Inventory Management System

1. **Explain why data structures and algorithms are essential in handling large inventories.**
   - Efficiency: Large inventories involve thousands of products. Efficient data structures help ensure fast lookups, updates, and deletions.
   - Scalability: Good algorithms and data structures help manage data growth without performance degradation.
   - Real-time Performance: Operations like finding, updating, or deleting a product need to be done quickly for smooth inventory control.

2. **Discuss the types of data structures suitable for this problem.**
   - HashMap: Ideal for fast access using productId as the key. Offers O(1) average time complexity for add, update, and delete.
   - ArrayList: Good for ordered lists, but slow for search (O(n)) and deletion.

3. **Analyze the time complexity of each operation (add, update, delete) in your chosen data structure**.
   - Add product : we use hashmap data structure and time complexity is O(1) on average
   - Update product : we use hashmap data structure an time complexity is O(1) on average
   - Delete product : we use hashmap data structure an time complexity is O(1) on average

4. **Discuss how you can optimize these operations.**

 Using TreeMap instead of HashMap if we need sorted inventory.

 Implement search by product name using a secondary map or Map<String, List<Product>>.

 Use concurrent data structures (e.g., ConcurrentHashMap) for multi-threaded environments.