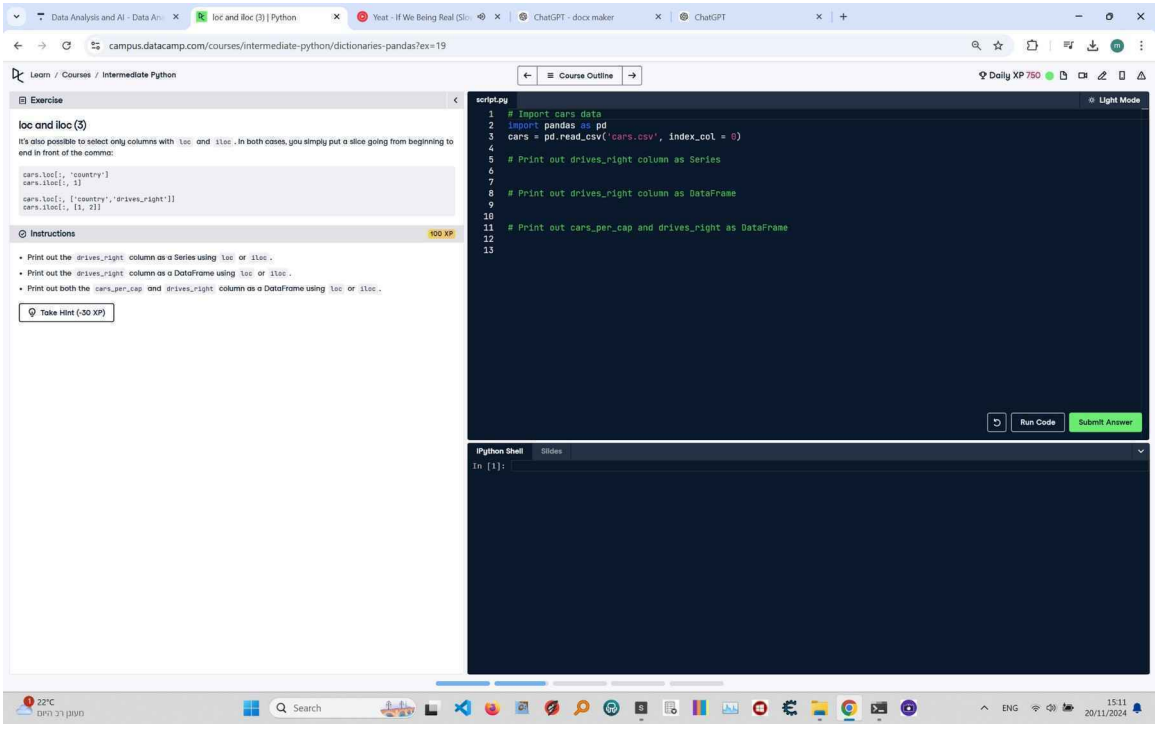
loc and iloc (3)



Question:

1. Print out the drives_right column as a Series using loc or iloc.
2. Print out the drives_right column as a DataFrame using loc or iloc.
3. Print out both the cars_per_cap and drives_right columns as a DataFrame using loc or iloc.

Answer:

```
# Import cars data
import pandas as pd
cars = pd.read_csv('cars.csv', index_col=0)

# Print out drives_right column as Series
print(cars['drives_right'])  # or cars.loc[:, 'drives_right']

# Print out drives_right column as DataFrame
print(cars[['drives_right']])  # or cars.loc[:, ['drives_right']]

# Print out cars_per_cap and drives_right columns as DataFrame
print(cars[['cars_per_cap', 'drives_right']])  # or cars.loc[:, ['cars_per_cap',
```

'drives_right']]

Code Explanation:

1. `cars['drives_right']`: Accesses the 'drives_right' column directly as a Series.
   Alternatively, `cars.loc[:, 'drives_right']` does the same using the loc method, where the colon selects all rows and the column name selects the column.
2. `cars[['drives_right']]`: Accesses the 'drives_right' column as a DataFrame.
   Alternatively, `cars.loc[:, ['drives_right']]` does the same using loc, where the colon selects all rows and the column name in a list selects the column as a DataFrame.
3. `cars[['cars_per_cap', 'drives_right']]`: Selects the 'cars_per_cap' and 'drives_right' columns as a DataFrame.
   Alternatively, `cars.loc[:, ['cars_per_cap', 'drives_right']]` does the same using loc, where the colon selects all rows and the column names in a list select the columns as a DataFrame.