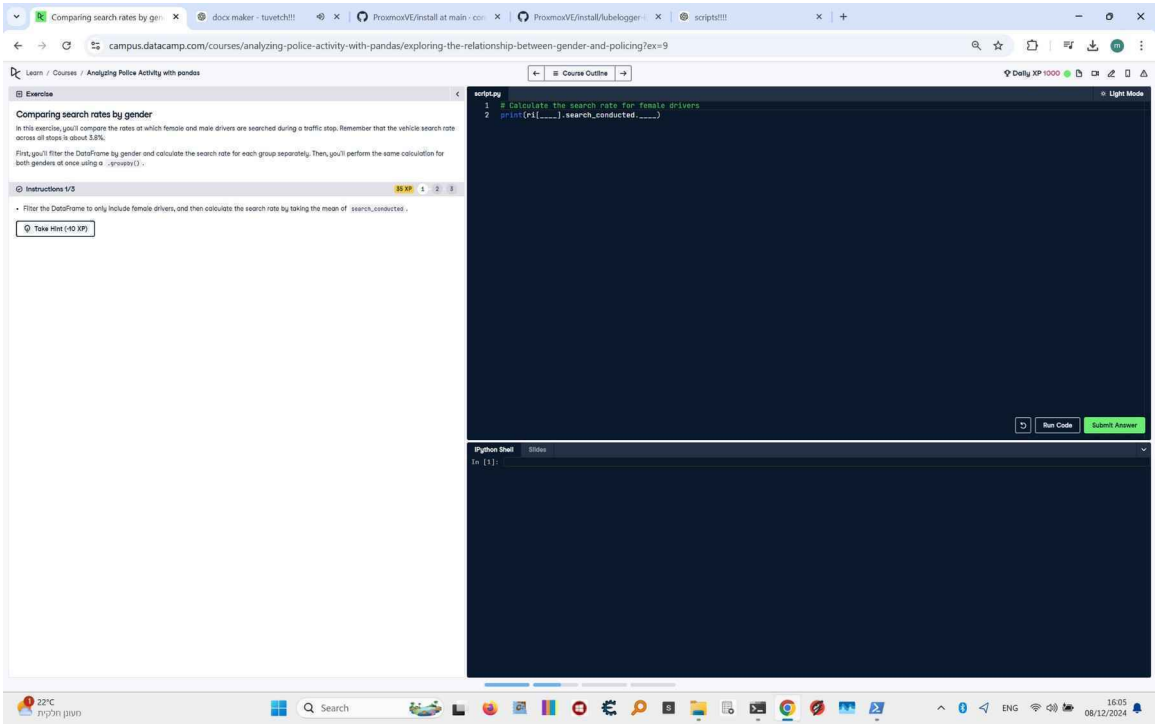
Comparing Search Rates by Gender



Task Description

1. Filter the DataFrame to only include female drivers, and then calculate the search rate by taking the mean of 'search_conducted'.
2. Filter the DataFrame to only include male drivers, and calculate the search rate similarly.
3. Use .groupby() to calculate the search rates for both genders in one step.

Code Solution

```
# Calculate the search rate for female drivers
female_search_rate = ri[ri['driver_gender'] == 'F']
['search_conducted'].mean()
print(female_search_rate)

# Calculate the search rate for male drivers
male_search_rate = ri[ri['driver_gender'] == 'M']
['search_conducted'].mean()
print(male_search_rate)

# Calculate search rates for both genders using groupby
gender_search_rate = ri.groupby('driver_gender')
['search_conducted'].mean()
print(gender_search_rate)
```

Code Explanation

1. The line 'female_search_rate = ri[ri['driver_gender'] == 'F']['search_conducted'].mean()' filters the DataFrame to include only female drivers and calculates the mean of the 'search_conducted' column, which represents the search rate for female drivers.
2. The line 'male_search_rate = ri[ri['driver_gender'] == 'M']['search_conducted'].mean()' filters the DataFrame to include only male drivers and calculates the mean of the 'search_conducted' column, representing the search rate for male drivers.
3. The line 'gender_search_rate = ri.groupby('driver_gender')['search_conducted'].mean()' groups the data by 'driver_gender' and calculates the mean of 'search_conducted' for each group (male and female), allowing for a concise calculation of search rates for both genders.