



Glossary – Module 07: Working with Data

This glossary covers key terms from working with core Python containers and Pandas for data manipulation.



Python Core Concepts

- **List:** A mutable, ordered Python container allowing duplicates.
 - **Tuple:** An immutable, ordered Python container.
 - **Set:** An unordered collection of unique elements.
 - **Dictionary:** A collection of key–value pairs.
-



Pandas Structures

- **Series:** A one-dimensional labeled array capable of holding any data type.
 - **DataFrame:** A two-dimensional, tabular data structure with labeled axes (rows and columns).
-



Loading and Exploring Data

- **`read_csv()`:** Loads a CSV file into a DataFrame.
 - **`.head()`:** Shows the first 5 rows of a DataFrame.
 - **`.info()`:** Prints summary of DataFrame including dtypes and missing values.
 - **`.describe()`:** Summary statistics for numeric (or all) columns.
-



Cleaning and Fixing Data

- **Missing Value (NaN):** A placeholder for missing or null entries.
 - **`.isnull()` / `.notnull()`:** Boolean mask of missing or present data.
 - **`.fillna()`:** Fills missing values with specified constant or method.
 - **`.dropna()`:** Drops rows or columns with missing values.
 - **`pd.to_numeric()`:** Converts a Series to numeric type; can coerce errors to NaN.
 - **`pd.to_datetime()`:** Converts string/object to datetime format.
-



Column Operations

- **`.rename()`:** Renames column(s) using a dictionary of `{old: new}`.
 - **`.drop(columns=...)`:** Removes column(s) from the DataFrame.
 - **`.apply()`:** Applies a function across a Series (column) or DataFrame.
-



Sorting and Filtering

- **`.sort_values()`:** Sorts rows by specified column(s).
- **Filtering:** Extracting rows based on conditional logic, e.g. `df[df["col"] > 10]`.
- **Boolean Masking:** Using logical operations (`&`, `|`, `~`) to filter rows.

- **.query()**: Filters rows using a SQL-like syntax, e.g. `df.query("score > 90")`.
-

Wrangling Lab Concepts

- **Feature Engineering**: Creating new columns from existing data to enhance analysis.
 - **GroupBy**: Aggregates rows based on a shared column value.
 - **.value_counts()**: Shows frequency counts of unique values in a Series.
 - **.groupby()**: Groups rows and applies aggregation functions like `.mean()`, `.sum()`, etc.
 - **.to_csv()**: Writes the DataFrame to a CSV file.
-

Use this glossary as a cheat sheet during assignments, labs, or quiz prep. Mastering these terms means you're well on your way to becoming a data wrangling ninja  