PANDAS BASIC

PANEL DATA

prepared by:

Gyro A. Madrona
Electronics Engineer

TOPIC OUTLINE

Pandas

Pandas DataFrame

Pandas Series

Common Operations in Pandas

PANDAS

PANDAS

pandas is an open-source Python library designed for efficient data manipulation and analysis. It provides data structures like **Series** and **DataFrames** to effectively clean, transform, and analyze large datasets and integrates seamlessly with other Python libraries, such as numpy and matplotlib.

https://pandas.pydata.org/

PANDAS PACKAGE

loading pandas package

```
import pandas as pd
```

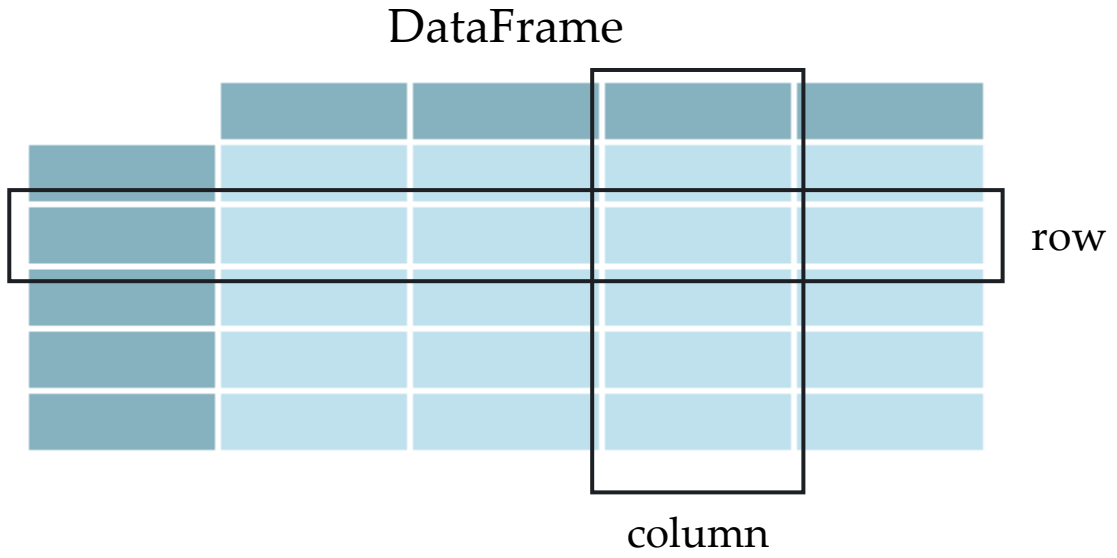
The community agreed alias for pandas is **pd** , so loading pandas as pd is assumed standard practice for all of the pandas documentation.

https://pandas.pydata.org/

PANDAS DATAFRAME

DataFrame is a **two-dimensional** data structure that stores data in columns, where each column contains values of a single data type (e.g., `int`, `str`, `float`). However, different columns can have different data types.

Pandas data table representation



CREATING DATAFRAME

```
import pandas as pd

data = {

    "Name":['Henry', 'Owen', 'Ada'],

    "Age":[22,35,58],

    "Sex":['M','M','F']

    }

df = pd.DataFrame(data)
```

DataFrame

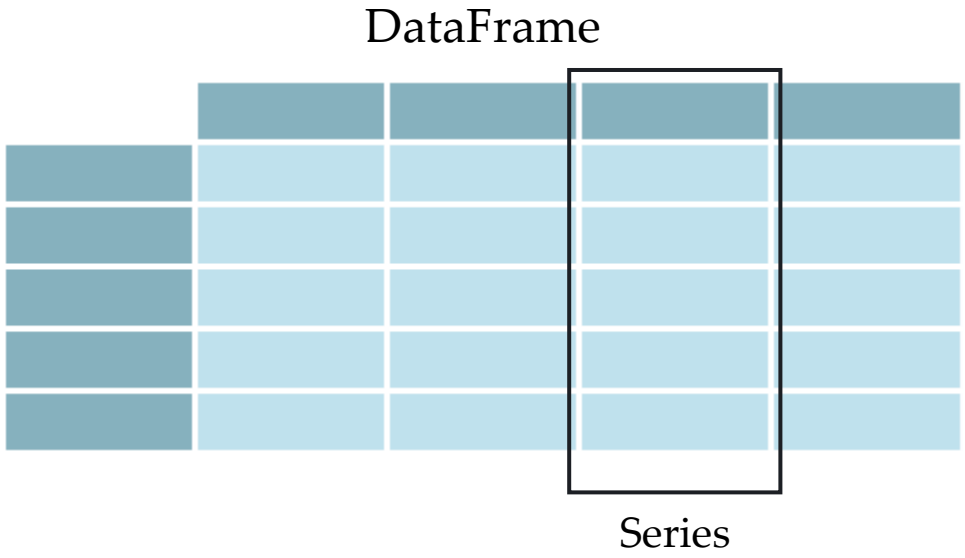
Name	Age	Sex
Henry	22	M
Owen	35	M
Ada	58	F

When using a Python dictionary of lists, the dictionary **keys** will be used as **column headers** and the values in each list as columns of the DataFrame.

PANDAS SERIES

pandas Series is a **one-dimensional** labeled array that can hold data of a single type (e.g., `int, float, str`).

pandas data table representation



Each column in a DataFrame is a Series.

SELECTING A COLUMN

df["Age"]

```
0    22
1    35
2    58
Name: Age, dtype: int64
```

df["Name"]

```
0    Henry
1     Owen
2      Ada
Name: Name, dtype: object
```

df["Sex"]

```
0    M
1    M
2    F
Name: Sex, dtype: object
```

DataFrame

Name	Age	Sex
Henry	22	M
Owen	35	M
Ada	58	F

Each column in a DataFrame is a Series.

CREATING A SERIES

```
import pandas as pd

name = pd.Series(["Henry","Owen","Ada"])

age = pd.Series([22,35,58])

sex = pd.Series(['M','M','F'])


df = pd.DataFrame({
    "Name":name,
    "Age":age,
    "Sex":sex
})
```

DataFrame

Name	Age	Sex
Henry	22	M
Owen	35	M
Ada	58	F

You can create a DataFrame from multiple Series.

COMMON OPERATIONS IN PANDAS

COMMON OPERATIONS

1. Reading Data

```
df = pd.read_csv('data.csv')  # read a CSV file
```

2. Viewing Data

```
df.head()  # display the first 5 rows

df.tail()  # display the last 5 rows

df.info()  # summary of the DataFrame

df.describe()  # statistical summary
```

3. Selecting Data

```
df['column_name']  # select a single column

df[['column1', 'column2']]  # select multiple columns

df.iloc[0]  # select row by index
```

COMMON OPERATIONS

4. Handling Missing Data

```
df.dropna()  # drop rows with missing values

df.fillna(0) # fill missing values with 0
```

5. Data Manipulation

```
df.sort_values('column_name')  # sort by column

df.groupby('column_name').mean()  # Group by column and calculate mean
```

6. Exporting Data

```
df.to_csv('output.csv', index=False)  # Export to CSV
```

LABORATORY