Contenido

[Introducción 2](#_Toc175125335)

[Pandas 2](#_Toc175125336)

[Requests 2](#_Toc175125337)

[sqlite3 2](#_Toc175125338)

[beautifulsoup4 3](#_Toc175125339)

[Inicio del laboratorio 3](#_Toc175125340)

[Tipos de Datos Estructurados y No Estructurados 7](#_Toc175125341)

[1. Datos Estructurados 7](#_Toc175125342)

[2. Datos No Estructurados 7](#_Toc175125343)

[¿Por qué es conveniente usar pandas para Ciencia de Datos? 7](#_Toc175125344)

[¿Cómo se pueden leer datos desde diversas fuentes y tipos de archivos? 7](#_Toc175125345)

## Introducción

Pandas

Pandas es una biblioteca de Python que generalmente se usa para la manipulación y análisis de datos. Además, permite trabajar con DataFrames.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

### Requests

requests es una biblioteca que permite realizar peticiones HTTP y se usa generalmente para pedir datos de APIs y procesarlos.

Texto

Descripción generada automáticamente

### sqlite3

sqlite3 permite interactuar con bases de datos SQLite desde Python y facilita el manejo de BBDD sin la necesidad de configurar un servidor. Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

### beautifulsoup4

beautifulsoup4 es utilizada para hacer web scraping, lo que significa extraer datos de sitios web al analizar y manipular elementos.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

## Inicio del laboratorio

Para esta practica necesitamos instalar algunas librerías que hagan falta:

Texto

Descripción generada automáticamente

Y se importan las librerías correspondientes

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza baja

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Texto

Descripción generada automáticamente Texto

Descripción generada automáticamente

## Tipos de Datos Estructurados y No Estructurados

### 1. Datos Estructurados

Los **datos estructurados** son aquellos que están organizados en un formato predefinido y fácilmente interpretable por máquinas. Un ejemplo típico son las bases de datos relacionales, donde los datos se organizan en tablas con filas y columnas. Estos datos suelen tener un esquema fijo, como nombres de columnas y tipos de datos específicos (por ejemplo, enteros, cadenas de texto, etc.).

### 2. Datos No Estructurados

Los **datos no estructurados** no siguen un formato específico y, por lo tanto, son más difíciles de organizar, procesar y analizar. Estos incluyen archivos de texto, correos electrónicos, imágenes, videos, audios y publicaciones en redes sociales.

### ¿Por qué es conveniente usar pandas para Ciencia de Datos?

pandas es una de las bibliotecas más importantes ya que permite la manipulación de datos en Python.

Una de sus principales características son su alta **facilidad al usar** grandes volúmenes de datos. Pues el tipo DataFrames permiten manejar datos tabulares de manera eficiente y realizar operaciones complejas con poco código. Otra ventaja es que se pueden realizar **operaciones** como filtrar, agrupar, agregar y transformar datos. Lo que lo hace ideal para realizar análisis, preparar datos o manipular grandes conjuntos de datos de forma rápida. Y por último se permite una gran cantidad de **formatos** ya que se puede leer y escribir datos desde/para una amplia gama de formatos, incluidos CSV, Excel, SQL, JSON, etc.

### ¿Cómo se pueden leer datos desde diversas fuentes y tipos de archivos?

Pandas generalmente tiene diversas formas de poder realizar la lectura de archivos diferentes y hasta ahora hemos visto la funcion que permite leer de uno de los formatos de datos más utilizados. Cargando archivos CSV utilizando pd.read\_csv(), lo que convierte el archivo en un DataFrame para su manipulación.

