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

Ans. The String which is created and can be changed or updated later.

```
StringBuilder mutableString = new StringBuilder("Hello");
System.out.println("Original string: " + mutableString);

// Appending to the string
mutableString.append(" World!");
System.out.println("After appending: " + mutableString);

// Inserting into the string
mutableString.insert(5, ", dear");
System.out.println("After inserting: " + mutableString);

// Deleting from the string
mutableString.delete(5, 11);
System.out.println("After deleting: " + mutableString);

// Updating the character at a specific index
mutableString.setCharAt(0, 'h');
System.out.println("After updating: " + mutableString);
```

2. **WAP to reverse a string**
Input: “PWSKILLS”
Output: “SLLIKSWP”

Ans.

```
public class Solution
{
   public static void main(String[] args)
        {
      String input = "PWSKILLS";
      String reversed = reverseString(input);
      System.out.println("Input: " + input);
      System.out.println("Reversed: " + reversed);
   }

   public static String reverseString(String str)
        {
      StringBuilder reversedString = new StringBuilder();
```

```
        for (int i = str.length() - 1; i >= 0; i--)
          {
           reversedString.append(str.charAt(i));
        }
        return reversedString.toString();
    }
}
```

3. **WAP to reverse a sentence while preserving the position**
Input: Think Twice
Output: “kniht eciwt”

Ans.

```
public class Solution
{
    public static void main(String[] args)
{
        String input = "Think Twice";
        String reversedSentence = reverseSentence(input);
        System.out.println("Input: " + input);
        System.out.println("Reversed: " + reversedSentence);
    }

    public static String reverseSentence(String sentence)
{
        String[] words = sentence.split(" ");
        StringBuilder reversedSentence = new StringBuilder();

        for (String word : words)
{
            StringBuilder reversedWord = new StringBuilder(word).reverse();
            reversedSentence.append(reversedWord).append(" ");
        }

        reversedSentence.setLength(reversedSentence.length() - 1); // Remove the
trailing space
        return reversedSentence.toString();
    }
}
```

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

Ans.

```
import java.util.Arrays;

public class Solution
{
   public static void main(String[] args)
{
      String input = "fedcba";
      String sortedString = sortString(input);
      System.out.println("Input: " + input);
      System.out.println("Sorted: " + sortedString);
   }

   public static String sortString(String str)
{
      char[] charArray = str.toCharArray();
      Arrays.sort(charArray);
      return new String(charArray);
   }
}
```