**Fourth Java Assignment Answers**

**1.WAP to remove Duplicates from a String.(Take any String ex with duplicates**

**character)**

//WAP to remove Duplicates from a String.(Take any String example with duplicates character)

public class Answer\_1 {

public static void main(String[] args) {

String word = "heelo";

String result = "";

boolean status =false;

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

char temp = word.charAt(i);

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

if(i==word.length()-1) {

break;

}

char temp\_2 = word.charAt(i+1);

if(temp==temp\_2) {

status=true;

}else {

status =false;

break;

}

}

if(status==false) {

result=result+word.charAt(i);

}

}

System.out.print(result);

}

}

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

//WAP to print Duplicates characters from the String.

public class Answer\_2 {

public static void main(String[] args) {

String str = "Helloogff";

String res ="";

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

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

if(j==i) {

j++;

}

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

res += str.charAt(j)+" ";

++i;

}

}

}

System.out.println("The letters Are :" +res);

}

}

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

//WAP to check if “2552” is palindrome or not.

public class Answer\_3 {

public static void main(String[] args) {

String str = "2552";

String res = "";

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

res +=str.charAt(i);

}

if(res.equals(str)) {

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

}

else

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

}

}

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

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

public class Answer\_4 {

public static void main(String[] args) {

String str = "#Mohan@";

int vowelChck =0;

int consonantChck =0;

int specialChck=0;

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

char temp = str.charAt(i);

if(temp>=65&&temp<=90) {

temp =(char) ((char) temp+32);

}

if(temp==97||temp==101||temp==105||temp==111||temp==117) {

vowelChck++;

}

else if((temp>=32&&temp<=47)||(temp>=58&&temp<=64)||(temp>=91&&temp<=96)||(temp>=123&&temp<=126)) {

specialChck++;

}

else {

consonantChck++;

}

}

System.out.println("The Vowels Are :"+vowelChck);

System.out.println("The Consonants Are :"+consonantChck);

System.out.println("The Special Characters Are :"+specialChck);

} }

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

//WAP to implement Anagram Checking least inbuilt methods being used.

public class Answer\_5 {

public static void main(String[] args) {

String name\_1 = "race";

String name\_2 = "care";

int checkCount =0;

for(int i=0;i<name\_1.length();i++) {

for(int j=0;j<name\_2.length();j++) {

if(name\_1.charAt(i)==name\_2.charAt(j)) {

checkCount++;

break;

}

}

}

if(checkCount==name\_1.length()) {

System.out.println("The Given Names Are Anagram");

}else {

System.out.println("The Given Names Are Not Angram");

}

}

}

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

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

public class Answer\_6 {

public static void main(String[] args) {

String details ="The five boxing wizards jump quickly";

String pangram = "";

String result ="";

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

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

pangram =pangram+details.charAt(i);

}

}

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

if(!result.contains(String.valueOf(pangram.charAt(i)))) {

result += String.valueOf(pangram.charAt(i));

}

}

if(result.length()==26) {

System.out.println("The Given Sentence is Pangram");

}else {

System.out.println("The Given Sentence is Not A Pangram");

}

}

}

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

//WAP to find if String contains all unique characters.

public class Answer\_7 {

public static void main(String[] args) {

String unique = "abcd10jk";

String chck = "";

boolean status =false;

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

char temp = unique.charAt(i);

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

if(i==unique.length()-1) {

break;

}

char temp\_2 = unique.charAt(i+1);

if(temp==temp\_2) {

status=true;

}else {

status =false;

break;

}

}

if(status) {

chck=chck+unique.charAt(i);

}

}

if(chck.length()==0) {

System.out.println("They Are All Unique Characters");

}else {

System.out.println("They Are Not Unique Characters");

}

}

}

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

//WAP to find the maximum occurring character in a String.

public class Answer\_8 {

public static void main(String[] args) {

String str = "Heeeeloofff";

String res ="";

int max =0;

int min = 0;

int fin =0;

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

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

if(j==i) {

j++;

}

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

res =res+str.charAt(j);

j++;

}

j++;

}

}

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

max=0;

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

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

if(i==0){

min++;

}else {

max++;

}

if(i==0) {

i += min-1;

}else if(i>res.length()-max){

i+=max-1;

}

}

}

if(i>=2) {

if(min>max) {

}else if(max>min) {

min =max;

fin=i;

}

}

if(i==1) {

if(min>max) {

fin =0;

}else if(max>min) {

fin=1;

min=max;

}

}

}

System.out.println("The Maximum Occurring Character is :"+res.charAt(fin));

}

}