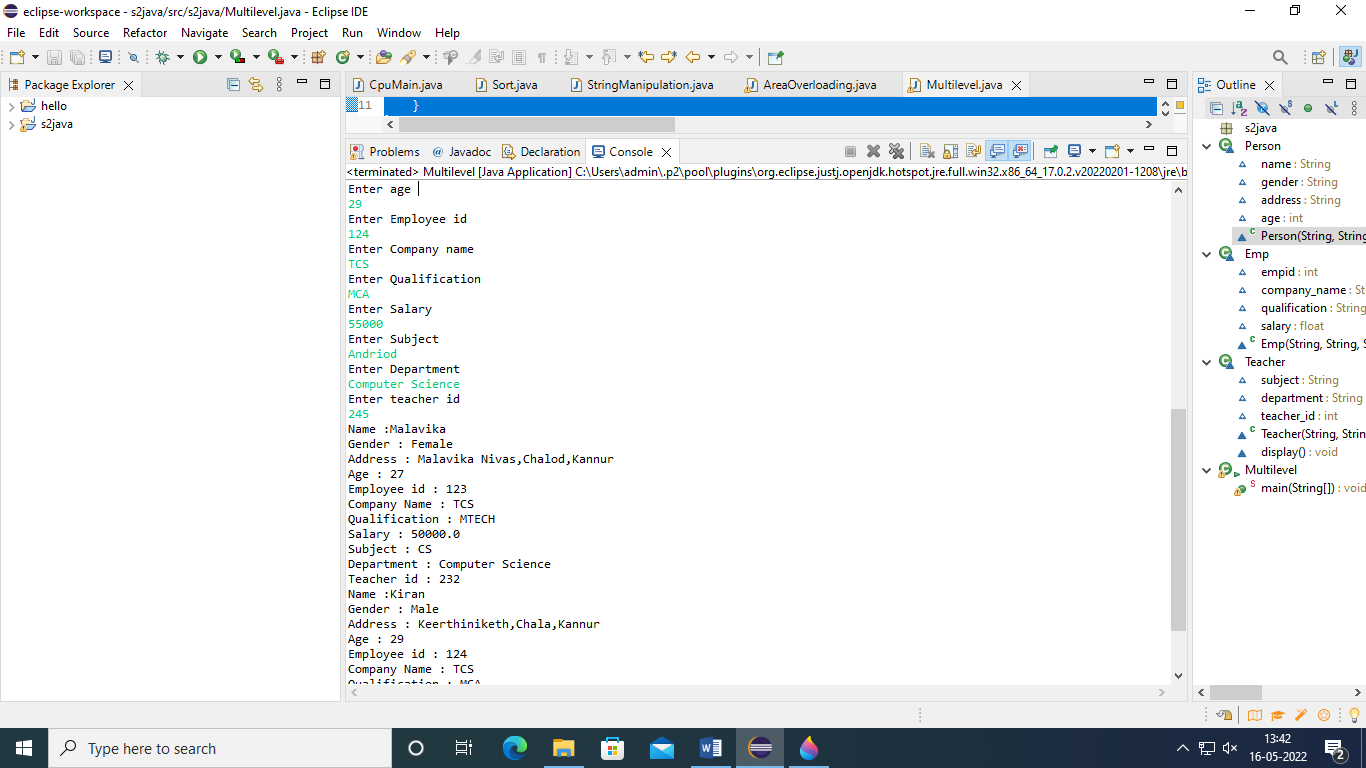
}

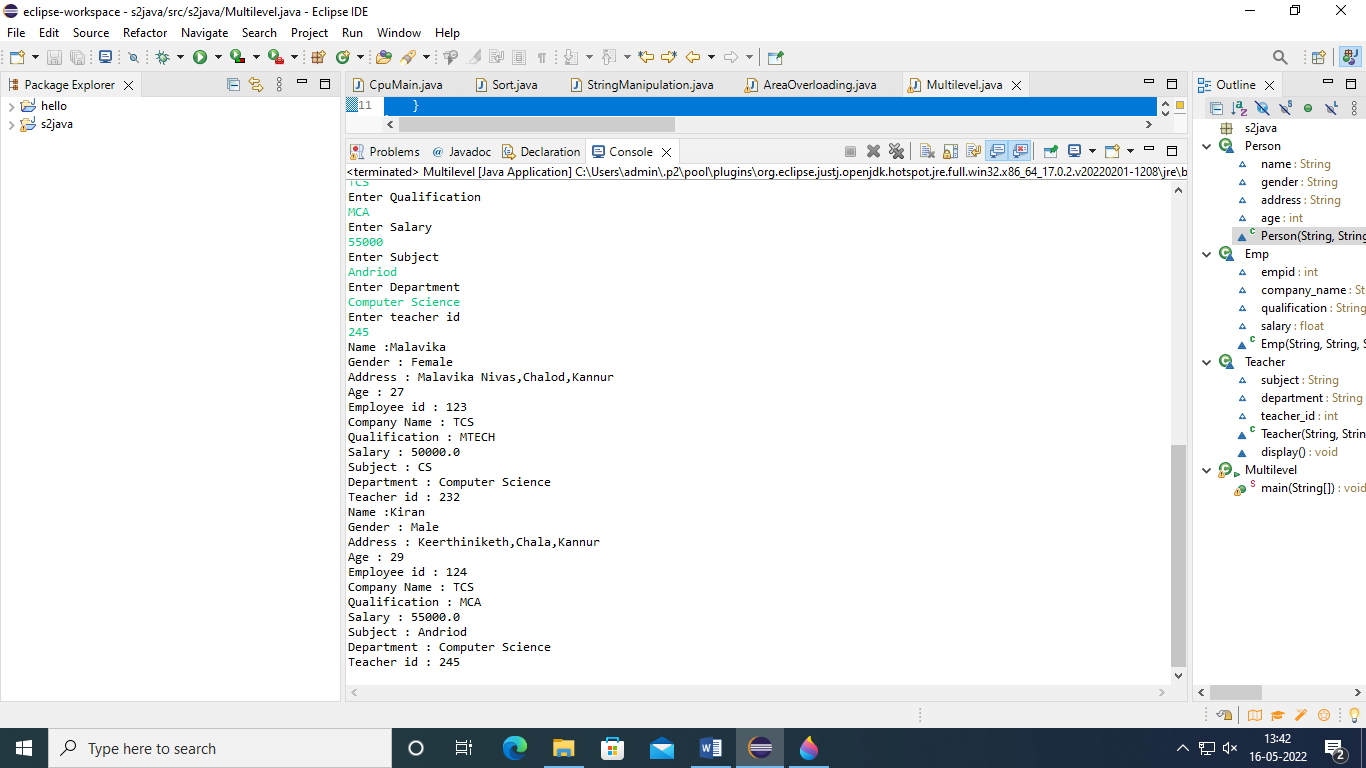
}

}

**Output**







**Program : 10**

**Aim :** Create an interface having prototypes of functions area() and perimeter(). Create two

classes Circle and Rectangle which implements the above interface. Create a menu driven

program to find area and perimeter of objects.

**Source Code**

package s2java;

import java.util.\*;

interface Shapes{

void area();

void perimeter();

}

class Circle1 implements Shapes{

int radius;

Circle1(int r){

this.radius=r;

}

public void area(){

System.*out*.println("Area of circle :"+3.14\*radius\*radius);

}

public void perimeter(){

System.*out*.println("Perimeter of circle : "+(2\*3.14\*radius));

}

}

class Rect implements Shapes{

int l,b;

Rect(int l,int b){

this.l=l;

this.b=b;

}

public void area(){

System.*out*.println("Area of Rectangle :"+l\*b);

}

public void perimeter() {

int p=2\*(l+b);

System.*out*.println("Perimeter of rectangle :"+p);

}

}

public class ShapesInterface{

public static void main(String[] args) {

Scanner sc=new Scanner(System.*in*);

int ch,r,l,b;

System.*out*.println("\n1-Circle\n2-Rectangle\n3-Exit");

while(true) {

System.*out*.println("Enter your choice\n");

ch=sc.nextInt();

switch(ch) {

case 1:

System.*out*.println("Enter the radius of circle :");

r=sc.nextInt();

Circle1 cobj=new Circle1(r);

cobj.area();

cobj.perimeter();

break;

case 2:

System.*out*.println("Enter the length and breadth of rectangle :");

l=sc.nextInt();

b=sc.nextInt();

Rect robj=new Rect(l,b);

robj.area();

robj.perimeter();

break;

case 3:System.*out*.println("Exiting");

System.*exit*(0);

default:System.*out*.println("Invalid Choice");

break;

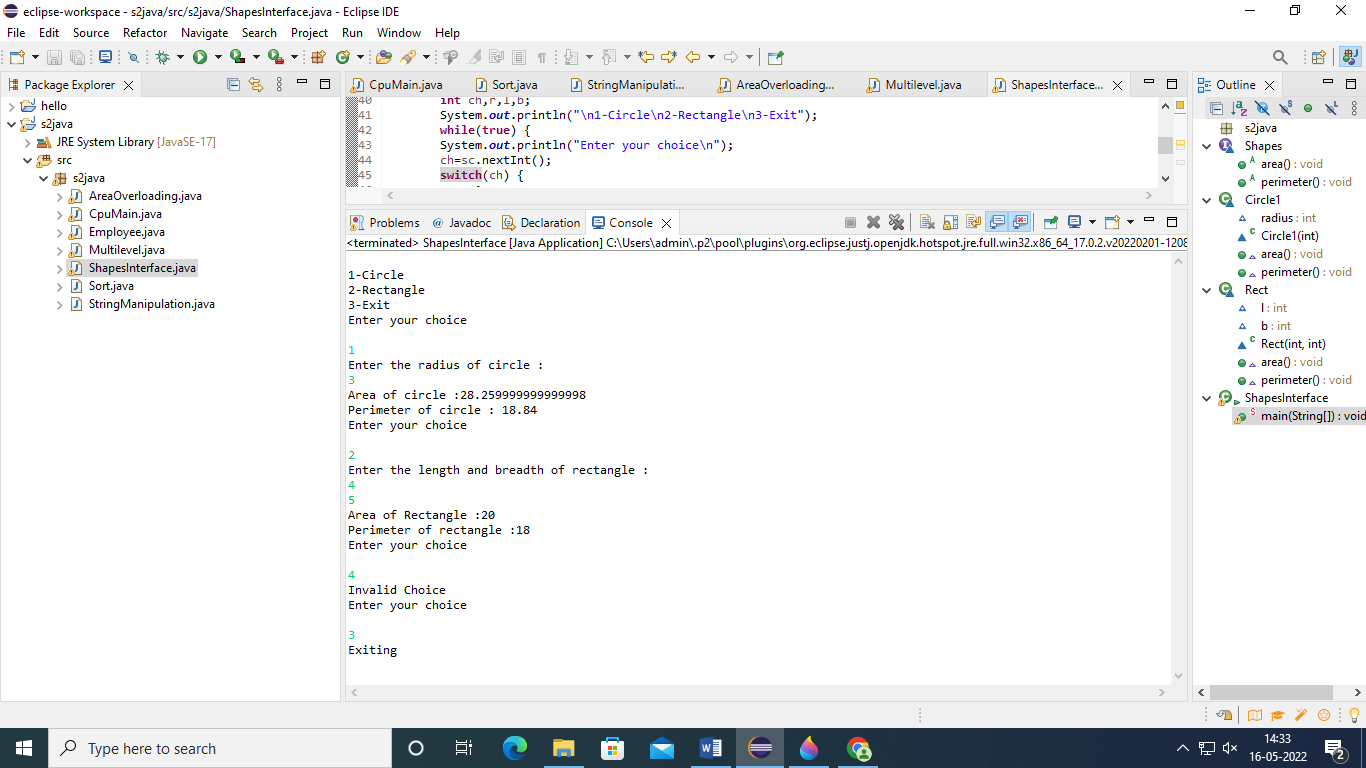
}

}

}

}

**Output**



**Program : 11**

**Aim :** Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle,Square and Circle. Test the package by finding the area of these figures.

**Source Code**

Package shapes

package Shapes;

public class Rectangle {

int l,b;

public Rectangle(int l,int b){

this.l=l;

this.b=b;

}

public void RectArea() {

System.*out*.println("Area of Rectangle :"+l\*b);

}

}

package Shapes;

public class Circle {

float r;

public Circle(float r){

this.r=r;

}

public void CirArea() {

System.*out*.println("Area of circle :"+(3.14\*r\*r));

}

}

package Shapes;

public class Square {

int s;

public Square(int s){

this.s=s;

}

public void SquArea() {

System.*out*.println("Area of square : "+s\*s);

}

}

package Shapes;

public class Triangle {

int h,b;

public Triangle(int h,int b){

this.h=h;

this.b=b;

}

public void TriArea() {

System.*out*.println("Area of Triangle : "+(0.5\*h\*b));

}

}

package s2java;

import Shapes.\*;

import java.util.\*;

public class AreaPackage {

public static void main(String[] args) {

Scanner sc=new Scanner(System.*in*);

int s,l,b,h,ch;

float r;

System.*out*.println("\n------Area---------\n1-Circle\n2-Rectangle\n3-Square\n4-Triangle\n5-Exit");

while(true) {

System.*out*.println("Enter your choice\n");

ch=sc.nextInt();

switch(ch) {

case 1:

System.*out*.println("Enter the radius of circle :");

r=sc.nextFloat();

Circle cobj=new Circle(r);

cobj.CirArea();

break;

case 2:System.*out*.println("Enter the length and breadth of rectangle :");

l=sc.nextInt();

b=sc.nextInt();

Rectangle robj=new Rectangle(l,b);

robj.RectArea();

break;

case 3:System.*out*.println("Enter the side of square :");

s=sc.nextInt();

Square sobj=new Square(s);

sobj.SquArea();

break;

case 4:System.*out*.println("Enter height and breadth of triangle :");

h=sc.nextInt();

b=sc.nextInt();

Triangle tobj=new Triangle(h,b);

tobj.TriArea();

break;

case 5:System.*out*.println("Exiting");

System.*exit*(0);

default:System.*out*.println("Invalid Choice");

break;

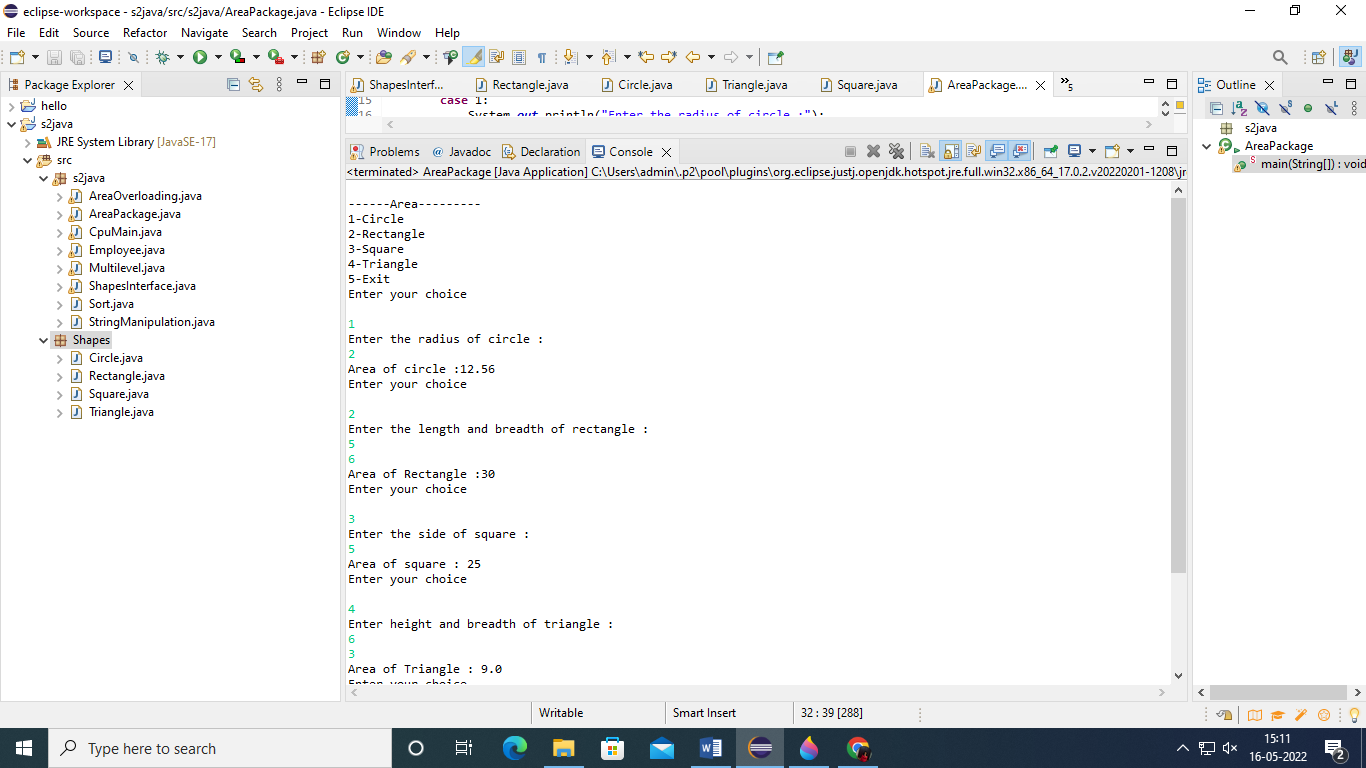
}

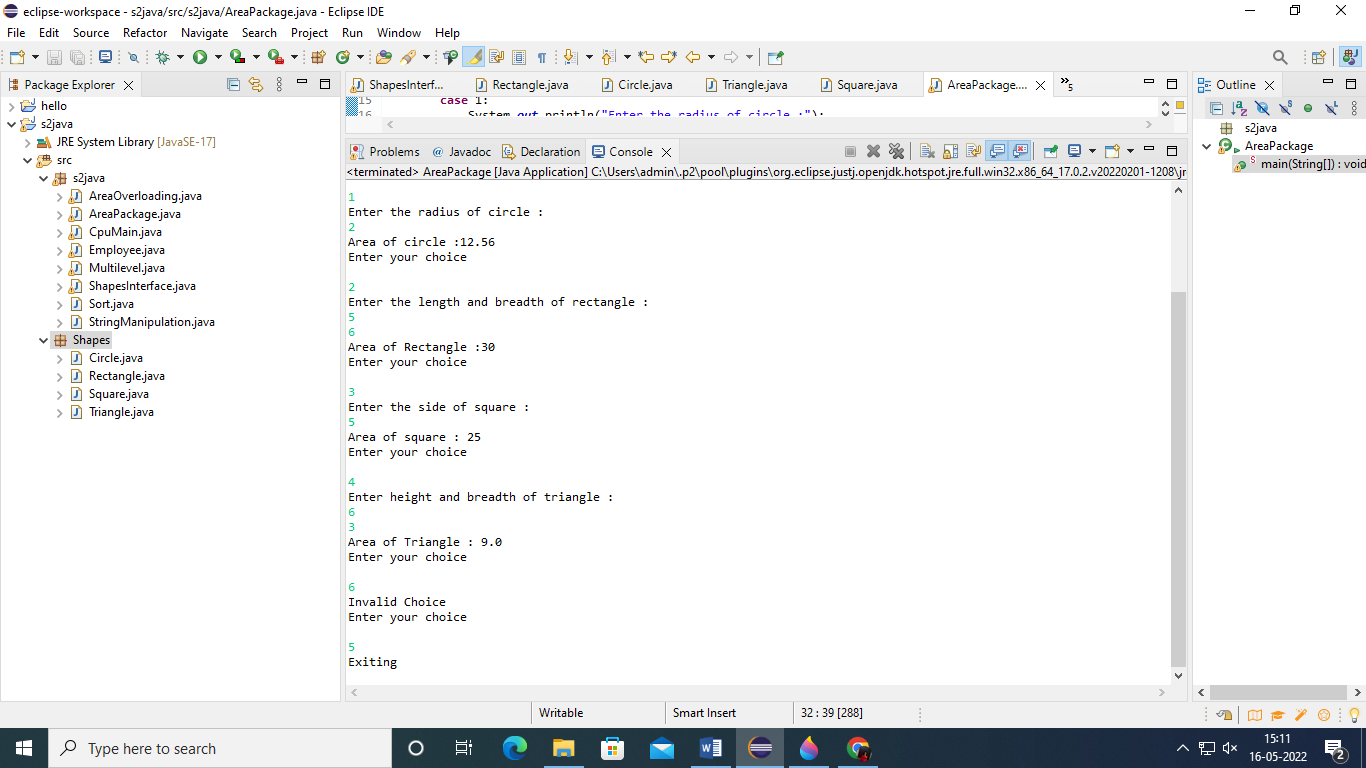
}

}

}

**Output**





**Program 12**

**Aim** : Write a user defined exception class to authenticate the user name and password.

**Source Code**

package s2java;

import java.util.\*;

import java.lang.Exception.\*;

class UsernameException extends Exception{

UsernameException(String msg){

System.*out*.println(msg);

}

}

class PasswordException extends Exception{

PasswordException(String msg){

System.*out*.println(msg);

}

}

public class LoginException {

public static void main(String[] args) {

Scanner s=new Scanner(System.*in*);

String username,password;

System.*out*.println("Enter username : ");

username=s.nextLine();

System.*out*.println("Enter password : ");

password=s.nextLine();

try {

if(username.length()<8)

throw new UsernameException("Username length must be greater than 8 characters\n");

else if(!password.equals("admin"))

throw new PasswordException("Incorrect password\n");

else

System.*out*.println("Login Successfull");

}

catch(UsernameException u) {

System.*out*.println(u);

}

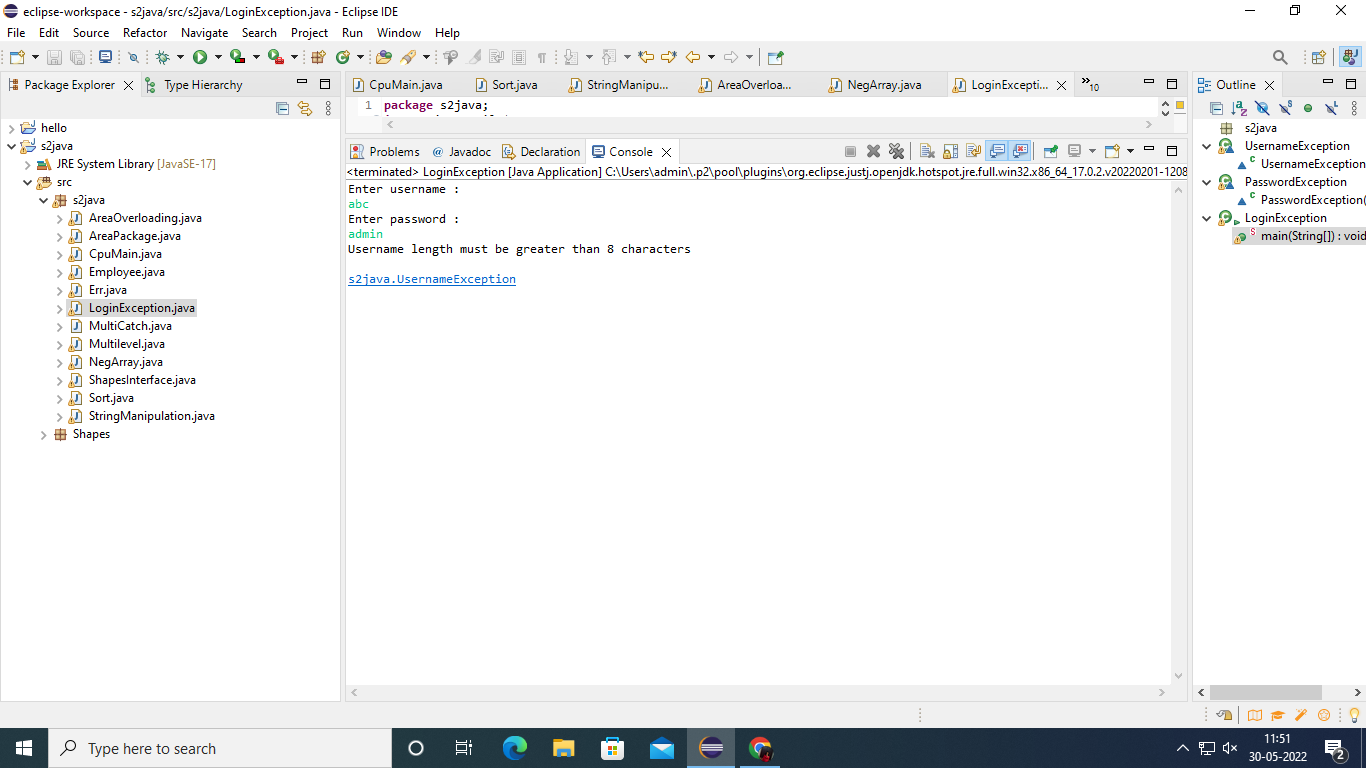
catch(PasswordException p) {

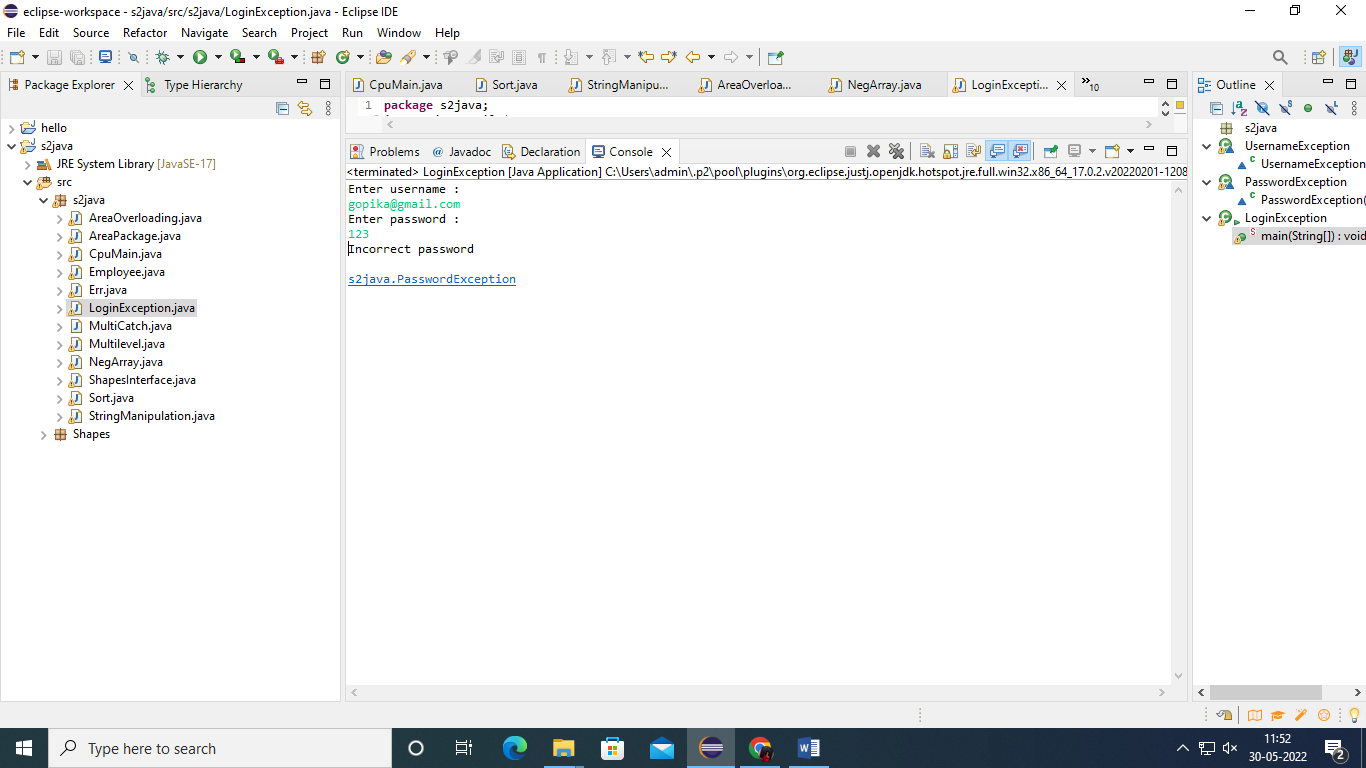
System.*out*.println(p);

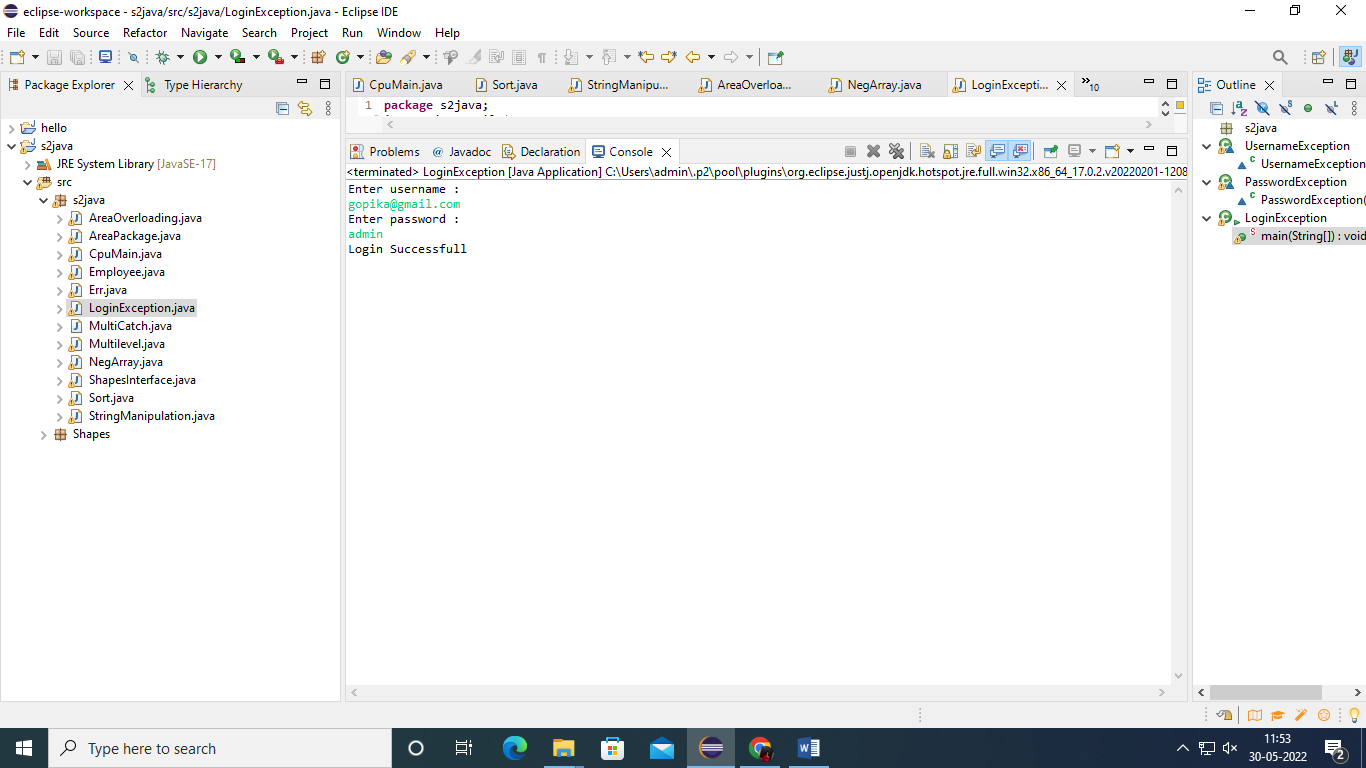
}

}}

**Output**







**Program : 13**

**Aim :**  Find the average of N positive integers, raising a user defined exception for each negative input.

**Source Code**

package s2java;

import java.util.\*;

import java.io.IOException;

class MyException extends Exception{

public MyException(String str) {

System.out.println(str);

}

}

public class SignException {

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

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

int n=s.nextInt();

int k=0,sum=0;

int numbers[]=new int[n];

while(n>0) {

try {

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

int num=s.nextInt();

if(num<0)

throw new MyException(" Number is negative ");

else {

numbers[k]=num;

sum=sum+num;

k++;

}

n--;

}

catch(MyException m) {

System.out.println(m);

}

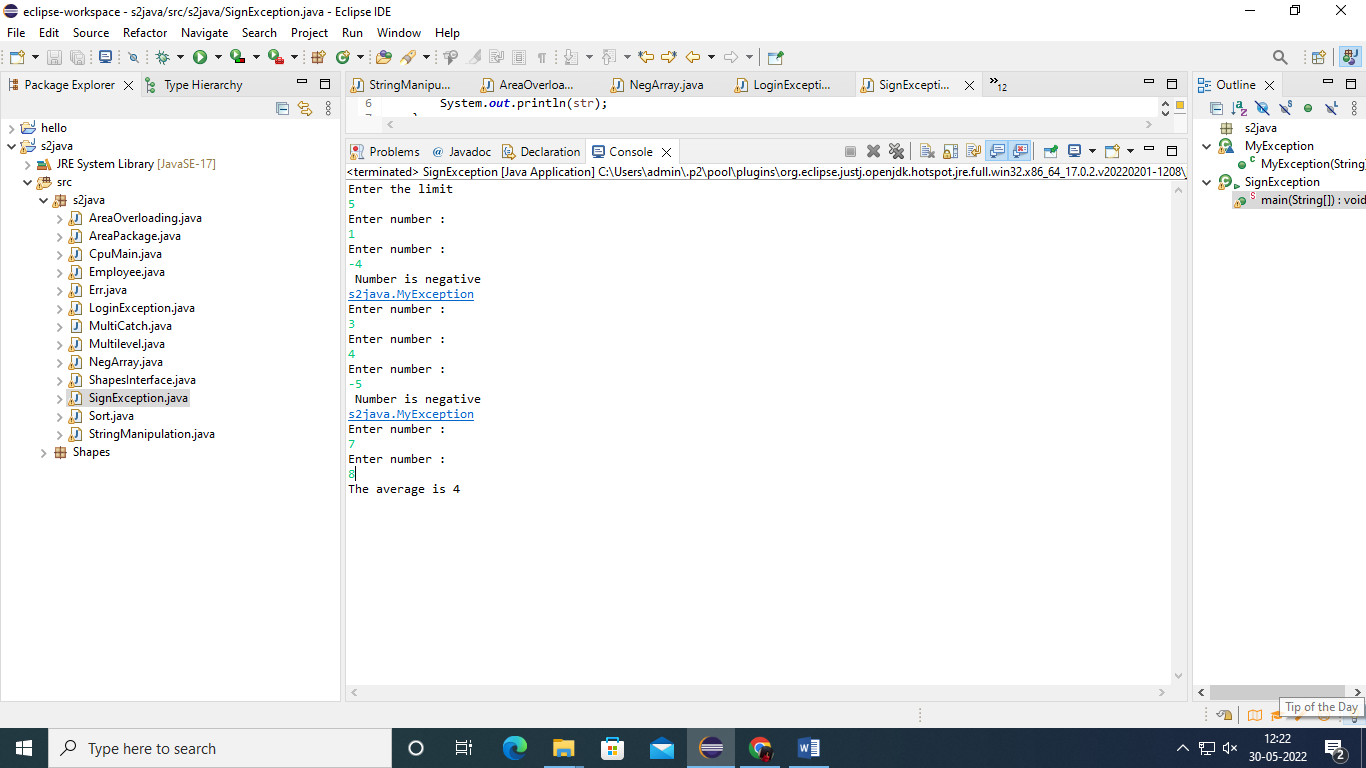
}

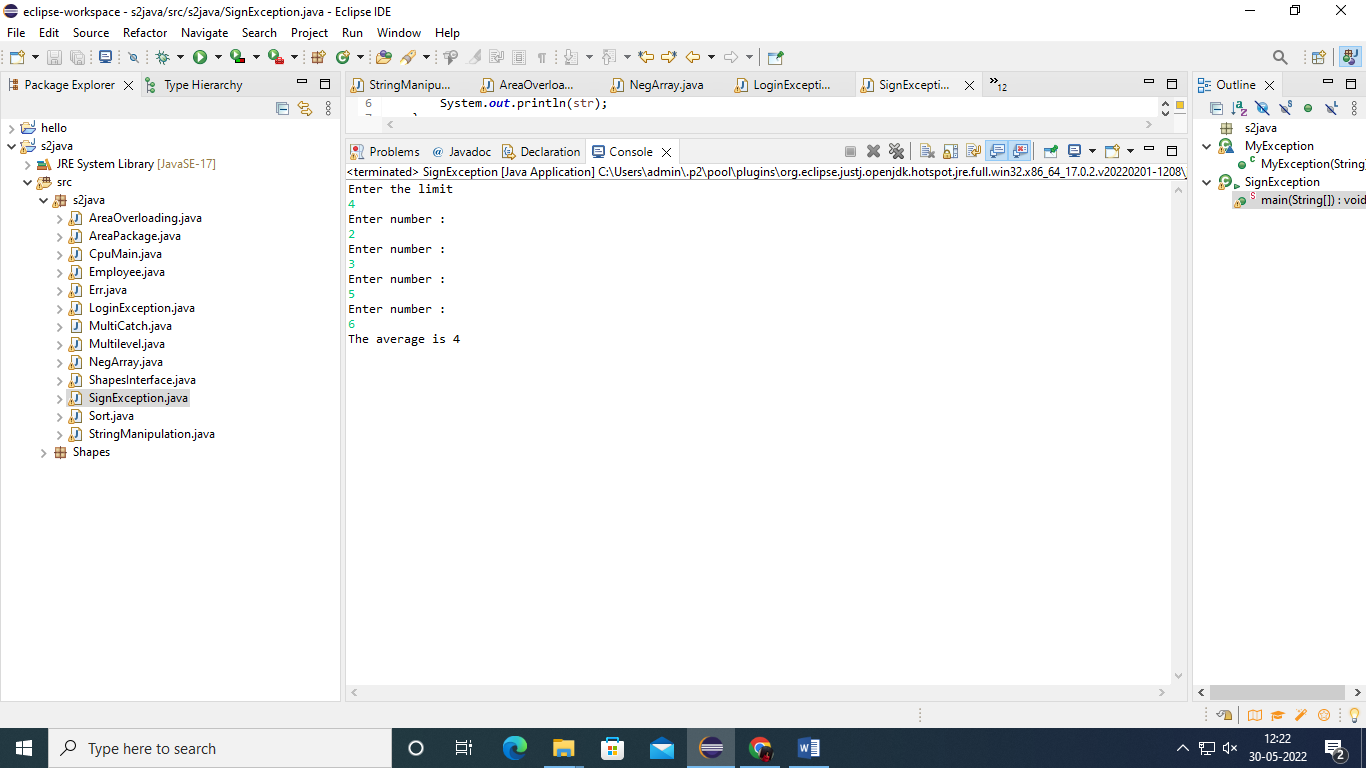
System.out.println("The average is " +sum/k);

}

}

**Output**





**Program 14**

**Aim :** Program to find maximum of three numbers using AWT.

**Source Code**

package s2java;

import java.awt.\*;

import java.awt.event.\*;

public class MaxAwt extends Frame {

TextField t1;

TextField t2;

TextField t3;

TextField t4;

Label l1,l2,l3,l4;

Button b1,b2;

MaxAwt(){

setTitle("MAXIMUM");

t1=new TextField();

l1=new Label(" Enter 1st No ");

l1.setBounds(100,45,145,20);

t1.setBounds(100,75,145,20);

t2=new TextField();

l2=new Label(" Enter 2nd No ");

l2.setBounds(100,110,145,20);

t2.setBounds(100,145,145,20);

t3=new TextField();

l3=new Label(" Enter 3rd No ");

l3.setBounds(100,170,145,20);

t3.setBounds(100,200,145,20);

t4=new TextField();

l4=new Label(" ");

l4.setBounds(100,240,145,20);

t4.setBounds(100,250,135,20);

b1=new Button("FIND");

b1.setBounds(105,278,70,40);

b2=new Button("EXIT");

b2.setBounds(195,278,70,40);

add(b1);

add(b2);

add(l1);

add(t1);

add(l2);

add(t2);

add(l3);

add(t3);

add(l4);

add(t4);

setSize(100,100);

setVisible(true);

b1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

int a=Integer.*parseInt*(t1.getText());

int b=Integer.*parseInt*(t2.getText());

int c=Integer.*parseInt*(t3.getText());

if(a>b && a>c) {

l4.setText("MAXIMUM = "+String.*valueOf*(a));

}

else if(b>c) {

l4.setText("MAXIMUM = "+String.*valueOf*(b));

}

else {

l4.setText("MAXIMUM = "+String.*valueOf*(c));

}

}

});

b2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

System.*exit*(0);;

}

});

}

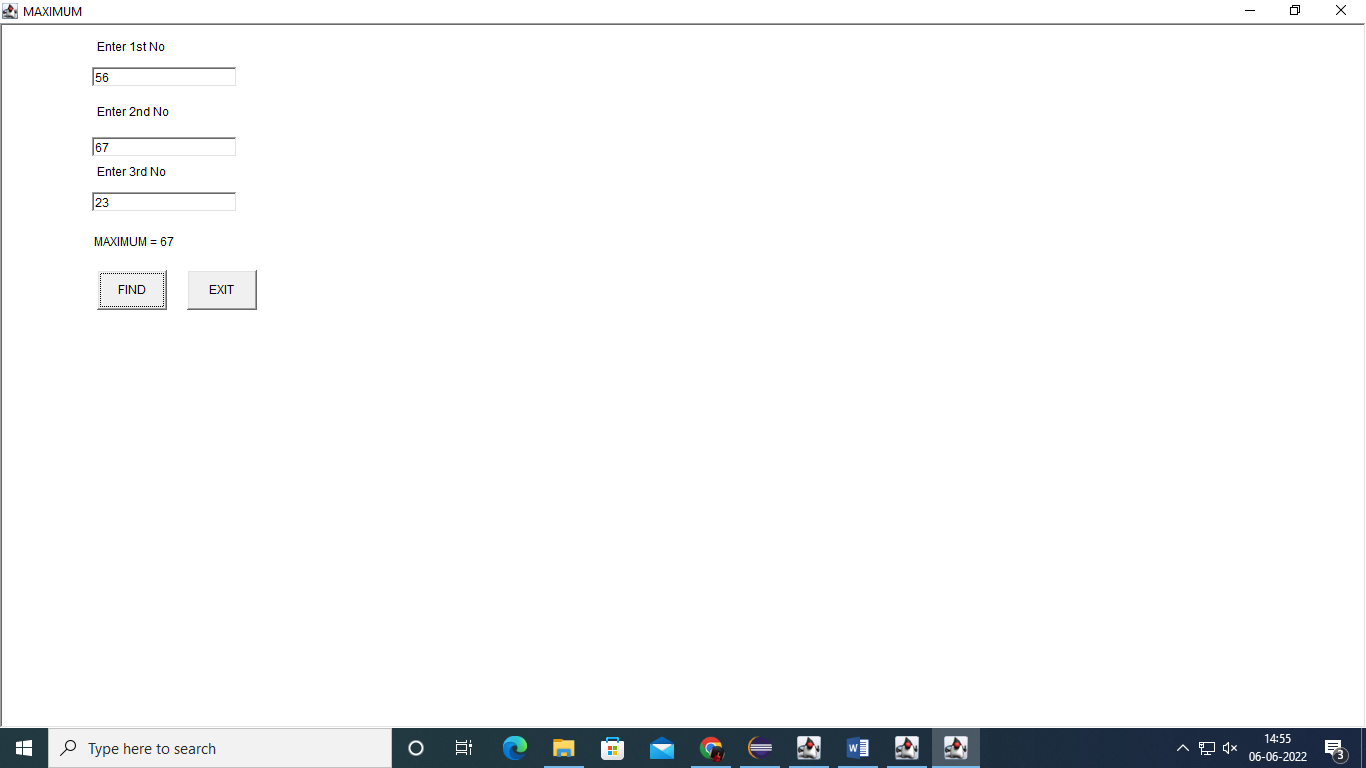
public static void main(String args[]) {

new MaxAwt();

}

}

**Output**



**Program 15**

**Aim :** Implement a simple calculator using AWT components.

**Source Code**

package s2java;

import java.awt.\*;

import java.awt.event.\*;

public class Calculator extends Frame {

TextField t1,t2,t3;

Label l1,l2,l3;

Button b1,b2,b3,b4,b5,b6;

Calculator(){

setTitle("CALCULATOR");

t1=new TextField();

l1=new Label("Enter 1st No");

l1.setBounds(100,45,145,20);

t1.setBounds(100,75,145,20);

t2=new TextField();

l2=new Label("Enter 2nd No");

l2.setBounds(100,110,145,20);

t2.setBounds(100,145,145,20);

t3=new TextField();

l3=new Label("Result");

l3.setBounds(100,169,145,20);

t3.setBounds(100,185,145,20);

b1=new Button("+");

b1.setBounds(65,200,80,40);

b2=new Button("-");

b2.setBounds(155,200,80,40);

b3=new Button("\*");

b3.setBounds(245,200,80,40);

b4=new Button("/");

b4.setBounds(95,250,80,40);

b5=new Button("%");

b5.setBounds(195,250,80,40);

b6=new Button("EXIT");

b6.setBounds(140,310,70,40);

add(b1);

add(b2);

add(b3);

add(b4);

add(b5);

add(b6);

add(l1);

add(t1);

add(l2);

add(t2);

add(l3);

add(t3);

setSize(400,400);

setVisible(true);

b1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

int a=Integer.*parseInt*(t1.getText());

int b=Integer.*parseInt*(t2.getText());

int c=a+b;

l3.setText(String.*valueOf*(a)+"+"+String.*valueOf*(b)+"="+String.*valueOf*(c));

}

});

b2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

int a=Integer.*parseInt*(t1.getText());

int b=Integer.*parseInt*(t2.getText());

int c=a-b;

l3.setText(String.*valueOf*(a)+"-"+String.*valueOf*(b)+"="+String.*valueOf*(c));

}

});

b3.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

int a=Integer.*parseInt*(t1.getText());

int b=Integer.*parseInt*(t2.getText());

int c=a\*b;

l3.setText(String.*valueOf*(a)+"\*"+String.*valueOf*(b)+"="+String.*valueOf*(c));

}

});

b4.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

int a=Integer.*parseInt*(t1.getText());

int b=Integer.*parseInt*(t2.getText());

int c=a/b;

l3.setText(String.*valueOf*(a)+"/"+String.*valueOf*(b)+"="+String.*valueOf*(c));

}

});

b5.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

int a=Integer.*parseInt*(t1.getText());

int b=Integer.*parseInt*(t2.getText());

int c=a%b;

l3.setText(String.*valueOf*(a)+"%"+String.*valueOf*(b)+"="+String.*valueOf*(c));

}

});

b6.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

System.*exit*(0);

}

});

}

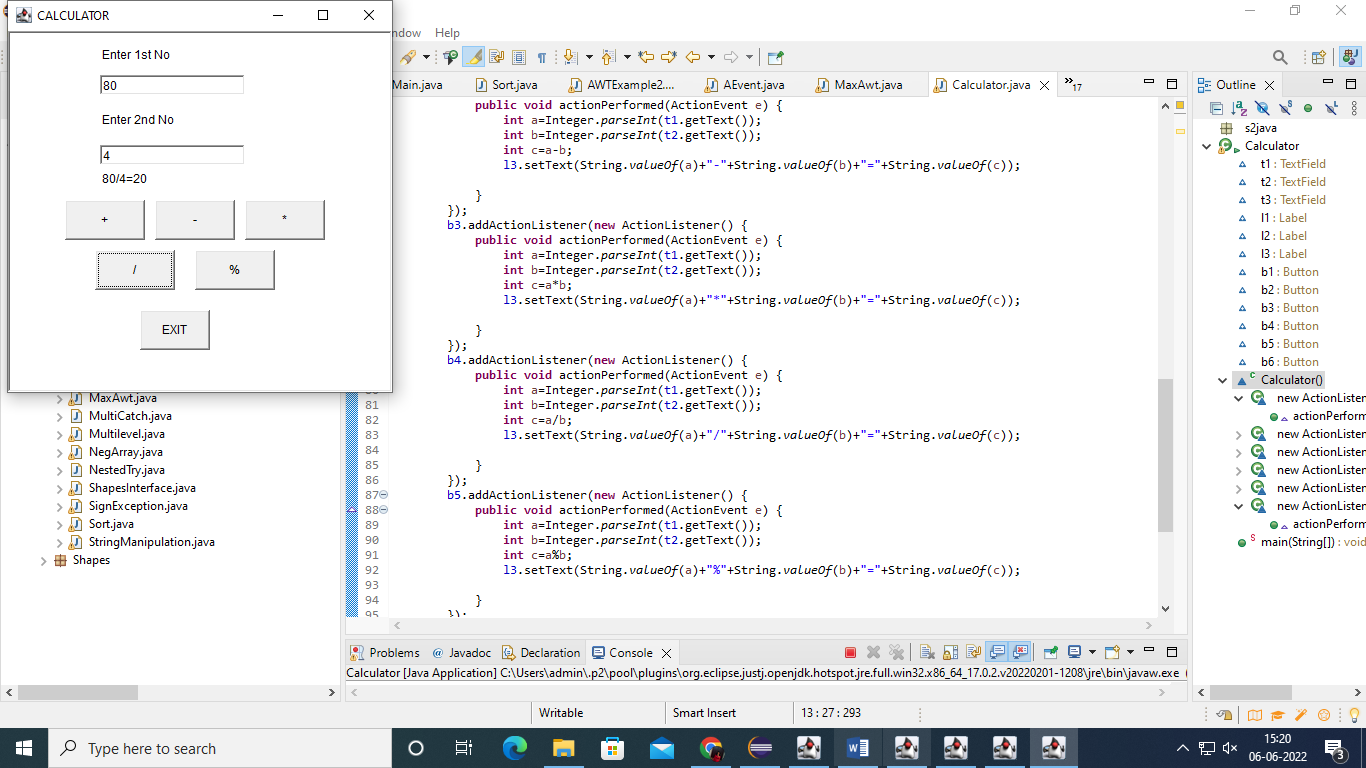
public static void main(String[] args) {

new Calculator();

}

}

**Output**



**Progarm : 16**

**Aim :** Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface).

**Source Code**

package s2java;

class FiboThread implements Runnable{

public void run() {

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

System.*out*.println("Fibbonacci Thread : "+a);

System.*out*.println("Fibbonacci Thread : "+a);

for(int h=1;h<=7;h++) {

c=a+b;

System.*out*.println("Fibbonacci Thread : "+c);

a=b;

b=c;

}

}

}

class EvenRangeThread implements Runnable{

public void run() {

int a=2,b=10;

for(int k=a;k<=b;k+=2) {

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

}

}

}

public class FiboEven {

public static void main(String[] args) {

FiboThread ft=new FiboThread();

EvenRangeThread et=new EvenRangeThread();

Thread t1=new Thread(ft);

Thread t2=new Thread(et);

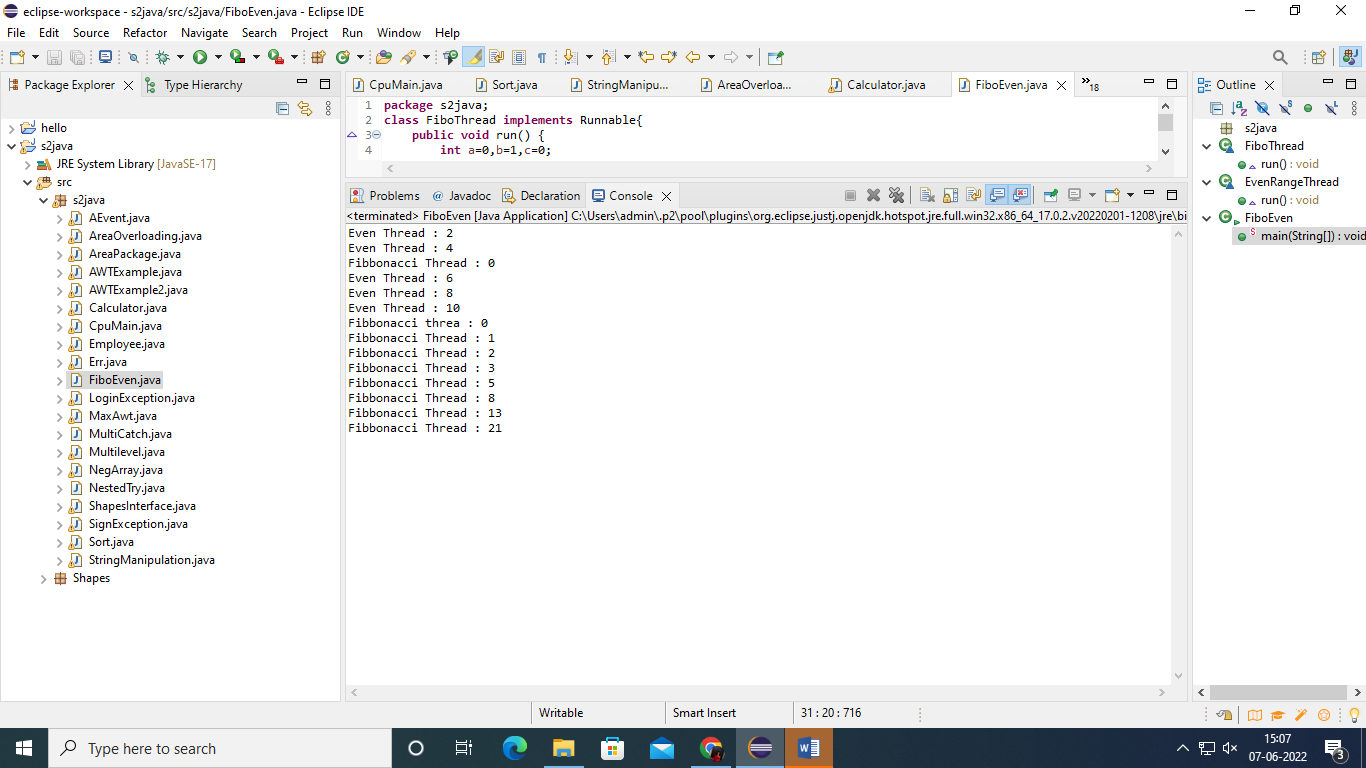
t1.start();

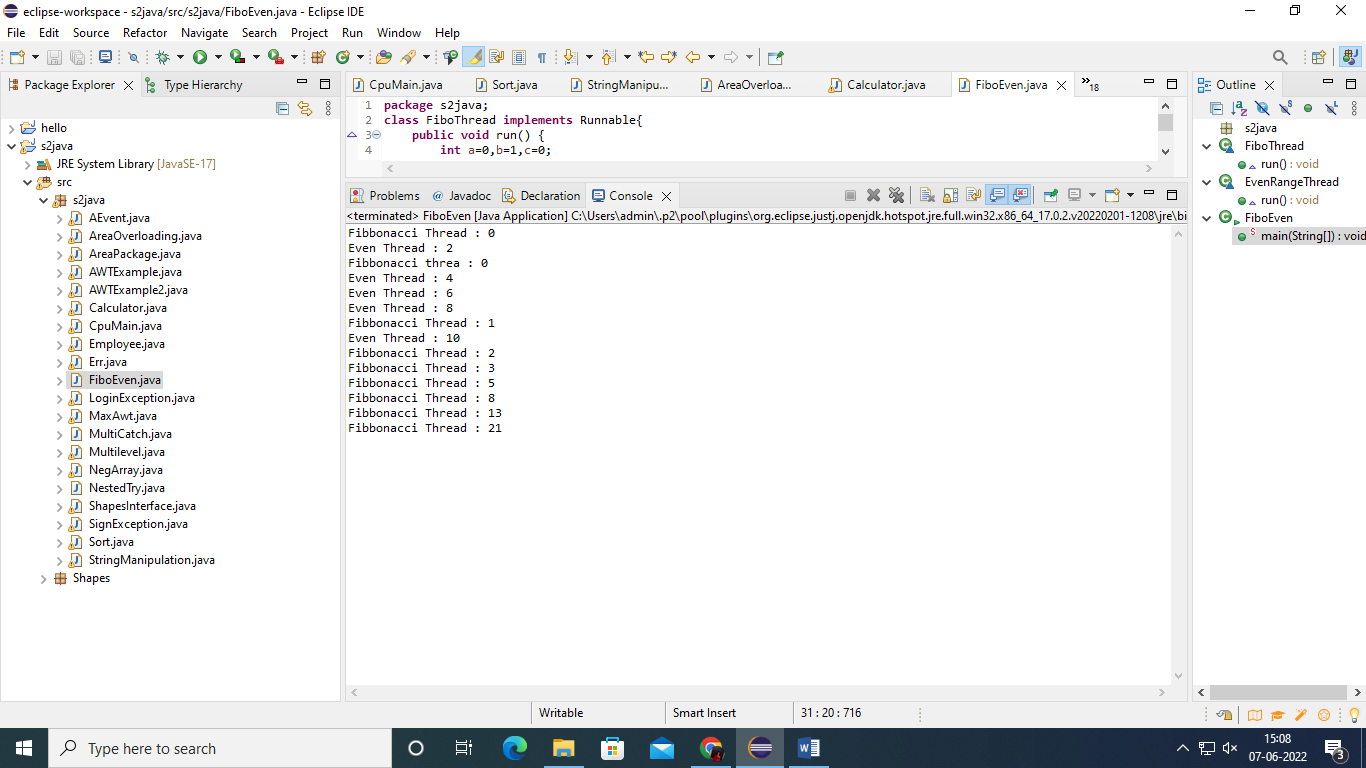
t2.start();

}

}

**Output**





**Program : 17**

**Aim** : Maintain a list of Strings using ArrayList from collection framework, perform built-in

operations.

**Source Code**

package lab;

import java.util.\*;

public class ArrayListExp {

public static void main ( String args[ ] ) {

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

list.add(" Tom" ) ;

list.add("Dick" ) ;

list.add("Harry") ;

list.add(" John" ) ;

list.add(" Henry" ) ;

System.out.println(list) ;

System.out.println(" Returning element :"+list.get( 2 )) ;

list.set( 2 ," Potter " ) ;

System.out.println ( " List after insertion of : newly inserted " ) ;

for ( String word : list )

System.out.println( word ) ;

Collections.sort(list) ;

System.out.println(" \nSorted list : " ) ;

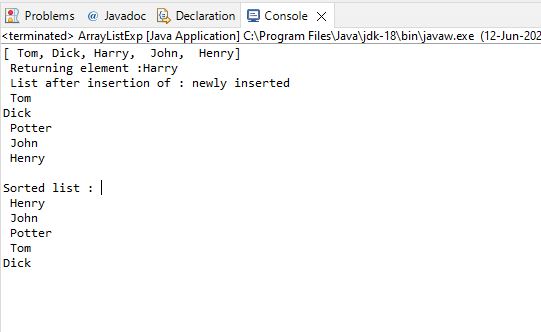
for(String word :list )

System.out.println( word ) ;

}

}

**Output**



**Program : 18**

**Aim:** Program to demonstrate the creation of queue object using the PriorityQueue class.

**Source Code**

package lab;

import java.util.PriorityQueue ;

import java.util.Queue ;

public class QPriorityQ {

public static void main(String args[ ] )

{

Queue<String> pq = new PriorityQueue <>();

pq.add ("Welcome" ) ;

pq.add( " have " ) ;

pq.add( " your " ) ;

pq.add ( " seat" ) ;

System.out.println( " Original Queue : " + pq ) ;

pq.remove( " your " ) ;

System.out.println( " After Remove " + pq ) ;

System.out.println(" Poll Method " + pq.poll( ) ) ;

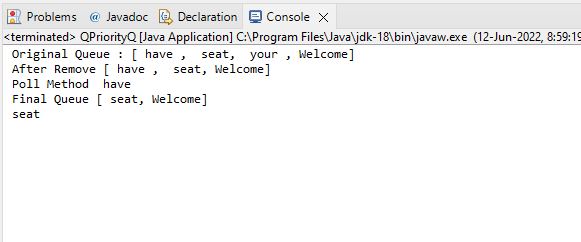
System.out.println( " Final Queue " + pq ) ;

System.out.println( pq.peek( ) ) ;

}

}

**Output**

****

**Program : 19**

**Aim :** Program to demonstrate the working of Map interface by adding, changing and removing elements.

**Source Code**

package lab;

import java.awt.\* ;

import java.util.\* ;

public class IllustrationMap {

public static void main ( String[ ] args )

{

Map<Integer , Integer > map= new HashMap<>();

map.put(1,10);

map.put(2 , 30 ) ;

map.put (3 ,20 ) ;

System.out.println ( "Original Map: " ) ;

for (Map.Entry<Integer , Integer > e : map.entrySet( ) )

System.out.println(e.getKey( ) +" "+ e.getValue ( ) ) ;

map.remove(2 ) ;

System.out.println("Map after removing 2: " ) ;

for (Map.Entry<Integer , Integer > e : map.entrySet ( ) )

System.out.println(e.getKey( ) +" "+e.getValue( ) ) ;

map.replace(3 , 500 ) ;

System.out.println("Map after changing value of 3 : " ) ;

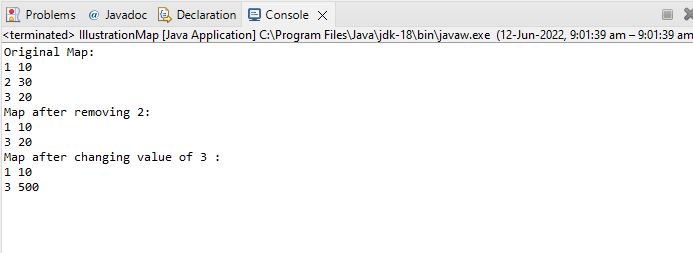
for (Map.Entry<Integer , Integer > e : map.entrySet ( ) )

System.out.println(e.getKey( ) + " "+ e.getValue ( ) ) ;

}

}

**Output**

****

**Program : 20**

**Aim :** Write a program to write to a file, then read from the file and display the contents on

the console.

**Source Code**

package lab;

import java.io.\* ;

class Console

{

public static void main(String[ ] args ) {

try {

FileWriter fw = new FileWriter(" abc.txt" ) ;

fw.write(" Once upon a time......");

fw.close( ) ;

System.out.println("\n File written ") ;

char c ;

FileReader fr=new FileReader(" abc.txt" ) ;

BufferedReader in=new BufferedReader( fr ) ;

String line=in.readLine( ) ;

while ( line != null )

{

System.out.println(line) ;

line=in.readLine( ) ;

}

in.close( ) ;

fr.close( ) ;

}

catch ( IOException e ) {

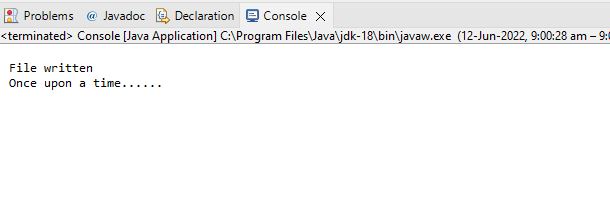
System.out.println(" There a r e some IOException " ) ;

}

}

}

**Output**

****