Akilesh K

k.akilesh123@gmail.com

Data engineering - Batch 1

Date: 01-01-24

DAY 10-PYTHON-DATA PROCESSING

Pandas for Data Processing

```
In [31]: print(df.describe())
```

```
              cgpa      year
count     6.000000  6.000000
mean      8.966667  2.000000
std       0.598888  0.632456
min       7.800000  1.000000
25%       9.025000  2.000000
50%       9.100000  2.000000
75%       9.250000  2.000000
max       9.500000  3.000000
```

- It provides functions and methods to handle missing data, and reshape datasets.
- It makes it easy to explore and understand your data.
- It allows you to filter and select data based on conditions.

Reading CSV Data using Pandas

In [16]:
```
import pandas as pd

csv_file_path = 'output.csv'

with open(csv_file_path, 'r') as file:
    content = file.read()
    print(content)
```

```
branch,cgpa,name,year
COE,9.0,Nikhil,2
COE,9.1,Sanchit,2
IT,9.3,Aditya,2
SE,9.5,Sagar,1
MCE,7.8,Prateek,3
EP,9.1,Sahil,2
```

Read Data from CSV Files to Pandas Dataframes

In [15]:
```
import pandas as pd

path = 'output.csv'

df = pd.read_csv(path)

print(df)
```

	branch	cgpa	name	year
0	COE	9.0	Nikhil	2
1	COE	9.1	Sanchit	2
2	IT	9.3	Aditya	2
3	SE	9.5	Sagar	1
4	MCE	7.8	Prateek	3
5	EP	9.1	Sahil	2

Filter Data in Pandas Dataframe using query

In [17]:
```
filtered_df = df.query('year > 1 and cgpa > 8.0')

print("\nFiltered DataFrame:")
print(filtered_df)
```

```
Filtered DataFrame:
  branch  cgpa     name  year
0    COE   9.0   Nikhil     2
1    COE   9.1  Sanchit     2
2     IT   9.3   Aditya     2
5     EP   9.1    Sahil     2
```

Get Count by Status using Pandas Dataframe APIs

In [1]:
```
import pandas as pd

data = {'status': ['Completed', 'InProgress', 'Completed', 'Pending', 'InProgress', 'Completed']}
df = pd.DataFrame(data)

status_counts = df['status'].value_counts()

print(status_counts)
```

```
status
Completed     3
InProgress    2
Pending       1
Name: count, dtype: int64
```

Get count by Month and Status using Pandas Dataframe APIs

In [2]:
```
import pandas as pd

data = {'date': ['2022-01-01', '2022-01-01', '2022-02-01', '2022-02-01', '2022-02-01'],
        'status': ['Completed', 'InProgress', 'Completed', 'Pending', 'InProgress']}
df = pd.DataFrame(data)

df['date'] = pd.to_datetime(df['date'])

df['month'] = df['date'].dt.to_period('M')

count_by_month_status = df.groupby(['month', 'status']).size().reset_index(name='count')

print(count_by_month_status)
```

```
     month      status  count
0  2022-01   Completed      1
1  2022-01  InProgress      1
2  2022-02   Completed      1
3  2022-02  InProgress      1
4  2022-02     Pending      1
```

Create Dataframes using dynamic column list on CSV Data

In [8]:
```
import pandas as pd

csv_file_path = 'industry.csv'
df = pd.read_csv(csv_file_path)

dynamic_column_list = ['column1', 'column3', 'column4']

selected_df = df[dynamic_column_list]

print(selected_df)
```

Performing Inner Join between Pandas Dataframes

In [9]:
```
import pandas as pd

df1 = pd.DataFrame({'ID': [1, 2, 3], 'Name': ['John', 'Alice', 'Bob']})
df2 = pd.DataFrame({'ID': [2, 3, 4], 'Age': [25, 30, 22]})

result_df = pd.merge(df1, df2, on='ID', how='inner')

print(result_df)
```

```
   ID   Name  Age
0   2  Alice   25
1   3    Bob   30
```

Perform Aggregations on Join results

In [10]:
```
import pandas as pd

df1 = pd.DataFrame({'ID': [1, 2, 3], 'Name': ['John', 'Alice', 'Bob']})
df2 = pd.DataFrame({'ID': [2, 3, 4], 'Age': [25, 30, 22]})

result_df = pd.merge(df1, df2, on='ID', how='inner')

aggregated_df = result_df.groupby('ID').agg({'Name': 'first', 'Age': 'mean'}).reset_index()

print(aggregated_df)
```

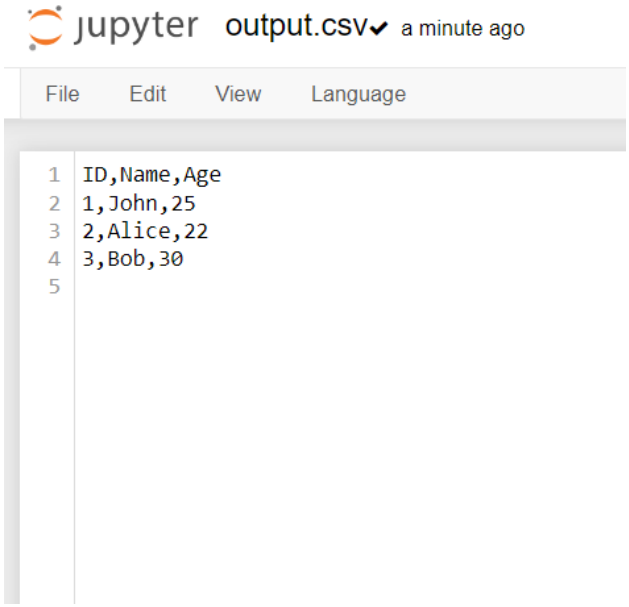
```
   ID   Name   Age
0   2  Alice  25.0
1   3    Bob  30.0
```

Sort Data in Pandas Dataframes

In [11]:
```
import pandas as pd

df = pd.DataFrame({'ID': [3, 1, 2, 4],
                   'Name': ['John', 'Alice', 'Bob', 'Eve'],
                   'Age': [25, 22, 30, 22]})

sorted_df = df.sort_values(by='ID')

print(sorted_df)
```

```
   ID   Name  Age
1   1  Alice   22
2   2    Bob   30
0   3   John   25
3   4    Eve   22
```

Writing Pandas Dataframes to Files



Write Pandas Dataframes to JSON Files