🌟 **Background**: At Hop Industries, we pride ourselves on the quality of our software. As we move into the e-commerce world, ensuring the reliability of our systems is crucial. This is where unit testing comes in!

The Scenario:

You've been handed a piece of code responsible for handling the stock management of products in the e-commerce store. Your job is to write unit tests to ensure this code works as expected. But beware! Rumor has it that there might be some bugs lurking around.

Expectations For the Stock Manager Class

1. **When adding stock**:
    - If a user tries to add stock for a new product, that product should be tracked and its stock should reflect the added amount.
    - If a user adds more stock for an already-tracked product, its stock should correctly accumulate.
    - A user should not be able to add a negative quantity of stock.
2. **When removing stock**:
    - If a user tries to remove stock from a product that isn't tracked (i.e., doesn't exist in our system), the operation should indicate failure (e.g., return `false`).
    - When a user requests removal of a quantity of stock that's available, the operation should succeed and the stock should reduce correctly.
    - If a user tries to remove more stock than what's available for a product, the operation should indicate failure.
    - When removing stock leads to a product having zero quantity left, it should still be tracked (perhaps for restocking purposes in the future).
3. **When querying for stock**:
    - If a user queries the stock of a product that's not tracked, the system should indicate that the product isn't found (e.g., return `-1` or a similar sentinel value).
    - When querying an existing product, it should return its current stock level.

Your Mission:

Write unit tests in the unit test project to ensure that the Stock Manager class fits all requirements. A single requirement may require more than one unit test to PROVE that the requirement is met. Look at the expectations for the stock manager class and use that as a starting point to figure out what tests you will need to write. All tests are required to pass.