

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

Activities and Learning

Task 1: Conexiones a Bases de Datos (10 minutos)

¿Cómo conectar Python con bases de datos?

Python puede conectarse a cualquier base de datos usando librerías específicas o ORMs como SQLAlchemy.

Tipos de conexiones:

PostgreSQL: psycopg2 o SQLAlchemy
MySQL: pymysql o SQLAlchemy
SQLite: Librería estándar de Python
SQL Server: pyodbc

Ejemplo básico de conexión:

```
import sqlite3

# Conexión simple a SQLite
conn = sqlite3.connect('mi_base.db')
cursor = conn.cursor()

# Ejecutar consulta
cursor.execute("SELECT * FROM usuarios")
resultados = cursor.fetchall()

conn.close()
```

Task 2: Extracción desde APIs (10 minutos)

¿Cómo obtener datos de APIs web?

Las APIs REST son una fuente común de datos. Python usa la librería requests para hacer llamadas HTTP.

Conceptos importantes:

Métodos HTTP: GET (obtener), POST (enviar), PUT (actualizar)
Autenticación: API keys, tokens, OAuth
Rate limiting: Límites de llamadas por tiempo
Paginación: Múltiples páginas de resultados

Ejemplo de llamada API:

```
import requests

# Llamada simple a API
response = requests.get('https://api.ejemplo.com/datos')
if response.status_code == 200:
    datos = response.json()
    print(f"Obtenidos {len(datos)} registros")
```

Task 3: Lectura de Archivos (10 minutos)

¿Cómo leer diferentes formatos de archivos?

Python puede leer archivos de datos en múltiples formatos usando librerías especializadas.

Formatos comunes:



 Dashboard Career Path Forms Profile Support

CSV: Archivos de texto separados por comas
JSON: Datos estructurados en formato JavaScript
Excel: Hojas de cálculo (.xlsx, .xls)
Parquet: Formato columnar eficiente

Ejemplo de lectura de CSV:

```
import csv

# Leer archivo CSV
with open('datos.csv', 'r') as archivo:
    lector = csv.DictReader(archivo)
    for fila in lector:
        print(f"Nombre: {fila['nombre']}, Edad: {fila['edad']}")
```

[Sign Out](#)