Basic Pandas

```
import pandas as pd
```

create dataframe from scratch

```
# create dataframe from scratch

raw_data = {
    "name": ['Alice', 'Bob', 'Charlie', 'Diana', 'Ethan'],
    "age": [28, 32, 25, 30, 27],
    "gender": ['F', 'M', 'M', 'F', 'M'],
    "country": ['USA', 'Canada', 'USA', 'UK', 'USA']
}

df = pd.DataFrame(raw_data)

df
```

	name	age	gender	country
0	Alice	28	F	USA
1	Bob	32	M	Canada
2	Charlie	25	M	USA
3	Diana	30	F	UK
4	Ethan	27	M	USA

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
## Add a new column
df["city"] = ["New York", "Toronto", "Los Angeles", "London", "Chicago"]


df
```

	name	age	gender	country	city
0	Alice	28	F	USA	New York
1	Bob	32	M	Canada	Toronto
2	Charlie	25	M	USA	Los Angeles
3	Diana	30	F	UK	London
4	Ethan	27	M	USA	Chicago

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df.shape
```

```
(5, 5)
```

```
 ## drop column city
 df = df.drop('city', axis = 1) # axis 0 == row, 1 == column


df
```

	name	age	gender	country
0	Alice	28	F	USA
1	Bob	32	M	Canada
2	Charlie	25	M	USA
3	Diana	30	F	UK
4	Ethan	27	M	USA

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
## remove index(row) = 2

df = df.drop(2, axis = 0)

df
```

	name	age	gender	country
0	Alice	28	F	USA
1	Bob	32	M	Canada
3	Diana	30	F	UK
4	Ethan	27	M	USA

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
# reset index
df = df.reset_index(drop = True)
df
```

	name	age	gender	country
0	Alice	28	F	USA
1	Bob	32	M	Canada
2	Diana	30	F	UK
3	Ethan	27	M	USA

Next steps: Generate code with df | View recommended plots | New interactive sheet

Rename columns

```
# rename column

list(df.columns)
```

```
['name', 'age', 'gender', 'country']
```

```
df.columns = ['first_name', 'age', 'gender', 'country']
df
```

	first_name	age	gender	country
0	Alice	28	F	USA
1	Bob	32	M	Canada
2	Diana	30	F	UK
3	Ethan	27	M	USA

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df['first_name']
```

	first_name
0	Alice
1	Bob
2	Diana
3	Ethan

dtype: object

Series vs. Dataframe

```
type(df['first_name'])
```

```
pandas.core.series.Series
def __init__(data=None, index=None, dtype: Dtype | None=None, name=None, copy: bool | None=None,
fastpath: bool | lib.NoDefault=lib.no_default) -> None
```

```
1      2
dtype: int64

Due to input data type the Series has a `view` on
the original data, so
the data is changed as well.
```

```
type(df)
```

```
pandas.core.frame.DataFrame
def __init__(data=None, index: Axes | None=None, columns: Axes | None=None, dtype: Dtype |
None=None, copy: bool | None=None) -> None
```

```
/usr/local/lib/python3.11/dist-packages/pandas/core/frame.py
Two-dimensional, size-mutable, potentially heterogeneous tabular data.

Data structure also contains labeled axes (rows and columns).
Arithmetic operations align on both row and column labels. Can be
thought of as a dict-like container for Series objects. The primary
```

```
# create a new series
s1 = pd.Series(['Charl', 25, 'M', 'USA'], index = ['first_name', 'age', 'gender', 'country'])
print(s1)
print(type(s1))
```

```
first_name    Charl
age              25
gender            M
country         USA
dtype: object
<class 'pandas.core.series.Series'>
```

```
df
```

	first_name	age	gender	country
0	Alice	28	F	USA
1	Bob	32	M	Canada
2	Diana	30	F	UK
3	Ethan	27	M	USA

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
# append s1 to df
df = pd.concat([df, s1.to_frame().T], ignore_index=True)
df
```

	first_name	age	gender	country
0	Alice	28	F	USA
1	Bob	32	M	Canada
2	Diana	30	F	UK
3	Ethan	27	M	USA
4	Charl	25	M	USA

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
# create new column
s2 = pd.Series(['New York', 'Toronto', 'Los Angeles', 'London', 'Chicago'])
df['city'] = s2
df
```

	first_name	age	gender	country	city
0	Alice	28	F	USA	New York
1	Bob	32	M	Canada	Toronto
2	Diana	30	F	UK	Los Angeles
3	Ethan	27	M	USA	London
4	Charl	25	M	USA	Chicago

Next steps: Generate code with df | View recommended plots | New interactive sheet

Write CSV file

```
# write CSV file
df.to_csv('mydata.csv', index =False)
```

Import csv file

```
# import csv file
df2 = pd.read_csv('data.csv')
df2
```

	id	name	city
0	1	John	London
1	2	Joe	Liverpool
2	3	Mary	Manchester
3	4	Anna	Swansea
4	5	David	London

Next steps: Generate code with df2 | View recommended plots | New interactive sheet

Import excel file

```
df3 = pd.read_excel('data.xlsx')
df3
```

	id	name	city
0	1	John	London
1	2	Joe	Liverpool
2	3	Mary	Manchester
3	4	Anna	Swansea
4	5	David	London

Next steps: Generate code with df3 | View recommended plots | New interactive sheet

Import json file

```
# import json file
df4 = pd.read_json('data.json')
df4
```

	ebook	language	amazonRating	myFavorite
0	Getting started with Python	python	4.89	True
1	Introduction to R	r	4.88	False
2	SQL for Beginners	sql	4.75	True

Next steps: Generate code with df4 | View recommended plots | New interactive sheet