



El futuro digital  
es de todos

MinTIC



Vigilada Mineducación



# CICLO I:

## Fundamentos de Programación en Python



**Hechos**  
QUE CONECTAN

# Sesión 15:

# Manejo de Archivos

DATOS DE ARCHIVOS

# Objetivos de la sesión

Al finalizar esta sesión estarás en capacidad de:

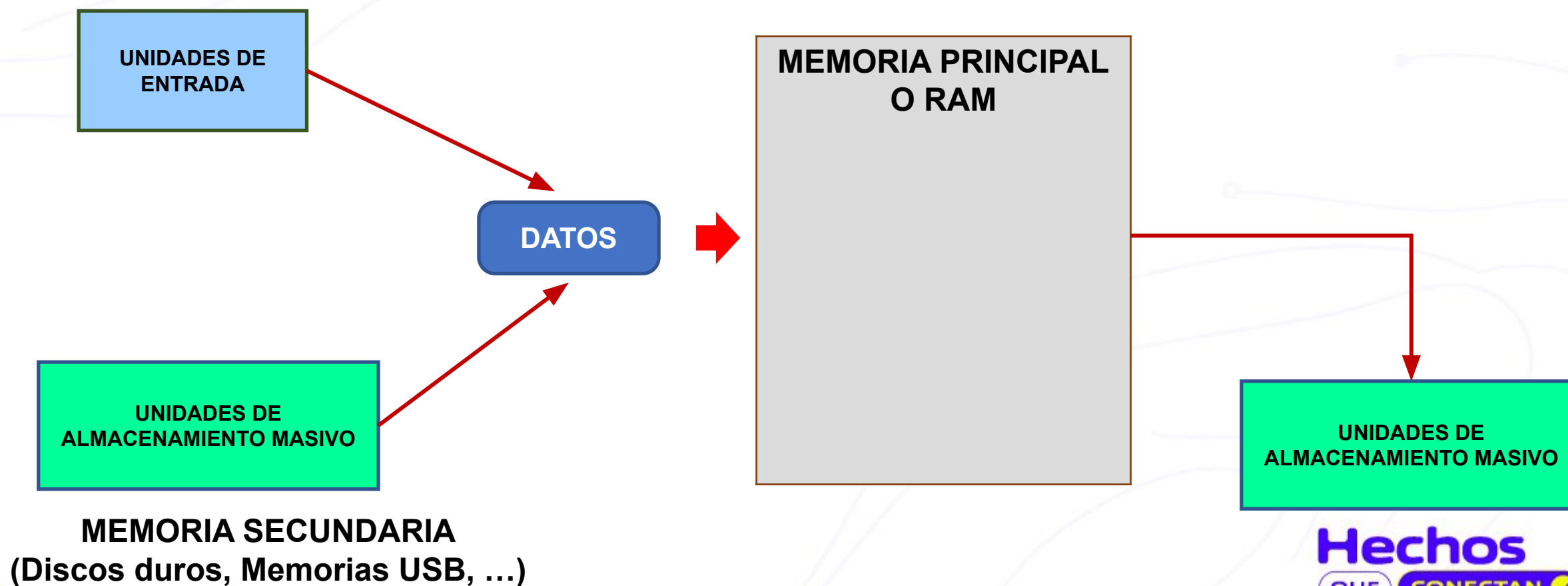
1. Construir un programa con una interfaz de consola o gráfica simple
2. Construir un programa que maneje archivos de texto para almacenar información persistente (formato TXT, JSON, CSV)

# Archivos: Definición

Un archivo es una colección de información (datos relacionados entre sí) almacenada en memoria secundaria. A diferencia de los arrays o arreglos un archivo puede almacenarse en un dispositivo auxiliar discos, etc. de forma que los datos obtenidos durante y después del manejo no se pierdan.

Los archivos son “Estructuras dinámicas” debido a que el tamaño ocupado en memoria puede ser modificado.

# Archivos: Diferencia entre Memoria Principal y Secundaria



# Archivos: Características generales

**Representables:** Los archivos suelen tener un nombre de máximo 255 caracteres y suelen ser representados en sistemas operativos de interfaz gráfica (como Windows) por un ícono determinado.

**Únicos por directorio:** En una misma carpeta o directorio no pueden existir dos archivos idénticos con el mismo nombre.

**Modificables:** Excepto aquellos que expresamente hayan sido protegidos contra modificación.

**Poseen un tamaño:** De acuerdo a la cantidad de información que un archivo contenga, éste tendrá un tamaño o “peso”, mensurable en KB, MB o incluso GB.

# Archivos: Tipos

## Archivos de texto

- Archivos de texto plano
- Archivos de configuración
- Archivos XML

## Archivos binarios



# Archivos: De texto

Cuando el contenido del archivo es exclusivamente caracteres de texto (se puede leer con un editor de texto básico)

Extensión	Tipo de Archivo
.txt	Texto plano
.xml	XML
.json	De intercambio de información
.props	De propiedades
.conf	De configuración
.sql	Script SQL
.srt	De subtítulo



# Archivos: De texto

A continuación algunos tipos de archivos de texto conocidos:

**Archivos de texto plano** son los que contienen texto libre y donde se puede escribir sin tener en cuenta ningún tipo de formato

Archivos CSV son los que tienen un formato específico que permite guardar los datos en un formato de tabla estructurada.

**Archivos de configuración** son los que contienen información de configuración para una aplicación. Tienen un formato específico.

**Archivos XML** son los que contienen información con etiquetas que le dan significado. Tienen unas reglas y formato más o menos definido.

# Archivos: Binarios

Son los archivos que no están compuestos solamente de texto. Pueden contener imágenes, videos, aunque también se puede considerar un archivo binario a uno de Microsoft Word en el que sólo se haya escrito algún texto puesto que, al almacenarse el archivo, el procesador de texto incluye alguna información binaria.

Extensión	Tipo de Archivo
.pdf	PDF
.jpg, .png	De imagen
.doc, .docx	Microsoft Word
.avi	Video
.ppt, .pptx	Microsoft PowerPoint
.pdf	PDF
.exe	Programa ejecutable

# Archivos: CRUD

La sigla o palabra CRUD se conforma con las primeras letras de las cuatro operaciones fundamentales de aplicaciones persistentes en sistemas de bases de datos, que son archivos binarios de propósito especial:

- Create (Crear registros)
- Read (Leer registros)
- Update (Actualizar registros)
- Delete (Borrar registros)

# Archivos: Pasos para interactuar

Existen tres (3) pasos básicos para interactuar con archivo:

## 1. Abrir el archivo

En Python para abrir un archivo usamos la función `open()`

```
f = open("data.txt", "r")
```

`f` es un objeto de tipo `<class '_io.TextIOWrapper'>`

## 2. Manipular o realizar operaciones con el archivo

## 2. Cerrar el archivo

En Python para cerrar un archivo usamos el método `close` del objeto

`'_io.TextIOWrapper'`

```
f.close()
```

# Archivos: Apertura

Existen dos formas básicas de acceder a un archivo:

1. Como un archivo de texto: línea por línea;
2. Como un archivo binario: byte por byte.

En Python, para abrir un archivo se usa la función **open**, que recibe el nombre del archivo a abrir.

```
archivo = open("archivo.txt")
```

Esta función intentará abrir el archivo con el nombre indicado. Si tiene éxito, devolverá una variable que permitirá manipular el archivo de diferentes formas.

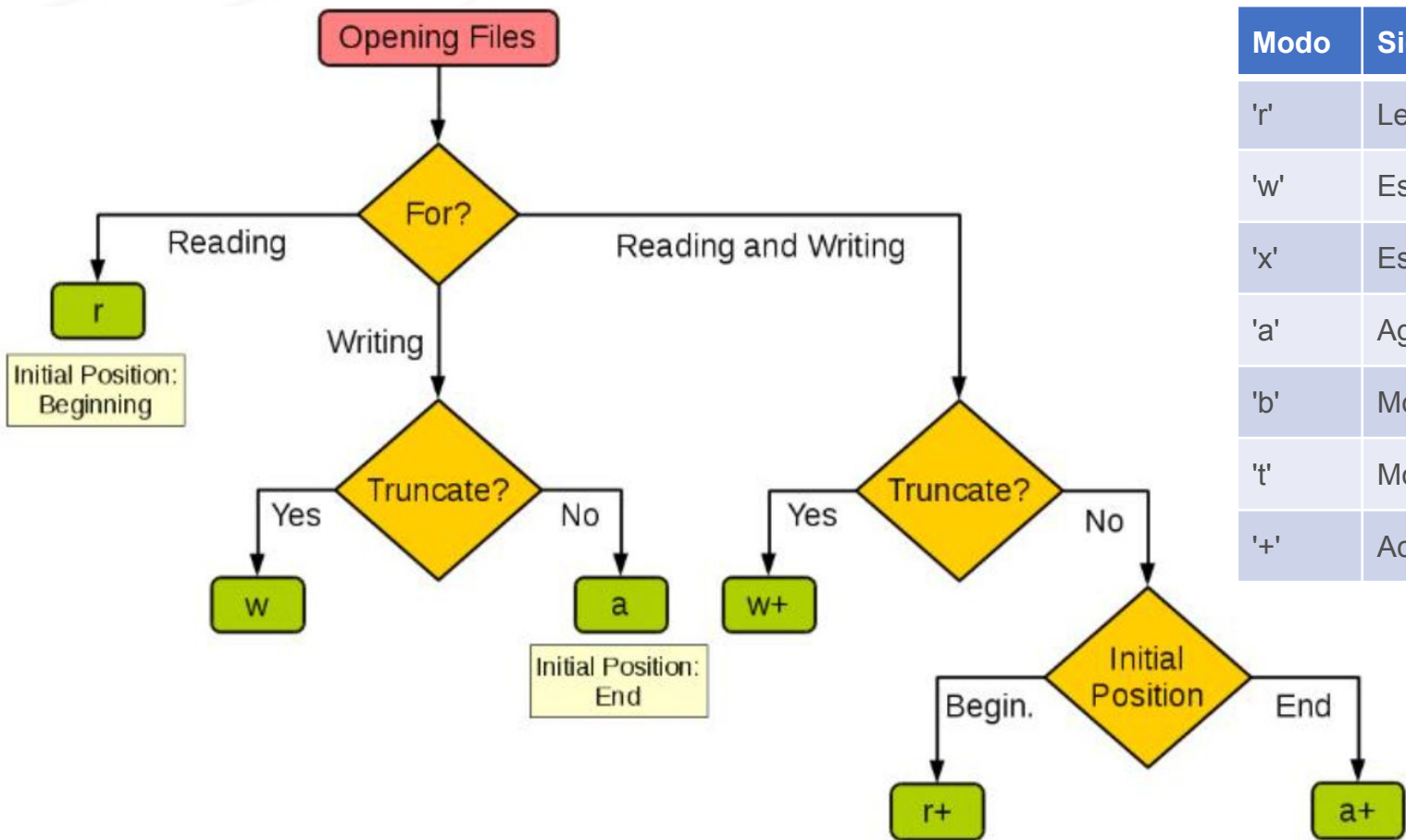
# Archivos: Modos de apertura

La función open recibe un parámetro opcional para indicar el modo en que se desea abrir el archivo. Los tres modos de apertura permitidos son:

1. De sólo lectura (*r*). No es posible realizar modificaciones sobre el archivo, solamente se puede leer su contenido.
2. De sólo escritura (*w*). El archivo es truncado (vaciado) si existe, y si no se crea.
3. Sólo escritura posicionándose al final del archivo (*a*). Se crea el archivo, si no existe, en caso de que exista se posiciona al final, manteniendo el contenido original.



# Archivos: Modos de apertura



Modo	Significado específico
'r'	Leer (predeterminado)
'w'	Escribir (truncar primero el contenido anterior)
'x'	Escribir, si el archivo ya existe generará una excepción
'a'	Agregar, escribir contenido al final de un archivo existente
'b'	Modo binario
't'	Modo de texto (predeterminado)
'+'	Actualización (tanto lectura como escritura)



# Archivos: Manejo de excepciones

- FileNotFoundError
- InterruptedError
- LookupError
- UnicodeDecodeError
- IsADirectoryError
- NotADirectoryError
- PermissionError
- OSError
- ProcessLookupError
- TimeoutError

# Archivos: Leer desde un archivo de texto

Existen varias formas para leer el contenido de un archivo:

1. Hacia una cadena con **read()**.
2. Leer una línea con **readline()**.
3. Obtener una lista conteniendo las líneas del archivo con **readlines()**.

Los tres métodos aceptan un parámetro entero opcional que define el número máximo de bytes a leer del archivo. Si este parámetro es negativo o simplemente se omite, **read** y **readlines** leerán todo el archivo y **readline** una línea completa sin importar su largo.

# COMPONENTE PRÁCTICO



El futuro digital  
es de todos

MinTIC

Misión  
TIC 2022

UN UNIVERSIDAD  
DEL NORTE

Vigilada Mineducación

**¡GRACIAS**

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

**Hechos**

QUE

CONECTAN

