**mean() function in Python / Pandas**

The **mean()** function calculates the **average** of numeric values.

**1 Python list example:**

numbers = [10, 20, 30, 40, 50]

avg = sum(numbers) / len(numbers)

print(avg) # Output: 30.0

* mean() essentially does sum(values) / count(values).

**2. Pandas series example:**

import pandas as pd

s = pd.Series([10, 20, 30, 40, 50])

print(s.mean())

**Output:**

30.0

* Pandas mean() works directly on **Series** or **DataFrame columns**.
* It **ignores NaN values** by default.

**3 Pandas DataFrame example:**

df = pd.DataFrame({

"Maths": [88, 92, 45, 67],

"Physics": [76, 85, 67, 90]

})

print(df["Maths"].mean()) # Average of Maths

print(df.mean()) # Average of all numeric columns

**Output:**

73.0

Maths 73.0

Physics 79.5

dtype: float64

* df.mean() calculates **mean column-wise** by default.
* You can also use axis=1 to get **row-wise mean**:

df.mean(axis=1)

**In short:**

mean() = average = sum of values ÷ number of values

* Works on **Python lists**, **NumPy arrays**, **Pandas Series**, and **DataFrame columns**.
* Ignores NaN by default in Pandas.