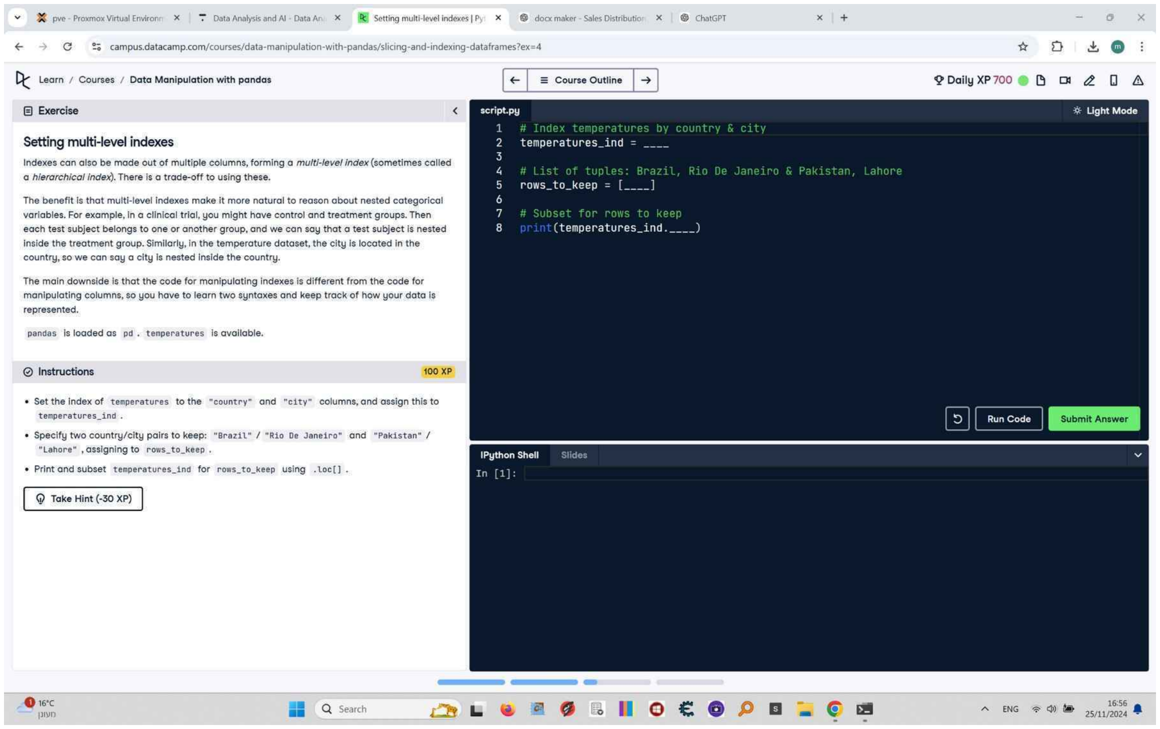
Setting Multi-Level Indexes



campus.datacamp.com/courses/data-manipulation-with-pandas/slicing-and-indexing-dataframes?ex=4

Learn / Courses / Data Manipulation with pandas

Exercise

Setting multi-level indexes

Indexes can also be made out of multiple columns, forming a *multi-level index* (sometimes called a *hierarchical index*). There is a trade-off to using these.

The benefit is that multi-level indexes make it more natural to reason about nested categorical variables. For example, in a clinical trial, you might have control and treatment groups. Then each test subject belongs to one or another group, and we can say that a test subject is nested inside the treatment group. Similarly, in the temperature dataset, the city is located in the country, so we can say a city is nested inside the country.

The main downside is that the code for manipulating indexes is different from the code for manipulating columns, so you have to learn two syntaxes and keep track of how your data is represented.

`pandas` is loaded as `pd`. `temperatures` is available.

Instructions 100 XP

- Set the index of `temperatures` to the `"country"` and `"city"` columns, and assign this to `temperatures_ind`.
- Specify two country/city pairs to keep: `"Brazil"` / `"Rio De Janeiro"` and `"Pakistan"` / `"Lahore"`, assigning to `rows_to_keep`.
- Print and subset `temperatures_ind` for `rows_to_keep` using `.loc[]`.

script.py

```python
# Index temperatures by country & city
temperatures_ind = ____

# List of tuples: Brazil, Rio De Janeiro & Pakistan, Lahore
rows_to_keep = [____]

# Subset for rows to keep
print(temperatures_ind.____)
```

```
In [1]:
```

Indexes can also be made out of multiple columns, forming a multi-level index (sometimes called a hierarchical index). There is a trade-off to using these.

The benefit is that multi-level indexes make it more natural to reason about nested categorical variables. For example, in a clinical trial, you might have control and treatment groups. Then each test subject belongs to one or another group, and we can say that a test subject is nested inside the treatment group. Similarly, in the temperature dataset, the city is located in the country, so we can say a city is nested inside the country.

The main downside is that the code for manipulating indexes is different from the code for manipulating columns, so you have to learn two syntaxes and keep track of how your data is represented.

pandas is loaded as pd. temperatures is available.

Final Answer

```python
# Index temperatures by country & city
temperatures_ind = temperatures.set_index(["country", "city"])
```

```
# List of tuples: Brazil, Rio De Janeiro & Pakistan, Lahore
rows_to_keep = [("Brazil", "Rio De Janeiro"), ("Pakistan", "Lahore")]

# Subset for rows to keep
print(temperatures_ind.loc[rows_to_keep])
```