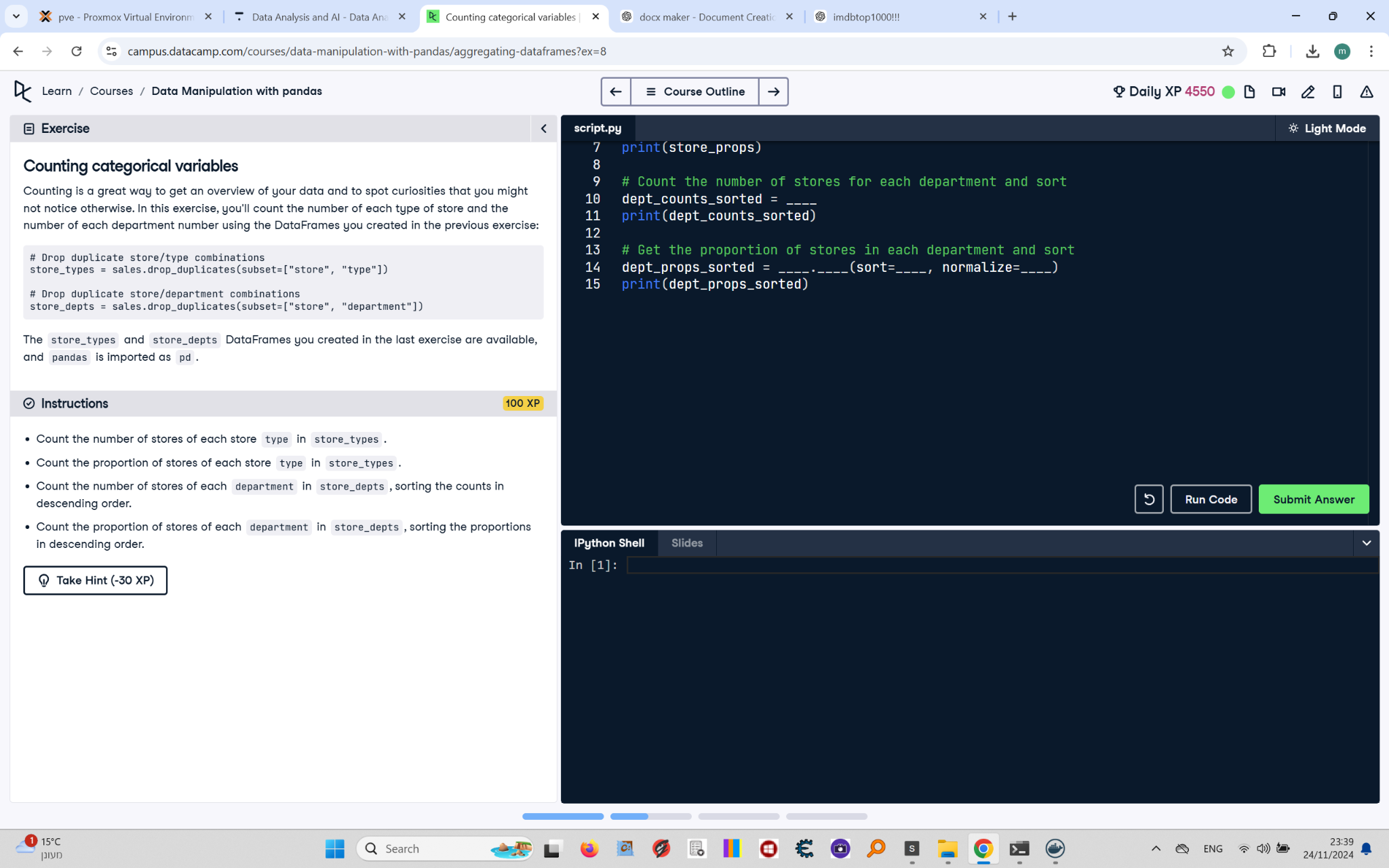
# Counting Categorical Variables (Solution)

This document includes the question, the solution, and a breakdown of the code provided in the screenshot.

## Uploaded Screenshot

Below is the screenshot of the task:



## Question

1. Count the number of stores of each store `type` in `store\_types`.  
2. Count the proportion of stores of each store `type` in `store\_types`.  
3. Count the number of stores of each `department` in `store\_depts`, sorting the counts in descending order.  
4. Count the proportion of stores of each `department` in `store\_depts`, sorting the proportions in descending order.

## Answer

# Count the number of stores of each store type  
store\_counts = store\_types['type'].value\_counts()  
print(store\_counts)  
  
# Count the proportion of stores of each store type  
store\_props = store\_types['type'].value\_counts(normalize=True)  
print(store\_props)  
  
# Count the number of stores of each department and sort  
dept\_counts\_sorted = store\_depts['department'].value\_counts(sort=True)  
print(dept\_counts\_sorted)  
  
# Count the proportion of stores in each department and sort  
dept\_props\_sorted = store\_depts['department'].value\_counts(sort=True, normalize=True)  
print(dept\_props\_sorted)

## Code Explanation

1. `store\_types['type'].value\_counts()`: Counts the occurrences of each store type in the `type` column of the `store\_types` DataFrame.

2. `value\_counts(normalize=True)`: Calculates the proportion (percentage) of each store type instead of absolute counts.

3. `store\_depts['department'].value\_counts(sort=True)`: Counts occurrences of each department in the `department` column of the `store\_depts` DataFrame, sorting by count.

4. `value\_counts(sort=True, normalize=True)`: Combines sorting with the calculation of proportions for each department.