**day-017-string-assignment**

**GitHub Link:**

<https://github.com/irahuldutta02/pw-skills-jdsd-assignments/tree/main/day-017-string-assignment/codes>

**1. What is Mutable String in Java Explain with an example**

Mutable means changing over time or that can be changed. In a mutable string, we can change the value of the string and JVM doesn’t create a new object. In a mutable string, we can change the value of the string in the same object.

To create a mutable string in java, Java has two classes StringBuffer and StringBuilder where the String class is used for the immutable string.

public class MutableString

{

public static *void* main (String[] *args*)

{

StringBuffer str1 = **new** StringBuffer("JavaGoal");

StringBuilder str2 = **new** StringBuilder("Learning");

System.out.println("Value of str1 before change :" + str1);

System.out.println("Value of str2 before change :" + str2);

str1.append(".com");

str2.append(" website");

System.out.println("Value of str1 after change :" + str1);

System.out.println("Value of str2 after change :" + str2);

}

}

**2. WAP to reverse a String**

**Input: "PWSKILLS"**

**Output: "SLLIKSPW"**

Question\_02.java

// 2. WAP to reverse a String

// Input: "PWSKILLS"

// Output: "SLLIKSPW"

class Question\_02 {

static String revString(String *str*) {

String nstr = "";

for (*int* i = 0; i < *str*.length(); i++) {

nstr = *str*.charAt(i) + nstr;

}

return nstr;

}

public static *void* main(String[] *args*) {

String str = "PWSKILLS";

String s1 = str.substring(0, 2);

String s2 = str.substring(2, 8);

System.out.println("Old String : " + str);

String nstr = revString(s2) + s1;

System.out.println("New String : " + nstr);

}

}

Output :



**3. WAP to reverse a sentence while preserving the position**

**Input: Think Twice**

**Output: "kniht eciwt"**

Question\_03.java

// 3. WAP to reverse a sentence while preserving the position

// Input: Think Twice

// Output: "kniht eciwt"

public class Question\_03 {

static String revString(String *str*) {

String nstr = "";

for (*int* i = 0; i < *str*.length(); i++) {

nstr = *str*.charAt(i) + nstr;

}

return nstr;

}

public static *void* main(String[] *args*) {

String str = "Think Twice", nstr = "";

str = str.toLowerCase();

String[] sArr = str.split(" ");

System.out.println("Original String : " + str);

for (*int* i = 0; i < sArr.length; i++) {

nstr = nstr + " " + revString(sArr[i]);

}

System.out.println("Reversed wString : " + nstr);

}

}

Output :



**4. WAP to sort a String Alphabetically**

Question\_04.java

// WAP to sort a String Alphabetically

import java.util.Arrays;

public class Question\_04 {

public static *void* main(String *args*[]) {

String str = "Rahul";

str = str.toLowerCase();

*char* charArray[] = str.toCharArray();

Arrays.sort(charArray);

System.out.println(**new** String(charArray));

}

}

Output :

