Strings in Java

Assignment Questions

Assignment Questions

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

   Ans :

```
public class Duplicates {
    public static void main (String[] args){
    String name = "letter";
    String s1 = "";
    for (int i=0; i<name.length();i++){
        char result = name.charAt(i);
       int v = s1.indexOf(result);
       if (v==-1){
        s1 = s1 + result;
       }
    }
    System.out.println (s1);
    }
}
```

   Output :

```
letr
```

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

   Ans :

```
public class DuplicateChar {
    public static void main (String[]args){
        String name = "letter";
        String s1 = "";
        String s2 = "";
        for (int i=0 ; i<name.length();i++){
            char result = name.charAt(i);
            int v = s1.indexOf(result);
            if (v==-1){
                s1 = s1 + result;
            }
            else {
                s2 = s2 + result + " ";
            }
        }
        if (s2.length()>0){
        System.out.println("The duplicate characters are : " + s2);
        }
        else {
            System.out.println ("There is not any duplicate
character in the string");
        }
    }
}
```

Output :

```
The duplicate characters are : t e
```

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

Ans :

```
public class Palindrome {
    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("It is palindrome");
            }
                else {
                    System.out.println("It is not the palindrome");
                }
            }
        }
```

Output :

```
    It is palindrome
```

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

Ans :

```
public class Calculate {
public static void main (String[]args){

      String name = "computer!*&#@";
      String s1 = "";
      String s2 = "";
      String s3 = "";
      String ch = "*{}[],=-().+;'/@&#!";
      String vowels = "aeiou";
      for(int i=0;i<name.length();i++){
       char result = name.charAt(i);
       int v = vowels.indexOf(result);
       int c = ch.indexOf(result);
      if (v>=0){
       s1 = s1 + result;
      }
      else if (c>=0) {
       s2 = s2 + result;
      }
      else {
       s3 = s3 + result;
      }
     }
```

```
      System.out.println ("The number of vowels are "  + s1.length());
      System.out.println ("The numbers of Special characters are " +
s2.length());
      System.out.println ("The number of consonants are " +
s3.length());
  }
}
```

Output :

```
    The number of vowels are 3
    The numbers of Special characters are 5
    The number of consonants are 5
```

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

Ans :

```
    public class Anagram{
        public static void main (String[]args){
            String name = "night";
            String s1 = "";
            String name1 = "thing";
            for (int i=0;i<name.length();i++){
                char result = name.charAt(i);
                int v = name1.indexOf(result);
                char result1 = name1.charAt(i);
                int c = name.indexOf (result1);
                if (v>=0){
                s1 = s1 + result;
                }
            }
            if (s1.equals(name)){
                System.out.println ("It is Anagram");
            }
            else {
                System.out.println("it is not the Anagram");
            }
        }
    }
```

Output :

```
It is Anagram
```

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

Ans :

```
public class Pangram {
    public static void main (String[]args){
        String name = "The five boxing wizards jump quickly";
        name = name.toLowerCase();
        name = name.replace (" ", "");
        String Alphabet = "abcdefghijklmnopqrstuvwxyz";
        String s1 = "";
         for (int i=0; i<Alphabet.length();i++){
         char result = Alphabet.charAt(i);
         int v = name.indexOf(result);
         if (v>=0){
         s1 = s1 + result;
         }
         }
         if (s1.equals(Alphabet)){
          System.out.println("It is pangram");
         }
         else {
          System.out.println("It is not the pangram");
         }
    }
}
```

Output :

```
It is pangram
```

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

Ans :

```
public class Unique {
    public static void main (String[]args){
        String name = "computers";
        String s1 = "";
        for (int i=0 ; i<name.length();i++){
            char result = name.charAt(i);
            int v = name.indexOf(result);
            int c = name.lastIndexOf(result);
            if (v==c){
              s1 = s1 + result;
            }
        }
        if (s1.equals(name)){
            System.out.println ("This string contain all unique
characters");
        }
        else{
            System.out.println ("This string dose not contain all
the unique characters");
        }
    }
}
```

Output :

```
This string contain all unique characters
```

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

Ans :

```
public class Repetition {
    public static void main (String[]args){
     String name = "letter";
     String s3 = "";
     String s2 = "";
     String s1 = "";
     for (int i=0; i<name.length();i++){
         char result = name.charAt(i);
```

```
            int v = name.indexOf(result);
            int c = name.lastIndexOf(result);
            if (v!=c){
               s1 = s1 + result + " ";
            }
           else {
               s2 = s2 + result;
           }
        }
        if (s2!=name){
           if (s1.length()>0){
               for (int j=0;j<s1.length();j++){
               char results = s1.charAt(j);
               int z = s3.indexOf(results);
               if(z==-1){
               s3 = s3 + results + " ";
                }
               }
               System.out.println ("Maximum occurring character in the
string are : " + s3 );
           }
           else{
               System.out.println("There is not any maximum occurring
character in the string");
           }
        }
       }
    }
```

Output :

```
    Maximum occurring character in the string are : e t
```