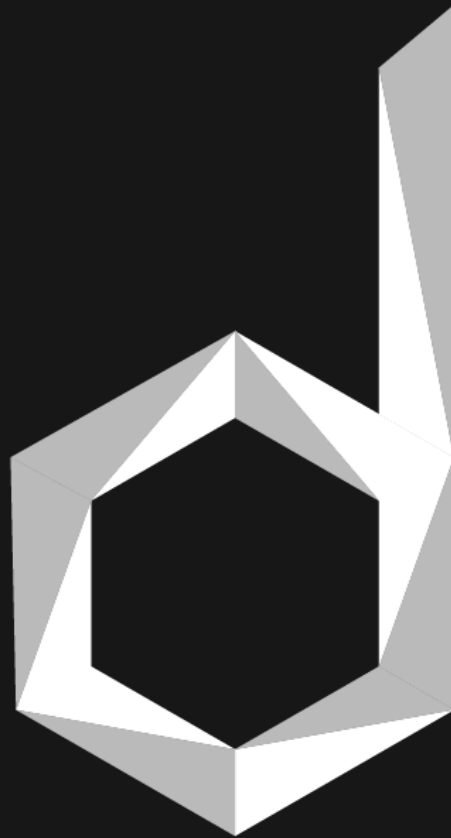


INGENIERÍA MECATRÓNICA



DI_CERO

DIEGO CERVANTES RODRÍGUEZ

INSTRUMENTACIÓN VIRTUAL

PYTHON 3.9.7, C# & LABVIEW

Abrir, Escribir y Leer un Archivo

Contenido

Instrucciones – Abrir, Escribir y Leer un Archivo:	2
Código Python – Visual Studio Code (Logo Azul):	2
Resultado del Código Python	3
Código C# (.Net Framework) – Visual Studio (Logo Morado):	4
Código con Resultado en Consola:	4
Resultado del Código C#	4
Diagrama LabVIEW:	5



Instrucciones – Abrir, Escribir y Leer un Archivo:

Genere un programa que escriba un mensaje y guárdelo en un archivo. Posteriormente, ábralo y lea su contenido. Por ejemplo, escriba la frase: ¡Hola mundo, cruel!, guárdela en un archivo y después recupérela e imprímala en la pantalla.

Pseudocódigo.

1. Escribir el mensaje en una variable del tipo string llamada stMessage.
2. Creación de un archivo llamado Mensaje.txt para escritura del mensaje.
3. Escribir el mensaje en el archivo.
4. Cerrar el archivo.
5. Abrir el archivo para su lectura.
6. Leer el contenido del archivo en la variable string llamada stRecoveredMessage.
7. Cerrar el archivo.
8. Imprimir en la pantalla la variable stRecoveredMessage.

Código Python – Visual Studio Code (Logo Azul):

```
# -*- coding: utf-8 -*-

#Comentario de una sola línea con el símbolo #, en Python para nada se deben poner acentos sino el programa
#puede fallar o imprimir raro en consola, la siguiente línea de código es para que no tenga error, pero aún
#así al poner un ángulo saldrá raro en consola, la línea debe ponerse tal cual como aparece y justo al inicio
#y cuando se vuelva a ejecutar el programa, se borrará lo que tenga escrito y escribirá algo nuevo.

#ABRIR, ESCRIBIR y LEER UN ARCHIVO: Genere un programa que escriba un mensaje y lo guarde en un archivo,
#posteriormente, ábralo y lea su contenido. Para probar el programa, escriba la frase: ¡Hola mundo, cruel!,
#en consola, luego guarde el mensaje en un archivo, después recupérela y finalmente imprímala en la pantalla.
#Con las siguientes líneas de código se abre un archivo, se escribe sobre él, se lee su contenido y se imprime
#en consola.

#INTRODUCIR UN MENSAJE EN CONSOLA:
#input(): Método que sirve para introducir un mensaje en consola, que en el programa se interpreta como un dato
#de tipo String.
stMessage = input("Ingresa el mensaje en consola: \n") #Variable que guarda un mensaje introducido en consola

#ABRIR y ESCRIBIR EN UN ARCHIVO:
#Variable que guarda el directorio y el nombre del archivo creado, se deben reemplazar los guiones\ por /
#Para leer una imagen o cualquier otro archivo se usa la dirección relativa o absoluta de un directorio:
# - Dirección relativa: Es una dirección que busca un archivo desde donde se encuentra la carpeta del
#   archivo python actualmente, esta se debe colocar entre comillas simples o dobles.
# - Dirección absoluta: Es una dirección que coloca toda la ruta desde el disco duro C o cualquier otro
```

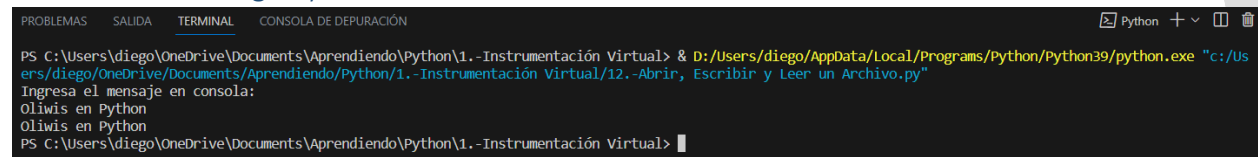
```

# que se esté usando hasta la ubicación del archivo, la cual se debe colocar entre comillas simples o
# dobles.
# .. : Significa que nos debemos salir de la carpeta donde nos encontramos actualmente.
# / : Sirve para introducirnos a alguna carpeta cuyo nombre se coloca después del slash.
# .ext : Se debe colocar siempre el nombre del archivo + su extensión.
filename = "Ejercicios Python con Archivos y Carpetas/12.-Abrir, Escribir y Leer un Archivo/Mensaje.txt"
#open(): Método que sirve para abrir un archivo cualquiera, para ello es necesario indicar dos parámetros,
#el primero se refiere a la ruta relativa o absoluta del archivo previamente creado y la segunda indica qué
#es lo que se va a realizar con él, el contenido del archivo se asigna a una variable.
# - w: Sirve para escribir en un archivo, pero borrará la información que previamente contenía el archivo.
# - a: Sirve para escribir en un archivo sin que se borre la info anterior del archivo, se llama append.
new_file = open(filename, 'w')
#var_file_open.write(): Método para colocar un string en un archivo previamente abierto con el método open().
new_file.write(stMessage) #Sirve para escribir el mensaje que ingresó el usuario en consola.
#var_file_open.close(): Método para cerrar un archivo previamente abierto con el método open(), es peligroso
#olvidar colocar este método, ya que la computadora lo considerará como si nunca hubiera sido cerrado, por lo
#cual no podré volver a abrirlo al dar clic sobre él.
new_file.close() #Sirve para cerrar el archivo que crea el archivo en el directorio indicado.

#ABRIR y LEER UN ARCHIVO:
#open(): Método que sirve para abrir un archivo cualquiera, para ello es necesario indicar dos parámetros,
#el primero se refiere a la ruta relativa o absoluta del archivo previamente creado y la segunda indica qué
#es lo que se va a realizar con él, el contenido del archivo se asigna a una variable.
# - r: Sirve para leer el contenido de un archivo.
my_file = open(filename, 'r')
#var_file_open.readline(): Método para leer la primera línea del contenido de un archivo previamente abierto
#con el método open().
stRecoveredMessage = my_file.readline() #Sirve para leer y guardar el mensaje que ingresó el usuario en consola.
#var_file_open.close(): Método para cerrar un archivo previamente abierto con el método open(), es peligroso
#olvidar colocar este método, ya que la computadora lo considerará como si nunca hubiera sido cerrado, por lo
#cual no podré volver a abrirlo al dar clic sobre él.
my_file.close() #Sirve para cerrar el archivo que crea el archivo en el directorio indicado.
#print(): Imprimir un mensaje en consola.
print(stRecoveredMessage)

```

Resultado del Código Python



```

PROBLEMAS  SALIDA  TERMINAL  CONSOLA DE DEPURACIÓN
Python + - □ □
PS C:\Users\diego\OneDrive\Documents\Aprendiendo\Python\1.-Instrumentación Virtual> & D:/Users/diego/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/diego/OneDrive/Documents/Aprendiendo/Python/1.-Instrumentación Virtual/12.-Abrir, Escribir y Leer un Archivo.py"
Ingresa el mensaje en consola:
Oliwis en Python
Oliwis en Python
PS C:\Users\diego\OneDrive\Documents\Aprendiendo\Python\1.-Instrumentación Virtual>

```

Código C# (.Net Framework) – Visual Studio (Logo Morado):

Código con Resultado en Consola:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
//Se usa la librería System.Numerics, agregada dando clic derecho en la parte de:
//Explorador de soluciones -> Referencias -> Agregar Referencia -> Nombre Librería.
using System.IO; //Librería utilizada para poder manipular archivos

namespace _8._Abrir__Escribir_y_Leer_un_Archivo
{
    class EditarArchivoConConsola
    {
        //CÓDIGO PARA LEER UN CREAD, ESCRIBIR Y LEER EL CONTENIDO DE UN ARCHIVO CUALQUIERA:
        static void Main(string[] args)
        {
            Console.WriteLine("Escribe un mensaje: ");
            /*El método Console.ReadLine() sirve para permitir al usuario que ingrese algo por consola, que será de
            tipo String por default*/
            string msg = Console.ReadLine();

            /*Para indicar una ruta en C# se debe poner un @ antes de indicar la ruta en cualquier string, es para que
            no den errores en el programa cuando se usen caracteres especiales en los nombres de la dirección del
            archivo y poder usar diagonales.
            Para leer una imagen o cualquier otro archivo se usa la dirección relativa o absoluta de un directorio:
            - Dirección relativa: Es una dirección que busca un archivo desde donde se encuentra la carpeta del
            archivo python actualmente, esta se debe colocar entre comillas simples o dobles.
            - Dirección absoluta: Es una dirección que coloca toda la ruta desde el disco duro C o cualquier otro
            que se esté usando hasta la ubicación del archivo, la cual se debe colocar entre comillas simples o
            dobles.
            ..      : Significa que nos debemos salir de la carpeta donde nos encontramos actualmente.
            /       : Sirve para introducirnos a alguna carpeta cuyo nombre se coloca después del slash.
            .ext    : Se debe colocar siempre el nombre del archivo + su extensión.*/
            string path = @"C:\Users\diego\OneDrive\Documents\Aprendiendo\C#\1.-Instrumentación
            Virtual\Archivos_Ejercicios_C#\7.-Abrir, Escribir y Leer un Archivo\MensajeC#.txt";
            /*El método using hace uso de la librería System.IO, este método debe usar un objeto de la clase StreamWriter
            y recibe como parámetro un string que dé una dirección de algún archivo.*/
            using (StreamWriter objetoClaseStreamWriter = new StreamWriter(path))
            {
                objetoClaseStreamWriter.WriteLine(msg);
            }

            //Lectura del archivo
            string mensajeRecuperado = File.ReadAllText(path);

            //Despliegue del mensaje recuperado en consola
            Console.WriteLine(mensajeRecuperado);

            } //Método main: Desde el método main se ejecutan todas las partes del proyecto
        } //Clase del programa
    } //Espacio de nombres: En esta parte del código se pueden declarar más de una clase que conforme el proyecto
}
```

Resultado del Código C#



```
C:\WINDOWS\system32\cmd.  +  v
Escribe un mensaje:
Oliwis en C#
Oliwis en C#
Presione una tecla para continuar . . . |
```

Diagrama LabVIEW:

