Strings in Java

Assignment Solutions

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

Ans: The String class is immutable, which means that once a String object is created, its value cannot be changed. However, there are other classes in Java, such as StringBuilder and StringBuffer, which are mutable and can be used to manipulate strings.
Here's an example of using a mutable string in Java with the
StringBuilder class:
StringBuilder sb = new StringBuilder("Hello");
sb.append(" World"); // append method adds the string " World" to the end of sb
System.out.println(sb); // prints "Hello World"
In this example, we create a StringBuilder object sb with an initial value of "Hello". We then use the append method to add the string " World" to the end of sb. Finally, we print the value of sb, which is "Hello World".

2. WAP to reverse a String

Input: "PWSKILLS"

Output: "SLLIKSPW"

Ans: public class StringReverser {

```
    public static void main(String[] args) {
        String str = "PWSKILLS";
        String result = reverseString(str);
        System.out.println("The original string is: " + str);
        System.out.println("The reversed string is: " + result);
    }

    public static String reverseString(String str) {
        StringBuilder sb = new StringBuilder(str);
        sb.reverse();
        return sb.toString();
    }
}
```

Output: The original string is: PWSKILLS
        The reversed string is: SLLIKSWP

3. WAP to reverse a sentence while preserving the position

Input: Think Twice

Output: "knihT eciwT"

Ans: public class SentenceReverser1 {

```
    public static void main(String[] args) {
                String str1="Think Twice";
                String str2="";
```

```
                String sarr[]=str1.split(" ");
                for(String elem: sarr)
                {

                        for(int i=elem.length()-1;i>=0;i--)
                        {
                                str2=str2+elem.charAt(i);
                        }
                        str2=str2 +" ";
                }
                System.out.println(str2);
        }
}
```

Output: knihT eciwTg is: SLLIKSWP

4. WAP to sort a String Alphabetically

Ans:
```
import java.util.Arrays;
    class SortingOrder {
    static void sortString(String str) {
    char []arr = str.toCharArray();
    Arrays.sort(arr);
    System.out.print(String.valueOf(arr));
      }
      public static void main(String[] args) {
      String s = "aabbzzncedt";
      sortString(s);
      }
    }
```

Output: aabbcdentzz