Name: Mohan Krishna Balaji

Reg No:192324221

Course Code: CSA0406

Course Name: Fundamental Data Science

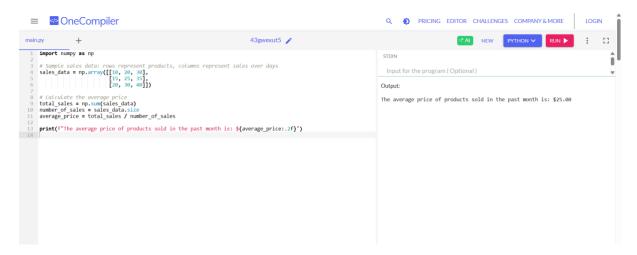
1. **Scenario:** You are working on a project that involves analyzing student performance data for a class of 32 students. The data is stored in a NumPy array named student_scores, where each row represents a student and each column represents a different subject. The subjects are arranged in the following order: Math, Science, English, and History. Your task is to calculate the average score for each subject and identify the subject with the highest average score.

Question: How would you use NumPy arrays to calculate the average score for each subject and determine the subject with the highest average score? Assume 4x4 matrix that stores marks of each student in given order.



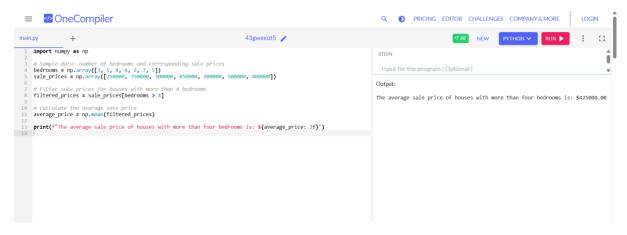
2. **Scenario:** You are a data analyst working for a company that sells products online. You have been tasked with analyzing the sales data for the past month. The data is stored in a NumPy array.

Question: How would you find the average price of all the products sold in the past month? Assume 3x3 matrix with each row representing the sales for a different product



3. **Scenario:** You are working on a project that involves analyzing a dataset containing information about houses in a neighborhood. The dataset is stored in a CSV file, and you have imported it into a NumPy array named house_data. Each row of the array represents a house, and the columns contain various features such as the number of bedrooms, square footage, and sale price.

Question: Using NumPy arrays and operations, how would you find the average sale price of houses with more than four bedrooms in the neighborhood?



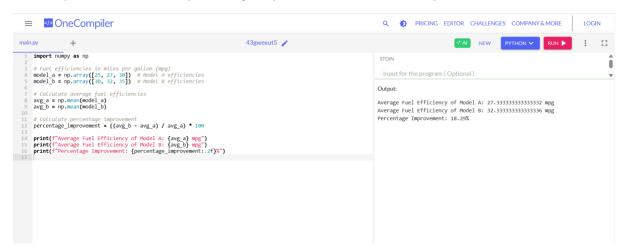
4. **Scenario:** You are working on a project that involves analyzing the sales performance of a company over the past four quarters. The quarterly sales data is stored in a NumPy array named sales_data, where each element represents the sales amount for a specific quarter. Your task is to calculate the total sales for the year and determine the percentage increase in sales from the first quarter to the fourth quarter.

Question: Using NumPy arrays and arithmetic operations calculate the total sales for the year and determine the percentage increase in sales from the first quarter to the fourth quarter?



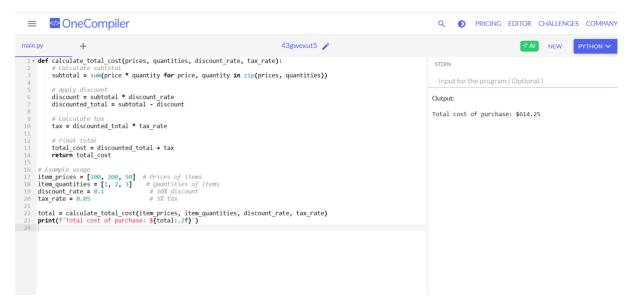
5. **Scenario:** You are a data analyst working for a car manufacturing company. As part of your analysis, you have a dataset containing information about the fuel efficiency of different car models. The dataset is stored in a NumPy array named fuel_efficiency, where each element represents the fuel efficiency (in miles per gallon) of a specific car model. Your task is to calculate the average fuel efficiency and determine the percentage improvement in fuel efficiency between two car models.

Question: How would you use NumPy arrays and arithmetic operations to calculate the average fuel efficiency and determine the percentage improvement in fuel efficiency between two car models?



6. **Scenario:** You are a cashier at a grocery store and need to calculate the total cost of a customer's purchase, including applicable discounts and taxes. You have the item prices and quantities in separate lists, and the discount and tax rates are given as percentages. Your task is to calculate the total cost for the customer.

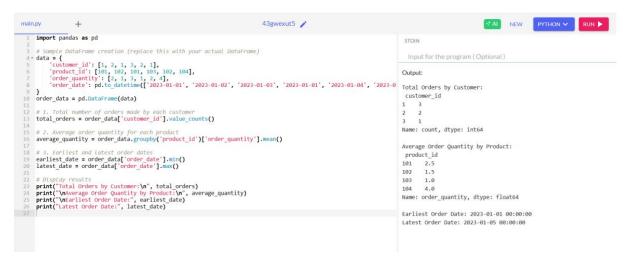
Question: Use arithmetic operations to calculate the total cost of a customer's purchase, including discounts and taxes, given the item prices, quantities, discount rate, and tax rate?



7. **Scenario:** You are working as a data analyst for an e-commerce company. You have been given a dataset containing information about customer orders, stored in a Pandas DataFrame named order_data. The DataFrame has columns for customer ID, order date, product name, and order quantity. Your task is to analyze the data and answer specific questions about the orders.

Question: Using Pandas DataFrame operations, how would you find the following information from the order data DataFrame:

- 1. The total number of orders made by each customer.
- 2. The average order quantity for each product.
- 3. The earliest and latest order dates in the dataset.



8. **Scenario:** You are a data scientist working for a company that sells products online. You have been tasked with analyzing the sales data for the past month. The data is stored in a Pandas data frame.

Question: How would you find the top 5 products that have been sold the most in the past month?

```
main.py + 43gwexut5 / RUN

| Import pandas as pd | from datetime import datetime, timedelta | from datetime import datetime product name; ['product A', 'Product B', 'Product B', 'Product D'], | from datetime import datetime | from date import date | from date |
```

9. **Scenario:** You work for a real estate agency and have been given a dataset containing information about properties for sale. The dataset is stored in a Pandas DataFrame named property_data. The DataFrame has columns for property ID, location, number of bedrooms, area in square feet, and listing price. Your task is to analyze the data and answer specific questions about the properties.

Question: Using Pandas DataFrame operations, how would you find the following information from the property_data DataFrame:

- 1. The average listing price of properties in each location.
- 2. The number of properties with more than four bedrooms.
- 3. The property with the largest are



10. **Scenario:** You are working on a data visualization project and need to create basic plots using Matplotlib. You have a dataset containing the monthly sales data for a company, including the month and corresponding sales values. Your task is to develop a Python program that generates line plots and bar plots to visualize the sales data.

Question:

- 1. How would you develop a Python program to create a line plot of the monthly sales data?
- 2: How would you develop a Python program to create a bar plot of the monthly sales data?

```
import matplotlib.pyplot as plt
      plt.figure(figsize=(10, 5))
plt.plot(months, sales, marker='o', color='green', linestyle='-', linewidth=2)
plt.title('Month')
plt.xlabel('Month')
plt.ylabel('Sales')
plt.grid(True)
plt.tight layout()
plt.show()
      plt.figure(figsize=(10, 5))
plt.bar(months, sales, color='orange', edgecolor='black')
plt.title('Monthly Sales - Bar Plot')
plt.xlabel('Monthl')
plt.ylabel('Sales')
plt.grid(axis='y', linestyle='--')
plt.tight layout()
plt.show()
                                                                        Monthly Sales - Line Plot
     4250
    3750
     3500
를
3
3250
    3000
    2750
                                                                         Monthly Sales - Bar Plot
    3500
     3000
를
2000
     1500
     1000
      500
                                                                                 Jun Jul
Month
                                                                                                         Aug
                                                                                                                     Sep
```