



El futuro digital  
es de todos

MinTIC



*Vigilada Mineducación*

# CICLO I:

## Fundamentos de programación en Python

**Misión  
TIC2022**





El futuro digital  
es de todos

MinTIC



Vigilada Mineducación

# Sesión 19:

# Manejo de archivos





## Objetivos de la sesión

Al finalizar esta sesión estarás en capacidad de:

- Construir un programa con una interfaz de consola o gráfica simple.
- Construir un programa que maneje archivos de texto para almacenar información persistente (formato JSON / CSV).



# Archivos

## Acceso a Archivos

- Crear un archivo
- Modo de apertura de los archivos
- Leer desde un archivo de texto
- Escribir en un archivo o Agregar información a un archivo
- Cerrar un archivo
- Persistencia de datos
- Diccionarios



# Archivos

## Escribir en un archivo

Presentamos dos formas comunes de escribir a un archivo:

1. Mediante cadenas: escribe una cadena al archivo.

**`archivo.write(cadena)`**

2. Mediante listas de cadenas: recibe una lista de líneas para escribir.

**`archivo.writelines(lista_de_cadenas)`**



# Archivos

## Cerrar un archivo

Al terminar de trabajar con el archivo, se cierra. Para cerrar un archivo se realiza lo siguiente:

**`archivo.close()`**

Esto libera el archivo para ser usado por otros programas, y además asegura que los cambios sobre él se guarden.

Python se encarga de cerrar todos los archivos que queden abiertos al final del programa, sin embargo, se recomienda como una buena práctica no dejar nada al azar y cerrar el archivo tan pronto se termine de usar.





# Archivos

## Persistencia de datos

Persistencia es la capacidad de poder guardar la información de un programa para volver a utilizarla en otro momento.

Esto se refiere a lo que los usuarios llaman como Guardar el archivo y después Abrir el archivo.

Para un programador, esto puede significar suele involucrar un proceso de serialización de los datos a un archivo o a una base de datos o a algún otro medio similar, y el proceso inverso de recuperar los datos a partir de la información serializada.



# Diccionarios

En Python, un diccionario es una colección no-ordenada de valores que son accedidos a través de una clave. Es decir, no se accede a la información mediante el índice numérico, como es el caso de las listas y tuplas, si no que es posible acceder a los valores a través de sus claves, las cuales pueden ser de diversos tipo.

Las claves son únicas dentro de un diccionario, no puede haber un diccionario que tenga dos veces la misma clave, si se asigna un valor a una clave que ya existe, se sobrescribe el valor anterior.





# Diccionarios – Algunos usos

- Contar el número de veces que una palabra aparece dentro de un texto.
- Para tener una agenda donde la clave es el nombre de la persona, y el valor es una lista con los datos correspondientes a esa persona.
- En general, los diccionarios sirven para crear bases de datos muy simples, en las que la clave es el identificador del elemento, y el valor son todos los datos del elemento a considerar.



# Diccionarios – Python

De la misma forma que con listas, es posible definir un diccionario directamente con los miembros que va a contener, o bien inicializar el diccionario vacío y luego agregar los valores de a uno o de a muchos.

Para definirlo junto con los miembros que va a contener, se encierra el listado de valores entre llaves, las parejas de clave y valor se separan con comas, y la clave y el valor se separan con `:`.

```
punto = {'x': 2, 'y': 1, 'z': 4}
```

Para declararlo vacío y luego ingresar los valores, se lo declara como un par de llaves sin nada en medio, y luego se asignan valores directamente a los índices.



El futuro digital  
es de todos

MinTIC



Vigilada Mineducación

# Ejercicios para practicar





El futuro digital  
es de todos

MinTIC

**UN** UNIVERSIDAD  
DEL NORTE

Vigilada Mineducación

**¡GRACIAS**  
**POR SER PARTE DE**  
**ESTA EXPERIENCIA**  
**DE APRENDIZAJE!**



**Mision**  
**TIC 2022**