

Extracción de Datos con Python - Día 1

in-progress

40 min

Learning Objectives

- 1 Entender métodos básicos de extracción de datos
- 2 Aprender a conectar con diferentes fuentes de datos
- 3 Comprender conceptos de autenticación y seguridad
- 4 Conocer estrategias para extracción eficiente

Theory

Practice

Evidence

Quiz

« Practical exercise to apply the concepts learned.

Ejercicio: Extraer datos de múltiples fuentes

Leer datos de un archivo CSV:

```
import csv

def leer_csv(ruta_archivo):
    datos = []
    with open(ruta_archivo, 'r') as f:
        lector = csv.DictReader(f)
        for fila in lector:
            datos.append(fila)
    return datos

# Uso
clientes = leer_csv('clientes.csv')
print(f"Leídos {len(clientes)} clientes")
```

Simular extracción de API:

```
import json

def extraer_api_simulada():
    # Simular respuesta de API
    datos_api = {
        "productos": [
            {"id": 1, "nombre": "Producto A", "precio": 100},
            {"id": 2, "nombre": "Producto B", "precio": 200}
        ]
    }
    return datos_api["productos"]

productos = extraer_api_simulada()
print(f"Extraídos {len(productos)} productos")
```

Conectar a la base de datos SQLite:

```
import sqlite3

def conectar_base_datos():
    conn = sqlite3.connect(':memory:') # Base temporal
    cursor = conn.cursor()

    # Crear tabla
    cursor.execute('''
        CREATE TABLE ventas (
            id INTEGER PRIMARY KEY,
            producto TEXT,
            cantidad INTEGER
        )
    ''')

    # Insertar datos de ejemplo
    cursor.execute("INSERT INTO ventas VALUES (1, 'Producto A', 10)")
    cursor.execute("INSERT INTO ventas VALUES (2, 'Producto B', 5)")

    # Leer datos
    cursor.execute("SELECT * FROM ventas")
    resultados = cursor.fetchall()
```



[Dashboard](#)[Career Path](#)[Forms](#)[Profile](#)[Support](#)

```
conn.close()
return resultados

ventas = conectar_base_datos()
print(f"Encontradas {len(ventas)} ventas")
```

Verificación: ¿Qué consideraciones de seguridad debes tener al conectar con bases de datos y APIs? ¿Cómo manejarías errores de conexión o respuestas inválidas?

Requerimientos:

- Conocimiento básico de Python
- Familiaridad con archivos y bases de datos
- Comprensión de conceptos HTTP

▼

[Sign Out](#)