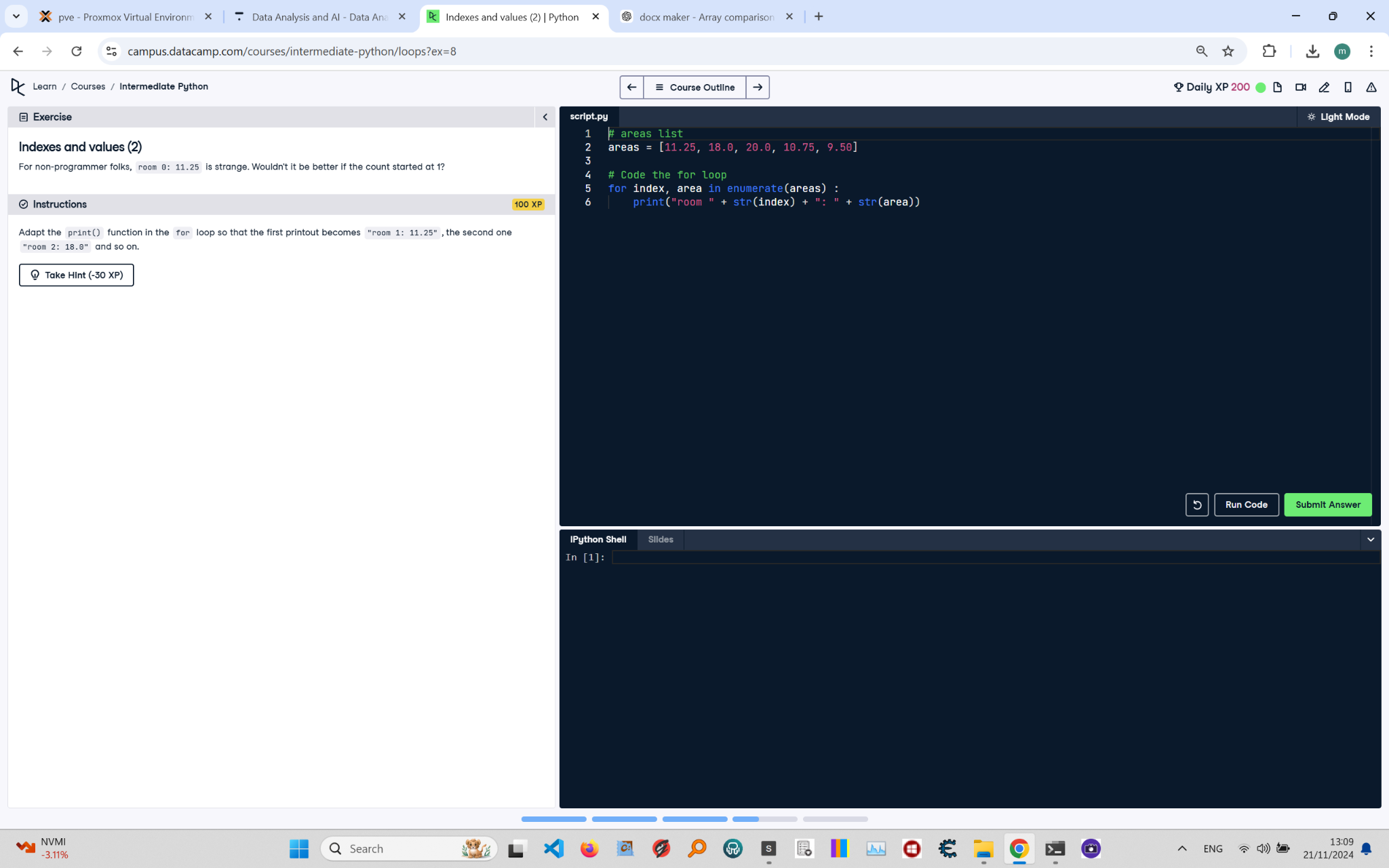
# Indexes and Values (2) in Python



\*\*Question:\*\*

For non-programmer folks, `room 0: 11.25` is strange. Wouldn't it be better if the count started at 1?  
  
\*\*Instructions:\*\*  
Adapt the `print()` function in the `for` loop so that the first printout becomes `room 1: 11.25`, the second one `room 2: 18.0`, and so on.

\*\*Answer:\*\*

Here is the Python code that solves the problem:

# areas list  
areas = [11.25, 18.0, 20.0, 10.75, 9.50]  
  
# Code the for loop  
for index, area in enumerate(areas):  
 print(f"room {index + 1}: {area}")

\*\*Explanation:\*\*

1. \*\*Initialization\*\*: The `areas` list is defined, containing the areas of different rooms in a house.  
2. \*\*Using enumerate()\*\*: The `enumerate()` function is used to loop over the list and get both the index and the value of each element.  
3. \*\*Adjusting the index\*\*: To make the count start at 1 instead of 0, `index + 1` is used in the `print()` statement.  
4. \*\*Print statement\*\*: The `print()` statement is updated to use an f-string, ensuring the output format is `room x: y`, where `x` is the index incremented by 1 and `y` is the element value.  
5. \*\*Execution\*\*: The loop runs once for each element in the `areas` list, printing the adjusted index and value for each room.