Pandas Cheatsheet

Pandas is a powerful library for data manipulation and analysis in Python.

---

1. Creating DataFrames and Series

- **pd.Series(data)** - Creates a one-dimensional Series.
- **pd.DataFrame(data)** - Creates a two-dimensional DataFrame.
- **pd.read_csv('file.csv')** - Reads a CSV file into a DataFrame.
- **pd.read_excel('file.xlsx')** - Reads an Excel file into a DataFrame.
- **pd.read_json('file.json')** - Reads a JSON file into a DataFrame.
- **df.to_csv('file.csv')** - Saves a DataFrame to a CSV file.
- **df.to_excel('file.xlsx')** - Saves a DataFrame to an Excel file.
- **df.to_json('file.json')** - Saves a DataFrame to a JSON file.

---

2. Viewing and Inspecting Data

- **df.head(n)** - Displays the first n rows.
- **df.tail(n)** - Displays the last n rows.
- **df.info()** - Displays concise DataFrame info.
- **df.describe()** - Summarizes numerical columns.
- **df.shape** - Returns the number of rows and columns.
- **df.columns** - Returns column names.
- **df.index** - Returns the index of the DataFrame.
- **df.dtypes** - Displays data types of each column.
- **df.isnull().sum()** - Checks for missing values.

---

3. Selecting and Filtering Data

- **df['column']** - Selects a single column.
- **df[['col1', 'col2']]** - Selects multiple columns.
- **df.loc[row_label, col_label]** - Selects data by label.
- **df.iloc[row_index, col_index]** - Selects data by position.
- **df[df['column'] > value]** - Filters rows based on condition.

- **df.query('column > value')** - Filters using a query string.

---

4. Modifying Data

- **df['new_col'] = value** - Adds a new column.
- **df.drop(columns=['col1'])** - Drops specified columns.
- **df.drop(index=[row_index])** - Drops specified rows.
- **df.rename(columns={'old': 'new'})** - Renames columns.
- **df.fillna(value)** - Fills missing values.
- **df.dropna()** - Drops missing values.
- **df.replace({old: new})** - Replaces values in a column.

---

5. Grouping and Aggregation

- **df.groupby('column').sum()** - Groups by a column and sums.
- **df.groupby('column').mean()** - Groups by a column and averages.
- **df.groupby('column').count()** - Groups by a column and counts.
- **df.pivot(index, columns, values)** - Pivots a DataFrame.
- **df.melt(id_vars, value_vars)** - Unpivots a DataFrame.

---

6. Sorting and Ordering

- **df.sort_values('column')** - Sorts values by a column.
- **df.sort_index()** - Sorts by index.

---

7. Merging and Joining

- **pd.concat([df1, df2])** - Concatenates DataFrames.
- **df1.merge(df2, on='column')** - Merges DataFrames on a column.
- **df1.join(df2, lsuffix='_left', rsuffix='_right')** - Joins DataFrames on index.

---

8. Applying Functions

- **df.apply(func, axis=1)** - Applies function to rows.
- **df.apply(func, axis=0)** - Applies function to columns.
- **df['column'].map(func)** - Maps a function to a column.

- **df['column'].apply(lambda x: x*2)** - Applies lambda function to a column.

---

9. Time Series Analysis

- **df['date'] = pd.to_datetime(df['date'])** - Converts column to datetime.
- **df.set_index('date')** - Sets a datetime index.
- **df.resample('M').sum()** - Resamples data to monthly frequency.

---

10. Exporting Data

- **df.to_csv('file.csv')** - Saves DataFrame to CSV.
- **df.to_excel('file.xlsx')** - Saves DataFrame to Excel.
- **df.to_json('file.json')** - Saves DataFrame to JSON.

Official documentation: https://pandas.pydata.org/docs/