**Assignment**

**Pandas for data processing**

Pandas is a powerful and popular Python library for data manipulation and analysis. It provides data structures like DataFrame and Series, which make it easy to work with structured data. Here are some key aspects of using Pandas for data processing:

1. **Data Structures:**
   * **DataFrame:** A 2-dimensional labelled data structure with columns that can be of different types. It's like a spreadsheet or SQL table.
   * **Series:** A 1-dimensional labelled array that can hold any data type.
2. **Loading Data:**
   * Pandas can read data from various file formats, such as CSV, Excel, SQL databases, and more.
   * Example: pd.read\_csv('filename.csv')
3. **Data Exploration:**
   * Inspecting the first or last few rows of the dataset using head() or tail().
   * Getting summary statistics using describe().
4. **Data Selection and Filtering:**
   * Selecting columns or rows using column names or boolean conditions.
   * Example: df['column\_name'] or df[df['column\_name'] > 10]
5. **Handling Missing Data:**
   * Pandas provides methods like dropna() to remove missing values and fillna() to fill or interpolate missing values.
6. **Data Transformation:**
   * Changing data types, renaming columns, and creating new columns.
   * Example: df['new\_column'] = df['column1'] + df['column2']
7. **Grouping and Aggregation:**
   * Grouping data based on certain criteria and performing aggregate functions on each group.
   * Example: df.groupby('category').mean()
8. **Merging and Joining DataFrames:**
   * Combining multiple DataFrames based on a common column or index.
   * Example: pd.merge(df1, df2, on='common\_column')
9. **Time Series Data:**
   * Pandas provides excellent support for working with time series data, including date/time indexing and resampling.

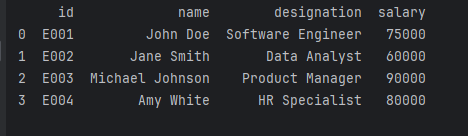
**Reading CSV data using Pandas**

**import pandas as pd**

**pandas\_file = pd.read\_csv("new\_file.csv")**

**print(pandas\_file.to\_string())**

**Output**

****

**Read data from CSv file to Pandas data frame**

**import numpy as np**

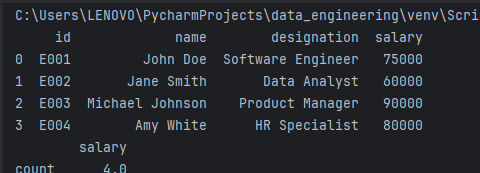
**import pandas as pd**

**pandas\_file = pd.read\_csv("new\_file.csv")**

**data = pd.DataFrame(pandas\_file)**

**print(data)**

**Output**

****

**Filter data in pandas dataframe using query**

**import pandas as pd**

**# Create a sample DataFrame**

**data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],**

**'Age': [25, 30, 22, 35],**

**'Salary': [50000, 60000, 45000, 70000]}**

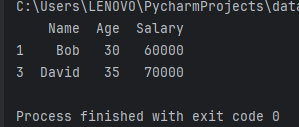
**df = pd.DataFrame(data)**

**# Use the query method to filter data**

**filtered\_df = df.query('Age > 25 and Salary > 50000')**

**print(filtered\_df)**

**Output**

****

**Get count by status using DataFrame API**

**import pandas as pd**

**# Create a sample DataFrame**

**data = {'ID': [1, 2, 3, 4, 5],**

**'Status': ['Completed', 'In Progress', 'Completed', 'Cancelled', 'In Progress']}**

**df = pd.DataFrame(data)**

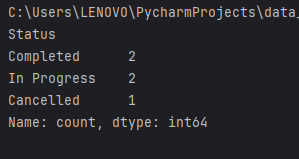
**# Get the count of each status**

**status\_counts = df['Status'].value\_counts()**

**# Print the result**

**print(status\_counts)**

**Output**

****

**Get count by Month and Status using dataFrame API**

**import pandas as pd**

**# Create a sample DataFrame**

**data = {'ID': [1, 2, 3, 4, 5],**

**'Date': ['2022-01-15', '2022-01-20', '2022-02-10', '2022-02-15', '2022-03-05'],**

**'Status': ['Completed', 'In Progress', 'Completed', 'Cancelled', 'In Progress']}**

**df = pd.DataFrame(data)**

**# Convert the 'Date' column to datetime**

**df['Date'] = pd.to\_datetime(df['Date'])**

**# Create 'Month' column**

**df['Month'] = df['Date'].dt.to\_period('M')**

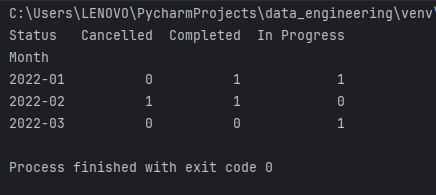
**# Get the count by month and status**

**count\_by\_month\_status = df.groupby(['Month', 'Status']).size().unstack(fill\_value=0)**

**# Print the result**

**print(count\_by\_month\_status)**

**Output**

****

**Create Dataframe using dynamic column list on csv data**

**import pandas as pd**

**import io**

**# Sample CSV data**

**csv\_data = "Name,Age,Salary\nAlice,25,50000\nBob,30,60000\nCharlie,22,45000"**

**# Define dynamic column names**

**dynamic\_columns = ['EmployeeName', 'EmployeeAge', 'EmployeeSalary']**

**# Create DataFrame**

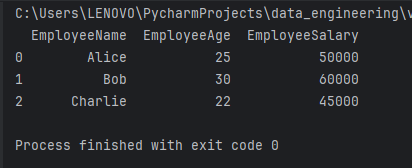
**df = pd.read\_csv(io.StringIO(csv\_data))**

**df.columns = dynamic\_columns**

**# Print the resulting DataFrame**

**print(df)**

**Output**

****

**Perform Inner join between Pandas dataFrame**

**import pandas as pd**

**# Create two sample DataFrames**

**df1 = pd.DataFrame({'ID': [1, 2, 3], 'Name': ['Alice', 'Bob', 'Charlie']})**

**df2 = pd.DataFrame({'ID': [2, 3, 4], 'Age': [25, 30, 22]})**

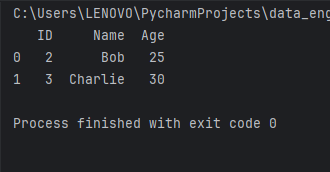
**# Perform inner join based on the 'ID' column**

**result\_df = pd.merge(df1, df2, on='ID', how='inner')**

**# Print the resulting DataFrame**

**print(result\_df)**

**Output**

****

**Perform aggregation on join results**

**import pandas as pd**

**# Create two sample DataFrames**

**df1 = pd.DataFrame({'ID': [1, 2, 3], 'Name': ['Alice', 'Bob', 'Charlie']})**

**df2 = pd.DataFrame({'ID': [2, 3, 4], 'Age': [25, 30, 22]})**

**# Perform inner join based on the 'ID' column**

**result\_df = pd.merge(df1, df2, on='ID', how='inner')**

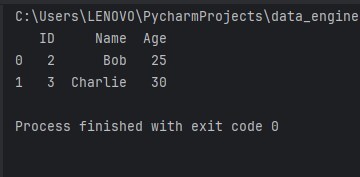
**# Perform aggregation on the joined DataFrame**

**aggregated\_df = result\_df.groupby('ID').agg({'Name': 'first', 'Age': 'mean'}).reset\_index()**

**# Print the resulting DataFrame**

**print(aggregated\_df)**

**Output**

****

**Sort data in a pandas dataframe**

**import pandas as pd**

**# Create a sample DataFrame**

**data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],**

**'Age': [25, 30, 22, 35],**

**'Salary': [50000, 60000, 45000, 70000]}**

**df = pd.DataFrame(data)**

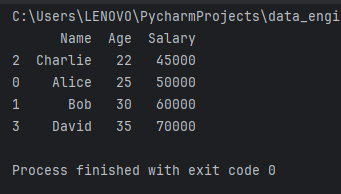
**# Sort DataFrame by a specific column (e.g., 'Age')**

**sorted\_df = df.sort\_values(by='Age')**

**# Print the sorted DataFrame**

**print(sorted\_df)**

**Output**

****

**Write Pandas dataframe to file**

**import pandas as pd**

**# Create a sample DataFrame**

**data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],**

**'Age': [25, 30, 22, 35],**

**'Salary': [50000, 60000, 45000, 70000]}**

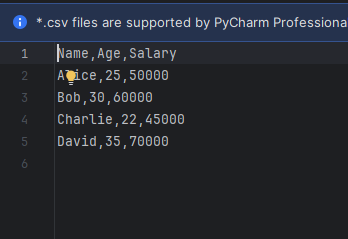
**df = pd.DataFrame(data)**

**# Write DataFrame to a CSV file**

**df.to\_csv('output.csv', index=False)**

**Output**

**output.csv**

****

**Write Pandas dataframe to JSON file**

**import pandas as pd**

**# Create a sample DataFrame**

**data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],**

**'Age': [25, 30, 22, 35],**

**'Salary': [50000, 60000, 45000, 70000]}**

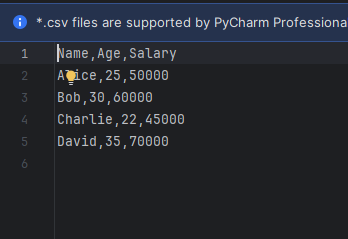
**df = pd.DataFrame(data)**

**# Write DataFrame to a CSV file**

**df.to\_csv('output.csv', index=False)**

**output**

**output.csv**

****