Complete Pandas Mastery Roadmap

Stage 0: Prerequisites

Skill	Tools
☑ Python basics	variables, loops, conditionals, functions
☑ NumPy basics	arrays, indexing, broadcasting
☑ Jupyter Notebook	basic usage, shortcuts

Stage 1: Pandas Basics (Foundation)

Goal: Understand Pandas data structures and how to work with them.

◇ Topics to Cover:

- What is Pandas and when to use it?
- Series & DataFrame (creation, indexing, slicing)
- `pd.Series()` and `pd.DataFrame()` objects
- Basic attributes: `.index`, `.columns`, `.shape`, `.dtype`
- Viewing data: `.head()`, `.tail()`, `.info()`, `.describe()`

Practice:

- Create DataFrames from dictionaries and lists
- Try `.iloc`, `.loc`, and direct column selection

Stage 2: Data Input/Output (I/O)

Goal: Load and export data in various formats.

◇ Topics to Cover:

- `pd.read_csv()`, `.read_excel()`, `.read_json()`
- `df.to_csv()`, `.to_excel()`, `.to_json()`
- Working with file paths and large files (`chunksize`, `iterator`)

Practice:

- Import a CSV from your PC and clean it
- Save cleaned version using `.to_csv()`

Stage 3: Data Manipulation & Cleaning

Goal: Handle real-world messy data like a pro.

◇ Topics to Cover:

- Selecting, filtering, conditional logic
- Adding, modifying, deleting columns/rows
- Handling missing values: `.isnull()`, `.dropna()`, `.fillna()`
- Renaming columns, replacing values
- Detecting & removing duplicates

▧ Practice:

- Clean a dataset (fill NAs, drop duplicates)
- Create a new derived column using `.apply()`

🔁 Stage 4: Data Transformation

Goal: Reshape and transform data for analysis.

◇ Topics to Cover:

- Sorting: `.sort_values()`, `.sort_index()`
- Grouping: `.groupby()` and `.agg()`
- Pivot Tables: `.pivot_table()`
- Reshaping: `.melt()`, `.stack()`, `.unstack()`
- String operations: `.str.lower()`, `.str.contains()`
- Date handling: `pd.to_datetime()`, `.dt.year`, `.dt.month`

▧ Practice:

- Group and analyze dataset by category
- Create a pivot table showing aggregated sales

🔗 Stage 5: Merging and Joining

Goal: Combine multiple datasets efficiently.

◇ Topics to Cover:

- Concatenation: `pd.concat()`
- Merging: `pd.merge()` (inner, outer, left, right joins)
- `.join()` and `set_index()` vs `reset_index()`

▧ Practice:

- Merge two datasets using a common key
- Try different join types and observe results

📐 Stage 6: Advanced Analysis

Goal: Perform complex analyses using Pandas.

◇ Topics to Cover:

- Multi-indexing
- Custom functions with `.apply()` and `lambda`
- Window functions: `.rolling()`, `.expanding()`
- Correlation matrix: `.corr()`, `.cov()`
- Value counts and binning: `.value_counts()`, `pd.cut()`, `pd.qcut()`

▧ Practice:

- Calculate rolling average on time-series data
- Use `.corr()` to find relationships

Stage 7: Visualization with Pandas

Goal: Quickly visualize data for insights.

◇ Topics to Cover:

- `.plot()` built-in wrapper (matplotlib backend)
- Line plot, bar chart, histogram, box plot
- Integrating with Seaborn and Matplotlib

▧ Practice:

- Plot bar chart of category vs frequency
- Plot time series and histogram

Stage 8: Performance & Optimization

Goal: Work efficiently with large datasets.

◇ Topics to Cover:

- Memory usage optimization: `.memory_usage()`
- Use of `categorical` data types
- Vectorization vs iteration
- Use `query()` and `eval()` for speed
- Chunk processing with `read_csv(chunksize=...)`

▧ Practice:

- Profile and reduce memory of a large DataFrame
- Use `query()` to filter efficiently

Stage 9: Real Projects & Practice

Goal: Apply Pandas in real-world datasets.

◇ Ideas:

- Analysis of IPL, COVID-19, or Indian census data

- EDA on a Kaggle dataset (Titanic, Netflix, etc.)
- Build an Excel automation tool with Pandas

📓 Practice:

- Take a Kaggle dataset, perform full cleaning + analysis
- Share the Jupyter notebook on GitHub or LinkedIn

🛠️ Tools & Resources

- 📓 Official Pandas Docs
- 🎥 Corey Schafer Pandas YouTube
- 📓 Pandas Cheat Sheet – DataCamp

🏁 Stage 10: Mastery

Now You're a Pandas Ninja If You Can: ☑ Clean any raw data
☑ Merge multiple dataframes efficiently
☑ Create pivot tables and grouped analysis
☑ Automate repetitive tasks using `.apply()` and `.groupby()`
☑ Work on 1M+ rows using chunking and optimization
☑ Explain Pandas concepts clearly to others