**Strings in Java**

**Assignments Questions**

**1. WAP(Write a Program) to remove Duplicates from a String.(Take any String example with duplicates character)**

public class stringAssignments3{

    public static void main(String []args){

        String s1="This is the surprise every one was waiting for";

        String s2="";

        char c[]=s1.toCharArray();

        for(int i =0;i<c.length;i++){

            int j;

            for(j=0;j<i;j++){

                if(c[i]==c[j]) break;

            }

            if(i==j){

                s2=s2+c[i];

            }

        }

        System.out.println(s2);

    }

}

**2. WAP to print Duplicates characters from the String**

public class stringAssignments3{

    public static void main(String []args){

        String s1= "aaabbcddeffffghhhhiiiij";

        // String s2="";

        char a[]=s1.toCharArray();

        for(int i=0;i<a.length;i++){

            int count=1;

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

                if(a[i]==a[j]){

                    count++;

                    a[j]='0';

                }

            }

            if(count>1 && a[i]!='0') System.out.println(a[i]);

        }

    }

}

**3. WAP to check if "2552" is palindrome or not.**

public class stringAssignments3{

    public static void main(String []args){

        String s1="2552";

        String s2="";

        for(int i=s1.length()-1;i>=0;i--){

            s2=s2+s1.charAt(i);

        }

        if(s1.equals(s2)) System.out.println("Palindrome");

        else System.out.println("Not Palindrome");

    }

}

**4. WAP to count the number of consonants, vowels, special characters in a String.**

public class stringAssignments3{

    public static void main(String []args){

        String s1="PW Skills is Just Awesome @#%$@$@";

        s1=s1.toLowerCase();

        int vowel=0,consonant=0,special=0;

        for(int i=0;i<s1.length();i++){

            char c=s1.charAt(i);

            if(c=='a' || c=='e' || c=='i' ||c=='o' ||c =='u') vowel++;

            else if(c>='a' && c<='z') consonant++;

            else{

                if(c!=' ') special++;

            }

        }

        System.out.println("Vowel = " +vowel);

        System.out.println("Consonant = " +consonant);

        System.out.println("Special Character = " +special);

    }

}

**5. WAP to implement Anagram Checking least inbuilt methods being used.**

import java.util.Arrays;

public class anagramP {

    public static void main(String[] args){

        String S1="School Master";

        String S2="The Classroom Java";

        S1=S1.replace(" ", "");

        S2=S2.replace(" ","");

        S1=S1.toLowerCase();

        S2=S2.toLowerCase();

        char[] a=S1.toCharArray();

        char[] b=S2.toCharArray();

        Arrays.sort(a);

        Arrays.sort(b);

        if(Arrays.equals(a, b)) System.out.println("ANAGRAM");

        else System.out.println("NOT ANAGRAM");

    }

}

**6. WAP to implement Pangram Checking with least inbuilt methods being used.**

public class pangramP{

    public static void main(String[] args){

        String s1=" The Quick brown fox jumps over the lazy dog";

        s1=s1.replace(" ", "");

        s1=s1.toUpperCase();

        char []a=s1.toCharArray();

        int []b=new int[26];

        for(int i=0;i<a.length;i++){

            b[a[i]-65]++;

        }

        boolean flag=true;

        for(int i=0;i<26;i++){

            if(b[i]==0){

                // int n=i+65;

                char temp=(char)(i+65);

                System.out.println(temp + " is not present");

                // break;

                flag=false;

            }

        }

        if(flag==true) System.out.println("PANGRAM");

        else System.out.println("NOT PANGRAM");

    }

}

**7. WAP to find if String contains all unique characters.**

public class stringAssignments3{

    public static void main(String []args){

                String s1= "abcdefghijklmnopqrstuvwxyaz";

                boolean flag=true;

                // String s2="";

                char a[]=s1.toCharArray();

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

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

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

                            flag=false;

                            break;

                        }

                    }

                }

                if(flag==true) System.out.println("All Unique");

                    else System.out.println("Some Repetition");

            }

        }

**8. WAP to find the maximum occurring character in a String**

public class stringAssignments3{

    public static void main(String []args){

        String s1= "aaaBBBBBBBcccccccccccccc$$$$$$$$$ccccccccc$$$$$$$$$$$cdef";

        char c[]=s1.toCharArray();

        int max=0;

        char maxm='0';

        for(int i=0;i<s1.length();i++){

            int count=1;

            for(int j=i+1;j<s1.length();j++){

                if(c[i]==c[j]) count++;

            }

            if(count>max)

            {

                max=count;

                maxm=c[i];

            }

        }

        System.out.println(maxm +" occured "+max+ " times");

    }

}

**Output :**

****