1. **Importance of Data Structures and Algorithms:**
   * Efficiently manage and access data in an inventory system.
   * Optimize operations (add, update, delete) as the inventory grows.
   * Ensure scalability with increasing data.
2. **Suitable Data Structures:**
   * **ArrayList:**
     + Pros: Fast access by index (suitable for smaller inventories).
     + Cons: Costly add/remove operations (O(n)).
   * **HashMap:**
     + Pros: Average-case O(1) time complexity (add, update, delete) due to hashing.
     + Cons: Extra memory for hashing and potential collisions.
   * **TreeMap:**
     + Pros: Sorted order, O(log n) time complexity (add, update, delete).
     + Cons: Slower than HashMap but useful for sorted data.
3. **Chosen Data Structure:**
   * For most inventory systems, use a HashMap for efficient operations.
4. **Time Complexity Analysis:**
   * Add Operation:
     + HashMap: Average-case O(1) due to hashing.
     + ArrayList: O(n) (checking duplicates, shifting).
   * Update Operation:
     + HashMap: Average-case O(1) (existence check, update).
     + ArrayList: O(n) (search, update).
   * Delete Operation:
     + HashMap: Average-case O(1) (existence check, removal).
     + ArrayList: O(n) (search, removal).
5. **Optimizations:**
   * Use HashMap with a good hash function.
   * Consider memory usage (adjust load factor).