**Assignment 9.1**

1. **Write a program to reverse a String without using the inbuilt method.**

**public class ReverseString {**

**public static String reverseString(String input) {**

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

**int left = 0;**

**int right = charArray.length - 1;**

**while (left < right) {**

**char temp = charArray[left];**

**charArray[left] = charArray[right];**

**charArray[right] = temp;**

**left++;**

**right--;**

**}**

**return String.valueOf(charArray);**

**}**

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

**String str1 = "hello";**

**String reversedStr1 = reverseString(str1);**

**System.out.println(reversedStr1);**

**String str2 = "world";**

**String reversedStr2 = reverseString(str2);**

**System.out.println(reversedStr2);**

**}**

**}**

1. **Write a java program to know if it is palindrome or not.**

**public class PalindromeChecker**

**{**

**public static boolean isPalindrome(String input)**

**{**

**input = input.replaceAll("\\s+", "").toLowerCase();**

**int left = 0;**

**int right = input.length() - 1;**

**while (left < right) {**

**if (input.charAt(left) != input.charAt(right))**

**{**

**return false;**

**}**

**left++;**

**right--;**

**}**

**return true;**

**}**

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

**{**

**String str1 = "level";**

**System.out.println(isPalindrome(str1));**

**String str2 = "Hello";**

**System.out.println(isPalindrome(str2));**

**String str3 = "A man a plan a canal Panama";**

**System.out.println(isPalindrome(str3));**

**String str4 = "12321";**

**System.out.println(isPalindrome(str4));**

**}**

**}**

1. **Write a java program to convert uppercase to lowercase and vice versa**

**public class CaseConverter {**

**public static String convertCase(String input) {**

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

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

**char c = charArray[i];**

**if (Character.isUpperCase(c)) {**

**charArray[i] = Character.toLowerCase(c);**

**} else if (Character.isLowerCase(c)) {**

**charArray[i] = Character.toUpperCase(c);**

**}**

**}**

**return String.valueOf(charArray);**

**}**

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

**String str1 = "Hello World";**

**System.out.println(convertCase(str1));**

**String str2 = "Java Programming";**

**System.out.println(convertCase(str2));**

**}**

**}**

1. **Write a java program to remove particular character from string**

**public class CharacterRemover {**

**public static String removeCharacter(String input, char character) {**

**String result = input.replace(String.valueOf(character), "");**

**return result;**

**}**

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

**String str1 = "Hello World";**

**char charToRemove1 = 'o';**

**System.out.println(removeCharacter(str1, charToRemove1));**

**String str2 = "Java Programming";**

**char charToRemove2 = 'a';**

**System.out.println(removeCharacter(str2, charToRemove2));**

**}**

**}**

1. **Write a java program to find the index of substring**

**public class SubstringFinder {**

**public static int findSubstringIndex(String input, String substring) {**

**// Use the indexOf method to find the index of the substring**

**int index = input.indexOf(substring);**

**return index;**

**}**

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

**String str = "Hello World";**

**String substring = "World";**

**int index = findSubstringIndex(str, substring);**

**System.out.println("Substring index: " + index); // Output: Substring index: 6**

**}**

**}**

**Assignments 9.2**

1. **What is a String in Java?**

In Java, a String is a built-in class that represents a sequence of characters. It is one of the most commonly used classes in Java and is part of the java.lang package, which is automatically imported into every Java program.

1. **Types of String in Java are?**

* Mutable String
* Immutable String

1. **How many can you create String objects in java?**

By using new Keyword

1. **What is a String Constant Pool?**

The String Constant Pool, also known as the String Pool, is a special area of memory in the Java heap where string literals are stored. It is a part of the Java Runtime Environment (JRE) and is used to optimise memory usage by reusing string literals with the same value.

1. **What do you mean by mutable and immutable Strings**

* Mutable Strings can Change
* Immutable Strings cannot be Changed

1. **Where exactly is String constant pool located**

Heap Area

**Assignment 9.2**

1. **What is Mutable String With Example**

In Java, the String class is immutable, meaning that once a String object is created, its value cannot be changed. However, there is a mutable alternative to String called StringBuilder (or StringBuffer for thread-safe operations), which allows you to modify the contents of a string dynamically.

public class MutableStringExample {

public static void main(String[] args) {

StringBuilder sb = new StringBuilder("Hello");

// Appending to the string

sb.append(", World!");

System.out.println(sb); // Output: Hello, World!

// Modifying characters at specific indices

sb.setCharAt(6, '-');

System.out.println(sb); // Output: Hello- World!

// Inserting a substring at a specific index

sb.insert(5, " there");

System.out.println(sb); // Output: Hello there- World!

// Deleting characters from the string

sb.delete(5, 11);

System.out.println(sb); // Output: Hello- World!

}

}

1. **WAP to reverse a String**

**Input: PWSKILLS**

**Output: SLLIKSWP**

public class StringReversal {

public static String reverseString(String input) {

StringBuilder reversed = new StringBuilder();

// Iterate through the characters of the input string in reverse order

for (int i = input.length() - 1; i >= 0; i--) {

reversed.append(input.charAt(i));

}

return reversed.toString();

}

public static void main(String[] args) {

String input = "PWSKILLS";

String reversedString = reverseString(input);

System.out.println("Input: " + input);

System.out.println("Output: " + reversedString);

}

}

1. **WAP to reverse a sentence while Preserving the position**

**Input: Think Twice**

**Output: knihT eciwT**

public class SentenceReversal {

public static String reverseSentence(String sentence) {

String[] words = sentence.split(" ");

StringBuilder reversedSentence = new StringBuilder();

// Reverse each word and append to the reversed sentence

for (String word : words) {

StringBuilder reversedWord = new StringBuilder(word);

reversedWord.reverse();

reversedSentence.append(reversedWord).append(" ");

}

// Remove the trailing space and return the reversed sentence

return reversedSentence.toString().trim();

}

public static void main(String[] args) {

String sentence = "Think Twice";

String reversedSentence = reverseSentence(sentence);

System.out.println("Input: " + sentence);

System.out.println("Output: " + reversedSentence);

}

}

1. **WAP to sort a String**

**import java.util.Arrays;**

**public class StringSort {**

**public static String sortString(String input) {**

**// Convert the string to an array of characters**

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

**// Sort the array of characters**

**Arrays.sort(charArray);**

**// Convert the sorted array back to a string**

**String sortedString = new String(charArray);**

**return sortedString;**

**}**

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

**String input = "programming";**

**String sortedString = sortString(input);**

**System.out.println("Input: " + input);**

**System.out.println("Sorted String: " + sortedString);**

**}**

**}**

**Assignment 9.4**

1. **Write a simple String Program to take input from user**

import java.util.Scanner;

public class UserStringInput {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Prompt the user to enter a string

System.out.print("Enter a string: ");

String inputString = scanner.nextLine();

}

}

1. **How do you Concatenate 2 Strings**

Using concat() method

String s1=”Naman”;

String s2=”Gill”;

String s3=s1.concat(s2);

1. **How do you find the length of string**

Using length() method

String s1=”Naman”;

Int l=s1.length();

1. **How to compare 2 Strings in java**

Using equals() method

String s1=”Naman”;

String s2=”Gill”;

boolean b=s1.equals(s2);

1. **Write a program to find length of "refrigerator"**

public class StringLength {

public static void main(String[] args) {

String word = "refrigerator";

int length = word.length();

System.out.println("The length of the string \"" + word + "\" is: " + length);

}

}

1. **WAP to check if e is present in umbrella**

public class CharacterCheck {

public static void main(String[] args) {

String word = "umbrella";

char target = 'e';

boolean isPresent = false;

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

if (word.charAt(i) == target) {

isPresent = true;

break;

}

}

if (isPresent) {

System.out.println("'" + target + "' is present in the string \"" + word + "\"");

} else {

System.out.println("'" + target + "' is not present in the string \"" + word + "\"");

}

}

}

1. **Write a program to remove all consonants from the String “Hello, have a good day”**

**public class RemoveConsonants {**

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

**String sentence = "Hello, have a good day";**

**String result = removeConsonants(sentence);**

**System.out.println("Original Sentence: " + sentence);**

**System.out.println("Result: " + result);**

**}**

**public static String removeConsonants(String sentence) {**

**StringBuilder result = new StringBuilder();**

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

**char ch = sentence.charAt(i);**

**if (isVowel(ch) || Character.isWhitespace(ch)) {**

**result.append(ch);**

**}**

**}**

**return result.toString();**

**}**

**public static boolean isVowel(char ch) {**

**ch = Character.toLowerCase(ch);**

**return ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u';**

**}**

**}**