String Part-3

Q1. WAP(Write a Program) to remove Duplicates from a String.(Take any String example with duplicates character)

```
public class Q1 {
   public static void main(String[] args) {

      String str1 = "programming";
      StringBuilder str2 = new StringBuilder();

      for(int i = 0; i < str1.length(); i++){
         char ch = str1.charAt(i);
         int idx = str1.indexOf(ch , i+1);
         if(idx == -1){
            str2.append(ch);
         }
      }
      System.out.println(str2);
      }
   }
```

Q2. WAP to print Duplicates characters from the String.

```
public class Q2 {
   public static void main(String[] args) {

      String str = "programming";
      StringBuilder str2 = new StringBuilder();

      for(int i = 0; i < str.length(); i++){
         char ch = str.charAt(i);
         int idx = str.indexOf(ch , i+1);
         if(idx > 1){
            str2.append(ch);
         }
      }
      System.out.println(str2);
   }
}
```

Q3 WAP to check if “2772” is palindrome or not.

```
public class Q3 {

   public static void main(String[] args) {
      String str1="2552";
      String str2 ="";
      for(int i=str1.length()-1;i>=0; i--)
         {
            str2 = str2 + str1.charAt(i);
         }
      if(str1.equals(str2))
```

```
        {
            System.out.println("It is a palindrome");
        }
      else
        {
            System.out.println("It is not a palindrome");
        }
   }
}
```

Q4.WAP to count the number of consonants, vowels, special characters in a String.

```
import java.util.*;

public class Q4 {
   public static void main(String[] args) {
      int vowels=0 ,consonants=0, specail=0;

      Scanner scan= new Scanner(System.in);
      System.out.println("Enter a String");

      String str = scan.nextLine();
      str =str.toLowerCase();

      for(int i = 0; i < str.length(); i++){
         char ch= str.charAt(i);
         if(ch=='a' || ch=='e'|| ch=='i' || ch=='o' || ch=='u' ){
            vowels++;
         }
         else if(ch >= 'a' && ch <= 'z'){
            consonants++;
         }
         else{
            specail++;
         }
      }
      System.out.println("Number of vowels are " + vowels);
      System.out.println("Number of consonants are " + consonants);
      System.out.println("Number of specail are " + specail);
   }
}
```

Q5. WAP to implement Anagram Checking least inbuilt methods being used.

```
import java.util.Arrays;

public class Q5 {
   public static void main(String[] args) {

      String str1 = "you Silent ";
      String str2 = "listen you   ";

      str1 = str1.replace(" ", "");
```

```
    str2 = str2.replace(" ", "");

    str1 = str1.toLowerCase();
    str2 = str2.toLowerCase();

    char []arr1 = str1.toCharArray();
    char []arr2 = str2.toCharArray();

    Arrays.sort(arr1);
    Arrays.sort(arr2);

    if(Arrays.equals(arr1, arr2)){
      System.out.println("It is an Anagram");
    }
    else{
      System.out.println("It is not an Anagram");
    }
  }
}
```

Q6. WAP to implement Pangram Checking with least inbuilt methods being used.

```
public class Q6 {
  public static void main(String[] args) {

    Boolean flag = false;
    String str = "Pack my box with five dozen liquor jugs";
    str = str.replace(" ", "");
    str = str.toLowerCase();
    char []ch = str.toCharArray();

    int ar[] = new int[26];

    for(int i = 0; i < ch.length; i++)
    {
      ar[ch[i] - 97] ++;
    }
    for(int i = 0; i < ar.length; i++ )
    {
      if(ar[i] == 0)
      {
        System.out.println("It is not a pangram");
        flag = true;
      }
    }
    if(flag == false)
    {
      System.out.println("It is a pangram");
    }
  }
}
```

Q7. WAP to find if String contains all unique characters.

```
public class Q7 {
   public static void main(String[] args) {

      Boolean flag = false;
      String str = "Mobile";
      str = str.toLowerCase();
      char ch[] = str.toCharArray();

      int ar[] = new int[127];

      for(int i = 0; i < ch.length; i++)
      {
         ar[ch[i] - 97]++;
      }
      for(int i = 0; i <  ar.length; i++)
      {
         if(ar[i] > 1)
         {
            System.out.println("It is not a Unique string");
            flag = true;
         }
      }
      if(flag == false)
      {
         System.out.println("All Uniqye characters");
      }
   }
}
```

Q8. WAP to find the maximum occurring character in a String

```
public class Q8 {
   public static void main(String[] args) {

      int max = 0;
      char res = 0;
      String str = "God Bless You";
      str = str.toLowerCase();
      char ch[] = str.toCharArray();

      int arr[] = new int[256];

      for(int i = 0; i < ch.length; i++)
      {
         arr[ch[i]]++;
      }

      for(int i = 0; i < ch.length; i++)
      {
        if(max < arr[ch[i]])
        {
            max = arr[ch[i]];
```

```
            res = str.charAt(i);
          }
        }
        System.out.println("The maximum occurring character is " + res);
        System.out.println(max + " Times");
    }
}
```