q.WRITE A PROGRAM TO CHECK IF THE STRING IS PANAGRAM OR NOT.

class Pat{

public static void main(String [] args)

{

String s1="abcdefghIjklmnopqrstuvwxyz";

char x='A';

char y='a';

String x1=Character.toString(x);

String x2=Character.toString(y);

while(s1.contains(x1)||s1.contains(x2))

{

if(x=='Z'||y=='z')

{

break;

}

else{

x++;

y++;

x1=Character.toString(x);

x2=Character.toString(y);

}

}

if(x=='Z'||y=='z')

{

System.out.println("The string is panagram");

}

else{

System.out.println("the string is not panagram");

}

}

}

Q2.print the string to be inserted to make the inserted string panagram

import java.util.Scanner;

class Pat{

public static void main(String [] args)

{

Scanner s=new Scanner(System.in);

System.out.println("please enter the string");

String s1=s.nextLine();

char x='A';

char y='a';

String x1=Character.toString(x);

String x2=Character.toString(y);

String s2=" ";

while(x<='Z'){

if(s1.contains(x1)||s1.contains(x2))

{

x++;

y++;

x1=Character.toString(x);

x2=Character.toString(y);

}

else

{

s1=s1+x1;

s2=s2+x1;

}

}

System.out.println("the string to be inserted is"+s2);

}

}

Q3.check if the string is lipogram panagram or not.

import java.util.Scanner;

class Pat{

public static void main(String [] args)

{

int i=0;

Scanner s=new Scanner(System.in);

System.out.println("please enter the string");

String s1=s.nextLine();

char x='A';

char y='a';

String x1=Character.toString(x);

String x2=Character.toString(y);

String s2=" ";

while(x<='Z'){

if(s1.contains(x1)||s1.contains(x2))

{

x++;

y++;

x1=Character.toString(x);

x2=Character.toString(y);

}

else

{

s1=s1+x1;

i++;

}

}

if(i==1)

{

System.out.println("the string is panagram lipogram");

}

else if(i==0)

{

System.out.println("the string is only panagram");

}

}

}

Q4.remove all the Punct from the string

import java.util.Scanner;

class Pat{

public static void main(String [] args)

{

Scanner s=new Scanner(System.in);

System.out.println("please enter the string");

String s1=s.nextLine();

s1=s1.replaceAll("\\p{Punct}","");

System.out.println(s1);

}

}

Q6.arrange the string in such manner that the two same characters are not going to be in the adjacent places.

import java.util.Scanner;

class Pat{

public static void main(String [] args)

{

Scanner s=new Scanner(System.in);

System.out.println("please enter the string");

String s1=s.nextLine();

String s2=" ";

boolean x4=true;

char[] a=s1.toCharArray();

int l=0;

int n=s1.length();

char x=a[0];

char[] c=new char[4];

int y=0;

int r=0;

int h=0;

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

{

if(a[i]==x)

{

l++;

}

else if(a[i]!=x)

{

c[r]=a[i];

r++;

y++;

}

}

System.out.println(y);

System.out.println(l);

if(l-1>n-l)

{

System.out.println("arrangment not possible");

x4=false;

}

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

{

int x3=0;

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

{

if(a[j]==c[i])

{

x3++;

if(x3-1>n-x3)

{

System.out.println("arrangment not possible");

x4=false;

}

}

}

}

int g=0;

if(x4){

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

{

if(g<=n){

a[g]=a[0];

g=g+2;

}

}

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

{

if(i%2!=0)

{

a[i]=c[h];

h++;

}

}

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

{

System.out.print(a[i]);

}

}

}

}

Q7.check wether the string is the number or not?

import java.util.\*;

class Pat{

public static boolean isNumber(String str){

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

if(Character.isDigit(str.charAt(i))==false){

return false;

}

}

return true;

}

public static void main(String [] args)

{

Scanner s=new Scanner(System.in);

System.out.println("please enter the string");

String s1=s.nextLine();

if(isNumber(s1))

{

System.out.println("it is number");

}

else

{

System.out.println("it is not number");

}

}

}

Q10.output the first latter of the words in the string in capital.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

Scanner sc=new Scanner(System.in);

System.out.println("please enter the String");

String s1=sc.nextLine();

s1=s1.toUpperCase();

char[] g=s1.toCharArray();

int n=s1.length();

int i=0;

while(i<n)

{

int j=0;

while(g[i]!=' ')

{

if(j==0)

{

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

}

i++;

j++;

}

i++;

}

}

}

Q11.write the program to make the number round up to the nearest 100s place.

Q12.write the program to change the set of string.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

Scanner sc=new Scanner(System.in);

System.out.println("please enter the String");

String s1=sc.nextLine();

int n=s1.length();

char[] g=new char[n];

int j=0;

char x='a';

String x1=Character.toString(x);

String Str=" ";

while(j<n)

{

if(s1.contains(x1))

{

x++;

x1=Character.toString(x);

}

else

{

Str=Str+x1;

j++;

x++;

x1=Character.toString(x);

}

}

System.out.println(Str);

}

}

Q20.program to check frequency of all the elements.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

int[] arr={10,20,30,40,50,60,10,10,10};

int[] arr1={1,1,1,1,1,1,1,1,1};

int n=arr.length;

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

{

if(arr1[i]!=0){

int k=1;

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

{

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

{

k++;

arr1[j]=0;

}

}

System.out.println("the frequency of"+arr[i]+"is"+k);

}

}

}

}

Q21.program to left rotation.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

int[] arr={10,20,30,40,50,60,10,10,10};

int n=arr.length;

Scanner scan=new Scanner(System.in);

System.out.println("print the number of left rotations");

int k=scan.nextInt();

int[] temp=new int[2\*n];

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

{

temp[i]=arr[i];

temp[n+i]=arr[i];

}

int m=temp.length;

System.out.println(m);

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

{

int l=temp[i];

System.out.print(l+" ");

}

}

}

Q23.program for right rotation.

public static void main(String [] args)

{

int[] arr={10,20,30,40,50,60,10,10,10};

int n=arr.length;

Scanner scan=new Scanner(System.in);

System.out.println("print the number of left rotations");

int k=scan.nextInt();

int[] temp=new int[2\*n];

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

{

temp[i]=arr[i];

temp[n+i]=arr[i];

}

int m=temp.length;

System.out.println(m);

for(int i=n-k;i<m-k;i++)

{

int l=temp[i];

System.out.print(l+" ");

}

}

}

Q21.program for the multiplication of the two matrices.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

int [][] arr={{10,10,10},{20,30,40},{30,40,50}};

int[] [] arr1={{10,10,10},{20,40,50},{40,50,80}};

int[][] sum={{0,0,0},{0,0,0},{0,0,0}};

int n=arr.length;

int n1=arr.length;

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

{

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

{

for(int j=0;j<n1;j++)

{

sum[k][i]+=arr[k][j]\*arr1[j][i];

}

System.out.print(sum[k][i]+" ");

}

System.out.println();

}

}

}

Q24.write a program to check if the matrix is sparse or not.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

int[][] arr={{1,0,0},{0,1,0},{0,0,1}};

int n=arr.length;

int k=0;

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

{

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

{

if(i!=j)

{

if(arr[i][j]==0){

k++;

}

}

else if(i==j)

{

if(arr[i][j]!=0)

{

k++;

}

}

}

}

if(k==9)

{

System.out.println("this is identity matrix ");

}

}

}

Q25.print the upper triangular matrix.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

int[][] arr={{1,1,1},{1,1,1},{1,1,1}};

int n=arr.length;

int k=0;

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

{

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

{

if(j>=i)

{

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

}

else{

System.out.print(k+" ");

}

}

System.out.println();

}

}

}

Q.program to divide string in 2 parts.

import java.util.Scanner;

class Pat1{

public static void main(String [] args)

{

String str1="aaaabbbbcc";

int n=str1.length();

int c=n/2;

int k=0;

while(k<n)

{

for(int i=k;i<c+k;i++)

{

System.out.print(str1.charAt(i));

}

System.out.println();

k=k+c;

}

}

}

Q30.program for computation of the string.

import java.util.Scanner;

class Pat1{

public static String swap(String a,int i,int j) {

char[] b=a.toCharArray();

char temp;

temp=b[i];

b[i]=b[j];

b[j]=temp;

return String.valueOf(b);

}

public static void main(String [] args)

{

String str1="abc";

int n=str1.length();

Permutation(str1,0,n);

}

public static void Permutation(String a,int start,int end)

{

if(start==end-1)

{

System.out.println(a);

}

else

{

for(int i=start;i<end;i++)

{

a=swap(a,start,i);

Permutation(a,start+1,end);

a=swap(a,start,i);

}

}

}

}

Q34.program to find all the substrings of the string.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str1="abc";

int n=str1.length();

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

{

System.out.println(str1.substring(i));

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

System.out.println(str1.substring(i,j));

}

}

}

}

Q.to find the largest sequence in the string.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

int o=0;

String str1="abcgeabcge";

int n=str1.length();

char[] b=str1.toCharArray();

String f=" ";

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

{

int p=1;

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

{

if(str1.charAt(i)==str1.charAt(j))

{

if(j+p<n){

while( j+p<n&& str1.charAt(i+p)==str1.charAt(j+p))

{

p++;

}

}

}

}

if(o<p){

o=p;

f=str1.substring(i,i+p);

}

}

System.out.println(f);

}

}

Q35.program to put special character on the space.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str1="ABC abc";

int n=str1.length();

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

{

if(str1.charAt(i)==' ')

{

str1=str1.replace(' ','a');

}

System.out.print(str1.charAt(i));

}

}

}

Q50.to check if the string is palindrome or not.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str1="pap";

int n=str1.length();

int k=0;

String str2="abc";

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

{

if(str1.charAt(i)!=str1.charAt(n-1-i))

{

System.out.println("this string is not palindrome");

break;

}

else{

k++;

}

}

if(k==n)

{

System.out.println("it is finish");

}

}

}

Q60.to check if the second string is the rotation of the first string.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str1="pap";

String str2="app";

int n=str1.length();

int k=0;

if(str1.length()!=str2.length())

{

System.out.println("the second string is not the permutation of the first one");

}

str1=str1.concat(str1);

if(str1.contains(str2))

{

System.out.println(" the str2 is the rotation of the str1");

}

System.out.println(str1.indexOf(str2));

}

}

Q.find the max and minimum repeating character in java

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str1="pap";

int n=str1.length();

String max=" ";

String min=" ";

int max1=0;

int min1=1;

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

{

if(str1.charAt(i)!=' '){

int l=1;

int k=0;

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

{

if(str1.charAt(i)==str1.charAt(j)&& str1.charAt(i)!=' ')

{

if(i==0)

{

l++;

}

k++;

}

}

if(k>max1){

max1=k;

max=max.replace(max.charAt(0),str1.charAt(i));

}

min1=l-1;

if(k<=min1)

{

min1=k;

min=min.replace(min.charAt(0),str1.charAt(i));

}

str1=str1.replace(str1.charAt(i),' ');

}

}

System.out.println(min);

System.out.println(max);

}

}

Q70.find the program to reverse the array.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str2=" ";

String str1="pap";

int n=str1.length();

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

{

str2=str2+str1.charAt(n-1-i);

}

System.out.println(str2);

}

}

Q.program to print the duplicate character in the string.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str2=" ";

String str1="pap";

int n=str1.length();

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

{

str2=str2+str1.charAt(n-1-i);

}

System.out.println(str2);

}

}

71.program to find duplicates in the string.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str2=" ";

String str1="pap";

int n=str1.length();

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

{

str2=str2+str1.charAt(n-1-i);

}

System.out.println(str2);

}

}

Q45.program to find the duplicate words in the java.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str1="pap bab pap";

String[] words=str1.split(" ");

int n=words.length;

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

{

int k=1;

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

{

if(words[i].equals(words[j])){

k++;

words[j]=" ";

}

}

if(k>1&& words[i]!=" ")

{

System.out.println("the"+words[i] +" is repeated "+k+"times");

}

}

}

}

Q.to find the frequency of the individual characters.

public static void main(String[] args)

{

String str1="pap bab pap";

String[] words=str1.split(" ");

int n=str1.length();

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

{

int k=1;

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

{

if(str1.charAt(i)==str1.charAt(j)){

k++;

}

}

if(k>1&& str1.charAt(i)!=' ')

{

System.out.println("the"+str1.charAt(i) +" is repeated "+k+"times");

}

str1=str1.replace(str1.charAt(i),' ');

}

}

}

Q50.program to remove the white spaces.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

String str1="abcgeabcge is";

int n=str1.length();

String y=" ";

char[] b=str1.toCharArray();

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

{

if(str1.charAt(i)!=' ')

{

System.out.print(str1.charAt(i));

}

}

}

}

Q40.to convert lower case to uppercase.

import java.util.Scanner;

class Pat1{

public static void main(String[] args)

{

fcString str1="ABC abc";

StringBuffer sb=new StringBuffer(str1);

int n=str1.length();

String y=" ";

char[] b=str1.toCharArray();

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

{

if(Character.isLowerCase(str1.charAt(i)))

{

sb.setCharAt(i,Character.toUpperCase(str1.charAt(i)));

}

if(Character.isUpperCase(str1.charAt(i)))

{

sb.setCharAt(i,Character.toLowerCase(str1.charAt(i)));

}

}

System.out.println(sb);

}

}

Q71.program to right rotate the array with the minimum time complexity.

public class Solution {

// Complete the rotLeft function below.

static int[] rotLeft(int[] a, int d) {

int y=0;

int j=a.length;

int[] b=new int[j];

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

b[i]=a[i];

}

y=d%j;

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

a[i-y]=b[i];

}

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

a[i+j-y]=b[i];

}

return a;

}