



Creación Manual de un Archivo txt o de TEXTO

Este año, estudiaremos dos tipos de archivos o ficheros: Los archivos de texto y los archivos binarios.

Un archivo de texto, contiene caracteres que son legibles por el ser humano y están guardados con una codificación (ASCII, UTF-8, ...).

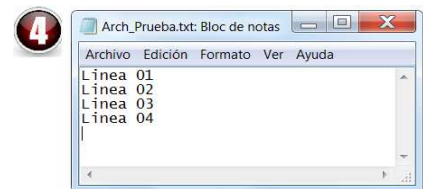
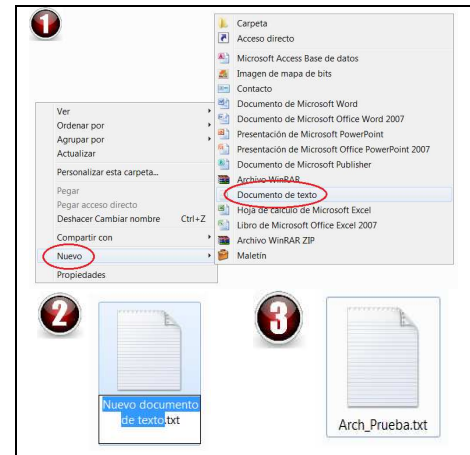
Un archivo binario, por el contrario, está compuesto por un flujo de bytes y solo tienen sentido en una PC y pueden ser leídos por los programas o aplicaciones para los que son creados. Un ejemplo de este tipo de archivos son los archivos que contienen imágenes, música, videos o incluso los mismos programas ejecutables.

Como crear un archivo txt con Windows. Si bien, la creación manual de un archivo de texto que acá aprenderán, no tiene que ver con la programación, y mucho menos es un comando Python, pero, es algo que necesitarás cuando comencemos a estudiar el proceso para leer estos archivos, ya que necesitaremos disponer de los archivos para leerles :).

Al comienzo, veremos la *lectura*, luego la *escritura* y finalizando el año "*la lectura, procesamiento de los datos leídos y posterior escritura de los resultados en un nuevo archivo de texto*".

Desde acá encontraras el "paso a paso" para crear archivos manualmente, archivos que luego leeremos con nuestros programas Python. Practica creando varios, ya que los usarás muy pronto.

- a) Primero nos posicionamos en la carpeta donde queremos crear nuestro Archivo de texto.
- b) Presiono Botón derecho del Mouse, esto provocará que una ventana de dialogo de *Windows* se abra. Ahora, en la ventana emergente, elije "Nuevo" y a continuación "**Documento de Texto**". Ver imagen 1.
- c) En el acto aparecerá el icono de un archivo de texto, y esperara a que ingresemos el nombre, el que sea de nuestra preferencia. Ver imagen 2. (poner un nombre sin espacios en blanco, ya que algunas versiones de Python, podrían presentar problemas)
- d) Escribimos el nombre que usaremos en nuestro programa, ya que será el que el programa busque. Ver imagen 3.
- e) Recuerda anotar la dirección o ruta de acceso al archivo ya que también lo necesitaras (**Si una vez creado el archivo le haces un clic, luego botón derecho, propiedades, podas ver la ubicación o ruta de acceso de este archivo**).
- f) **Repito:** debes anotar la dirección o ruta de acceso para llegar hasta el archivo, ya que también lo necesitarás (**Si una vez creado el archivo le haces un clic, luego botón derecho, propiedades, podas ver la ubicación o ruta de acceso de este archivo**).
- g) Escribir datos en el Archivo de Texto: Simplemente haces un doble clic en el archivo, o lo abres con el **Block de Notas** de Windows. Una vez abierto el archivo con el editor, procedes a escribir en cada línea lo que quieras y esperas leer desde tu programa Python. Con un **Enter**, das por terminada cada una de las líneas y continúas escribiendo en la siguiente. Ver imagen 4.
- h) Cuando hayas terminado de escribir todas las líneas necesaria, recuerda grabarlo; y para realizar esta acción, ingresas a la opción del menú "**Archivo**" (File) y seleccionar "**Grabar**" (Save). Y listo, has terminado. cierra la ventana y a otra cosa.



Desde este momento, ya puedes usar el archivo de texto desde tu programa Python.



Lectura de un Archivo de Texto Plano con un Programa Python

Si bien este proceso será explicado en clase, a continuación te dejo el código para que lo copies, y analices en el IDE Python. Recomiendo realizar todas las pruebas necesarias para entender bien todo el proceso. Recuerda que tienes la dirección para que lo bajes directamente de la web de tu curso, en:

Nombre del programa: `Codigo/08_Archivos_001_TXT_Plano_Lectura_001.txt`



Creación y Uso de Archivos TXT

5A - 2025

```
# -*- coding: utf-8 -*-  
.....  
Creado 24/04/2022  
Mientras se pueda leer, Proceso lo que lei  
@author: Horacio  
.....  
# Uso del import time, para generar tiempo de espera entre línea y línea  
import time  
  
print("-----")  
print("    Otra forma de Procesar archivos con Python.")  
print("    Mientras se pueda leer, Proceso (Parte 01)")  
print("Leo línea una línea y espero 3 segundos antes de leer la próxima")  
print("línea, para permitir que visualices lo que se lee en el archivo")  
print("-----")  
  
# Armo dirección y nombre de Archivo para trabajar  
# Camino = "d:/M_de_Falla/Programacion/Prog_2023/" #Aca la direccion que quiera  
Camino = "" # deajo en blanco y Python entiende que es la carpeta de trabajo  
Nombre = "Personas.txt" # Nombre Fisico del archivo (el que veo en el disco)  
Archivo_Fisico = Camino + Nombre # Armo Dirección más nombre de Archivo  
  
Archivo_Logico = open(Archivo_Fisico, "r") # Abro el Archivo.  
  
# Si se puede, Leerá la primera línea y quedará listo para leer próxima línea,  
# pero si no se puede, retorna NADA  
Linea = Archivo_Logico.readline()  
  
# Cuando el archivo se acaba (ya no hay mas que leer), en línea se guardará  
# "" es decir Nada, y el while entenderá que la condición es falsa y dará  
# por terminado el ciclo (ya NO repite más)  
while Linea: # Mientras Línea contenga algo, en el  
    print(Linea) # archivo ya Tiene un \n al final de cada línea  
    time.sleep(3) # Demora de tres segundos  
    Linea = Archivo_Logico.readline()  
  
if not Linea: # Pregunto: Esta Vacía?  
    print("Colorin Colorado, el Archivo ha Terminado")  
  
# Siempre cierro el archivo  
Archivo_Logico.close()  
  
# Termina Programa  
print("\n-----")  
print("Programa Terminado.")
```

Antes de comenzar las pruebas del programa en tu PC, recomiendo que cambies o configures el IDE en Spyde. El documento lo encontrar en el mismo lugar que abriste este

Un extra a saber, cuando abres archivos de Texto (encoding="utf-8")

Hasta el momento, se ha explicado (y siempre funcionó) que a un archivo podemos abrirlo para lectura así:

```
Archivo_Logico = open( Archivo Fisico, "r")
```

Esto asume que tu archivo ha sido escrito con las reglas definidas por "utf-8", ahora bien, si quieres evitar posibles fallos o errores de lectura y asegurarte que realmente se lo lea e interprete con esta norma, puedes abrir tus archivos de texto agregando el parámetro: **encoding="utf-8"**. Es decir ábrelo así:

```
Archivo_Logico = open( Archivo Fisico, "r", encoding="utf-8")
```

Explicación: El parámetro encoding="utf-8" indica a tu programa cómo interpretar los caracteres del archivo (la "codificación de texto").

Al poner ese parámetro, le dices a Python: "lee este archivo como texto en UTF-8".

Que resulta ser el estándar actual en la mayoría de los archivos y sistemas.



Creación y Uso de Archivos TXT

5A - 2025

Y por las dudas, estarás garantizando que se lean bien los acentos (á, é), eñes (ñ) y símbolos especiales.

Y si NO PONES ese parámetro, Si tu archivo solo tiene letras normales (a-z, A-Z) y números, funcionará igual. Pero si el archivo tiene acentos, ñ o símbolos raros, **podría dar el error** (UnicodeDecodeError) o mostrarte caracteres extraños.

Escritura de un archivo de texto plano con un programa Python

Escribamos muchos registros (Nombre y edad) que iremos ingresando por teclado de a uno. Terminamos la carga presionando **ENTER**, sin cargar un nombre.

```

Registro = "" # Defino cadena de caracteres, que usaré como registro
Ruta="" # Dirección Vacía, lee o escribe archivo guardado junto con mi programa
Archi="Nombres_00.txt"
Archivo=Ruta+Archi
Logico=open(Archivo, "w") # Abro Archivo para escribir
Nombre=input("Nombre: ")
while Nombre:
    Edad= int(input("Edad: "))
    Registro = Nombre + "/" + str(Edad) + "\n" # Armo Registro
    Logico.write(Registro)
    Nombre=input("Nombre: ")
Logico.close() # RECUERDA. Siempre cierro Archivo

```



RESUMEN de Trabajo con Archivos

Modos de apertura

r	Lectura
w	Escritura
	Hay otras formas de abrir archivos (para realizar otras tareas), que por ahora no analizaremos.

Lugar donde se encuentra y nombre	Camino = "" # dejo en blanco y Python entiende que es la carpeta de trabajo Nombre = "Personas.txt" # Nombre Físico del archivo (el que veo en el disco) Archivo_Físico = Camino + Nombre # Armo Dirección más nombre de Archivo
Dos formas de Apertura	Archivo_Logico = open(Archivo_Físico, "MODO APERTURA ") with open(Mi_Archivo_txt, 'MODO APERTURA') as f_txt:
Lectura/Escritura	Linea = Archivo_Logico.readline() f_txt.write(Linea + "\n")



Creación y Uso de Archivos TXT

5A - 2025

Proceso leyendo	<pre>Linea = Archivo_Logico.readline() while Linea: # Mientras Linea contenga algo print(Linea) #en el archivo ya Tine un \n al final de cada linea Linea = Archivo_Logico.readline()</pre>
Proceso Escribiendo	<p>Termina con la lógica que TÚ quieras. Podría ser igual a la de lectura, si cargara datos leídos por teclado. Cuidado, deberás usar f_txt.write</p>
Innecesario	<pre>if not Linea: # Pregunto: Esta Vacía? print("Puedo saber si termino de leer")</pre> <p>En cualquier momento puedo preguntar si leyó NADA. Esto sucede cuando se acabo el archivo y ya no quedan registros para leer.</p>
Escribiendo	<p>Termina con la lógica que yo quiera</p>
Finaliza y no hay nada mas que hacer	<pre># Siempre cierro el archivo Archivo_Logico.close()</pre>