VISUAL PROGRAMMING LAB 02
WORKBOOK

Instructor
Md. Rashedul Islam

SUBMIT WITHIN: 12:45 PM (TODAY, 21 September 2022)

SEND PROGRAMS IMAGES TO

rashed.class.official17@gmail.com

Email Subject: LAB02_CSC440_ID_FALL2022

1. **Demonstrate example of nested if else statement**

Syntax:

```
if(test_condition1){
        //code section 1
        if(test_condition_a){
                //code section a
}
else{
        //code section b
}
}
else if(test_condition2){
{
        //code section 2
}
else if(test_condition3){
        //code section 3
}
...
else{
        //else code section
}
```

See the syntax above, here we are checking test_condition_a within the test_conditon1 block, if test_condition1 will be true, "code section a" will be executed.

```
// C# program to demonstrate example of
// nested if else statement
using System;
using System.IO;
using System.Text;

namespace IncludeHelp
{
    class Test
    {
        // Main Method
        static void Main(string[] args)
        {
            //input a character and check whether it is vowel or consonant
            //but before it - check input character is an aplhabet or not
            char ch;
            Console.Write("Enter a character: ");
            ch = Console.ReadLine()[0];

            //check ch is an alphabet or not
            if((ch>='A' && ch<='Z') || (ch>='a' && ch<='z')){
                Console.WriteLine("{0} is a valid alphabet", ch);
                //checking for vowel or consonant
                if (ch == 'A' || ch == 'a' || ch == 'E' || ch == 'e' ||
                    ch == 'I' || ch == 'i' || ch == 'O' || ch == 'o' ||
                    ch == 'U' || ch == 'u')
                {
                    Console.WriteLine("{0} is a vowel", ch);
                }
                else
                {
                    Console.WriteLine("{0} is a consonant", ch);
                }
            }
            else{
                Console.WriteLine("{0} is not a valid alphabet",ch);
            }

            //hit ENTER to exit the program
            Console.ReadLine();
        }
    }
}
```

2. **Demonstrate example of nested conditional operator**

C# (or other programming languages also) allows to use a conditional/ternary operator within another conditional/ternary operator.

Syntax:

```
(logical_test1) ?
   ((logical_test2)? True_block : false_block) :  false_block_outer;
```

If logical_test1 is true then logical_test2 will be checked, if it is true then "true_block" executes, else "false_block" executes, and if logical_test1 is false then "false_block_outer" will be executed.

Note: Inner conditional operator can be used in any block as per the requirement.

C# example for nested conditional/ternary operator

```
// C# program to demonstrate example of
// nested conditional operator
using System;
using System.IO;
using System.Text;

namespace IncludeHelp
{
    class Test
    {
        // Main Method
        static void Main(string[] args)
        {
            //finding largest of three  numbers
            int a;
            int b;
            int c;

            //input numbers
            Console.Write("Enter first number : ");
            a = Convert.ToInt32(Console.ReadLine());
            Console.Write("Enter second number: ");
            b = Convert.ToInt32(Console.ReadLine());
            Console.Write("Enter third number : ");
            c = Convert.ToInt32(Console.ReadLine());

            //finding largest number
            int large = (a>b)?((a>c)?a:c):(b>c?b:c);

            Console.WriteLine("Largest number is {0}", large);

            //hit ENTER to exit the program
            Console.ReadLine();
        }
    }
}
```

3. **Program to check given strings are equal or not using equal to (==) operator**

C# code for string comparison

Here, we are asking for two strings input from the user and checking them whether they are equal or not using == operator and also ignoring the case.

```
using System;
using System.IO;
using System.Text;

namespace IncludeHelp
{
    class Test
    {
        // Main Method
        static void Main(string[] args)
        {
            string str1;
            string str2;

            //input strings
            Console.Write("Enter a string: ");
            str1 = Console.ReadLine();
            Console.Write("Enter another string: ");
            str2 = Console.ReadLine();

            //comparing strings
            if (str1 == str2)
                Console.WriteLine("\"{0}\" and \"{1}\" are equal", str1, str2);
            else
                Console.WriteLine("\"{0}\" and \"{1}\" are not equal", str1, str2);

            //another way
            if ((str1 == str2) == true)
                Console.WriteLine("\"{0}\" and \"{1}\" are equal", str1, str2);
            else
                Console.WriteLine("\"{0}\" and \"{1}\" are not equal", str1, str2);

            //comparing by ignoring the case
            //convert both strings in the same case
            //either in uppercase or lowercase
            Console.WriteLine("By ignoring case...");
            if(str1.ToUpper() == str2.ToUpper())
                Console.WriteLine("\"{0}\" and \"{1}\" are equal", str1, str2);
            else
                Console.WriteLine("\"{0}\" and \"{1}\" are not equal", str1, str2);

            //hit ENTER to exit the program
            Console.ReadLine();
        }
    }
}
```

4. STRING Class Operations

Demonstrate the example of Copy method of String class

String.Copy() Method

It is a method of String class which is used to create a new instance of String class copy value of already created String.

Syntax:

String String.Copy(String str);
Above method return newly created instance with value of passed string.

Example of String.Copy() in C#

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main()
        {
            String str1;
            String str2;

            Console.Write("Enter string : ");
            str1 = Console.ReadLine();
            str2 = String.Copy(str1);


            Console.WriteLine("Value is str1 : " + str1);
            Console.WriteLine("Value of str2 : " + str2);
        }
    }
}
```

How to trim leading spaces of string using String.TrimStart()

String.TrimStart()

This method returns string after removing the leading spaces.

Syntax:

String String.TrimStart();
Example:

Input string is: " This is a sample string "
Output string is: "This is a sample string "
Consider the program:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main()
        {
            String str1 = " This is a sample string  "; ;
            String str2;

            str2 = str1.TrimStart();

            Console.WriteLine("Trimmed string is:(" + str2+")");

        }
    }

}
```

How to pad string from right using String.PadRight()

String.PadRight()
Method returns padded string from right.

Syntax:

String String.PadRight(int totalLength, char ch);
Here,
totalLength : This parameter specifies total length of string after padding.
ch : This parameter specifies a character which will be used to pad string from right side.

Example:

Input string: "This is a sample string"
Padding string with '#' and total string length will be 30
Output string: "This is a sample string#######"
Explanation: Input string's length was 23 and to make it 30, program added 7 characters (#) in the end.

Consider the program:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main()
        {
            String str1 = "This is a sample string"; ;
            String str2;


            str2 = str1.PadRight(30, '#');

            Console.WriteLine("String after right padding:(" + str2+")");

        }
    }

}
```

Check whether string ends with given substring or not using String.EndsWith()

Given a string and we have to check whether it is ending with a given substring or not?

String.EndsWith()
This method checks whether string is ending with given substring or not?

Syntax:

Bool String.EndsWith(String subStr);

Here,

subStr is the substring to be checked.
Bool is the Boolean value, it's a return type of this method, if string is ending with substring subStr method will return true and if it is not ending with substring subStr method will return false.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main()
        {
            String str1;
            bool flag = false;

            Console.Write("Enter String : ");
            str1 = Console.ReadLine();

            flag = str1.EndsWith("india");

            if(flag==true)
                Console.WriteLine("String ends with india");
            else
                Console.WriteLine("String does not end with india");

        }
    }

}
```

How to replace a character with another character in a string

Given a string and we have to replace a character with another character in string.

String.Replace()
Method returns modified string after replacing a character with another.

Syntax:

String String.Replace(char oldChar, char newChar);
Here,
oldChar : is the character to be replaced.
newChar : is the character to replace.

Consider the program:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main()
        {
            String str1;
            String str2;

            Console.Write("Enter String : ");
            str1 = Console.ReadLine();

            str2 = str1.Replace('i','I');

            Console.WriteLine("String after replace method : " + str2);
        }
    }
}
```

Count the frequency of the specified word in the given string

Here we will create a program to count the frequency of the specified word in the given string.

Program:

```
//Program to count the frequency of the
//specified word in the given string.

using System;

class WordFrequency
{
    public static int CountWordFrequncy(string sentence, string word)
    {
        int cnt = 0;
        int i = 0;
        while ((i = sentence.IndexOf(word, i)) != -1)
        {
            i += word.Length;
            cnt++;
        }
        return cnt;
    }
    public static void Main()
    {
        string sentence;

        Console.Write("Enter the Sentence: ");
        sentence = Console.ReadLine();

        Console.WriteLine(CountWordFrequncy(sentence, "the"));
    }
}
```

Extract only numbers from a specified string using the Split() method

Here we extract number from a given string using the Split() method of Regex class with the help of regular expressions.

Program:

```
//C# program to extract only numbers from a
//specified string using Split() method

using System;
using System.Text.RegularExpressions;

class SplitDemo
{
    static void Main()
    {
        string[] numbers;
        string str = "Cow has 4 legs, one cow may produce approx 10 ltr milk per day";

        numbers = Regex.Split(str, @"\D+");

        Console.WriteLine("Numbers in given string:");
        foreach (string num in numbers)
        {
            Console.WriteLine(num);
        }
    }
}
```

Explanation:

Here, we created a SplitDemo class that contains the Main() method. The Main() method is the entry point of the program. Here we created a string str initialized with a sentence.

numbers = Regex.Split(str, @"\D+");
The Split() method extract data based on specified regular expression, here we extract only digits from the specified string. And then printed the extracted numbers using the "foreach" loop on the console screen.

Print the list of all possible substrings of a specified string

Here, we will find the all-possible substrings and then print them on the console screen.

Program:

```
//C# program to print the list of all
//possible substrings of a specified string.

using System;

class Demo
{
    static void GetSubStrings(string str)
    {
        int j=0;
        int i=0;

        Console.WriteLine("Possible sub-strings are :");
        for (i = 1; i <= str.Length; i++)
        {
            for (j = 0; j <= str.Length - i; j++)
            {
                Console.WriteLine(str.Substring(j, i));
            }
        }
    }
    public static void Main()
    {
        string str;

        Console.Write("Enter the String : ");
        str = Console.ReadLine();

        GetSubStrings(str);
    }
}
```

Explanation:

Here, we created a class Demo that contains two static methods GetSubstrings() and Main().

The GetSubstrings() method is used to find all possible substring based on a given string and then print them on the console screen.

The Main() method is the entry point for the program execution. Here we read a string and then find possible substrings using the GetSubstrings() method.

Program to count the lines in a given string

Here, we will count the lines in a specified string.

Program:

```
//C# program to count the lines in a given string.

using System;

class Program
{
    static int CountLines(string str)
    {
        int lines = 1;
        int index = 0;
        while (true)
        {
            index = str.IndexOf('\n', index);
            if (index < 0)
                break;
            lines++;
            index++;
        }
        return lines;
    }
    static void Main()
    {
        string str="Mumbai Indians\nDelhi Capitals\nRajsthan Royals";

        int lines = 0;

        Console.WriteLine("Lines:");
        Console.WriteLine(str);
        lines=CountLines(str);
        Console.WriteLine("Total lines in a string: "+lines);
    }

}
```

Explanation:

Here, we created a Program class that contains two static methods CountLines() and Main().

In the CountLines() method, we find the "\n" characters to count lines in a specified string using the IndexOf() method. This method returns the count of lines to the Main() method.

In the Main() method, we created the string str initialized with "Mumbai Indians\nDelhi Capitals\nRajsthan Royals" and then count lines and print the count on the console screen.

Observation: ***What if we convert \n to a space. Show the output for this observation and explain in the output in your own words.***