ASSIGNMENT 3:

Question 2(a). Read the string from the keyboard and write the length of the string with the label.

Code :

// See https://aka.ms/new-console-template for more information

using System;

public class practise

{

public static void Main()

{

int ans=0;

Console.Write("write your string : ");

string p = Console.ReadLine();

p += '\0'; //for null terminator;

int i = 0;

while (p[i] != '\0') {

ans++;

i++;

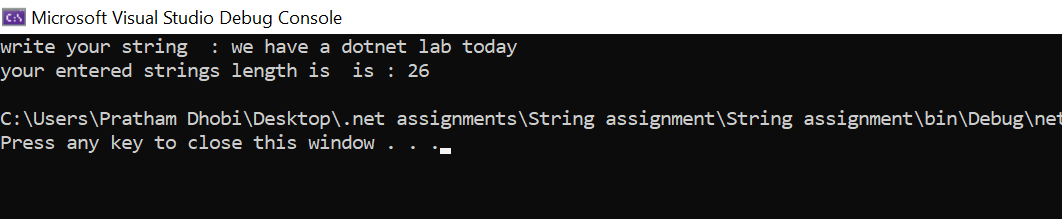
}

Console.Write("your entered strings length is is : {0}\n ", ans);

}

}

Output:



ASSIGNMENT 3:

Question 2(b). Read a sentence from a line of input, and print whether it represents a declarative sentence (i.e. ending in a period),interrogatory sentence (ending in a question mark),or an exclamation (ending in exclamation point) or is not a sentence (anything else).

Code :

// See https://aka.ms/new-console-template for more information

using System;

public class practise

{

public static void Main()

{

bool n;

do

{

Console.Write("Enter true to wrrite and check string else Enter anything : ");

n = bool.Parse(Console.ReadLine());

Console.Write("write your string : ");

string p = Console.ReadLine();

if (p.EndsWith('.'))

Console.Write("-->The wrriten string is a declarative sentence.");

else if (p.EndsWith('?'))

Console.Write("-->The wrriten string is a interrogatory sentence.");

else if (p.EndsWith('!'))

Console.Write("-->The wrriten string is a exclamation sentence.");

else

Console.Write("-->The wrriten string is not a sentence.");

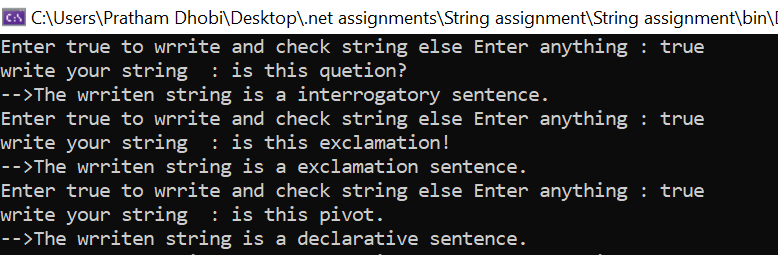
Console.WriteLine();

} while (n);

}

}

Output:



ASSIGNMENT 3:

Question 2(c). Read a whole name from a single line of user input . assume first and last names are separated by a space (no middle name).print last name first followed by the first name. for example , if the input Is “marcel proust”, the outputis “proust, marcel”.

Code:

// See https://aka.ms/new-console-template for more information

using System;

using System.Runtime.InteropServices;

public class practise

{

public static void Main()

{

string Reverse(string s,int x,int check)

{

string n="";

int j = x-1;

while (j >= check)

{

n += s[j];

j--;

}

return n;

}

Console.Write("write first and last name : ");

string name = Console.ReadLine();

int z = name.Length;

name = Reverse(name,z,0);

string ans="";

int p = 0;

int check = 0;

int space = 0;

for (int i = 0; i<name.Length; i++)

{

if (name[i]==' ' )

{

p = i;

ans+= Reverse(name,p,check);

check = i+1;

}

else if (i==name.Length-1)

{

p = i+1;

ans += Reverse(name, p, check);

}

if (name[i]==' ')

{ ans += ',';

ans += ' ';

}

}

Console.Write("The output of desired string is : ");

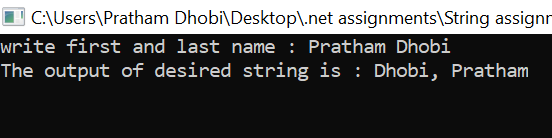
Console.Write(ans);

Console.ReadLine();

}

}

Output:



ASSIGNMENT 3:

Question 2(d). Improve the previous part ,so it also allows a single name without change. if there are two parts of the name ,it should work as in the original version.

Code:

// See https://aka.ms/new-console-template for more information

using System;

using System.Runtime.InteropServices;

public class practise

{

public static void Main()

{

string Reverse(string s,int x,int check)

{

string n="";

int j = x-1;

while (j >= check)

{

n += s[j];

j--;

}

return n;

}

int g = 0;

do

{

Console.Write("Do you want to take and write the name then press any digit greater than zero : ");

g = Convert.ToInt32(Console.ReadLine());

Console.Write("write first and last name : ");

string name = Console.ReadLine();

int z = name.Length;

name = Reverse(name, z, 0);

string ans = "";

int p = 0;

int check = 0;

int space = 0;

bool isSpace = false;

for (int i = 0; i < name.Length; i++)

{

if (name[i] == ' ')

{

isSpace = true;

p = i;

ans += Reverse(name, p, check);

check = i + 1;

}

else if (i == name.Length - 1)

{

if (isSpace == false)

{

int y = name.Length;

ans = Reverse(name, y, 0);

}

else

{

p = i + 1;

ans += Reverse(name, p, check);

}

}

if (name[i] == ' ')

{

ans += ',';

ans += ' ';

}

}

Console.Write("The output of desired string is : ");

Console.Write(ans);

Console.WriteLine();

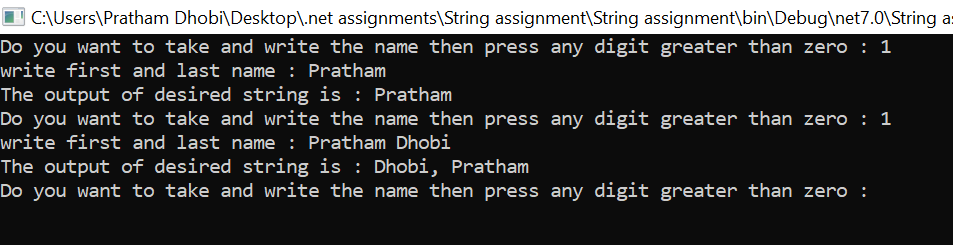
} while(g>0);

Console.ReadLine();

}

}

Output:



ASSIGNMENT 3:

Question 3. Enum Management Programs.

Code:

[Flags]

public enum Season

{

Spring= 0b\_0000\_0000,

Summer= 0b\_0000\_0001,

Autumn= 0b\_0000\_0010,

Winter= 0b\_0000\_0100

}

public class EnumConversionExample

{

public static void Main()

{

Season a = Season.Spring;

Console.WriteLine($"Integral value of {a} is {(int)a}"); // output: Integral value of Autumn is 2

Season b = Season.Summer;

Console.WriteLine($"Integral value of {b} is {(int)b}");

Season c= Season.Autumn;

Console.WriteLine($"Integral value of {c} is {(int)c}");

Season d = Season.Winter;

Console.WriteLine($"Integral value of {d} is {(int)d}");

var e = (Season)0;

Console.WriteLine($"who lies in {(int)e} index of enum {e}"); // output: Summer

var f = (Season)1;

Console.WriteLine($"who lies in {(int)f} index of enum {f}"); // output: 4

var g = (Season)2;

Console.WriteLine($"who lies in {(int)g} index of enum {g}");

var h = (Season)4;

Console.WriteLine($"who lies in {(int)h} index of enum {h}");

Season mine = Season.Summer | Season.Winter ;

Console.WriteLine($"my favorite season is {mine}");

Season mine1 = Season.Spring ;

Console.WriteLine($"i dont like {mine1}");

Season mine2 = Season.Autumn;

Console.WriteLine($"I am confused in {mine2}");

Console.ReadLine();

}

}

Output:

