class StringMethodsDemo.java 1

C: > Users > KALYAN > Desktop > SEM3 > java > class StringMethodsDemo.java > StringMethodsDemo

```java
class StringMethodsDemo
{
    Run | Debug
    public static void main(String[] args)
    {
        String str = "Java Programming";
        String str2 = "java programming";
        System.out.println(x: " String Methods Demonstration \n");
        //length()
        System.out.println("1. Length of string: " + str.length());
        //charAt()
        System.out.println("2. Character at index 5: " + str.charAt(index: 5));
        //substring()
        System.out.println("3. Substring(5): " + str.substring(beginIndex: 5));
        System.out.println(" Substring(5, 11): " + str.substring(beginIndex: 5, endIndex: 11));
        //equals() and equalsIgnoreCase()
        System.out.println("4. Equals: " + str.equals(str2));
        System.out.println(" Equals Ignore Case: " + str.equalsIgnoreCase(str2));
        //compareTo()
        System.out.println("5. CompareTo (lexicographically): " + str.compareTo(str2));
        //toUpperCase() and toLowerCase()
        System.out.println("6. Uppercase: " + str.toUpperCase());
        System.out.println(" Lowercase: " + str.toLowerCase());
        //trim()
        String spaced = " Hello World ";
        System.out.println("7. Trimmed string: '" + spaced.trim() + "'");
        //startsWith() and endsWith()
        System.out.println("8. Starts with 'Java': " + str.startsWith(prefix: "Java"));
        System.out.println(" Ends with 'ing': " + str.endsWith(suffix: "ing"));
        //contains()
        System.out.println("9. Contains 'gram': " + str.contains(s: "gram"));
        //indexOf() and lastIndexOf()
        System.out.println("10. Index of 'a': " + str.indexOf(ch: 'a'));
        System.out.println(" Last index of 'a': " + str.lastIndexOf(ch: 'a'));
        //replace() and replaceAll()
        System.out.println("11. Replace 'Java' with 'Python': " + str.replace(target: "Java", replacement: "Python"));
        System.out.println(" Replace all 'a' with '*': " + str.replaceAll(regex: "a", replacement: "*"));
```

class StringMethodsDemo.java 1

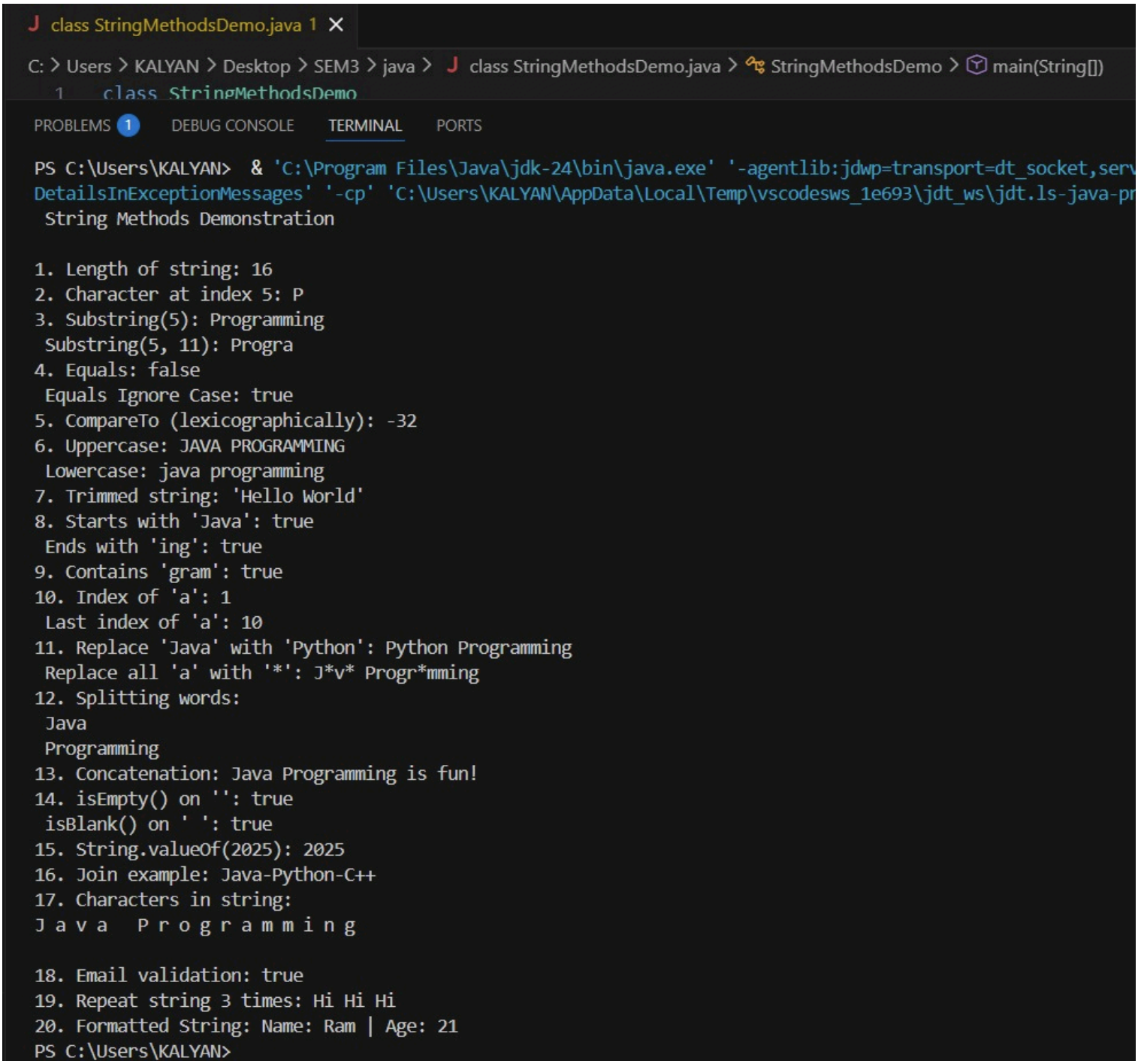
C: > Users > KALYAN > Desktop > SEM3 > java > class StringMethodsDemo.java > StringMethodsDemo

```java
class StringMethodsDemo
    public static void main(String[] args)
        System.out.println(" Replace all 'a' with '*': " + str.replaceAll(regex: "a", replacement: "*"));
        //split()
        System.out.println(x: "12. Splitting words:");
        String[] words = str.split(regex: " ");
        for (String w : words)
            System.out.println(" " + w);
        //concat()
        System.out.println("13. Concatenation: " + str.concat(str: " is fun!"));
        //isEmpty() and isBlank()
        String emptyStr = "";
        String blankStr = " ";
        System.out.println("14. isEmpty() on '': " + emptyStr.isEmpty());
        System.out.println(" isBlank() on ' ': " + blankStr.isBlank());
        //valueOf()
        int num = 2025;
        System.out.println("15. String.valueOf(2025): " + String.valueOf(num));
        //join()
        System.out.println("16. Join example: " + String.join(delimiter: "-", ...elements: "Java", "Python", "C++"));
        //toCharArray()
        System.out.println(x: "17. Characters in string:");
        char[] chars = str.toCharArray();
        for (char c : chars)
            System.out.print(c + " ");
        System.out.println();
        //matches() (regex)
        String email = "test123@gmail.com";

        System.out.println("\n18. Email validation: " + email.matches(regex: "[a-z0-9]+@[a-z]+\\.[a-z]{2,3}"));
        //repeat()
        System.out.println("19. Repeat string 3 times: " + "Hi ".repeat(count: 3));
        //format()
        System.out.println("20. Formatted String: " + String.format(format: "Name: %s | Age: %d",...args: "Ram", 21));

    }
}
```

Ln 2, Col 2 Spaces: 4 UTF-8 { } Java


class StringMethodsDemo.java 1
C: > Users > KALYAN > Desktop > SEM3 > java > class StringMethodsDemo.java > StringMethodsDemo > main(String[])
1 class StringMethodsDemo
PROBLEMS 1
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\KALYAN> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,serv
DetailsInExceptionMessages' '-cp' 'C:\Users\KALYAN\AppData\Local\Temp\vscodesws_1e693\jdt_ws\jdt.ls-java-pr
String Methods Demonstration
1. Length of string: 16
2. Character at index 5: P
3. Substring(5): Programming
Substring(5, 11): Progra
4. Equals: false
Equals Ignore Case: true
5. CompareTo (lexicographically): -32
6. Uppercase: JAVA PROGRAMMING
Lowercase: java programming
7. Trimmed string: 'Hello World'
8. Starts with 'Java': true
Ends with 'ing': true
9. Contains 'gram': true
10. Index of 'a': 1
Last index of 'a': 10
11. Replace 'Java' with 'Python': Python Programming
Replace all 'a' with '*': J*v* Progr*mming
12. Splitting words:
Java
Programming
13. Concatenation: Java Programming is fun!
14. isEmpty() on '': true
isBlank() on ' ': true
15. String.valueOf(2025): 2025
16. Join example: Java-Python-C++
17. Characters in string:
J a v a P r o g r a m m i n g
18. Email validation: true
19. Repeat string 3 times: Hi Hi Hi
20. Formatted String: Name: Ram | Age: 21
PS C:\Users\KALYAN>

class StringMethodsDemo.java 1 | class StringBufferMethodsDemo.java 1

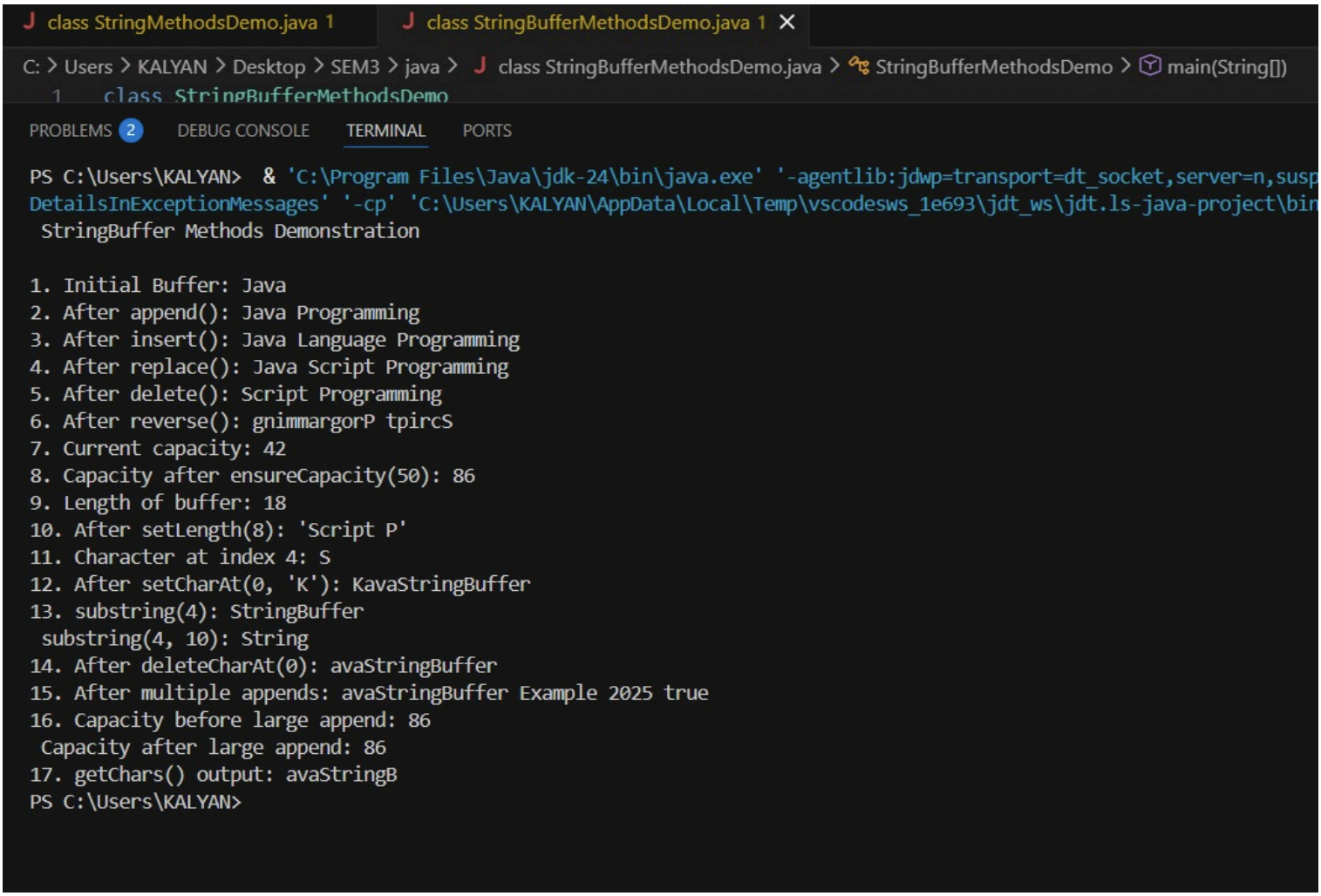
C: > Users > KALYAN > Desktop > SEM3 > java > class StringBufferMethodsDemo.java > StringBufferMethodsDemo > main(String[])

```java
class StringBufferMethodsDemo
{
    Run | Debug
    public static void main(String[] args)
    {
        System.out.println(x: " StringBuffer Methods Demonstration \n");
        //Creating a StringBuffer
        StringBuffer sb = new StringBuffer(str: "Java");
        System.out.println("1. Initial Buffer: " + sb);
        //append() - Add text at the end
        sb.append(str: " Programming");
        System.out.println("2. After append(): " + sb);
        //insert() - Insert text at a specific position
        sb.insert(offset: 4, str: " Language");
        System.out.println("3. After insert(): " + sb);
        //replace() - Replace characters between start and end indices
        sb.replace(start: 5, end: 13, str: "Script");
        System.out.println("4. After replace(): " + sb);
        //delete() - Remove characters between given indices
        sb.delete(start: 0, end: 5);
        System.out.println("5. After delete(): " + sb);
        //reverse() - Reverse the entire sequence
        sb.reverse();
        System.out.println("6. After reverse(): " + sb);
        sb.reverse(); // Restore original content
        //capacity() - Get the current capacity of the buffer
        System.out.println("7. Current capacity: " + sb.capacity());
        //ensureCapacity() - Increase capacity if needed
        sb.ensureCapacity(minimumCapacity: 50);
        System.out.println("8. Capacity after ensureCapacity(50): " + sb.capacity());
        //length() - Current number of characters
        System.out.println("9. Length of buffer: " + sb.length());
        //setLength() - Truncate or extend the length
        sb.setLength(newLength: 8);
        System.out.println("10. After setLength(8): '" + sb + "'");
        //charAt() - Get character at specific index
        sb.setLength(newLength: 0);
```

class StringMethodsDemo.java 1 | class StringBufferMethodsDemo.java 1

C: > Users > KALYAN > Desktop > SEM3 > java > class StringBufferMethodsDemo.java > StringBufferMethodsDemo > main(String[])

```
1   class StringBufferMethodsDemo
3       public static void main(String[] args)
36          sb.setLength(newLength: 0);
37          sb.append(str: "JavaStringBuffer");
38          System.out.println("11. Character at index 4: " + sb.charAt(index: 4));
39          //setCharAt() - Change a character at a specific index
40          sb.setCharAt(index: 0, ch: 'K');
41          System.out.println("12. After setCharAt(0, 'K'): " + sb);
42          //substring() - Extract part of the sequence
43          System.out.println("13. substring(4): " + sb.substring(start: 4));
44          System.out.println(" substring(4, 10): " + sb.substring(start: 4, end: 10));
45          //deleteCharAt() - Remove character at given index
46          sb.deleteCharAt(index: 0);
47          System.out.println("14. After deleteCharAt(0): " + sb);
48          //append() with multiple types (method overloading)
49          sb.append(str: " Example ").append(i: 2025).append(c: ' ').append(b: true);
50          System.out.println("15. After multiple appends: " + sb);
51          //capacity grows automatically when exceeded
52          System.out.println("16. Capacity before large append: " + sb.capacity());
53          sb.append(str: " This text is to check automatic capacity increment.");
54          System.out.println(" Capacity after large append: " + sb.capacity());
55          //getChars() - Copy characters into an array
56          char[] arr = new char[10];
57          sb.getChars(srcBegin: 0, srcEnd: 10, arr, dstBegin: 0);
58          System.out.print(s: "17. getChars() output: ");
59          for (char c : arr)
60              System.out.print(c);
61          System.out.println();
62      }
63  }
```


class StringMethodsDemo.java 1
class StringBufferMethodsDemo.java 1
C: > Users > KALYAN > Desktop > SEM3 > java > class StringBufferMethodsDemo.java > StringBufferMethodsDemo > main(String[])
1 class StringBufferMethodsDemo
PROBLEMS 2
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\KALYAN> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,susp
DetailsInExceptionMessages' '-cp' 'C:\Users\KALYAN\AppData\Local\Temp\vscodesws_1e693\jdt_ws\jdt.ls-java-project\bin
StringBuffer Methods Demonstration
1. Initial Buffer: Java
2. After append(): Java Programming
3. After insert(): Java Language Programming
4. After replace(): Java Script Programming
5. After delete(): Script Programming
6. After reverse(): gnimmargorP tpircS
7. Current capacity: 42
8. Capacity after ensureCapacity(50): 86
9. Length of buffer: 18
10. After setLength(8): 'Script P'
11. Character at index 4: S
12. After setCharAt(0, 'K'): KavaStringBuffer
13. substring(4): StringBuffer
substring(4, 10): String
14. After deleteCharAt(0): avaStringBuffer
15. After multiple appends: avaStringBuffer Example 2025 true
16. Capacity before large append: 86
Capacity after large append: 86
17. getChars() output: avaStringB
PS C:\Users\KALYAN>