Python with Pandas

Urban Data Analysis Course

Week- 3

Topics Covered

DataFrames: Introduction and Manipulation

Data Visualization: Histograms, Bar Plots, Pie Charts, Heatmaps, and Line Plots

Working with Lists: Data Cleaning and Transformation

Introduction

- **Objective**: Learn how to use Python with Pandas for data manipulation, cleaning, and visualization.
- **Tools Used**:
    - Pandas for DataFrames
    - Matplotlib and Seaborn for Visualizations
    - Geopandas for Geospatial Data (optional)
- **Dataset**: NYC Bridge Strike Data

Import	import pandas as pd # For data manipulation
Import	import seaborn as sns # For data visualization
Import	import matplotlib.pyplot as plt # For plotting graphs
Import	import geopandas as gpd # For working with geospatial data

Loading Data

df =
pd.read_csv('NYC_
Bridge_Strike_Data
_20250207.csv')

print(df.head()) #
Display the first few
rows

print(df.info()) #
Overview of data
types and non-null
counts

Explanation:

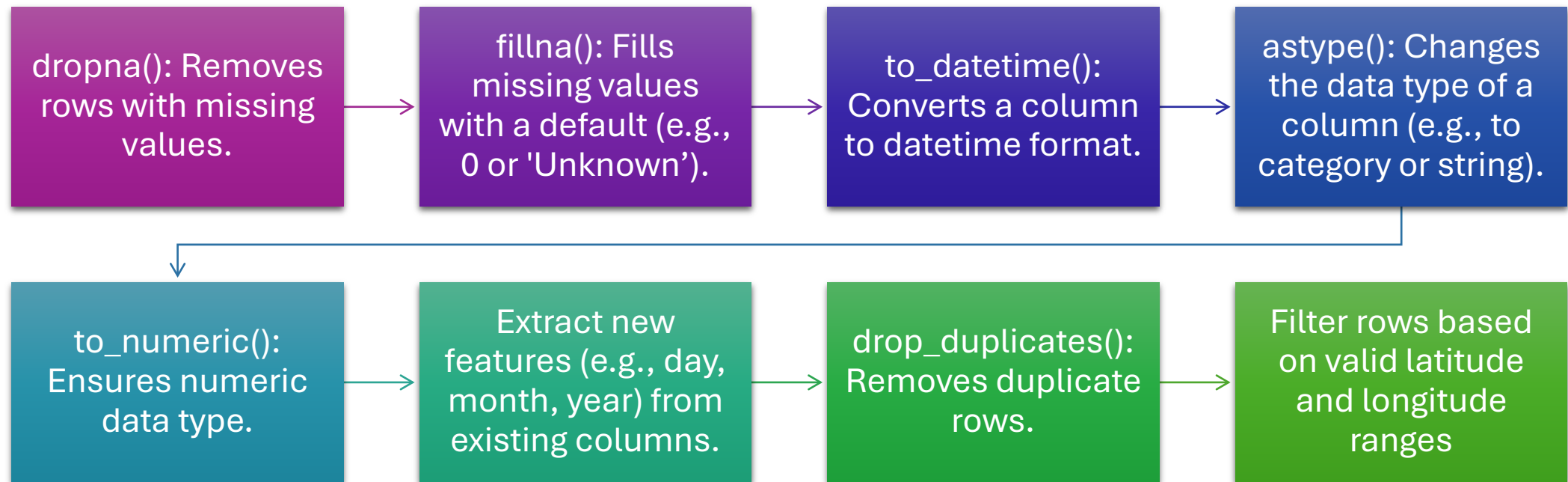read_csv(): Loads data from a CSV file into a DataFrame.

head(): Displays the first 5 rows.

info(): Provides metadata (e.g., column names, data types, non-null counts).

Summary Table

Method	Library	Use Case	Example
pd.read_csv()	Pandas	CSV files	df = pd.read_csv('data.csv')
pd.read_excel()	Pandas	Excel files	df = pd.read_excel('data.xlsx')
pd.read_json()	Pandas	JSON files	df = pd.read_json('data.json')
pd.read_sql()	Pandas	SQL databases	df = pd.read_sql('SELECT * FROM table')
pd.read_parquet()	Pandas	Parquet files	df = pd.read_parquet('data.parquet')
np.loadtxt()	NumPy	Numerical text files	data = np.loadtxt('data.txt')
json.load()	Built-in	JSON files	data = json.load(file)
gpd.read_file()	Geopandas	Geospatial data	gdf = gpd.read_file('data.shp')

Data Processing

dropna(): Removes rows with missing values.	fillna(): Fills missing values with a default (e.g., 0 or 'Unknown').	to_datetime(): Converts a column to datetime format.	astype(): Changes the data type of a column (e.g., to category or string).

to_numeric(): Ensures numeric data type.	Extract new features (e.g., day, month, year) from existing columns.	drop_duplicates(): Removes duplicate rows.	Filter rows based on valid latitude and longitude ranges

Data Visualization

- Bar plots are used to visualize the distribution of categorical data.

- Pie charts are used to show proportions of categorical data.

- Heatmaps are used to visualize relationships between two categorical variables.

- Line plots are used to show trends over time.
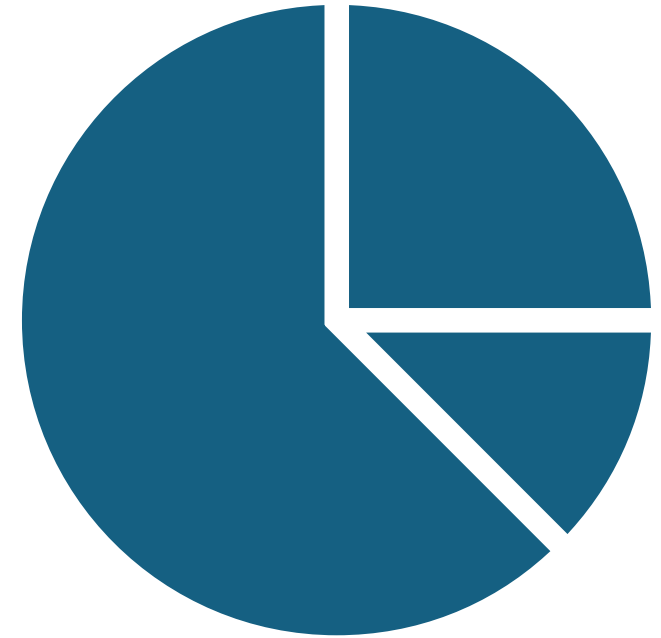
Working with Lists

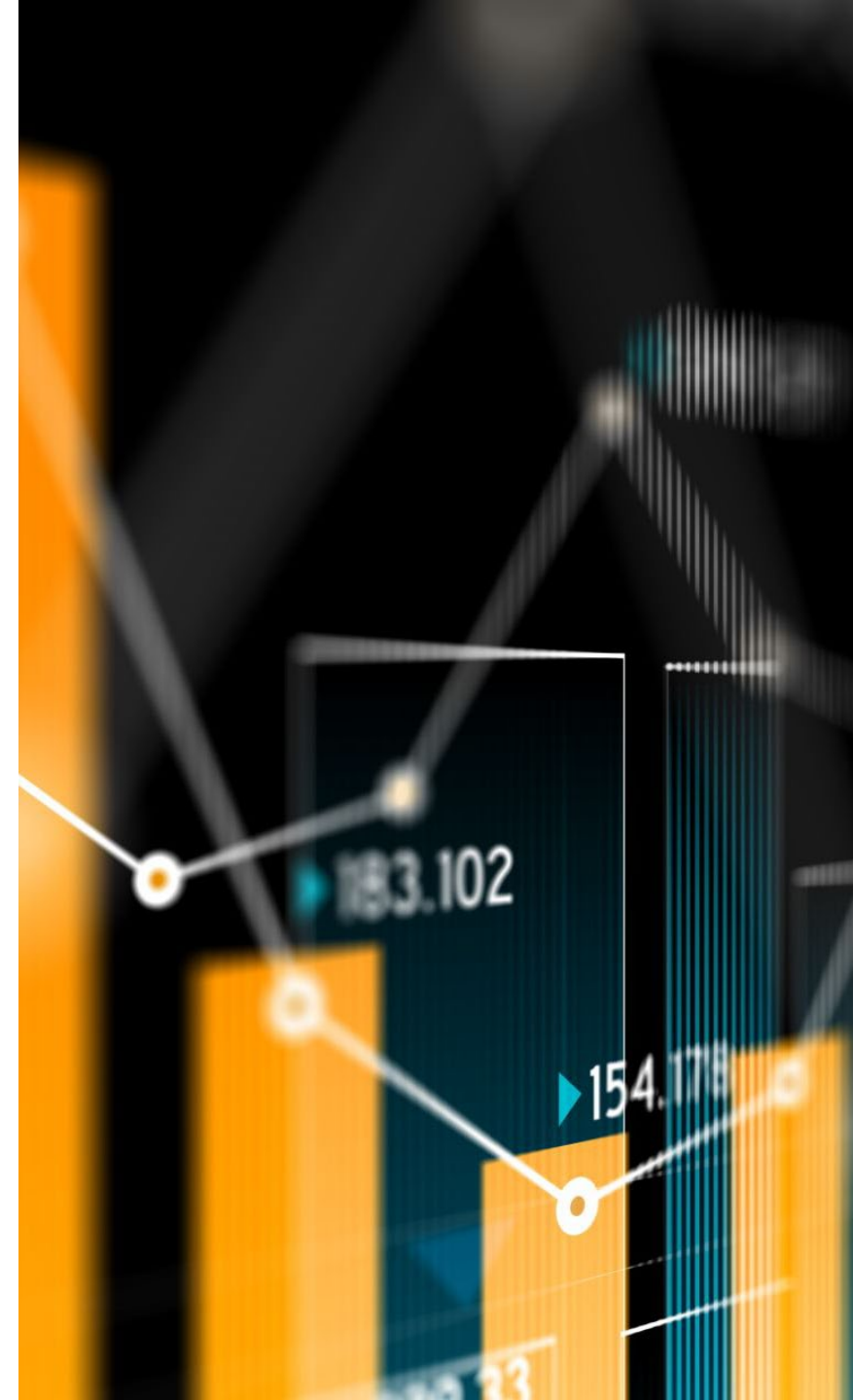# Example: Convert a column to a list

borough_list = df['BOROUGH'].tolist()

print(borough_list[:5]) # Display first 5 elements

- **Explanation**:
  - tolist(): Converts a DataFrame column to a Python list.
  - Lists are useful for further data manipulation or analysis.

- **Key Takeaways**:
  - Pandas is a powerful tool for data manipulation and cleaning.
  - Visualizations (e.g., bar plots, pie charts, heatmaps) help uncover insights.
  - Working with lists enables flexible data handling.

Next Steps: