

# Los Datos del Mundo: Análisis con Python

Guía interactiva para leer, explorar y analizar datos usando Pandas.

ID	NAME	VALUE	TIMESTAMP	STATUS
001	"Sensor_A"	42.5	2024-05-20T10:00:00Z	OK
002	"Sensor_B"	24.5	2024-05-20T10:00:00Z	OK
003	"Sensor_C"	42.5	2024-05-20T10:00:00Z	Alert Red
004	"Sensor_D"	32.5	2024-05-20T10:00:00Z	Alert Red

# El lenguaje de los datos

Ya conoces Python para automatizar tareas, programar juegos y manipular variables simples.



Automatización



Juegos

{tx\_xs}

Variables simples

- Ahora usarás su superpoder principal: procesar y analizar grandes volúmenes de datos mediante su amplio ecosistema de herramientas.

```
ID_001, 'Data_Science', 1024  
ID_002, 'Analysis_Tool', 251
```

ID_001	...	2021-01-01	100	1014	...	...	...	...	...
ID_002	...	2021-01-01	100	1014	...	...	...	...	...
ID_003	...	2021-01-01	100	1014	...	...	...	...	...
ID_004	...	2021-01-01	100	1014	...	...	...	...	...
ID_005	...	2021-01-01	100	1014	...	...	...	...	...
ID_006	...	2021-01-01	100	1014	...	...	...	...	...
ID_007	...	2021-01-01	100	1014	...	...	...	...	...
ID_008	...	2021-01-01	100	1014	...	...	...	...	...
ID_009	...	2021-01-01	100	1014	...	...	...	...	...
ID_010	...	2021-01-01	100	1014	...	...	...	...	...
ID_011	...	2021-01-01	100	1014	...	...	...	...	...
ID_012	...	2021-01-01	100	1014	...	...	...	...	...
ID_013	...	2021-01-01	100	1014	...	...	...	...	...
ID_014	...	2021-01-01	100	1014	...	...	...	...	...
ID_015	...	2021-01-01	100	1014	...	...	...	...	...
ID_016	...	2021-01-01	100	1014	...	...	...	...	...
ID_017	...	2021-01-01	100	1014	...	...	...	...	...
ID_018	...	2021-01-01	100	1014	...	...	...	...	...
ID_019	...	2021-01-01	100	1014	...	...	...	...	...
ID_020	...	2021-01-01	100	1014	...	...	...	...	...
ID_021	...	2021-01-01	100	1014	...	...	...	...	...
ID_022	...	2021-01-01	100	1014	...	...	...	...	...
ID_023	...	2021-01-01	100	1014	...	...	...	...	...
ID_024	...	2021-01-01	100	1014	...	...	...	...	...
ID_025	...	2021-01-01	100	1014	...	...	...	...	...

```
>>> python3 analyze_data.py
```

! ¿Qué historias esconden los datos hoy?

# De Caos a Claridad: CSV vs. Tabla

```
Rango de edad,Sí sabe,No sabe,No informa,No informa.1
De 0 a 9 años*,40.6,10.31,1.43,
De 10 a 19 años,97.3,1.37,1.33,
De 20 a 29 años,96.57,1.71,1.72,
De 30 a 39 años,95.96,2.69,1.35,
De 40 a 49 años,94.5,4.24,1.26,
De 50 a 59 años,92.47,6.28,1.25,
De 60 a 69 años,88.71,10.34,0.96,
```

Conversión  
Visual

	A	B	C	D	E
1	Rango de edad	Sí sabe	No sabe	No informa	No informa.1
2	De 0 a 9 años*	40.6	10.31	1.43	
3	De 10 a 19 años	97.3	1.37	1.33	
4	De 20 a 29 años	96.57	1.71	1.72	
5	De 30 a 39 años	95.96	2.69	1.35	
6	De 40 a 49 años	94.5	4.24	1.26	
7	De 50 a 59 años	92.47	6.28	1.25	
8	De 60 a 69 años	88.71	10.34	0.96	
9					

## CSV (Valores Separados por Comas).

Archivo de texto plano. Ocupa muy poco espacio en memoria, ideal para almacenar grandes volúmenes.

## Formato de Tabla.

Representación ordenada que permite visualizar la información de forma clara para el cerebro humano.

# Anatomía de un DataFrame

**Dataframe:** Estructura de datos en forma de tabla con filas y columnas, utilizada para organizar y analizar información en Python.

**Columnas.** Cada columna contiene un tipo específico de dato (números, texto, fechas).

**Nombres de Columnas.** Etiquetas que identifican la variable medida.

**Filas.** Cada fila representa un registro único (ej. un estudiante).

	ID	Nombre	Edad	Carrera	Promedio
0	E001	Ana García	20	Ingeniería	8.9
1	E002	Carlos López	22	Arquitectura	9.2
2	E003	Barrey Rolaers	23	Ingenieria	9.1
3	E004	Anna Bidlo	21	Arquitectura	9.9
...	...	...	...	...	...
...	...	...	...	...	...

```
>>> python3 analyze_data.py
```

# Elige tu entorno de trabajo



**Google Colab**

**Tipo**  
Nube (Requiere Internet)

**Archivos**  
Se borran al cerrar sesión

**Ideal para**  
Trabajo rápido colaborativo



**Anaconda / Jupyter**

**Tipo**  
Local (Requiere Instalación)

**Archivos**  
Se guardan en tu PC

**Ideal para**  
Proyectos grandes y offline



**Python Online**

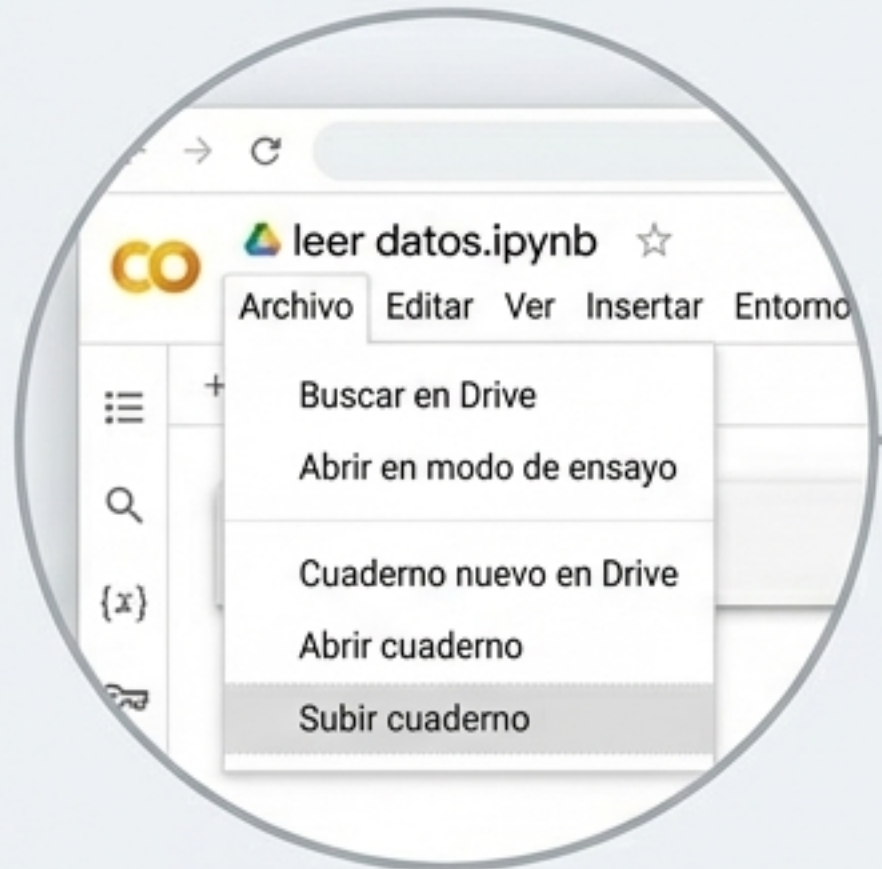
**Tipo**  
Web Express (Navegador)

**Archivos**  
Descarga manual

**Ideal para**  
Dispositivos móviles / sin instalación

```
>>> python3 environment_setup.py
```

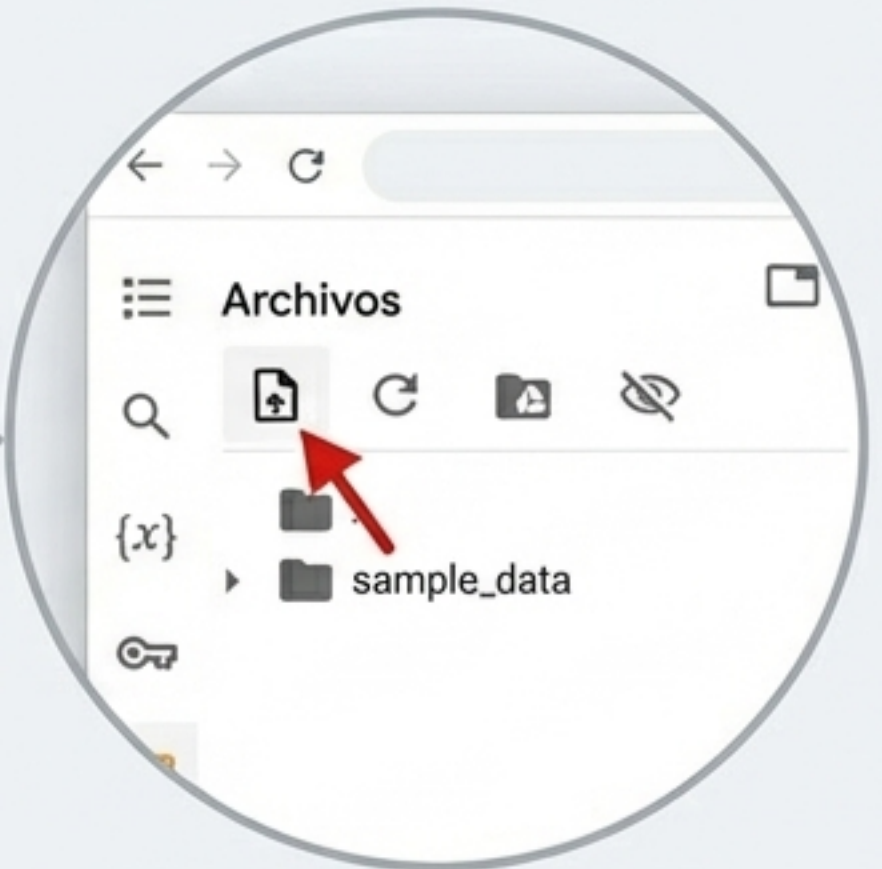
# Setup: Google Colab (La Nube)



**Paso 1:** Archivo -> Subir cuaderno (.ipynb).



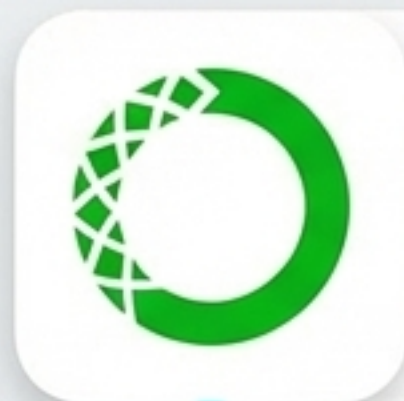
**Paso 2:** Clic en el ícono de "Carpeta" en la barra lateral izquierda.



**Paso 3:** Clic en "Subir al almacenamiento de sesión" y selecciona tu CSV.

**¡ADVERTENCIA!** Los archivos cargados aquí se eliminarán cuando finalice tu sesión. Asegúrate de tener copias de seguridad.

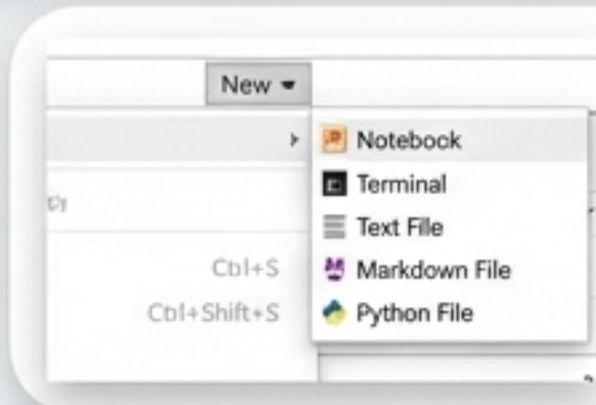
# Setup: Anaconda y Jupyter (Local)



Paso 1: Abre la aplicación **Anaconda Navigator** (busca el círculo verde en tu PC).



Paso 2: Encuentra la tarjeta de **Jupyter Notebook** y haz clic en **Launch**.

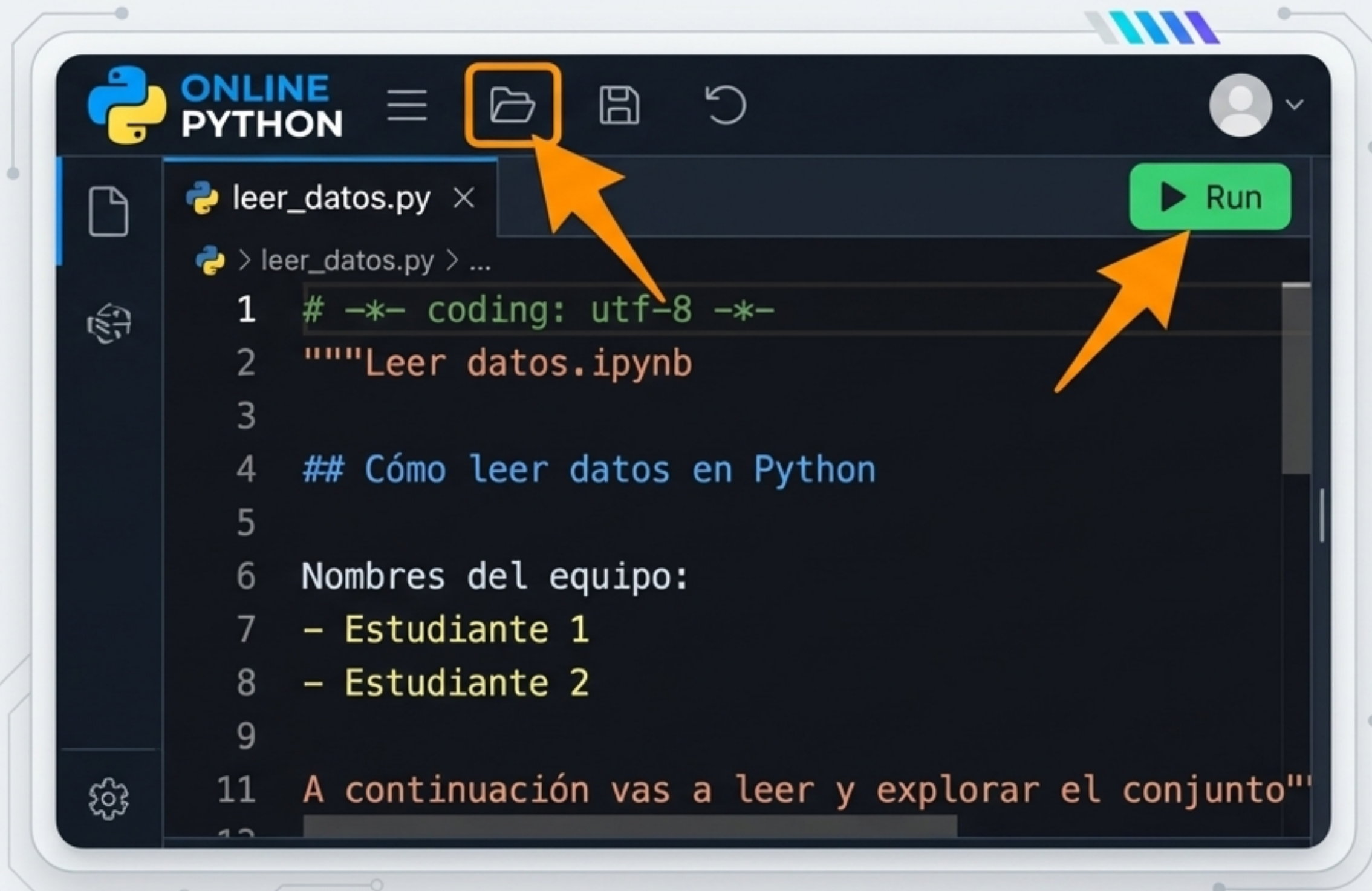


Paso 3: En el navegador, ve a **New -> Notebook**.

## Concept Note:

Un **"Notebook"** o **cuaderno interactivo** combina código, texto y visualizaciones en un solo documento.

# Setup: Python Online (Express)



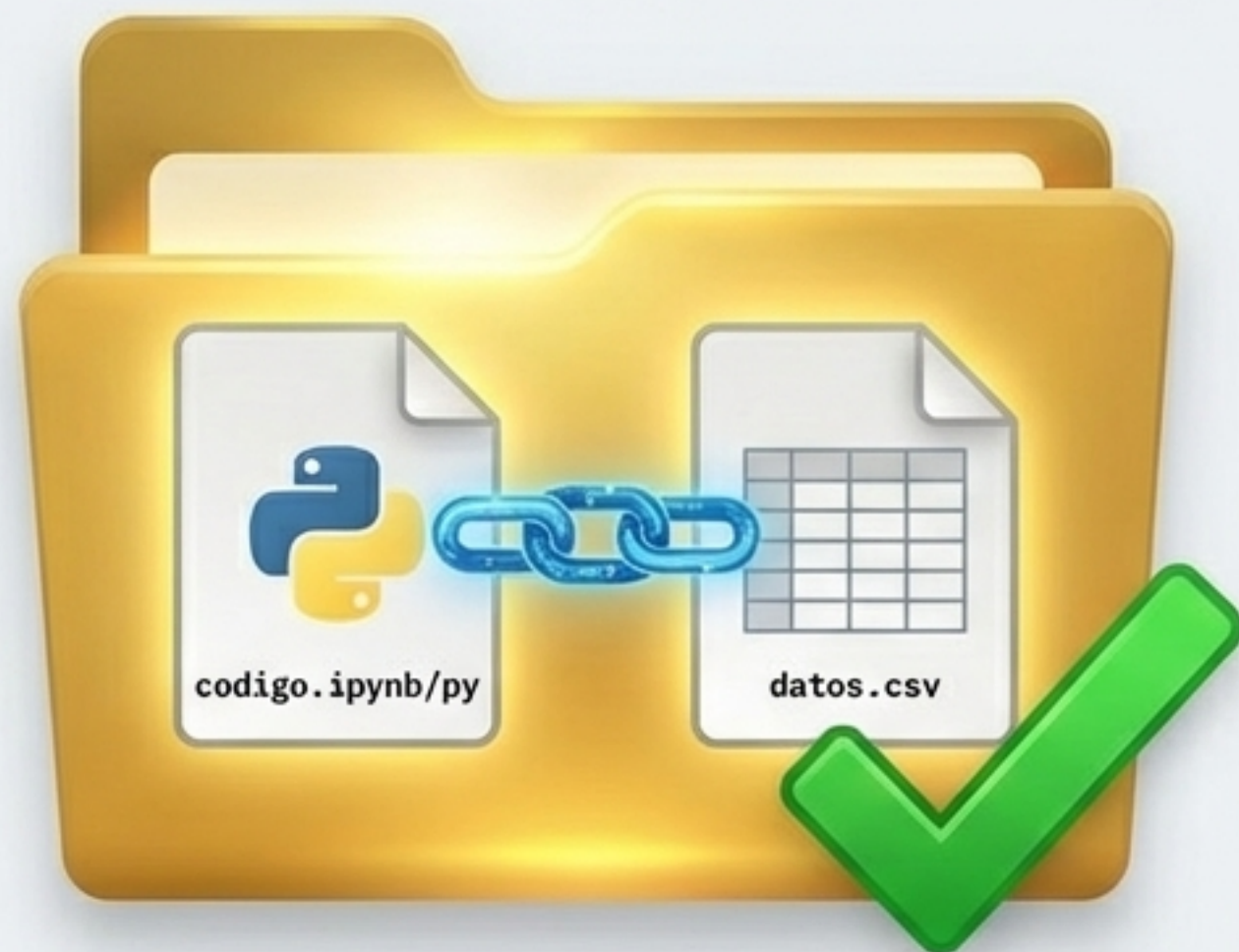
Paso 1: Ingresa a [www.online-python.com](http://www.online-python.com).

Paso 2: Haz clic en el ícono de **Abrir** (la pequeña carpeta en la barra superior).

Paso 3: Selecciona y carga primero tu archivo de código (**leer\_datos.py**) y luego tu archivo de datos (**pruebas\_saber.csv**).

Paso 4: Haz clic en **Run** para ejecutar.

## La Regla de Oro del Explorador



Para leer archivos en Python, es **estrictamente necesario** que tanto el código como el archivo de datos vivan en la **MISMA CARPETA**.

➔ Crea una carpeta llamada **Guía 2 - Datos** en tus **Descargas** y mueve ambos archivos allí antes de empezar.

# Invocando a Pandas

## El Paquete:

Traemos la herramienta de análisis.

```
import pandas as pd  
  
saber = pd.read_csv("pruebas_saber.csv")
```

## El Apodo:

Le damos un nombre corto para escribir menos después.

## La Variable:

Creamos un espacio en memoria para guardar nuestra tabla.

## La Función Mágica:

El comando que transforma el texto plano en un DataFrame.

# Tu Cheat Sheet de Exploración



## **.head(10)**

Muestra las primeras 10 filas de la tabla. Útil para dar un vistazo rápido.



## **.info()**

Revela los detalles técnicos: tipos de datos, cantidad de columnas y memoria usada.



## **.tail(10)**

Muestra las últimas 10 filas de la tabla.



## **.shape**

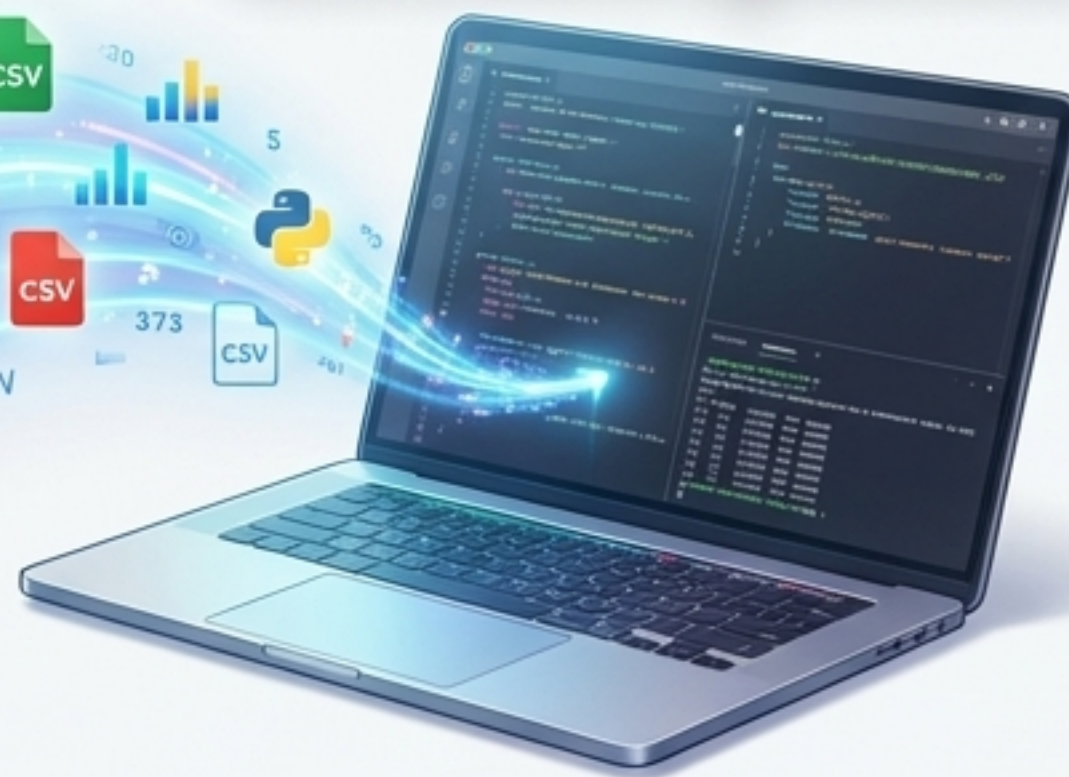
Te dice el tamaño exacto de la tabla en formato (filas, columnas).

# La Expedición: Datos Abiertos

Es el momento de buscar datos reales. Los Datos Abiertos son información pública gubernamental diseñada específicamente para ser usada, reutilizada y redistribuida libremente.



code x  
¡El gobierno de Colombia tiene miles de tablas esperando ser exploradas!



# Flujo de Extracción de Datos

## Selecciona un Tema

🔍 Buscar catálogo...

Explora el catálogo en **datos.gov.co** (ej. Educación, Salud, Cultura).

## Filtra

- Conjuntos
- Filtra
- Conjuntos de datos

Asegúrate de seleccionar el tipo de vista "Conjuntos de datos".

## Visualiza



Haz clic en la pestaña "Datos" para previsualizar la tabla en el navegador.

## Exporta

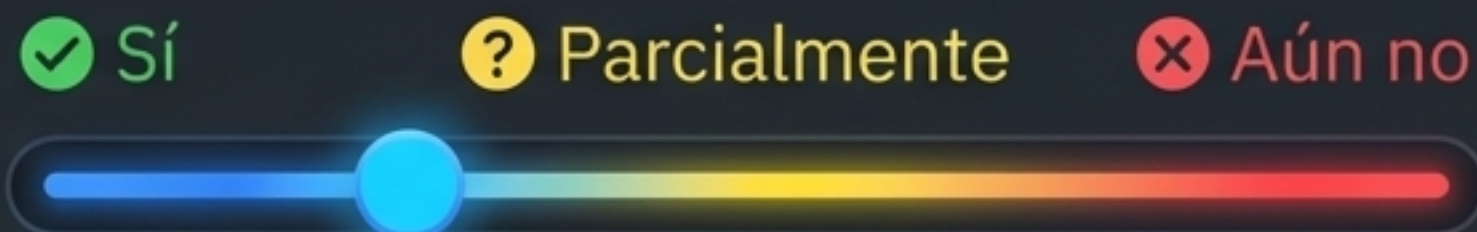
📄 Exportar

Usa el botón azul para descargar tu botín en formato **CSV**.

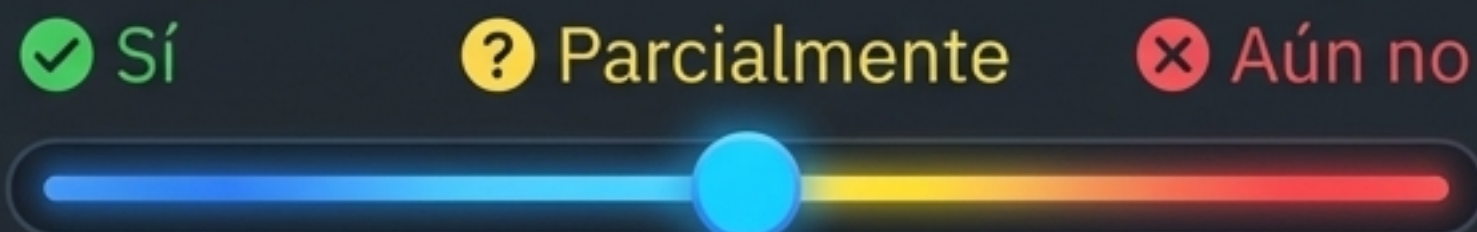
# Checkpoint de Aprendizaje

## Dashboard

¿Reconozco y diferencio las estructuras de datos, incluyendo archivos CSV y DataFrames?



¿Puedo escribir y ejecutar código utilizando Pandas para leer un archivo CSV en Python?



*Si tu respuesta es 'Parcialmente' o 'Aún no', regresa al Cheat Sheet (Slide 11) o verifica la Regla de Oro (Slide 9).*

# Tiquete de Salida



Piensa en el conjunto de datos reales que acabas de descargar de datos.gov.co... **¿Qué preguntas podrías responder con estos datos?**

¿Cómo conectarías este nuevo poder de análisis con lo que ya sabías hacer en Python?



>>> SESIÓN\_01\_COMPLETADA. BUEN TRABAJO.