5. Will it effect your results from #1?
=> Solution:

tep 1: Understanding Question #1

In **#1**, we determined that the original function had a runtime complexity of:

$$T(n) = \Theta(n^2)$$

This was based on the summation:

$$\text{T(n)}=\sum_{i=1}^{n}\sum_{j=1}^{n} 1 = n^2$$

- Now, in **#4**, we modified the function by adding an extra operation:

$$y = i + j$$

which changed the number of operations per iteration **from 1 to 2**, leading to:

$$\text{T(n)}=\sum_{i=1}^{n}\sum_{j=1}^{n} 2 = 2n^2$$

Step 2: Does This Affect Our Results from #1?

1. **Big-O Analysis ($\boldsymbol{O(n^2)}$)**
   - **No change.** The function is still bounded above by $\boldsymbol{O(n^2)}$)
   - The constant factor 2 is ignored in Big-O notation.
2. **Big-Omega Analysis ($\boldsymbol{\Omega(n^2)}$)**
   - **No change.** The function still has a lower bound of $\Omega(n^2)$.
3. **Big-Theta Analysis ($\boldsymbol{\Theta(n^2)}$)**
   - **No change.** Since the upper and lower bounds remain n^2 , the function remains $\Theta(n^2)$

Step 3: Will This Affect Your Results?

- **Asymptotically? No.**
  - The function still follows $\Theta(n^2)$, and all the theoretical results from #1 remain valid.
- **Empirically (Measuring Execution Time)? Yes.**
  - The actual runtime would be about **twice as long**, but it does **not** change the **asymptotic complexity**.

Final Answer:

No, it does not affect our results from #1 because **Big-O, Big-Omega, and Big-Theta ignore constant factors**.
However, in actual measurements, the function would run slightly slower

⇨ Step 2: Analyzing the Complexity

- The added statement $y = i + j$ is a simple assignment.
- The **loop structure remains the same** $(O(n^2))$
- The total number of operations per iteration of the inner loop **increases from 1 to 2**.

Thus, the new number of operations is:

$$T(n)=\sum_{i=1}^{n}\sum_{j=1}^{n} 2 = 2n^2$$

⇨ Step 3: Will the Algorithm Take Longer?

- Yes, but **only by a constant factor**.

- The **growth rate** remains $O(n^2)$, so the asymptotic complexity does **not** change.
- However, **in practice**, execution time **will** increase because each inner loop iteration now performs **two** operations instead of one.
- The difference would be **small** but noticeable when timing large n