



! TUTORÍA WEB SCRAPING CON PYTHON

¿Qué es el Web Scraping?

El **Web Scraping** es el proceso de:

Acceder automáticamente a una página web

Leer su contenido (HTML)

Extraer información concreta

Transformarla en datos estructurados

Librerías principales que vamos a usar

Selenium

- Controla un navegador web real
- Carga páginas dinámicas
- Ejecuta JavaScript
- Simula la navegación de un usuario



BeautifulSoup

- Analiza HTML y XML
- Navegación y búsqueda sencillas
- Maneja documentos mal formados
- Extracción de datos rápida



Pandas

- Organiza los datos en tablas (DataFrames)
- Permite limpiar, filtrar y actualizar datos
- Facilita el almacenamiento y análisis



Librerías de apoyo, para estos ejemplos:

- time: pausas para carga de páginas
- datetime: registro de fechas
- numpy: gestión de valores nulos
-

¿Cómo trabajan Selenium y BeautifulSoup juntos?

Flujo general



Selenium

Carga la página web

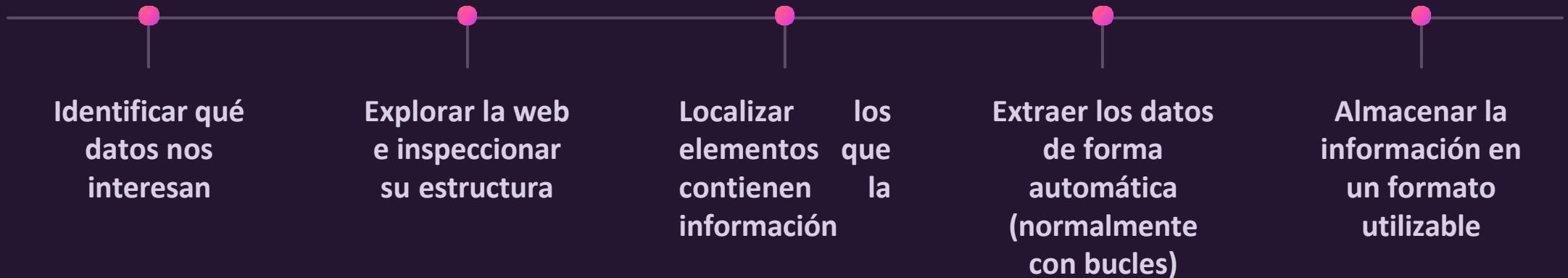
BeautifulSoup

Extracción de datos



Idea clave general: Selenium abre la web, BeautifulSoup la interpreta.

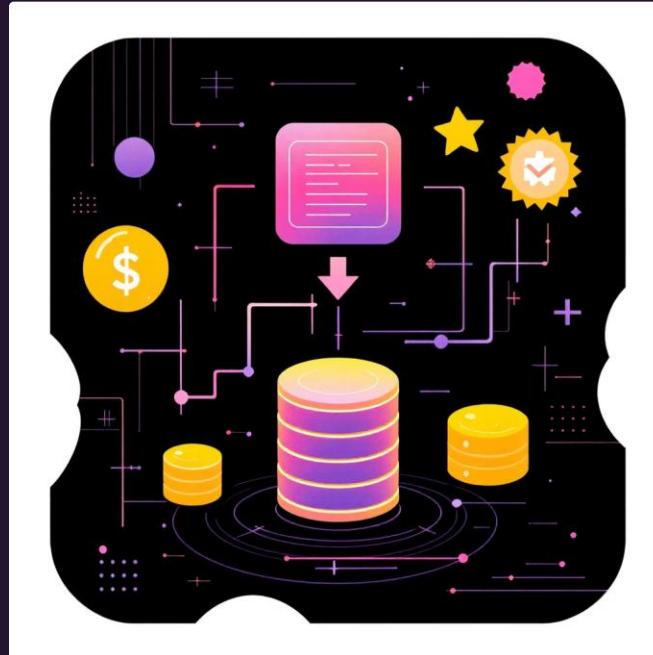
Proceso general de Web Scraping



Algunas aplicaciones que vamos a ver:

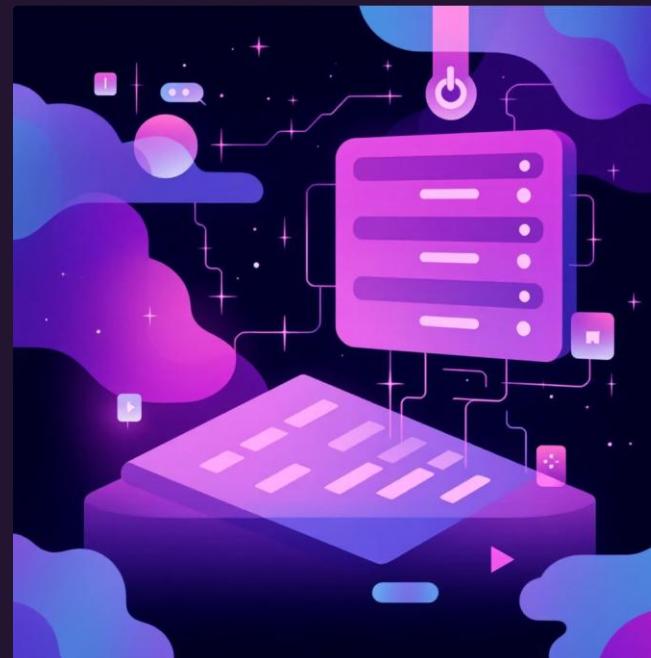
Crear datos

- Construcción de bases de datos desde una web



Corregir datos

- Completar valores nulos
- Actualizar información incompleta



Enriquecer datos

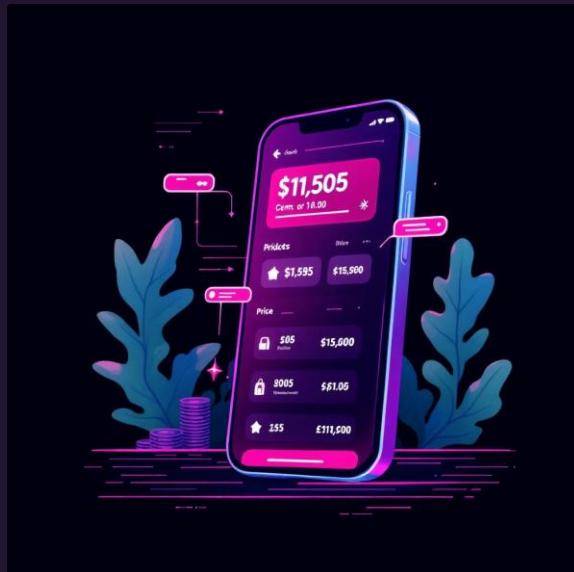
- Añadir contexto
- Traducir códigos técnicos a información legible



¿Qué vamos a ver en esta tutoría?

3 ejemplos prácticos de Web Scraping:

1 → Crear una base de datos de móviles y precios desde una web en diferentes fechas (por ejemplo, para analizar variaciones de precios en el tiempo...)



2 Gestión de valores nulos en una base de datos que contiene películas y fechas de estreno con nulos.



3 + Enriquecer una base de datos con información externa, en este caso de zonas FAO de pesca, asignarle su nombre a través de su ID.



El foco estará en entender cómo funciona, cómo aplicarlo y qué posibilidades ofrece.