Pandas Operations Guide

Pandas Operations

1. Creating DataFrames and Series:
   - Series: A one-dimensional array, similar to a list or column.
   - DataFrame: A two-dimensional table, similar to a spreadsheet or SQL table.

2. Reading Data:
   - read_csv(): Read data from a CSV file into a DataFrame.
   - read_excel(): Read data from an Excel file into a DataFrame.
   - read_sql(): Read data from a SQL database into a DataFrame.

3. Data Selection and Indexing:
   - Selecting a column: df['Age']
   - Selecting multiple columns: df[['Name', 'Age']]
   - Selecting rows by index (iloc): df.iloc[0]
   - Selecting rows by label (loc): df.loc[0]

4. Data Cleaning:
   - Handling missing data:
     - Fill missing values: df.fillna(0)
     - Drop rows with missing values: df.dropna()
   - Renaming columns: df.rename(columns={'OldName': 'NewName'})
   - Replacing values: df.replace('old_value', 'new_value')

5. Data Transformation:

- Sorting: df.sort_values(by='Age', ascending=False)
- Changing column types: df['Age'] = df['Age'].astype(float)
- Apply functions: df['Age'] = df['Age'].apply(lambda x: x + 1)

6. Aggregating and Grouping Data:

- Groupby: df.groupby('Age').mean()
- Summing values: df['Age'].sum()
- Aggregating data: df.groupby('Age').agg({'Salary': ['sum', 'mean']})

7. Merging and Joining:

- Concatenating data: pd.concat([df1, df2], axis=0)
- Merging dataframes: pd.merge(df1, df2, on='common_column')

8. Pivot Tables:

- Creating a Pivot Table: df.pivot_table(values='Salary', index='Age', columns='Department', aggfunc='mean')

9. Plotting Data:

- Using pandas built-in plotting: df['Age'].plot(kind='bar')

10. Working with Time Series:

- Convert a column to datetime: df['Date'] = pd.to_datetime(df['Date'])
- Resampling: df.resample('M').sum()

11. Handling Duplicates:

- Dropping duplicates: df.drop_duplicates()

- Checking for duplicates: df.duplicated()