1.Find the second highest:

public class secondMax {

public static double findmax(double[] Cgparry) {

double highest = Double.*MIN\_VALUE*;

for (double Cgp: Cgparry ) {

if (Cgp > highest) {

highest = Cgp;

}

}

return highest;

}

public static double findsecondmax(double[] Cgparry) {

double secHighest = Double.*MIN\_VALUE*;

double highest = Double.*MIN\_VALUE*;

for ( double cgp : Cgparry) {

if (cgp > secHighest) {

secHighest=highest;

highest = cgp ;

}

}

return highest;

}

public static void main(String[] args) {

double[] Cgparry = {3.50, 3.52, 3.43, 3.63, 3.48, 3.32, 3.30, 3.60, 3.86, 3.75};

double higest = *findmax*(Cgparry);

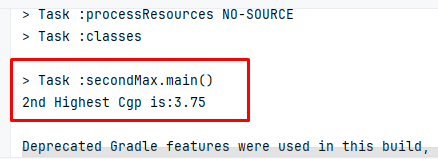
double SecHighest = *findsecondmax*(Cgparry);

*//System.out.println("Max Cgp is:" +higest);*

System.*out*.println("2nd Highest Cgp is:" +SecHighest);

}

}



2. Take a CGPA as user input. Now from the given array find if your input CGPA is present using binary search algorithm ::

import java.util.Scanner;

public class SearchCgp {

public static void main(String[] args) {

double[] Cgparry = {3.50, 3.52, 3.43, 3.63, 3.48, 3.32, 3.30, 3.60, 3.86, 3.75};

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

System.*out*.print("Enter CGPA to search: ");

double targetCGPA = scanner.nextDouble();

*search*(Cgparry, targetCGPA);

}

public static void search(double []Cgparry , double cgp){

boolean status =false;

for (double i = 0.00; i<Cgparry.length; i++){

if(Cgparry[(int) i] == cgp ){

status = true;

break;

}

}

if(status){

System.*out*.println("found");

}

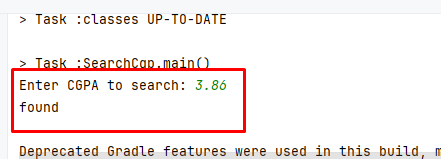
else {

System.*out*.println("Not Found");

}

}

}



3. Write a program that will break down the amount and count notes for any given amount.

import java.util.Scanner;

public class BreakNote {

public static void main(String[] args) {

int[] notes = {1000, 500, 200, 100, 50, 20, 10, 5, 2, 1};

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

System.*out*.print("Enter the amount: ");

int amount = scanner.nextInt();

*breakeNote*(amount, notes);

}

public static void breakeNote( int amount, int []notes) {

*// for(int note : notes){*

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

int note = notes[i];

int count = amount / note;

amount = amount % note;

if (count > 0) {

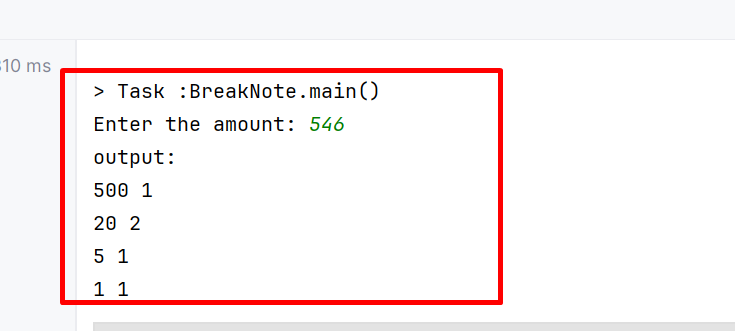
System.*out*.println(note + " " + count);

}

}

}

}



4.Take input as height of 10 babies in cm. Now find out the 2 lowest height of babies. (Don’t use Arrays.sort() method)

import java.util.Arrays;

import java.util.Scanner;

public class Secmin {

public static void main(String[] args) {

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

int[] babyHeights = new int[10];

System.*out*.println("Enter the heights of 10 babies in centimeters:");

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

System.*out*.print("Baby " + (i + 1) + ": ");

babyHeights[i] = scanner.nextInt();

}

*sortCGPA*( babyHeights);

System.*out*.println("Two lowest heights of babies:");

for(int i= 0 ; i<2;i++){

System.*out*.println("Baby " + (i + 1) + ": " + babyHeights[i] + " cm");

}

}

public static void sortCGPA( int [] babyHeights ) {

for(int i = 0 ; i<babyHeights.length -1;i++){

for(int j = 0; j<babyHeights.length -i -1;j++){

if(babyHeights[j] > babyHeights[ (j+1)]){

int temp= babyHeights[j];

babyHeights[ j]= babyHeights[ j+1];

babyHeights[j+1] = temp;

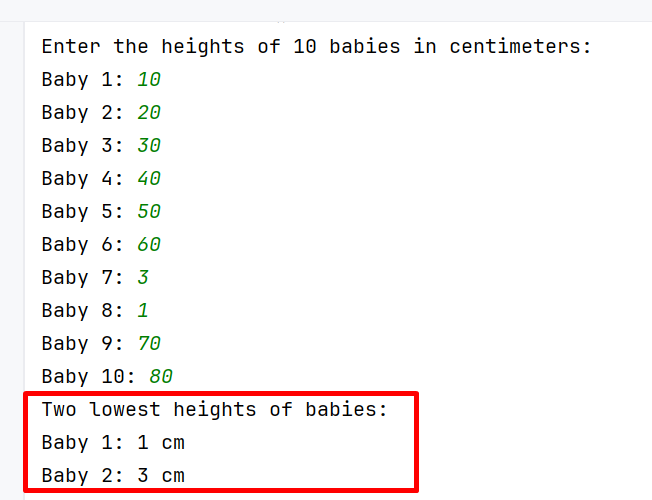
}

}

}

}

}



5. Sort the above scores from according to the order of highest CGPA (don’t use Arrays.sort() method, do it programmatically)

public class ArryS {

public static void main(String[] args) {

double[] Cgpa = {3.50, 3.52, 3.43, 3.63, 3.48, 3.32, 3.30, 3.60, 3.86, 3.75};

*sortCGPA*(Cgpa);

for (double i = 0; i<Cgpa.length;i++){

double Sort = Cgpa[(int) i];

System.*out*.println(Sort);

}

}

public static void sortCGPA(double[] Cgpa) {

for(double i = 0 ; i<Cgpa.length -1;i++){

for(double j = 0; j<Cgpa.length -i -1;j++){

if(Cgpa[(int)j] < Cgpa[(int) (j+1)]){

double temp= Cgpa[(int) j];

Cgpa[(int) j]= Cgpa[(int) j+1];

Cgpa[(int)j+1] = temp;

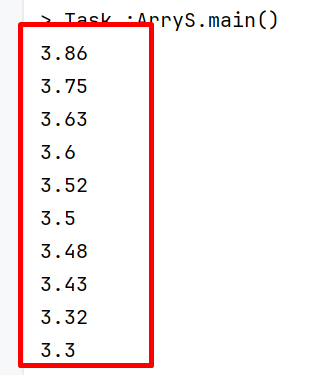
}

}

}

}

}



6.Peramid:

import java.util.Scanner;

public class Peramid {

public static void main(String[] args) {

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

Scanner scanner = new Scanner(System.in);

int user = scanner.nextInt();

System.out.println("Output : ");

for (int i = user; i >= 1; i--) {

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

System.out.print(j);

}

System.out.println();

}

for (int i = 2; i <= user; i++) {

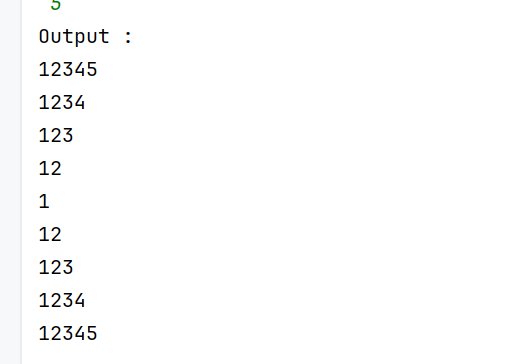
for (int j = 1; j <= i; j++) {

System.out.print(j);

}

System.out.println();

}}}



7. . Count number of words, number of characters without spaces, number of vowels and consonant from the given string:

Input: I am a SQA Engineer

import java.util.Scanner;

public class CountString {

public static void main(String[] args) {

System.*out*.println("Please Enter a sentece :");

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

String n= scanner.nextLine();

int wordcount= n.split("\\s+").length;

int countChar= n.replaceAll("\\s","").length();

int voiel= 0;

int consonent = 0 ;

for(char ch:n.toLowerCase().toCharArray()){

if(ch>='a'&& ch<='z' || ch>='A'&&ch<='Z'){

if(ch=='a'||ch=='e'||ch=='i'||ch=='o'||ch=='u'){

voiel++;

}

else {

consonent++;

}

}

}

System.*out*.println("Total word:"+ wordcount);

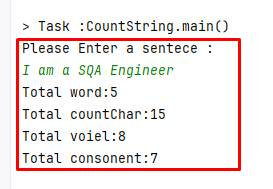
System.*out*.println("Total countChar:" + countChar);

System.*out*.println("Total voiel:"+ voiel);

System.*out*.println("Total consonent:"+ consonent);

}

}



8. Write a program that will take integer numbers as user input continuously and print the sum of numbers until user input q from the keyboard. When user input q, the program will be quit. If the user inputs another character, then the program will ask to input the number again

import java.util.Scanner;

public class Enterq {

public static void main(String[] args) {

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

int sum = 0;

System.*out*.println("Enter integer numbers. Type 'q' to quit.");

while (true) {

System.*out*.print("Enter a number: ");

if (scanner.hasNextInt()) {

int number = scanner.nextInt();

sum += number;

} else {

String input = scanner.next();

if (input.equals("q")) {

break;

} else {

System.*out*.println("Invalid input. Please enter a valid integer or 'q' to quit.");

}

}

}

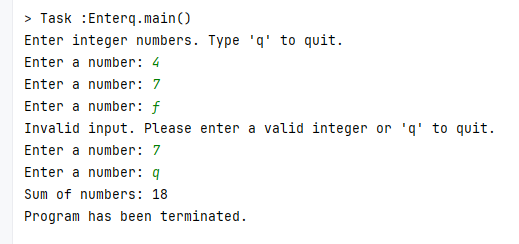
System.*out*.println("Sum of numbers: " + sum);

System.*out*.println("Program has been terminated.");

scanner.close();

}

}



9.A core i 7 laptop price is 85000 tk and a gaming mouse price is 2500 tk. If I buy the laptop and 1 piece mouse, what will be my total cost after giving 15% discount

import java.util.Scanner;

public class discountCalculate {

public static void main(String[] args) {

*// System.out.println("Enter Laptop Quantity : ");*

*// Scanner scanner = new Scanner(System.in);*

*// int laptopQuantiy = scanner.nextInt();*

*// System.out.println("Enter Mouse Quantity : ");*

*// Scanner scanner1 = new Scanner(System.in);*

*// int mouseQuantity = scanner.nextInt();*

double laptopprice = 85000;

double mouseprice = 2500;

int laptopquantity = 1;

int mousequantity = 1;

*// System.out.println("Enter Discount Amount : ");*

*// Scanner scanner2= new Scanner(System.in);*

*// double discount = scanner.nextDouble();*

double discountamount = 15.0;

double beforediscountprice = (laptopprice \* laptopquantity) + (mouseprice \* mousequantity);

double afterdiscountprice = (beforediscountprice \* discountamount) / 100.00;

double discountprice= (beforediscountprice-afterdiscountprice);

System.*out*.println("Total Laptop Price: " +laptopprice);

System.*out*.println("Total Mouse Price: " +mouseprice);

System.*out*.println("Total Laptop Quantity: " +laptopquantity);

System.*out*.println("Total Mouse Quantity: "+mousequantity);

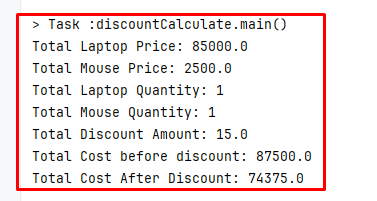
System.*out*.println("Total Discount Amount: " +discountamount);

System.*out*.println("Total Cost before discount: " +beforediscountprice);

System.*out*.println("Total Cost After Discount: "+discountprice);

}

}



10.Generate random 10 integer numbers in an array and print out all the numbers from the array and also print the max and min number from the array.

import java.util.Arrays;

import java.util.Random;

public class RandomNumber {

public static void main(String[] args) {

int[] numbers = new int[10];

Random random = new Random();

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

numbers[i] = random.nextInt(100);

}

System.*out*.println("Generated random numbers:");

System.*out*.println(Arrays.*toString*(numbers));

int max = *findMax*(numbers);

int min = *findMin*(numbers);

System.*out*.println("Maximum number: " + max);

System.*out*.println("Minimum number: " + min);

}

private static int findMax(int[] array) {

int max =0;

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

int num =array[i];

if (num > max) {

max = num;

}

}

return max;

}

private static int findMin(int[] array) {

int min = array[0];

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

int num = array[i];

if (num < min) {

min = num;

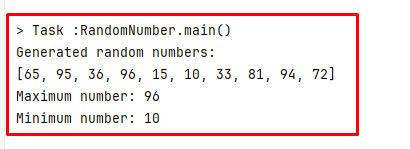
}

}

return min;

}

}



11.Print the numbers which are not duplicate from the given array

import java.util.Map;

public class NonDublicateNum {

public static void main(String[] args) {

int[] numbers = {1, 1, 2, 3, 1, 2, 4, 5, 6, 5, 4, 7};

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

boolean status = false;

for(int j = 0; j<numbers.length;j++){

if(i!=j && numbers[i]==numbers[j]){

status =true;

break;

}

}

if(!status){

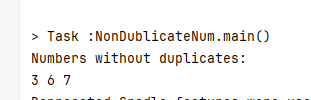
System.*out*.println(numbers[i]);

}

}

}

}



12.Find the number of occurrences of characters in a String

import java.util.Scanner;

public class Assignment9 {

public static void main(String[] args) {

System.*out*.println("Enter a Syntence: ");

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

String name= scanner.nextLine();

String lowerCase = name.toLowerCase();

int [] occurrences = new int[128];

char[] lowercasarrry= lowerCase.toCharArray();

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

if(Character.*isLetter*(lowercasarrry[i])){

occurrences [lowercasarrry[i]]++ ;

}

}

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

if(occurrences[i]>0){

int count= occurrences[i];

System.*out*.println((char) i+" "+count);

}

}

}

}

