Week-1 : Java Lab programs

Name : BAKKACHENNAIAHGARI SRI SARANYA

Reg no : RA2411028010119

Sec : V1

-----------------------------------------------------------------------------------------------------------------------------

1.Write a program to find and return the length of a string without using the length() method.

Program:

import java.util.Scanner;

public class StringLengthFinder {

    public static int getLength(String str) {

        int count = 0;

        try {

            while (true) {

                str.charAt(count);

                count++;

            }

        } catch (Exception e) {

            return count;

        }

    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

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

        String input = scanner.next();

        int customLength = getLength(input);

        int builtInLength = input.length();

        System.out.println("Length (custom method): " + customLength);

        System.out.println("Length (built-in method): " + builtInLength);

        scanner.close();

    }

}

Output:

Enter a string: Hlo World

Length (custom method): 3

Length (built-in method): 3

2. Write a program to split the text into words, compare the result with the split() method and

display the result.

Program:

import java.util.Scanner;

public class CustomSplit {

    public static int getLength(String str) {

        int count = 0;

        try {

            while (true) {

                str.charAt(count);

                count++;

            }

        } catch (Exception e) {

            return count;

        }

    }

    public static String[] customSplit(String str) {

        int length = getLength(str);

        int spaceCount = 0;

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

            if (str.charAt(i) == ' ') {

                spaceCount++;

            }

        }

        int[] spaceIndexes = new int[spaceCount + 2];

        spaceIndexes[0] = -1;

        int index = 1;

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

            if (str.charAt(i) == ' ') {

                spaceIndexes[index++] = i;

            }

        }

        spaceIndexes[index] = length;

        String[] words = new String[spaceCount + 1];

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

            int start = spaceIndexes[i] + 1;

            int end = spaceIndexes[i + 1];

            StringBuilder word = new StringBuilder();

            for (int j = start; j < end; j++) {

                word.append(str.charAt(j));

            }

            words[i] = word.toString();

        }

        return words;

    }

    public static boolean compareArrays(String[] a, String[] b) {

        if (a.length != b.length) return false;

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

            if (!a[i].equals(b[i])) return false;

        }

        return true;

    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

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

        String input = scanner.nextLine();

        String[] builtInSplit = input.split(" ");

        String[] customSplit = customSplit(input);

        boolean isSame = compareArrays(builtInSplit, customSplit);

        System.out.println("\nBuilt-in split:");

        for (String word : builtInSplit) {

            System.out.println(word);

        }

        System.out.println("\nCustom split:");

        for (String word : customSplit) {

            System.out.println(word);

        }

        System.out.println("\nAre both splits equal? " + isSame);

        scanner.close();

    }

}

Output:  
Enter a sentence: My self Hlo Wor

ld

Built-in split:

My

self

Hlo

World

Custom split:

My

self

Hlo

World

Are both splits equal? true

3. Write a program to split the text into words and return the words along with their lengths in a

2D array/

Program:

import java.util.Scanner;

public class WordLengthTable {

    public static int getLength(String str) {

        int count = 0;

        try {

            while (true) {

                str.charAt(count);

                count++;

            }

        } catch (Exception e) {

            return count;

        }

    }

    public static String[] customSplit(String str) {

        int length = getLength(str);

        int spaceCount = 0;

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

            if (str.charAt(i) == ' ') {

                spaceCount++;

            }

        }

        int[] spaceIndexes = new int[spaceCount + 2];

        spaceIndexes[0] = -1;

        int index = 1;

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

            if (str.charAt(i) == ' ') {

                spaceIndexes[index++] = i;

            }

        }

        spaceIndexes[index] = length;

        String[] words = new String[spaceCount + 1];

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

            int start = spaceIndexes[i] + 1;

            int end = spaceIndexes[i + 1];

            StringBuilder word = new StringBuilder();

            for (int j = start; j < end; j++) {

                word.append(str.charAt(j));

            }

            words[i] = word.toString();

        }

        return words;

    }

    public static String[][] getWordsWithLengths(String[] words) {

        String[][] result = new String[words.length][2];

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

            result[i][0] = words[i];

            result[i][1] = String.valueOf(getLength(words[i]));

        }

        return result;

    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

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

        String input = scanner.nextLine();

        String[] words = customSplit(input);

        String[][] wordData = getWordsWithLengths(words);

        System.out.println("\nWord\tLength");

        System.out.println("---------------");

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

            String word = wordData[i][0];

            int length = Integer.parseInt(wordData[i][1]);

            System.out.println(word + "\t" + length);

        }

        scanner.close();

    }

}

Output:

Enter a sentence: Hlo world

Word Length

---------------

Hlo 3

world 5

0

4. Write a program to find vowels and consonants in a string and display the count of Vowels

and Consonants in the string.

Program:

import java.util.Scanner;

public class VowelConsonantChecker {

    public static String getCharType(char c) {

        if (c >= 'A' && c <= 'Z') {

            c = (char)(c + 32);

        }

        if (c >= 'a' && c <= 'z') {

            if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u') {

                return "Vowel";

            } else {

                return "Consonant";

            }

        } else {

            return "Not a Letter";

        }

    }

    public static String[][] analyzeString(String str) {

        int len = 0;

        try {

            while (true) {

                str.charAt(len);

                len++;

            }

        } catch (Exception e) {}

        String[][] result = new String[len][2];

        for (int i = 0; i < len; i++) {

            char ch = str.charAt(i);

            result[i][0] = String.valueOf(ch);

            result[i][1] = getCharType(ch);

        }

        return result;

    }

    public static void displayResult(String[][] data) {

        System.out.println("\nCharacter\tType");

        System.out.println("-------------------------");

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

            System.out.println(data[i][0] + "\t\t" + data[i][1]);

        }

    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

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

        String input = scanner.nextLine();

        String[][] analysis = analyzeString(input);

        displayResult(analysis);

        scanner.close();

    }

}

Output:

Enter a string: Welcome to step p

rograms

Character Type

-------------------------

W Consonant

e Vowel

l Consonant

c Consonant

o Vowel

m Consonant

e Vowel

Not a Letter

t Consonant

o Vowel

Not a Letter

s Consonant

t Consonant

e Vowel

p Consonant

Not a Letter

p Consonant

r Consonant

o Vowel

g Consonant

r Consonant

a Vowel

m Consonant

s Consonant.

5. Write a program to take user input for the age of all 10 students in a class and check

whether the student can vote depending on his/her age is greater or equal to 18.

Program:

import java.util.Scanner;

import java.util.Random;

public class VotingEligibility {

    public static int[] generateRandomAges(int n) {

        Random rand = new Random();

        int[] ages = new int[n];

        for (int i = 0; i < n; i++) {

            ages[i] = rand.nextInt(90) + 10; // Generates 2-digit age between 10 and 99

        }

        return ages;

    }

    public static String[][] checkVotingEligibility(int[] ages) {

        String[][] result = new String[ages.length][2];

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

            result[i][0] = String.valueOf(ages[i]);

            if (ages[i] < 0) {

                result[i][1] = "false";

            } else if (ages[i] >= 18) {

                result[i][1] = "true";

            } else {

                result[i][1] = "false";

            }

        }

        return result;

    }

    public static void displayTable(String[][] data) {

        System.out.println("\nStudent\tAge\tCan Vote");

        System.out.println("------------------------------");

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

            System.out.println((i + 1) + "\t" + data[i][0] + "\t" + data[i][1]);

        }

    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter number of students: ");

        int n = scanner.nextInt();

        int[] studentAges = generateRandomAges(n);

        String[][] result = checkVotingEligibility(studentAges);

        displayTable(result);

        scanner.close();

    }

}

Output:

Enter number of students: 5

Student Age Can Vote

------------------------------

1 44 true

2 74 true

3 69 true

4 35 true

5 45 true

6. Write a program to trim the leading and trailing spaces from a string using the charAt()

method

Program:

import java.util.Scanner;

public class CustomTrim {

    public static int[] findTrimIndexes(String str) {

        int start = 0;

        int end = 0;

        int len = 0;

        try {

            while (true) {

                str.charAt(len);

                len++;

            }

        } catch (Exception e) {

        }

        while (start < len && str.charAt(start) == ' ') {

            start++;

        }

        end = len - 1;

        while (end >= 0 && str.charAt(end) == ' ') {

            end--;

        }

        return new int[] {start, end};

    }

    public static String customSubstring(String str, int start, int end) {

        StringBuilder sb = new StringBuilder();

        for (int i = start; i <= end; i++) {

            sb.append(str.charAt(i));

        }

        return sb.toString();

    }

    public static boolean compareStrings(String a, String b) {

        int len1 = 0, len2 = 0;

        try {

            while (true) {

                a.charAt(len1);

                len1++;

            }

        } catch (Exception e) {}

        try {

            while (true) {

                b.charAt(len2);

                len2++;

            }

        } catch (Exception e) {}

        if (len1 != len2) return false;

        for (int i = 0; i < len1; i++) {

            if (a.charAt(i) != b.charAt(i)) return false;

        }

        return true;

    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter a string with leading and trailing spaces:\n> ");

        String input = scanner.nextLine();

        int[] indexes = findTrimIndexes(input);

        String customTrimmed = customSubstring(input, indexes[0], indexes[1]);

        String builtInTrimmed = input.trim();

        System.out.println("\nCustom Trimmed String: '" + customTrimmed + "'");

        System.out.println("Built-in Trimmed String: '" + builtInTrimmed + "'");

        boolean isSame = compareStrings(customTrimmed, builtInTrimmed);

        System.out.println("\nAre both trimmed results same? " + isSame);

        scanner.close();

    }

}

Output:

Enter a string with leading and trailing spaces:

> Hlo World

Custom Trimmed String: 'Hlo World'

Built-in Trimmed String: 'Hlo World'

Are both trimmed results same? true