Pandas

Pandas es una de las bibliotecas de Python más populares y ampliamente utilizadas para el análisis de datos y la manipulación de datos estructurados. Fue desarrollada originalmente por Wes McKinney en 2008 y desde entonces se ha convertido en una herramienta esencial en el mundo de la ciencia de datos y el análisis de datos.

Las características principales de Pandas incluyen:

Estructuras de Datos: Pandas proporciona dos estructuras de datos principales: Series y DataFrames.

Series: Es una estructura unidimensional que puede contener datos de cualquier tipo (números, texto, fechas, etc.). Cada elemento de una Serie tiene una etiqueta asociada llamada índice.

DataFrames: Son estructuras de datos bidimensionales que se asemejan a una tabla de base de datos o una hoja de cálculo. Los DataFrames constan de filas y columnas, y pueden contener datos heterogéneos.

Lectura y Escritura de Datos: Pandas es capaz de leer datos desde una variedad de fuentes, incluyendo archivos CSV, Excel, bases de datos SQL y más. También permite escribir datos en diferentes formatos.

Manipulación de Datos: Pandas ofrece una amplia gama de funciones y métodos para limpiar, transformar y manipular datos. Esto incluye filtrar datos, eliminar duplicados, rellenar valores nulos, y más.

Indexación y Selección: Permite acceder y seleccionar datos de manera eficiente utilizando etiquetas de índice o índices enteros.

Agregación y Estadísticas: Facilita el cálculo de estadísticas descriptivas, como sumas, medias, medianas, varianzas, y permite realizar agregaciones más complejas mediante groupby().

Visualización de Datos: Pandas se integra bien con bibliotecas de visualización como ***Matplotlib*** y ***Seaborn*** para crear gráficos y visualizaciones de datos.

Manipulación de Fechas y Tiempo: Ofrece funcionalidades para trabajar con datos de fecha y hora de manera efectiva.

Combinación y Fusión de Datos: Permite combinar DataFrames y Series utilizando operaciones de fusión y concatenación.

Operaciones Matemáticas y Estadísticas: Facilita el cálculo de operaciones matemáticas y estadísticas en datos numéricos.

Pandas es ampliamente utilizado en campos como la ciencia de datos, la inteligencia de negocios, la ingeniería financiera, la investigación académica y muchas otras áreas donde se requiere el análisis y manipulación de datos. Es una herramienta esencial para cualquier persona que trabaje con datos en Python.

DataFrame

Un DataFrame es una estructura de datos bidimensional en la biblioteca de Python llamada Pandas. Es una de las estructuras de datos más fundamentales y utilizadas en análisis de datos y ciencia de datos. Un DataFrame

se asemeja a una tabla en una base de datos o una hoja de cálculo de Excel, y se utiliza para almacenar y manipular datos de manera organizada y eficiente.

Las características clave de un DataFrame incluyen:

Filas y Columnas: Un DataFrame consta de filas y columnas, donde cada fila representa una observación o un registro de datos, y cada columna representa una variable o atributo.

Etiquetas de Columna: Cada columna en un DataFrame tiene una etiqueta o nombre que la identifica. Estas etiquetas de columna se utilizan para acceder y manipular los datos en la columna.

Datos Homogéneos: A diferencia de las listas de Python, las columnas en un DataFrame generalmente contienen datos del mismo tipo, lo que facilita el análisis y la manipulación de los datos.

Indexación: Los DataFrames suelen tener un índice que identifica de manera única cada fila. Este índice se utiliza para acceder a filas específicas en el DataFrame.

Flexibilidad: Los DataFrames pueden contener una variedad de tipos de datos, incluyendo números, texto, fechas y más. También pueden manejar valores faltantes de manera eficiente.

Pandas es una biblioteca muy utilizada en Python para trabajar con DataFrames. Permite leer datos desde diferentes fuentes, realizar operaciones de filtrado, selección y agregación, fusionar y combinar DataFrames, y realizar análisis de datos de manera eficiente. Los DataFrames son esenciales para tareas de limpieza y preparación de datos, análisis exploratorio, y modelado de datos en proyectos de ciencia de datos y análisis de datos.

Funciones de Pandas

1. Estructuras de Datos

a. Series

Creación de Series: Puedes crear una Serie especificando una lista de datos y, opcionalmente, un índice.

```
import pandas as pd

data = [1, 2, 3, 4]
serie = pd.Series(data, index=['a', 'b', 'c', 'd'])
```

b. DataFrames

Creación de DataFrames: Puedes crear un DataFrame especificando un diccionario de Series o listas, donde cada clave representa el nombre de la columna.

```
data = {'Nombre': ['Juan', 'María', 'Pedro'],
        'Edad': [25, 30, 22]}
df = pd.DataFrame(data)
```

2. Lectura y Escritura de Datos

a. Lectura de Datos

Lectura de un archivo CSV: Pandas puede leer datos desde archivos CSV y otras fuentes.

```
df = pd.read_csv('datos.csv')
```

b. Escritura de Datos

Pandas permite escribir los resultados en diferentes formatos como csv, json, html entre muchos más.
Escritura en un archivo CSV: Puedes guardar un DataFrame en un archivo CSV.

```
df.to_csv('nuevo_datos.csv', index=False)
```

3. Manipulación de Datos

a. Filtrado de Datos

Filtrar filas: Puedes filtrar filas basadas en una condición.

```
df_filtrado = df[df['Edad'] > 25]
```

b. Eliminación de Duplicados

Eliminar filas duplicadas: Pandas permite eliminar filas duplicadas en un DataFrame.

```
df_sin_duplicados = df.drop_duplicates()
```

c. Rellenar Valores Nulos

Rellenar valores nulos: Puedes rellenar valores nulos con un valor específico.

```
df['Edad'].fillna(0, inplace=True)
```

4. Indexación y Selección

a. Selección por Etiquetas

Selección por etiquetas: Puedes seleccionar filas y columnas utilizando etiquetas de índice.

```
df.loc['Fila1', 'Columna1']
```

b. Selección por Posición

Selección por posición: Puedes seleccionar filas y columnas utilizando índices enteros.

```
df.iloc[0, 1]
```

5. Agregación y Estadísticas

a. Agregación

Agregación con groupby: Puedes agregar datos utilizando la función groupby() y luego aplicar funciones de agregación como sum(), mean(), max(), etc.

```
df.groupby('Categoría')['Ventas'].sum()
```

b. Estadísticas Descriptivas

Cálculo de estadísticas descriptivas: Puedes usar funciones como mean(), median(), max(), min(), std(), var(), etc., para calcular estadísticas descriptivas de columnas numéricas.

```
media = df['Puntaje'].mean()
```

sum(): Calcula la suma de valores en cada columna numérica.

```
df.sum()
```

count(): Calcula el número de valores no nulos en cada columna.

```
df.count()
```

mean(): Calcula la media (promedio) de valores en cada columna numérica.

```
df.mean()
```

max(): Encuentra el valor máximo en cada columna numérica.

```
df.max()
```

min(): Encuentra el valor mínimo en cada columna numérica

```
df.min()
```

Estas funciones complementan las capacidades de Pandas para el análisis y manipulación de datos. Con ellas, puedes realizar operaciones matemáticas y estadísticas, así como combinar y fusionar datos de diferentes fuentes en tus proyectos de análisis de datos.