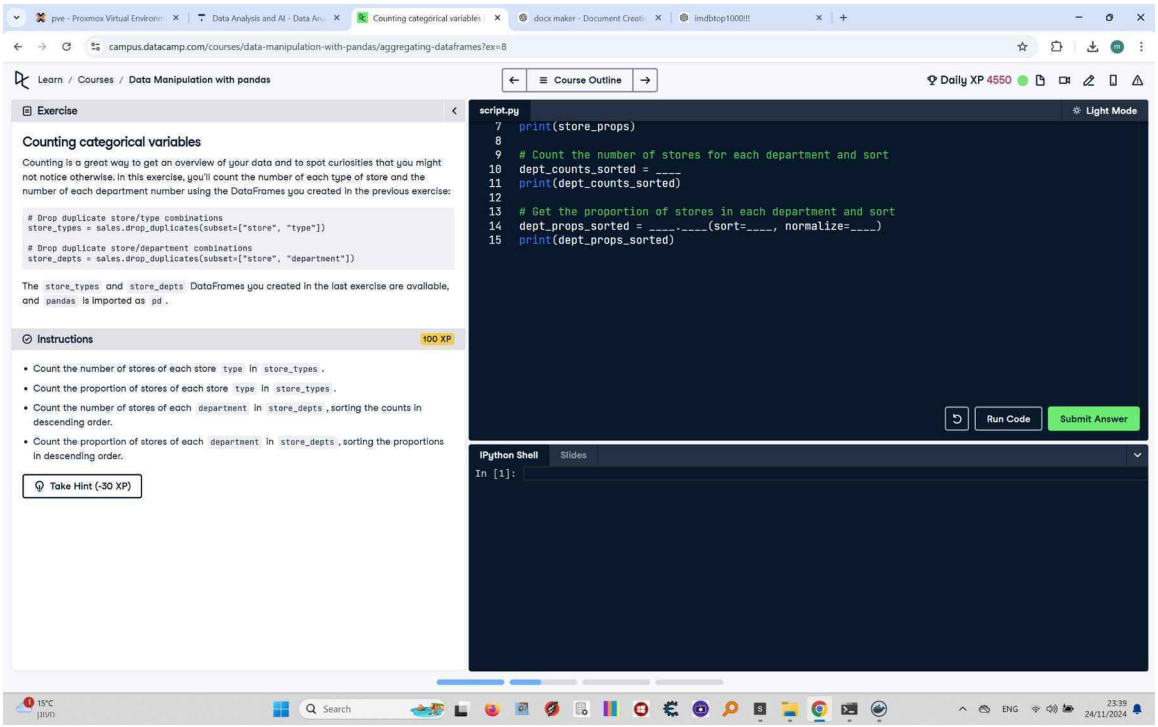
Counting Categorical Variables (Solution)

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.