**Exercise 1: Inventory Management System**

**Understanding the Problem:**

Efficient data storage and retrieval are crucial for handling large inventories because they directly impact the performance and scalability of the inventory management system. Effective data structures and algorithms ensure:

* **Quick Access**: Fast retrieval of product information.
* **Efficient Updates**: Swift updating of product quantities, prices, and other details.
* **Scalability**: The system can handle an increasing number of products without performance degradation.
* **Optimized Space**: Efficient use of memory.

**Suitable Data Structures for Inventory Management**

* **ArrayList**: Provides fast access and update operations but can be inefficient for frequent additions and deletions if the list size is large.
* **HashMap**: Offers average O(1) time complexity for add, update, and delete operations, making it ideal for inventory management where quick lookup and modifications are necessary.
* **TreeMap**: Maintains sorted order and allows log(n) time complexity for most operations but can be slower than a HashMap for large datasets.

For this problem, **HashMap** is chosen due to its average O(1) time complexity for insertions, deletions, and lookups.

**Analysis:**

**HashMap:**

* Add: O(1)
* Update: O(1)
* Delete: O(1)

**Optimization Discussion:** A HashMap is effective for quick lookups and updates. However, if order matters or frequent iterations are required, alternatives like ArrayList or sorted structures might be considered.