Coding Challenge-2
Python

Name: Ajay Chaudhary
Batch: Data Engineering (Batch-1)

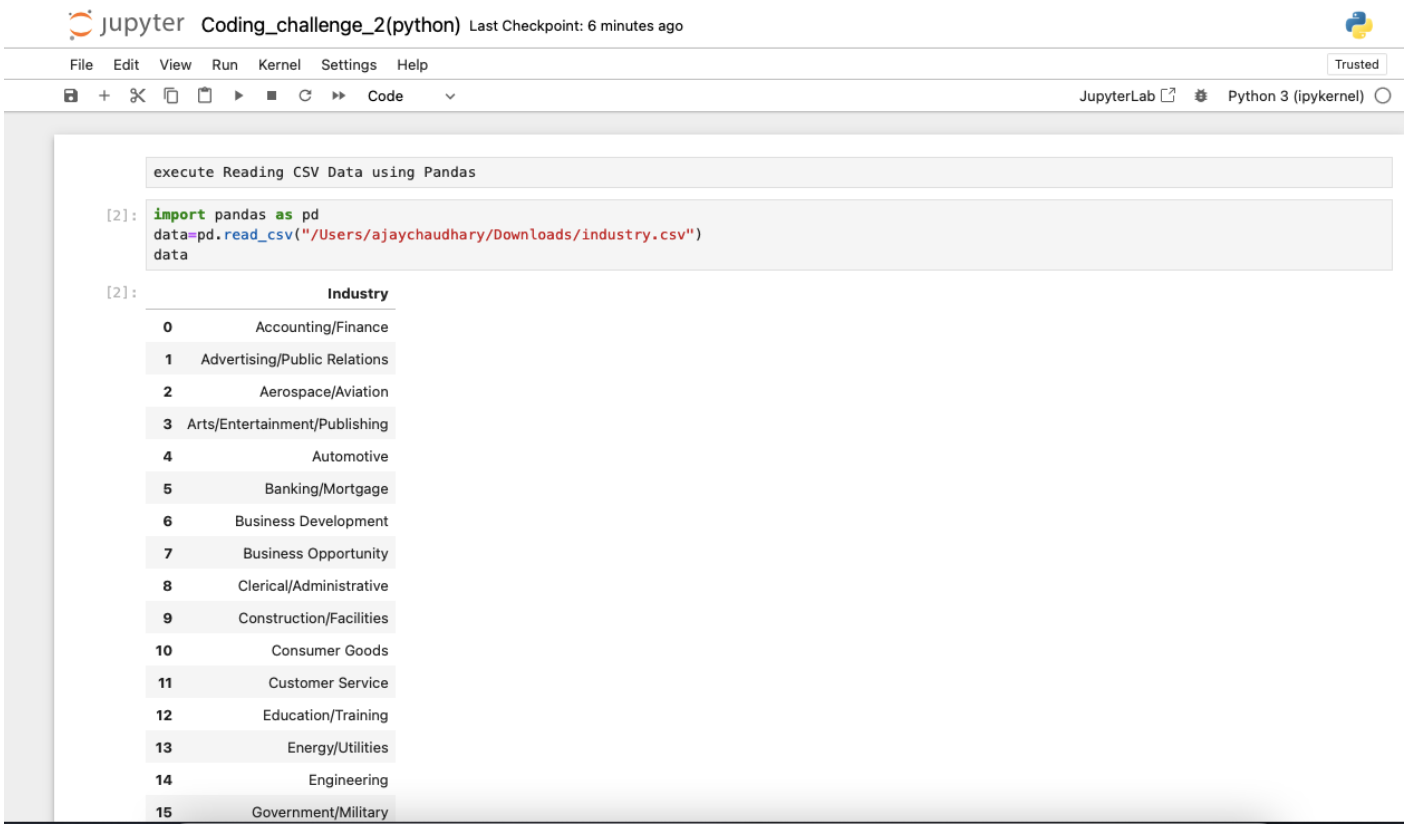
Explain Pandas for Data Processing & execute Reading CSV Data using Pandas

Pandas is a popular Python library for data manipulation and analysis. It provides data structures like Series and DataFrame, along with a wide range of functions for data cleaning, exploration, and transformation. Pandas is particularly useful for working with tabular data, making it a powerful tool for data processing and analysis.

Import Pandas: This line imports the Pandas library and assigns it the alias pd.

Read CSV File: The pd.read_csv() function is used to read data from a CSV file.

Display the DataFrame: The line 'data' shows the output.



	Industry
0	Accounting/Finance
1	Advertising/Public Relations
2	Aerospace/Aviation
3	Arts/Entertainment/Publishing
4	Automotive
5	Banking/Mortgage
6	Business Development
7	Business Opportunity
8	Clerical/Administrative
9	Construction/Facilities
10	Consumer Goods
11	Customer Service
12	Education/Training
13	Energy/Utilities
14	Engineering
15	Government/Military

Read Data from CSV Files to Pandas Dataframes

Importing the pandas library – this line will import the pandas library and give a alias name as pd– 'import pandas as pd'

Reading CSV data into a DataFrame - df = pd.read_csv("data.csv") this will read the pre-existing CSV file.

Displaying DataFrame - And finally the print statement will print the output

[4]:
```
# Importing the pandas library
import pandas as pd

# Reading CSV data into a DataFrame
df = pd.read_csv("data.csv")

# Displaying DataFrame
print(df)
```

```
    Name  M1 Score  M2 Score
0    Mic        65        85
1  Tyson        50        60
2   Bill        85        98
```

Filter Data in Pandas Dataframe using query.

[5]:
```
#Importing the Pandas library
import pandas as pd

# Creating a sample DataFrame
data = {
    'Name': ['James', 'William', 'Charles', 'Hazel', 'Thomas'],
    'Age': [22, 28, 26, 40, 35],
    'Salary': [60000, 450000, 52000, 65000, 75000]
}

df = pd.DataFrame(data)

# Using the query method to filter data
filtered_df = df.query('Age > 30 and Salary > 50000')

# Displaying the filtered DataFrame
print(filtered_df,"\n")

# Filter data where Age is between 20 and 30 and Salary is greater than 45000
filtered_df = df.query('20 <= Age <= 30 and Salary > 50000')

# Displaying the filtered DataFrame
print(filtered_df,"\n")
```

```
     Name  Age  Salary
3   Hazel   40   65000
4  Thomas   35   75000

      Name  Age  Salary
0    James   22   60000
1  William   28  450000
2  Charles   26   52000
```

Execute with one example Lambda Functions in Python

Lambda functions in Python are anonymous functions created using the lambda keyword. A lambda function can take any number of arguments, but can only have one expression.

[7]:
```
# implementation of lambda function
# lambda function for adding two numbers
lambda_add = lambda x, y: x + y

result_lambda = lambda_add(5, 3)

print("Result from lambda function:", result_lambda)


#another example for lambda function
# List of numbers
numbers = [1, 2, 3, 4, 5]

# Using the map function with a lambda function
squared_numbers_lambda = list(map(lambda x: x ** 2, numbers))
print("Second example")
# Displaying the results
print("Original numbers:", numbers)
print("Squared numbers (lambda function):", squared_numbers_lambda)
```

```
Result from lambda function: 8
Second example
Original numbers: [1, 2, 3, 4, 5]
Squared numbers (lambda function): [1, 4, 9, 16, 25]
```

Read JSON Strings to Python dicts or lists

1. Import json module and then define JSON string
2. Converting JSON string to Python dictionary by passing it to json.loads() in parameter.
3. Print the dictionary and their values using the keys as seen in the output.

[8]:
```
# Import JSON module
import json

# Define JSON string
jsonString = '{ "id": 1001, "name": "Ajay Chaudhary", "course": "Data Engineering"}'

# Convert JSON String to Python
student_details = json.loads(jsonString)

# Print Dictionary
print(student_details)

# Print values using keys
print(student_details['name'])
print(student_details['course'])
```

```
{'id': 1001, 'name': 'Ajay Chaudhary', 'course': 'Data Engineering'}
Ajay Chaudhary
Data Engineering
```