1.Write a program to remove duplicates from the string.
ANSWER:-

```
import java.util.Scanner;
class Duplicates{
        public static void main(String[] args) {
           Scanner sc=new Scanner(System.in);
           System.out.println(" enter the string : ");
           String s1=sc.next();
           StringBuilder sb=new StringBuilder();
           for(int i=0;i<s1.length();i++){
              char ch=s1.charAt(i);
              int idx=s1.indexOf(ch,i+1);
              if(idx==-1){
                 sb.append(ch);
              }
           }
           System.out.println("before removing duplicates : "+s1);
           System.out.println(" after removing duplicates :"+sb);
        }
}
```

Output:-

```
 enter the string :
pppppppppppp
before removing duplicates : pppppppppppp
 after removing duplicates :p
```

2.Write a program to print duplicate characters from the string ?
ANSWER:-

```
import java.util.Scanner;
public class PrintDuplicates {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.print(" enter the string : ");
        String s1=sc.next();
        System.out.println();
        int n=s1.length();
        int count[]=new int[300];
        for(int i=0;i<n;i++){
            count[s1.charAt(i)]++;
        }
        for(int i=0;i<300;i++){
            if(count[i]>1){
                System.out.println((char)(i)+" count = "+count[i]);
            }
        }
    }
```

```
}
```

Outpt:-
enter the string : aabccdee

a count = 2
c count = 2
e count = 2

3.Write a program to check if “2552” is palindrome or not ?
ANSWER:-

```
import java.util.Scanner;
public class Palin2552 {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.print(" enter the string to check palindrome or not : ");
        String s1=sc.next();
        System.out.println();
        String s2="";
        for(int i=s1.length()-1;i>=0;i--){
            s2=s2+s1.charAt(i);
        }
        if(s1.equals(s2)){
            System.out.println(s1+" is palindrome.");
        }
        else{
            System.out.println(s1+" is not palindrome.");
        }
    }
}
```

Output:-
enter the string to check palindrome or not : 2552

2552 is palindrome.

4.Write a program to count the number of consonants ,vowels,special characters in a string ?
ANSWER:-

```
import java.util.Scanner;
public class CountAll {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.println(" enter the string : ");
        String s1=sc.nextLine();
        s1=s1.toLowerCase();
        int vowels=0,consonants=0,specialCharacters=0,digits=0;
        char ch;
```

```
        for(int i=0;i<s1.length();i++){
            ch=s1.charAt(i);
            if(ch>='a' && ch<='z'){
              if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o'  || ch == 'u'){
                vowels++;
              }
              else{
                consonants++;
              }
            }
            else if(ch >= '0' && ch<='9'){
                digits++;
            }
            else{
                specialCharacters++;
            }
        }
        System.out.println(" vowels in "+s1+" are : "+vowels);
        System.out.println(" consonants in "+s1+" are :"+consonants);
        System.out.println("digits in "+s1+" are :"+digits);
        System.out.println(" special characters in "+s1+" are :"+specialCharacters);
    }
}
```

Output:-

```
 enter the string :
pwskills java dsa english batch 2023-2025.
 vowels in pwskills java dsa english batch 2023-2025. are : 7
 consonants in pwskills java dsa english batch 2023-2025. are :20
digits in pwskills java dsa english batch 2023-2025. are :8
 special characters in pwskills java dsa english batch 2023-2025. are :7
```

5.Write a program to implement Anagram checking least inbuilt methods being used ?
ANSWER:-

```
import java.util.Arrays;

public class Anagram {
    public static void main(String[] args) {
        String s1="the classroom";
        String s2="school master";
        s1=s1.replace(" ", "");
        s2=s2.replace(" ","");
        s1=s1.toLowerCase();
        s2=s2.toLowerCase();
        char ar1[]=s1.toCharArray();
        char ar2[]=s2.toCharArray();
        Arrays.sort(ar1);
```

```
        Arrays.sort(ar2);

        if(Arrays.equals(ar1,ar2)){
            System.out.println("The words are in anagram");
        }
        else{
            System.out.println(" The words are not in not anagram");
        }
    }
}
```

Output:-
The words are in anagram

6.Write a program to implement pangram checking with least inbuilt methods being used ?
ANSWER:-

```
public class PangramCheck {

        public static boolean isPangram(String input) {
            input = input.toLowerCase();
            for (char ch = 'a'; ch <= 'z'; ch++) {
                if (input.indexOf(ch) < 0) {
                    return false;
                }
            }
            return true;
        }
    public static void main(String[] args) {


            String input = "The quick brown fox jumps over the lazy dog";
            System.out.println(PangramCheck.isPangram(input));


    }

}
```

Output:-
true

7.Write a program to find if String contains all unique characters ?
ANSWER:-

```
public class UniqueCharacters {
    public static boolean hasUniqueCharacters(String input) {
        for (int i = 0; i < input.length(); i++) {
            char ch = input.charAt(i);
```

```
            for (int j = i + 1; j < input.length(); j++) {
                if (input.charAt(j) == ch) {
                    return false;
                }
            }
        }
        return true;
    }

   public static void main(String[] args) {
    String input = "hello";
    System.out.println(UniqueCharacters.hasUniqueCharacters(input));
}


}
```

Output:-
false

8.Write a program to find maximum occurring character in a string ?
ANSWER:-

```
public class MaxOccurringCharacter {
    public static char maxOccurringCharacter(String input) {
        int[] count = new int[256];
        for (int i = 0; i < input.length(); i++) {
            count[input.charAt(i)]++;
        }
        int max = Integer.MIN_VALUE;
        char result = ' ';
        for (int i = 0; i < count.length; i++) {
            if (count[i] > max) {
                max = count[i];
                result = (char) i;
            }
        }
        return result;
    }

      public static void main(String[] args) {
    String input = "hello";
    System.out.println(MaxOccurringCharacter.maxOccurringCharacter(input));
}
}
```

Output:-
l