

pandas-1

November 11, 2025

1 Pandas

- It introduces two key objects Series, DataFrames
- A series is a sequence of data like list in python or 1D array in NumPy
- DataFrame is the multiple series of common length with common index, bound together in a single tabular object

```
[2]: # Installing pandas
```

```
! pip install pandas --break-system-packages
```

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pandas in /home/bsh/.local/lib/python3.13/site-packages (2.2.3)

Requirement already satisfied: numpy>=1.26.0 in
/home/bsh/.local/lib/python3.13/site-packages (from pandas) (2.2.2)

Requirement already satisfied: python-dateutil>=2.8.2 in
/usr/lib/python3.13/site-packages (from pandas) (2.9.0)

Requirement already satisfied: pytz>=2020.1 in /usr/lib/python3.13/site-packages
(from pandas) (2025.2)

Requirement already satisfied: tzdata>=2022.7 in
/home/bsh/.local/lib/python3.13/site-packages (from pandas) (2025.1)

Requirement already satisfied: six>=1.5 in /usr/lib/python3.13/site-packages
(from python-dateutil>=2.8.2->pandas) (1.17.0)

```
[3]: import pandas as pd
```

```
[4]: from pandas import Series, DataFrame
```

```
[5]: # Series using list
```

```
s1 = Series([1, 2, 3, 4, 5])  
s1
```

```
[5]: 0    1  
     1    2  
     2    3  
     3    4
```

```
4    5
dtype: int64
```

```
[6]: s2 = Series(["Ram", "Shyam", "Hari", "Gopal"])
s2
```

```
[6]: 0    Ram
1    Shyam
2    Hari
3    Gopal
dtype: object
```

```
[7]: # Using custom index

idx = pd.Index(["Kathmandu", "Lalitpur", "Bhaktapur", "Pokhara", "Chitwan"])
idx
```

```
[7]: Index(['Kathmandu', 'Lalitpur', 'Bhaktapur', 'Pokhara', 'Chitwan'],
dtype='object')
```

```
[20]: population = Series([88888, 12345, 42342, 42312, 52311], index=idx,
↳ name="Population")
population
```

```
[20]: Kathmandu    88888
Lalitpur        12345
Bhaktapur       42342
Pokhara         42312
Chitwan         52311
Name: Population, dtype: int64
```

```
[9]: population["Kathmandu"]
```

```
[9]: np.int64(88888)
```

```
[11]: # Series using dictionary (district and its area)

area = Series({
    "Kathmandu": 123.5,
    "Lalitpur": 312.1,
    "Chitwan": 2131
}, name="Area")

area
```

```
[11]: Kathmandu    123.5
Lalitpur       312.1
Chitwan       2131.0
```

Name: Area, dtype: float64

```
[14]: # DataFrame using matrix
```

```
import numpy as np
import pandas as pd

m = np.arange(9).reshape(3, 3)
print(m)

df = pd.DataFrame(m)
print(df)
```

```
[[0 1 2]
 [3 4 5]
 [6 7 8]]
   0  1  2
0  0  1  2
1  3  4  5
2  6  7  8
```

```
[16]: # Adding index and columns
```

```
# index = left column
# columns = top row
```

```
df = pd.DataFrame(
    m,
    index = ["a", "b", "c"],
    columns = ["p", "q", "r"]
)
df
```

```
[16]:    p  q  r
a  0  1  2
b  3  4  5
c  6  7  8
```

```
[17]: # Q. Make a dataframe having index names and columns age, weight, height
```

```
[22]: # Using series
```

```
df = pd.DataFrame([population, area])
df
```

```
[22]:
```

	Kathmandu	Lalitpur	Bhaktapur	Pokhara	Chitwan
Population	88888.0	12345.0	42342.0	42312.0	52311.0
Area	123.5	312.1	NaN	NaN	2131.0

```
[23]: df = pd.DataFrame([population, area])  
df.T
```

```
[23]:
```

	Population	Area
Kathmandu	88888.0	123.5
Lalitpur	12345.0	312.1
Bhaktapur	42342.0	NaN
Pokhara	42312.0	NaN
Chitwan	52311.0	2131.0

```
[ ]:
```