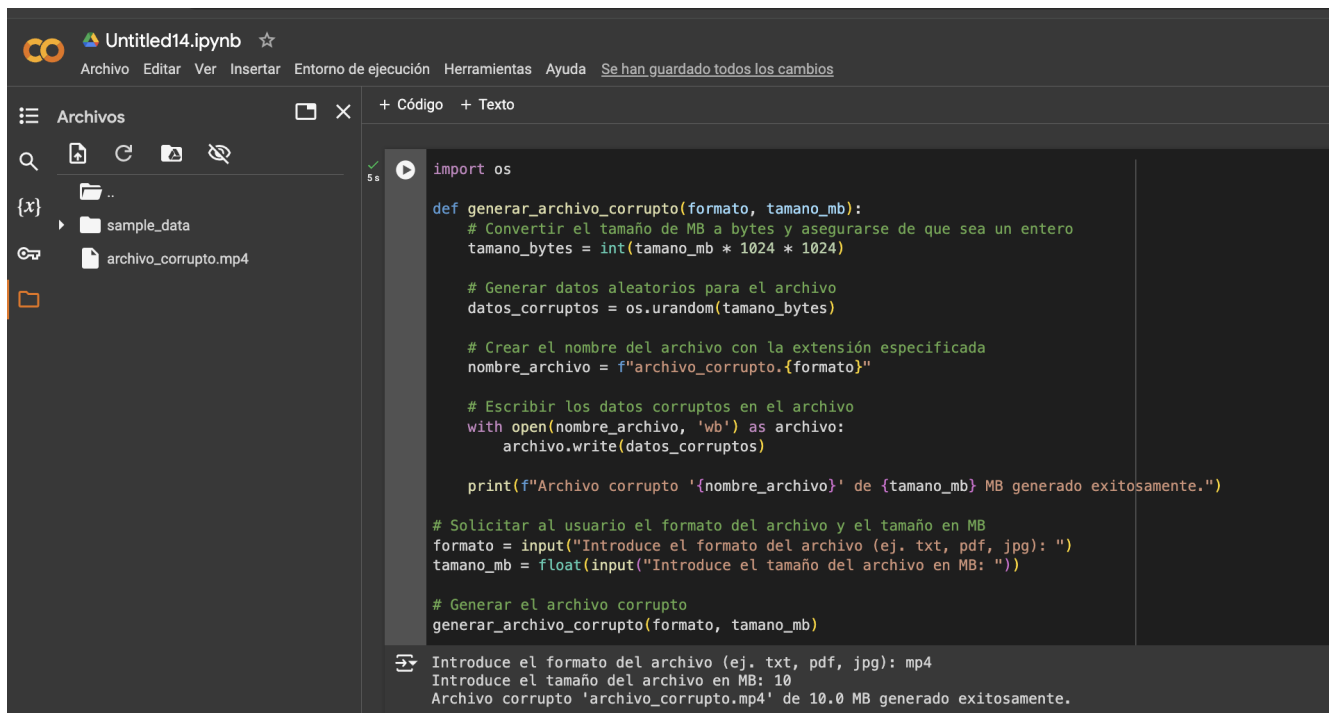


Script en Python que permite al usuario especificar el formato y el tamaño para generar un archivo corrupto

[crayon-6a9d4f7c46bc5517790374/]



The screenshot shows a Jupyter Notebook titled 'Untitled14.ipynb'. The left sidebar displays a file explorer with a folder named 'sample_data' and a file named 'archivo_corrupto.mp4'. The main area contains a Python script that defines a function to generate a corrupted file. The script uses the 'os' module to generate random data and write it to a file. It prompts the user for the file format and size in MB, and then generates the corrupted file. The output of the script is displayed at the bottom of the notebook.

```
import os

def generar_archivo_corrupto(formato, tamano_mb):
    # Convertir el tamaño de MB a bytes y asegurarse de que sea un entero
    tamano_bytes = int(tamano_mb * 1024 * 1024)

    # Generar datos aleatorios para el archivo
    datos_corruptos = os.urandom(tamano_bytes)

    # Crear el nombre del archivo con la extensión especificada
    nombre_archivo = f"archivo_corrupto.{formato}"

    # Escribir los datos corruptos en el archivo
    with open(nombre_archivo, 'wb') as archivo:
        archivo.write(datos_corruptos)

    print(f"Archivo corrupto '{nombre_archivo}' de {tamano_mb} MB generado exitosamente.")

# Solicitar al usuario el formato del archivo y el tamaño en MB
formato = input("Introduce el formato del archivo (ej. txt, pdf, jpg): ")
tamano_mb = float(input("Introduce el tamaño del archivo en MB: "))

# Generar el archivo corrupto
generar_archivo_corrupto(formato, tamano_mb)
```

Introduce el formato del archivo (ej. txt, pdf, jpg): mp4
Introduce el tamaño del archivo en MB: 10
Archivo corrupto 'archivo_corrupto.mp4' de 10.0 MB generado exitosamente.