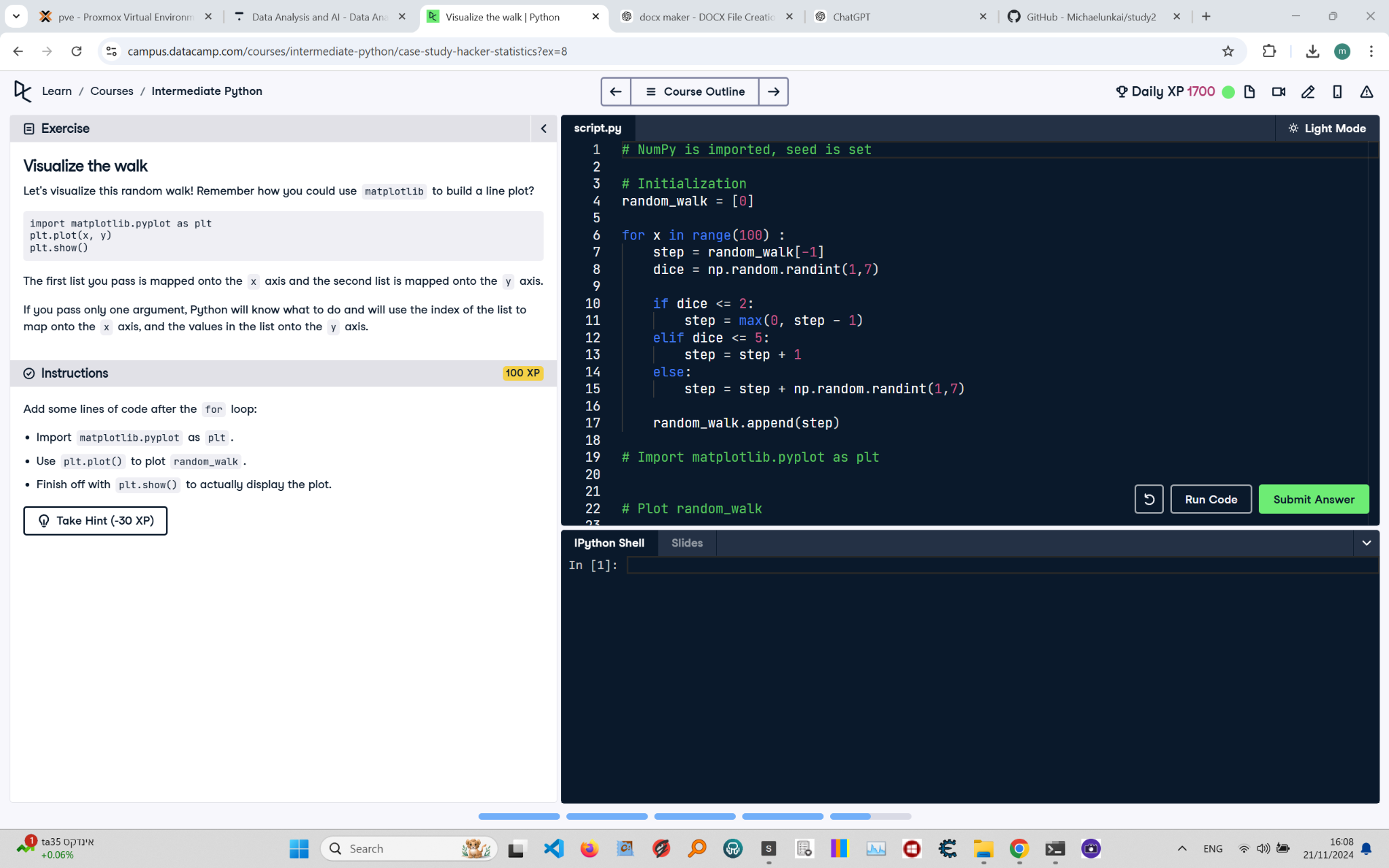
# Visualize the Walk



Below is the exercise on 'Visualize the Walk' from the Python course. The image includes the instructions, code, and task details.

Solution:

# NumPy is imported, seed is set  
import numpy as np  
np.random.seed(123)  
  
# Initialize random\_walk  
random\_walk = [0]  
  
# Complete the for loop  
for x in range(100): # Loop runs 100 times  
 # Set step: last element in random\_walk  
 step = random\_walk[-1]  
   
 # Roll the dice  
 dice = np.random.randint(1, 7)  
   
 # Determine next step using max to prevent step from going below 0  
 if dice <= 2:  
 step = max(0, step - 1) # Move down but ensure step doesn't go below 0  
 elif dice <= 5:  
 step = step + 1 # Move up  
 else:  
 step = step + np.random.randint(1, 7) # Move up by a random value  
   
 # Append next\_step to random\_walk  
 random\_walk.append(step)  
  
# Import matplotlib.pyplot as plt  
import matplotlib.pyplot as plt  
  
# Plot random\_walk  
plt.plot(random\_walk)  
  
# Show the plot  
plt.show()

Explanation:

1. Import numpy as np and set the random seed using np.random.seed(123) to ensure reproducibility.

2. Initialize random\_walk as a list containing the first step, 0.

3. Use a for loop that runs 100 times to simulate the steps of the random walk:

- Get the current step as the last element of the random\_walk list using random\_walk[-1].

- Roll the dice using np.random.randint(1, 7) to generate a random integer between 1 and 6.

- Use max() to ensure that step doesn't go below 0 when dice <= 2. This prevents negative steps:

- If dice is 1 or 2, use max(0, step - 1) to decrease step by 1 but not below 0.

- If dice is 3, 4, or 5, increase step by 1.

- If dice is 6, roll the dice again and add the new result to step.

4. Append the updated step to the random\_walk list.

5. Import matplotlib.pyplot as plt to visualize the random walk.

6. Use plt.plot(random\_walk) to plot the random walk on the y-axis, with the indices of random\_walk automatically plotted on the x-axis.

7. Call plt.show() to display the plot.