Pandas is an open-source software library designed for **data manipulation** and **analysis**. It provides data structures like series and DataFrames to easily clean, transform and analyze large datasets and integrates with other Python libraries, such as NumPy and Matplotlib.

It offers functions for data transformation, aggregation and visualization, which are important for analysis. Created by Wes McKinney in 2008, Pandas widely used by data scientists, analysts and researchers worldwide. Pandas revolves around two primary Data structures: Series (1D) for single columns and DataFrame (2D) for tabular data enabling efficient data manipulation.

Pandas is an open-source software library for the Python programming language, primarily used for data manipulation and analysis. It provides powerful and flexible data structures designed to work with tabular and time-series data.

Key Features and Concepts:

* **Data Structures:**
  + **Series:** A one-dimensional labeled array capable of holding any data type (integers, strings, floating point numbers, Python objects, etc.). It can be thought of as a single column in a spreadsheet.
  + **DataFrame:** A two-dimensional labeled data structure with columns of potentially different types. It is similar to a spreadsheet or a SQL table, providing rows and columns for organized data.
* **Data Manipulation and Analysis:**

Pandas offers a wide range of functionalities for:

* + **Data Loading and Saving:** Reading and writing data from various formats like CSV, Excel, SQL databases, JSON, etc.
  + **Data Cleaning:** Handling missing data (e.g., filling or dropping null values), removing duplicates, and correcting inconsistencies.
  + **Data Transformation:** Filtering, sorting, grouping, merging, joining, and reshaping data.
  + **Data Aggregation:** Calculating summary statistics like mean, median, sum, count, etc.
  + **Time Series Analysis:** Specific tools for working with time-indexed data.

**Basic Setup**

import pandas as pd

s1 = pd.Series([1, 2, 3, 4], index=['a', 'b', 'c', 'd'])

s2 = pd.Series([10, 20, 30, 40], index=['b', 'c', 'd', 'e'])

**➕ Arithmetic Operations**

Pandas aligns Series on the index before performing operations:

# Element-wise addition

s1 + s2

# Subtraction

s1 - s2

# Multiplication

s1 \* s2

# Division

s1 / s2

**🔁 Fill Missing Values During Operations**

# Fill missing values with 0 before addition

s1.add(s2, fill\_value=0)

# Other methods:

# s1.sub(s2, fill\_value=0)

# s1.mul(s2, fill\_value=0)

# s1.div(s2, fill\_value=1)

**🔍 Comparison Operations**

s1 > 2

s1 == s2

s1 < s2

**🔢 Aggregation Functions**

s1.sum()

s1.mean()

s1.std()

s1.max()

s1.min()

s1.median()

**🧹 Element-wise Functions**

s1.apply(lambda x: x \*\* 2)

s1.map({1: 'one', 2: 'two'}) # Map values using dict

**🎯 Index-based Selection**

s1['a'] # Access by label

s1[0] # Access by position

s1[['a', 'c']]

s1[1:3] # Slice (like lists)

**🧪 Filtering**

s1[s1 > 2] # Keep values greater than 2

**🔄 Sorting**

s1.sort\_values() # Sort by values

s1.sort\_index() # Sort by index

**🔗 Combining Series**

pd.concat([s1, s2])

s1.combine\_first(s2) # Fill missing values in s1 with those from s2