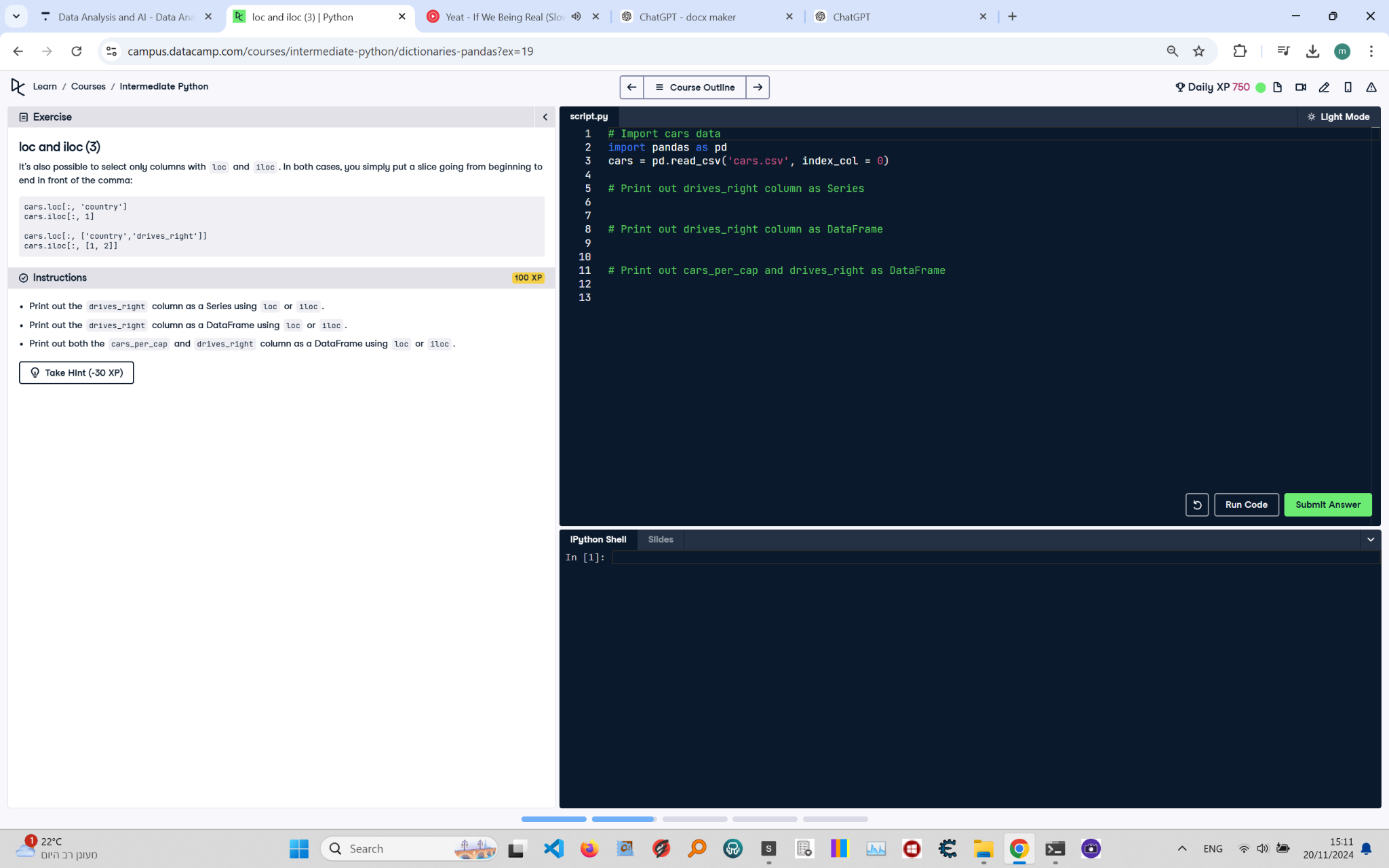
# loc and iloc (3)



## 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.