



# VIT<sup>®</sup>

**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

## **Computer Programming Java**

### **MOODLE LAB TASK 4**

**Faculty: Sathya Raj R**

**Slot: L3+L4+L29+L30**

**Venue: SJT 515**

**NAME: Soumyojyoti Saha**

**Reg ID: 21BCE4007**

## String Questions

**Q1. Write a Java program to find duplicate values in an array of string values.**

### CODE:

```
public class DuplicateFinder {  
    public static void main(String[] args) {  
        // Declare and initialize a string array 'my_array'.  
        String[] my_array = {"apple", "mango", "banana", "apple", "orange", "mango",  
            "guava"};  
        // Array to track elements that have been checked as duplicates.  
        boolean[] printed = new boolean[my_array.length];  
  
        // Iterate through the elements of the string array.  
        for (int i = 0; i < my_array.length - 1; i++) {  
            for (int j = i + 1; j < my_array.length; j++) {  
                // Check if two string elements are equal and not the same element.  
                if (my_array[i].equals(my_array[j]) && !printed[j]) {  
                    // Print the duplicate element and mark it as printed.  
                    System.out.println("Duplicate Element is : " + my_array[j]);  
                    printed[j] = true; // Mark this element as processed.  
                }  
            }  
        }  
    }  
}
```

## OUTPUT:

```
strings > J DuplicateFinder.java > ...
1  public class DuplicateFinder {
    Run | Debug
2  public static void main(String[] args) {
3      // Declare and initialize a string array 'my_array'.
4      String[] my_array = {"apple", "mango", "banana", "apple", "orange", "mango", "guava"};
5
6      // Array to track elements that have been checked as duplicates.
7      boolean[] printed = new boolean[my_array.length];
8
9      // Iterate through the elements of the string array.
10     for (int i = 0; i < my_array.length - 1; i++) {
11         for (int j = i + 1; j < my_array.length; j++) {
12             // Check if two string elements are equal and not the same element.
13             if (my_array[i].equals(my_array[j]) && !printed[j]) {
14                 // Print the duplicate element and mark it as printed.
15                 System.out.println("Duplicate Element is : " + my_array[j]);
16                 printed[j] = true; // Mark this element as processed.
17             }
18         }
19     }
20 }
21 }
22
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> cd "c:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings" & if (\$?) { java DuplicateFinder }
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> Duplicate Element is : apple  
Duplicate Element is : mango
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings>

**Q2. Write a Java program to concatenate a given string to the end of another string. String 1: Hello, String 2: World. The concatenated string: Hello World**

**CODE:**

```
public class StringConcatenation {  
    public static void main(String[] args) {  
        String str1 = "Hello";  
        String str2 = "World";  
        String concatenatedString = concatenateStrings(str1, str2);  
        System.out.println("The concatenated string: " + concatenatedString);  
    }  
  
    public static String concatenateStrings(String str1, String str2) {  
        return str1 + " " + str2;  
    }  
}
```

## OUTPUT:

```
strings > J StringConcatenation.java > ...
1  public class StringConcatenation {
    Run | Debug
2  public static void main(String[] args) {
3      String str1 = "Hello";
4      String str2 = "World";
5      String concatenatedString = concatenateStrings(str1, str2);
6      System.out.println("The concatenated string: " + concatenatedString);
7  }
8
9  public static String concatenateStrings(String str1, String str2) {
10     return str1 + " " + str2;
11 }
12 }
13
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> cd "c:\Users\Soumyojyoti oncatenation.java" ; if (\$?) { java StringConcatenation }  
The concatenated string: Hello World
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings>

**Q3. Write a Java program to compare a given string to another string, ignoring case considerations.**

**CODE:**

```
public class StringComparison {  
    public static void main(String[] args) {  
        String str1 = "Soumyojyoti";  
        String str2 = "soumyojyoti";  
  
        boolean isEqual = compareStringsIgnoreCase(str1, str2);  
        System.out.println("Strings are equal (ignoring case): " + isEqual);  
    }  
  
    public static boolean compareStringsIgnoreCase(String str1, String str2) {  
        return str1.equalsIgnoreCase(str2);  
    }  
}
```

## OUTPUT:

```
strings > J StringComparison.java > StringComparison > main(String[])
1  public class StringComparison {
    Run | Debug
2      public static void main(String[] args) {
3          String str1 = "Soumyojyoti";
4          String str2 = "soumyojyoti";
5
6          boolean isEqual = compareStringsIgnoreCase(str1, str2);
7          System.out.println("Strings are equal (ignoring case): " + isEqual);
8      }
9
10     public static boolean compareStringsIgnoreCase(String str1, String str2) {
11         return str1.equalsIgnoreCase(str2);
12     }
13 }
14
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> cd "c:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\" ; if (\$?) { java StringComparison }  
Strings are equal (ignoring case): true
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings>

**Q4. Write a Java program to count a given character's total number of occurrences in a string without using any loop.**

**CODE:**

```
public class CharacterCount {  
    public static void main(String[] args) {  
        String str = "Hello World";  
        char character = 'o';  
        int count = countCharacterOccurrences(str, character);  
        System.out.println("Total occurrences of '" + character + "': " + count);  
    }  
  
    public static int countCharacterOccurrences(String str, char character) {  
        return str.length() - str.replace(String.valueOf(character), "").length();  
    }  
}
```



## OUTPUT:

```
strings > J CharacterCount.java > CharacterCount > countCharacterOccurrences(String, char)
1  public class CharacterCount {
    Run | Debug
2      public static void main(String[] args) {
3          String str = "Hello World";
4          char character = 'o';
5          int count = countCharacterOccurrences(str, character);
6          System.out.println("Total occurrences of '" + character + "': " + count);
7      }
8
9      public static int countCharacterOccurrences(String str, char character) {
10         return str.length() - str.replace(String.valueOf(character), replacement:"").length();
11     }
12 }
13
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> cd "c:\Users\Soumyojyoti Saha\OneDrive erCount.java" ; if (\$?) { java CharacterCount }
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> Total occurrences of 'o': 2
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings>

**Q5. Write a Java program to replace each substring of a given string that matches the given regular expression with the given replacement. Sample string: "He is good person and he is an active person." In the above string replace all the person with boy.**

**CODE:**

```
public class StringReplacement {  
  
    // Define the main method.  
    public static void main(String[] args) {  
        // Declare and initialize a string variable.  
        String str = "He is good person and he is an active person.";  
  
        // Replace all occurrences of 'fox' with 'cat'.  
        String new_str = str.replaceAll("person", "boy");  
  
        // Display the original and modified strings for comparison.  
        System.out.println("Original string: " + str);  
        System.out.println("New String: " + new_str);  
    }  
}
```

## OUTPUT:

```
strings > J StringReplacement.java > StringReplacement > main(String[])
1 public class StringReplacement {
2
3     // Define the main method.
4     public static void main(String[] args) {
5         // Declare and initialize a string variable.
6         String str = "He is good person and he is an active person.";
7
8         // Replace all occurrences of 'fox' with 'cat'.
9         String new_str = str.replaceAll(regex:"person", replacement:"boy");
10
11        // Display the original and modified strings for comparison.
12        System.out.println("Original string: " + str);
13        System.out.println("New String: " + new_str);
14    }
15 }
16
17
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> cd "c:\Users\Soumyojyoti Saha\replacement.java" ; if (\$?) { java StringReplacement }
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings>

Original string: He is good person and he is an active person.  
New String: He is good boy and he is an active boy.

**Q6. Write a Java Program to reverse a string and check is it palindrome string or not.**

**CODE:**

```
public class PalindromeChecker {

    // Method to check if a string is a palindrome
    public static boolean isPalindrome(String str) {
        // Using StringBuilder to reverse the string
        StringBuilder rev = new StringBuilder(str).reverse();

        // Check if the reversed string is equal to the original string
        return str.equals(rev.toString());
    }

    public static void main(String[] args) {
        // Input string
        String str = "madam";

        // Convert the string to lowercase
        str = str.toLowerCase();

        // Check if the string is a palindrome
        boolean isPalindrome = isPalindrome(str);
        System.out.println(isPalindrome);
    }
}
```

## OUTPUT:

```
strings > J PalindromeChecker.java > PalindromeChecker > main(String[])
1  public class PalindromeChecker {
2
3      // Method to check if a string is a palindrome
4      public static boolean isPalindrome(String str) {
5          // Using StringBuilder to reverse the string
6          StringBuilder rev = new StringBuilder(str).reverse();
7
8          // Check if the reversed string is equal to the original string
9          return str.equals(rev.toString());
10     }
11
12     Run | Debug
13     public static void main(String[] args) {
14         // Input string
15         String str = "madam";
16
17         // Convert the string to lowercase
18         str = str.toLowerCase();
19
20         // Check if the string is a palindrome
21         boolean isPalindrome = isPalindrome(str);
22         System.out.println(isPalindrome);
23     }
24 }
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> cd "c:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings" & java PalindromeChecker
true
PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> 
```

**Q7. Java program to change the string into uppercase letters and lowercase letters.**

**CODE:**

```
public class CaseConversion {  
    public static void main(String[] args) {  
        String str = "Soumyojyoti Saha";  
  
        String upperCaseString = toUpperCase(str);  
        String lowerCaseString = toLowerCase(str);  
  
        System.out.println("Uppercase: " + upperCaseString);  
        System.out.println("Lowercase: " + lowerCaseString);  
    }  
  
    public static String toUpperCase(String str) {  
        return str.toUpperCase();  
    }  
  
    public static String toLowerCase(String str) {  
        return str.toLowerCase();  
    }  
}
```

## OUTPUT:

```
strings > J CaseConversion.java > CaseConversion > main(String[])
1  public class CaseConversion {
    Run | Debug
2  public static void main(String[] args) {
3  String str = "Soumyojyoti Saha";
4
5  String upperCaseString = toUpperCase(str);
6  String lowerCaseString = toLowerCase(str);
7
8  System.out.println("Uppercase: " + upperCaseString);
9  System.out.println("Lowercase: " + lowerCaseString);
10 }
11
12 public static String toUpperCase(String str) {
13     return str.toUpperCase();
14 }
15
16 public static String toLowerCase(String str) {
17     return str.toLowerCase();
18 }
19 }
20
```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS

- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> cd "c:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings" & if (\$?) { java CaseConversion }
- Uppercase: SOUMYOJYOTI SAHA
- Lowercase: soumyojyoti saha
- PS C:\Users\Soumyojyoti Saha\OneDrive - vit.ac.in\Desktop\java sem 7\strings> █

END