**Assignment - 13**

**Strings In Java(Assignment - 3)**

**1. Write a program to remove duplicates from a String?**

Ans:

public class Q1 {

public static void main(String[] args) {

String s1 = "Anurag Verma";

s1 = s1.replace(" ", "");

s1 = s1.toLowerCase();

String s2 = "";

char[] arr1 = s1.toCharArray();

char[] arr2 = new char[26];

for (int i = 0; i <= arr1.length - 1; i++) {

if (arr2[arr1[i] - 97] == 0) {

arr2[arr1[i] - 97]++;

s2 = s2 + arr1[i];

}

}

System.*out*.println("String With Duplicate Characters: "+s1);

System.*out*.println("String Without Duplicate Characters: "+s2);

}

}

**2. Write a program to print duplicate characters?**

Ans:

public class Q2 {

public static void main(String[] args) {

String s1 = "Anuragg Vermaaaaaaaa";

s1 = s1.replace(" ", "").toLowerCase();

char[] s = s1.toCharArray();

char[] s2 = new char[26];

int count;

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

s2[s[i] - 97]++;

if(s2[s[i]-97] == 2){

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

}

}

}

}

**3. Write a program to check whether the given String is Palindrome or not?**

Ans:

import java.util.Scanner;

public class Q3 {

public static void main(String [] args){

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

System.*out*.println("Enter A String:");

String s1 = sc.nextLine();

s1 = s1.toLowerCase();

String [] arr1 = s1.split("");

String s2 = "";

for(int i = arr1.length-1; i >= 0; i--){

s2 = s2 + arr1[i];

}

if(s1.equals(s2)){

System.*out*.println("The Given String Is Palindrome:");

}else{

System.*out*.println("The Given String Is Not A Palindrome:");

}

}

}

**4. Write a program to count the number of consonants, vowels and Special characters in a String?**

Ans:

import java.util.Scanner;

public class Q4 {

public static void main(String [] args){

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

System.*out*.println("Enter A String:");

String s = sc.nextLine();

s = s.toLowerCase();

s = s.replace(" ", "");

char [] arr = s.toCharArray();

int vowels = 0;

int consonants = 0;

int digits = 0;

int specialChars = 0;

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

if(arr[i] == 'a' || arr[i] == 'e' || arr[i] == 'i' || arr[i] == 'o' || arr[i] == 'u'){

vowels++;

}else if(arr[i] >= 'a' && arr[i] <= 'z' ) {

consonants++;

}else{

specialChars++;

}

}

System.*out*.println("The Number Of Vowels: "+vowels);

System.*out*.println("The Number Of Consonants: "+consonants);

System.*out*.println("The Number Of Digits and Special Characters: "+specialChars);

}

}

**5. Write a program to implement anagram checking with least inbuilt methods being used?**

Ans:

import java.util.Arrays;

import java.util.Scanner;

public class Q5 {

public static void main(String [] args) {

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

System.*out*.println("Enter String1:");

String s1 = sc.nextLine();

System.*out*.println("Enter String2:");

String s2 = sc.nextLine();

s1 = s1.replace(" ", "");

s2 = s2.replace(" ", "");

char[] arr1 = s1.toLowerCase().toCharArray();

Arrays.*sort*(arr1);

char []arr2 = s2.toLowerCase().toCharArray();

Arrays.*sort*(arr2);

if(Arrays.*equals*(arr1,arr2)){

System.*out*.println("It Is A Anagram");

}else{

System.*out*.println("It Is Not A Anagram");

}

}

}

**6. Write a program to implement Pangram checking with least Inbuilt methods?**

Ans:

public class Q6 {

public static void main(String [] args){

boolean flag = false;

String s = "THE QUICK BROWN FOX JUMPS OVER LAZY DOG";

s = s.replace(" ", "");

char[] arr = s.toCharArray();

int[] sarr = new int[26];

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

sarr[arr[i] - 65]++;

}

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

if (sarr[i] == 0) {

System.*out*.println("Not A Pangram");

flag = true;

}

}

if (flag == false) {

System.*out*.println("A Pangram");

}

}

}

**7. Write a program to find if String contains all Unique Characters?**

Ans:

import java.util.Scanner;

public class Q7 {

public static void main(String [] args){

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

System.*out*.println("Enter A String: ");

String s = sc.nextLine();

char []arr = s.toLowerCase().toCharArray();

int [] arr2 = new int[26];

int count = 0;

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

if(arr2[arr[i]-97] == 0) {

arr2[arr[i] - 97]++;

}

else{

count++;

}

}

if(count < 1) {

System.*out*.println("String Contains Unique Characters:");

}else{

System.*out*.println("String Doesn't Contain Unique Characters:");

}

}

}

**8. Write a program to find the maximum occurring character in a string?**

Ans:

import java.util.Arrays;

public class Q8 {

public static void main(String[] args) {

String s = "anurag verrrma";

char[] sarr = s.toCharArray();

int[] arr = new int[sarr.length];

char minChar = s.charAt(0);

char maxChar = s.charAt(0);

int i, j, min, max;

//Count each word in given string and store in array freq

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

arr[i] = 1;

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

if (sarr[i] == sarr[j] && sarr[i] != ' ' && sarr[i] != '0') {

arr[i]++;

//Set string[j] to 0 to avoid printing visited character

sarr[j] = '0';

}

}

}

max = arr[0];

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

if (max < arr[i]) {

max = arr[i];

maxChar = sarr[i];

}

}

System.*out*.println("Maximum occurring character: " + maxChar);

}

}