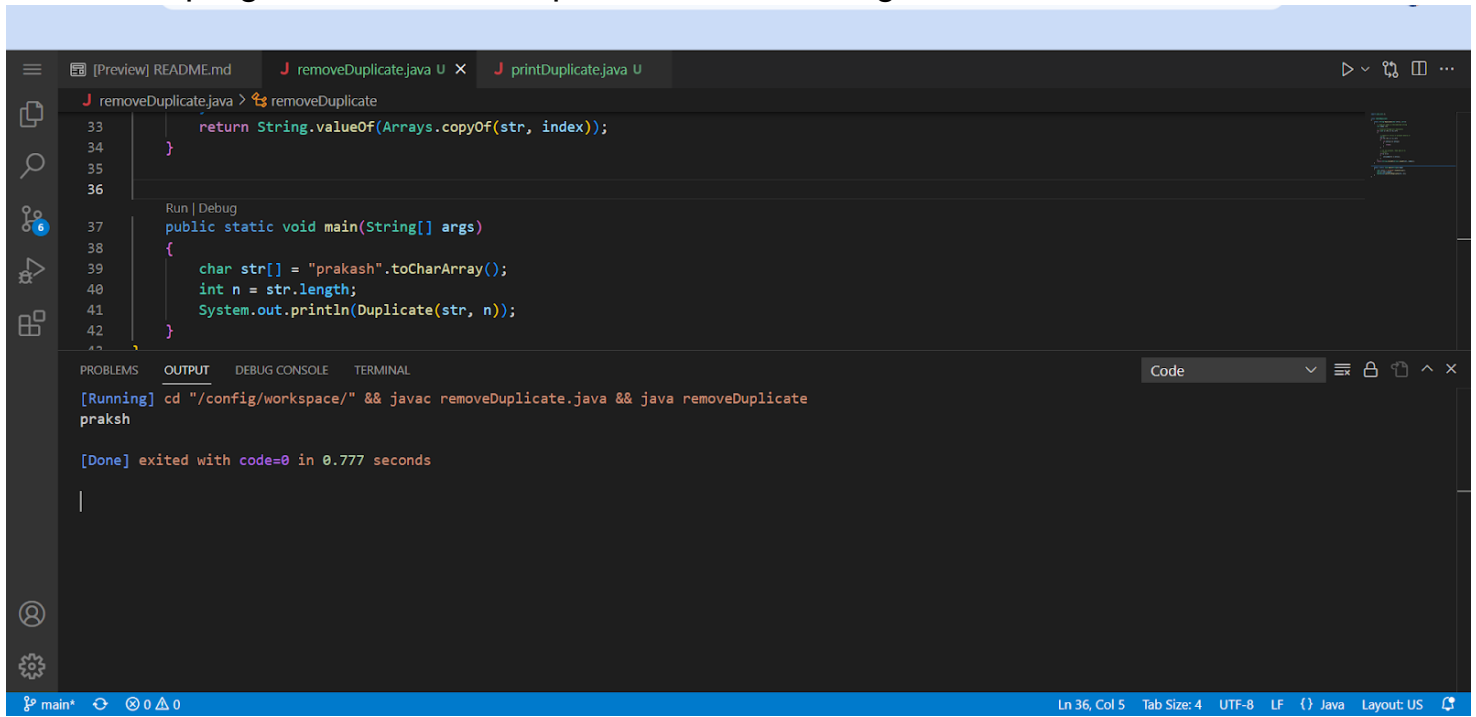


Strings in Java Part-3

Assignment Questions

1->Write a program to remove duplicates from a string ?



The screenshot shows an IDE with a Java file named `removeDuplicate.java`. The code defines a `Duplicate` method that takes a character array and its length, and a `main` method that tests it with the string "prakash". The output shows the result "praksh".

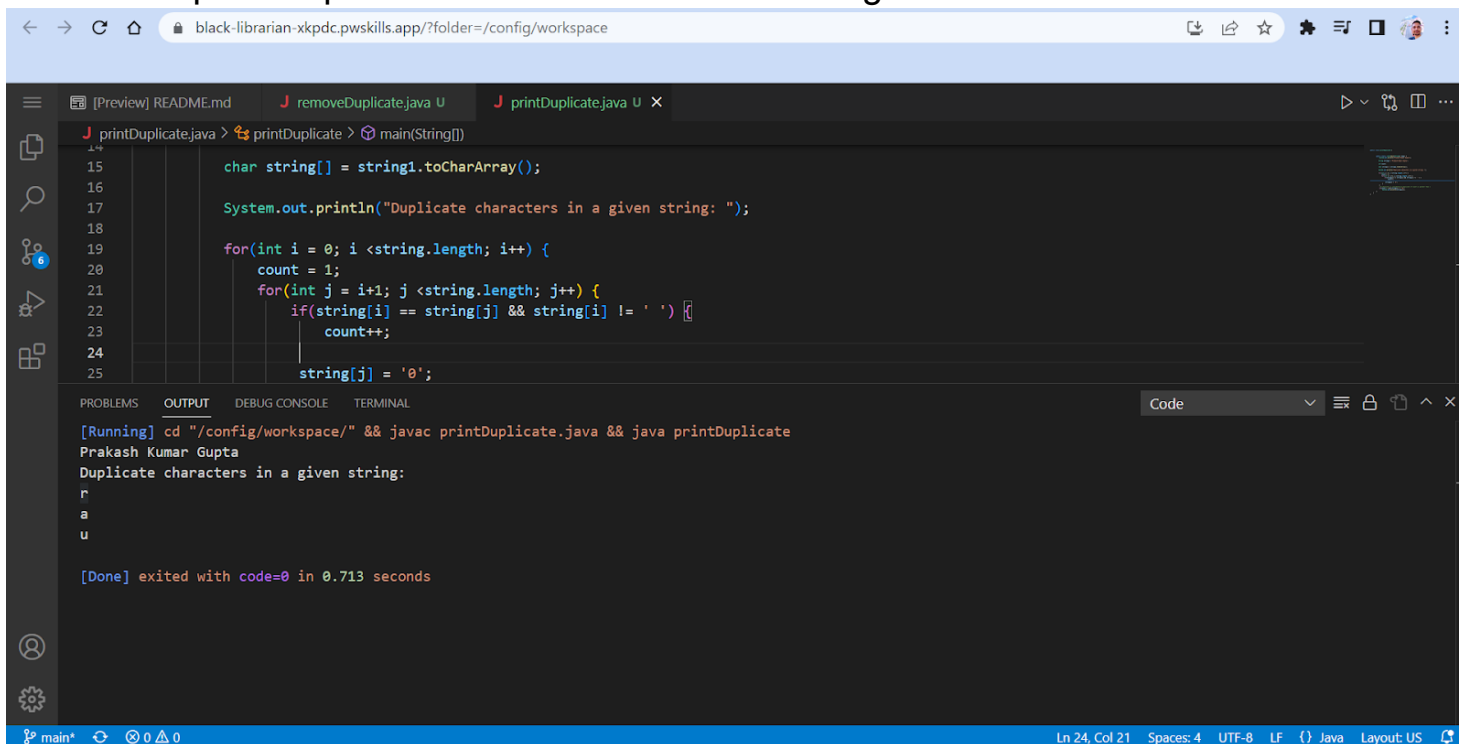
```
33     return String.valueOf(Arrays.copyOf(str, index));
34 }
35
36
37 Run | Debug
38 public static void main(String[] args)
39 {
40     char str[] = "prakash".toCharArray();
41     int n = str.length;
42     System.out.println(Duplicate(str, n));
43 }
```

OUTPUT

```
[Running] cd "/config/workspace/" && javac removeDuplicate.java && java removeDuplicate
praksh

[Done] exited with code=0 in 0.777 seconds
```

2->WAP to print Duplicates characters from the string ?



The screenshot shows an IDE with a Java file named `printDuplicate.java`. The code defines a `main` method that takes a string, converts it to a character array, and prints the duplicate characters. The output shows the string "Prakash Kumar Gupta" and the duplicate characters "a" and "u".

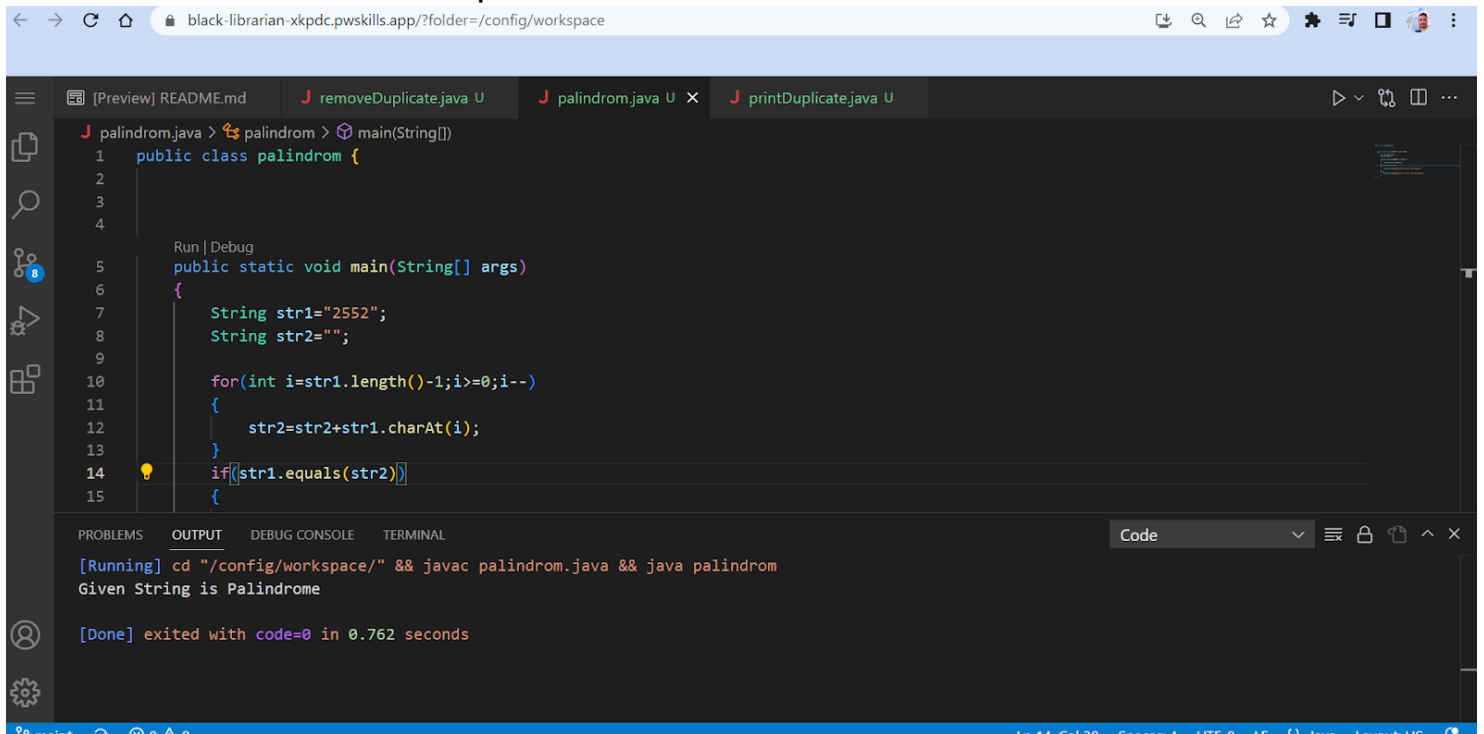
```
15 char string[] = string1.toCharArray();
16
17 System.out.println("Duplicate characters in a given string: ");
18
19 for(int i = 0; i < string.length; i++) {
20     count = 1;
21     for(int j = i+1; j < string.length; j++) {
22         if(string[i] == string[j] && string[i] != ' ') {
23             count++;
24         }
25         string[j] = '0';
26     }
27 }
```

OUTPUT

```
[Running] cd "/config/workspace/" && javac printDuplicate.java && java printDuplicate
Prakash Kumar Gupta
Duplicate characters in a given string:
a
u

[Done] exited with code=0 in 0.713 seconds
```

3->WAP to count if "2552" is palindrome or not ?



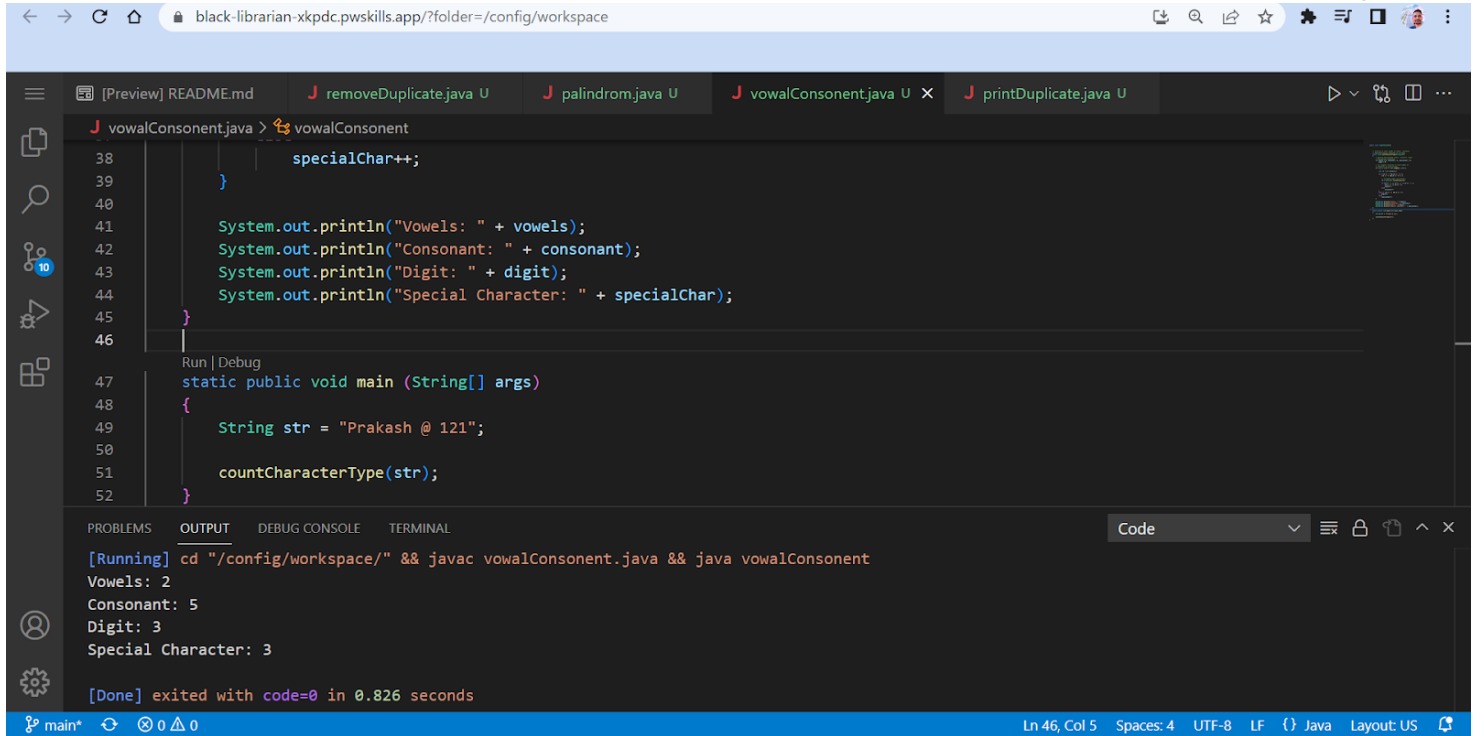
```
[Preview] README.md J removeDuplicate.java U J palindrom.java U X J printDuplicate.java U
J palindrom.java > palindrom > main(String[])
1 public class palindrom {
2
3
4
5 Run | Debug
6 public static void main(String[] args)
7 {
8     String str1="2552";
9     String str2="";
10
11     for(int i=str1.length()-1;i>=0;i--)
12     {
13         str2=str2+str1.charAt(i);
14     }
15     if(str1.equals(str2))
16     {
17
18     }
19 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Code

```
[Running] cd "/config/workspace/" && javac palindrom.java && java palindrom
Given String is Palindrome

[Done] exited with code=0 in 0.762 seconds
```

4->WAP to count the number of consonants , vowels , special characters in a string ?



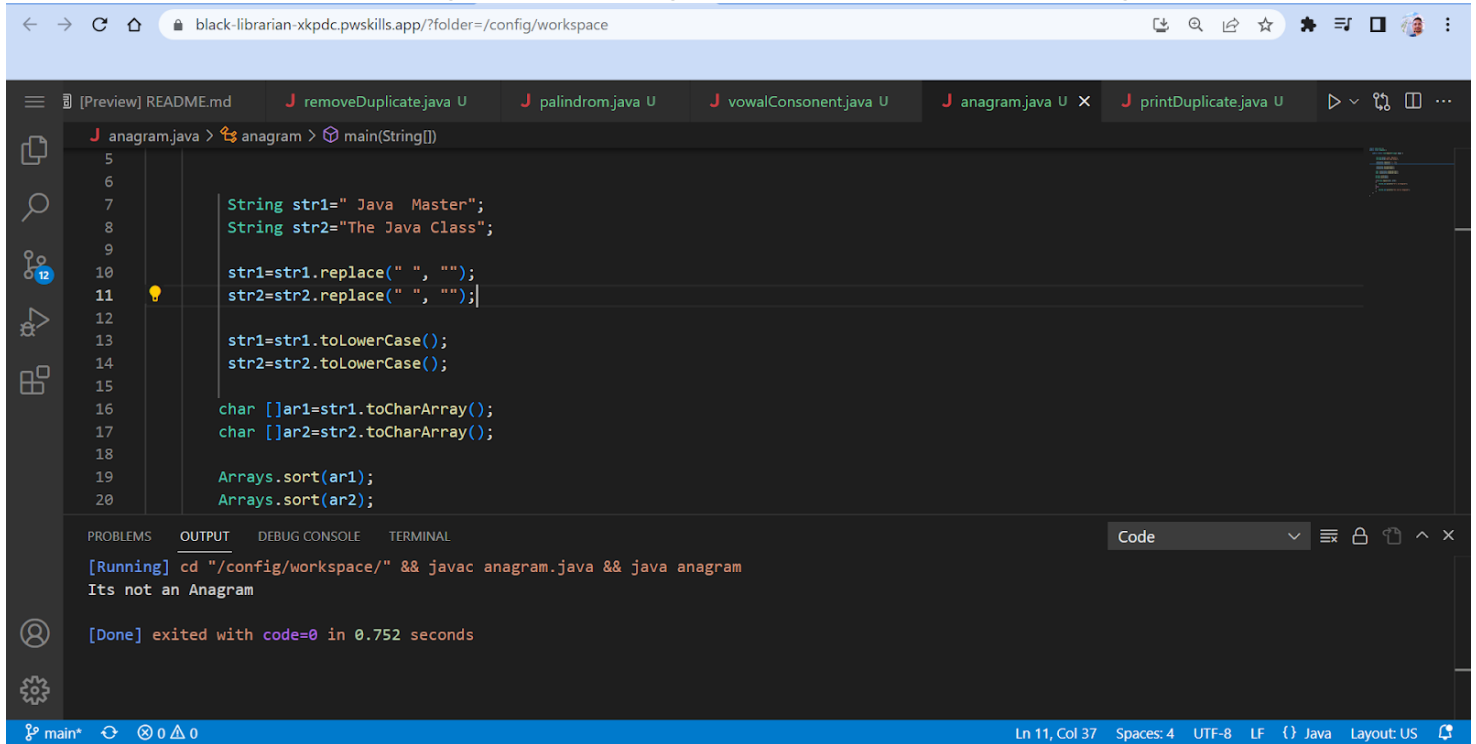
```
[Preview] README.md J removeDuplicate.java U J palindrom.java U J vowelConsonent.java U X J printDuplicate.java U
J vowelConsonent.java > vowelConsonent
38     specialChar++;
39 }
40
41 System.out.println("Vowels: " + vowels);
42 System.out.println("Consonant: " + consonant);
43 System.out.println("Digit: " + digit);
44 System.out.println("Special Character: " + specialChar);
45 }
46
47 Run | Debug
48 static public void main (String[] args)
49 {
50     String str = "Prakash @ 121";
51
52     countCharacterType(str);
53 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Code

```
[Running] cd "/config/workspace/" && javac vowelConsonent.java && java vowelConsonent
Vowels: 2
Consonant: 5
Digit: 3
Special Character: 3

[Done] exited with code=0 in 0.826 seconds
```

5->WAP to implement Anagram checking least inbuilt methods being used ?

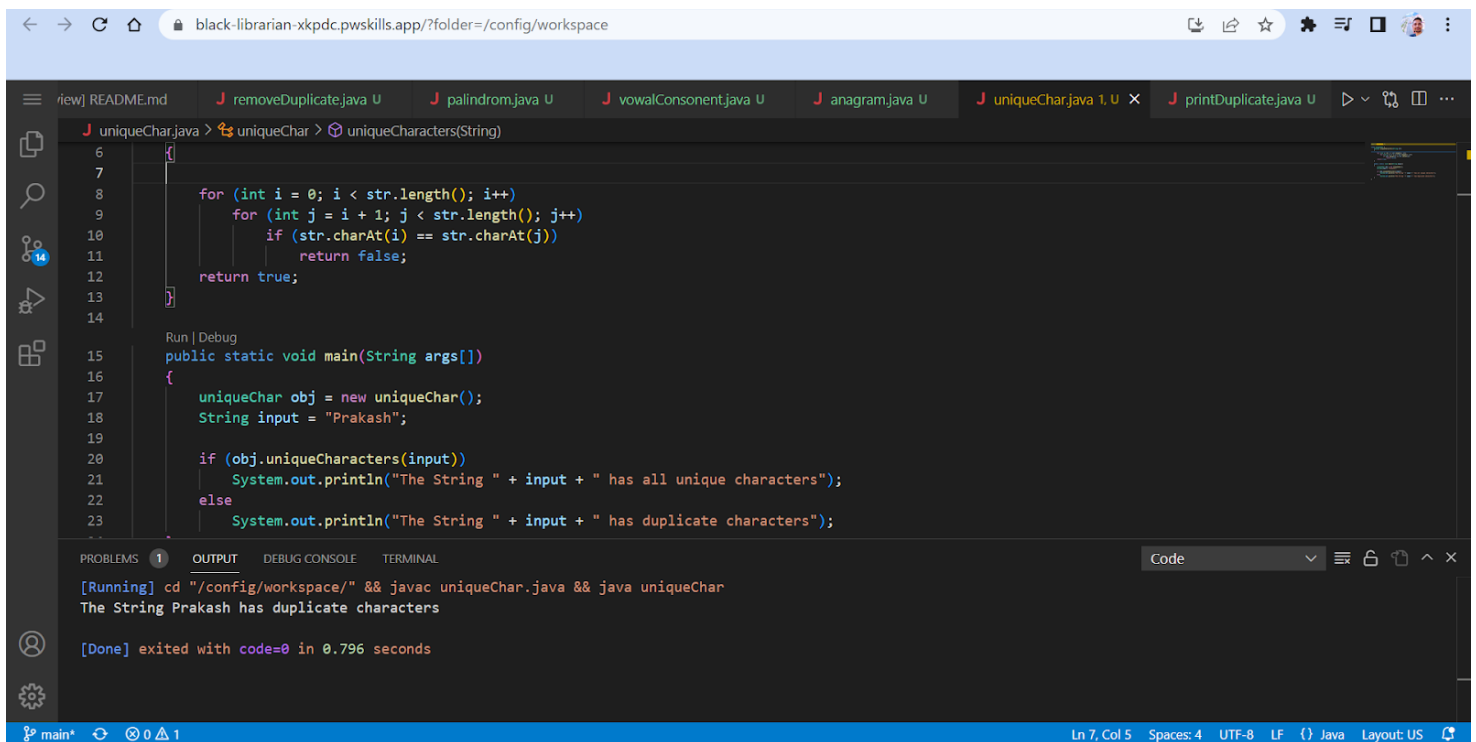


The screenshot shows a code editor with a file named `anagram.java`. The code implements an anagram checker by replacing spaces with empty strings, converting to lowercase, and sorting the character arrays. The output window shows the program running and outputting "Its not an Anagram".

```
5  
6  
7     String str1=" Java  Master";  
8     String str2="The Java Class";  
9  
10    str1=str1.replace(" ", "");  
11    str2=str2.replace(" ", "");  
12  
13    str1=str1.toLowerCase();  
14    str2=str2.toLowerCase();  
15  
16    char []ar1=str1.toCharArray();  
17    char []ar2=str2.toCharArray();  
18  
19    Arrays.sort(ar1);  
20    Arrays.sort(ar2);
```

OUTPUT
[Running] cd "/config/workspace/" && javac anagram.java && java anagram
Its not an Anagram
[Done] exited with code=0 in 0.752 seconds

6->WAP to find if string contain all unique characters ?



The screenshot shows a code editor with a file named `uniqueChar.java`. The code implements a method to check if a string has all unique characters using nested loops. The output window shows the program running and outputting "The String Prakash has duplicate characters".

```
6  
7  
8     for (int i = 0; i < str.length(); i++)  
9     {  
10        for (int j = i + 1; j < str.length(); j++)  
11        {  
12            if (str.charAt(i) == str.charAt(j))  
13                return false;  
14        }  
15    }  
16    return true;  
17  
18    public static void main(String args[])  
19    {  
20        uniqueChar obj = new uniqueChar();  
21        String input = "Prakash";  
22  
23        if (obj.uniqueCharacters(input))  
24            System.out.println("The String " + input + " has all unique characters");  
25        else  
26            System.out.println("The String " + input + " has duplicate characters");  
27    }
```

OUTPUT
[Running] cd "/config/workspace/" && javac uniqueChar.java && java uniqueChar
The String Prakash has duplicate characters
[Done] exited with code=0 in 0.796 seconds

7->WAP to find the maximum occurring character in a string ?

The screenshot shows a web-based IDE interface with a browser address bar at the top displaying "black-librarian-xkpcdc.pwskills.app/?folder=/config/workspace". The IDE has a dark theme and a sidebar on the left with icons for Explorer, Search, Run and Debug, Extensions, and Settings. The top of the IDE features a tab bar with several open files: "eDuplicate.java U", "palindrom.java U", "vowalConsonent.java U", "anagram.java U", "uniqueChar.java 1. U", "maxOccurance.java U x", and "printDuplicate.java U". The active file is "maxOccurance.java", which contains the following Java code:

```
19
20
21     {
22         if (max_count < count)
23         {
24             max_count = count;
25             ans = s.charAt(i-1);
26         }
27         count = 1;
28     }
29     else
30     {
31         count++;
32     }
33     System.out.println("prakash");
34     System.out.println("Maximum occurring character is "+ans);
35 }
36
37
```

Below the code editor, there is a panel with tabs for "PROBLEMS", "OUTPUT", "DEBUG CONSOLE", and "TERMINAL". The "OUTPUT" tab is selected, showing the following text:

```
[Running] cd "/config/workspace/" && javac maxOccurance.java && java maxOccurance
prakash
Maximum occurring character is a

[Done] exited with code=0 in 0.823 seconds
```

The bottom status bar of the IDE shows "main*", "0 1", "Ln 32, Col 41", "Spaces: 4", "UTF-8", "LF", "Java", and "Layout: US".

Note:- Please create a Google Document and write your answers and upload the shareable link of the Google Document with view access during the submission of the assignment.

Cracking the Coding Interview in JAVA - Foundation