using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace AccentureTest

{

class Program

{

////////////////////////////////////////////////////////////////////////////////////////////////////

/// <summary> Main entry-point for this application. </summary>

///

/// <remarks> Amiya, 22-08-2019. </remarks>

///

/// <param name="args"> An array of command-line argument strings. </param>

////////////////////////////////////////////////////////////////////////////////////////////////////

static void Main(string[] args)

{

#region Qns1

// Difference between ‘this’ and ‘let’ and when to use what, explain with examples.

// let varName = “check the scope”;

// let is used to declare variable in Angular Type script, its scope is within the block.

// outside block it will not be accessable.

// console.log(window.varName); //undefined

// let variables are usually used when there is a limited use of those variables.

//for (let i = 0; i < 10; i++)

//{

// console.log(i); //i is visible thus is logged in the console as 0,1,2,....,9

//}

//console.log(i); //throws an error as "i is not defined" because i is not visible

#endregion

#region QNS2

//Write a function which accepts an array of strings and returns a map of character to count of the

//character(including spaces and special characters).

//Eg: input: [‘hello world’, ‘hello world’] output: { h: 2, e: 2, l: 6, ‘ ‘: 1, o: 4 …… }

String inputString = "amiya kumar rout";

//Approach 1 using Dictionary

OccurenceOfCharInString.charCountUsingDictionary(inputString);

//Approach 2 using LINQ

OccurenceOfCharInString.charCountUsingDictionary(inputString);

#endregion

#region QNS3

//Write a function to return all the keys present in an object at any level.

//Eg: input: { a: 5, b: { c: { d: 10 } } }

//output: [a, b, c, d]

// Need some more time to attend this question. The apporach will be using recursive function to get the result.

#endregion

#region Qns3

//Write a function to reverse every word of a string.

//Using split and reverse

ReverseWords.ReverseString();

// Without using split and reverse and in O(1) space(since strings are immutable, assume that the input is a character array).

#endregion

#region Ans4

//Write a function to remove duplicate strings from an array of strings

string[] strInputArray = { "amiya", "rout", "kumar", "rout", "amiya" };

RemoveDuplicateString.removeDups(strInputArray);

#endregion

Console.ReadLine();

}

}

////////////////////////////////////////////////////////////////////////////////////////////////////

/// <summary> An occurence of character in string. </summary>

/// This will find the occurances of character in the input string.

/// <remarks> Amiya, 22-08-2019. </remarks>

////////////////////////////////////////////////////////////////////////////////////////////////////

class OccurenceOfCharInString

{

#region UsingDictionary

internal static void charCountUsingDictionary(string str)

{

Dictionary<char, int> dict = new Dictionary<char, int>();

int max = 0;

foreach (char c in str.ToCharArray())

{

int i;

dict.TryGetValue(c, out i);

i++;

if (i > max)

{

max = i;

}

dict[c] = i;

}

foreach (KeyValuePair<char, int> chars in dict)

{

Console.WriteLine("{0}: {1}", chars.Key, chars.Value);

}

}

#endregion

#region UsingLInq

internal static void charCountUsingLINQ(string inputString)

{

var charGroups = (from c in inputString

group c by c into g

select new

{

cname = g.Key,

count = g.Count(),

}).OrderByDescending(c => c.count);

foreach (var group in charGroups.OrderBy(c => c.cname))

{

Console.WriteLine(group.cname + ": " + group.count);

}

}

#endregion

}

class FindKeyInObject

{

internal static void FindObject()

{

}

}

////////////////////////////////////////////////////////////////////////////////////////////////////

/// <summary> A reverse words. </summary>

/// this will print string in reverse order.

/// <remarks> Amiya, 22-08-2019. </remarks>

////////////////////////////////////////////////////////////////////////////////////////////////////

public class ReverseWords

{

internal static void ReverseString()

{

string[] inputStringArray = "Please find below coding problem statement for technical role in Accenture".Split(' ');

// Apporach 1: using loop & string operation

string outputString = "";

for (int i = inputStringArray.Length - 1; i >= 0; i--)

{

outputString += inputStringArray[i] + " ";

}

Console.Write("Reversed String:\n");

Console.Write(outputString.Substring(0, outputString.Length - 1));

// Approach 2 : Using reverse function

var revereseString = inputStringArray.Reverse();

// Approach 3 : Without using split and reverse and in O(1) space

// (since strings are immutable, assume that the input is a character array).

//

string inputString = "Please find below coding problem statement for technical role in Accenture";

char[] inputCharArray = inputString.ToCharArray();

var reverseCharArray = inputCharArray.Reverse();

// I was trying to complet using sorted list.

}

}

////////////////////////////////////////////////////////////////////////////////////////////////////

/// <summary> A remove duplicate string. </summary>

/// This will accept input string array and display after removing duplicate string.

/// <remarks> Amiya, 22-08-2019. </remarks>

////////////////////////////////////////////////////////////////////////////////////////////////////

class RemoveDuplicateString

{

/\* Method removes duplicate characters from the string

This function work in-place and fills null characters

in the extra space left \*/

internal static void removeDups(string[] strInputArray)

{

string[] strAfterRemoveDuplicate = strInputArray.Distinct().ToArray();

foreach (string str in strAfterRemoveDuplicate)

Console.Write(str + " ");

Console.ReadLine();

}

}

}