## Step 1: Understand the Problem

1. 1. Explain why Data Structures and Algorithms Are Essential in handling large inventories.

In an inventory system, data structures and algorithms ensure fast, efficient, and scalable management of product records. Without them, operations like searching, adding, or updating products could become slow and error-prone as data grows — especially in large warehouses.

1. 2. Discuss the types of data structures suitable for this problem.

* Dictionary (HashMap)

Best for fast lookups using productId.  
Time complexity: O(1) for add, update, delete, and search.

* List (ArrayList)

Simple, maintains order.  
Time complexity: O(n) for search, update, delete.

* LinkedList

Efficient for insertions/deletions at known positions.  
Not ideal for search-heavy use.

* Binary Search Tree (BST)

Good if ordered traversal or range queries are needed.

## Step 4: Analysis

### Time Complexity (Using Dictionary):

|  |  |
| --- | --- |
| Operation | Time Complexity |
| Add | O(1) |
| Update | O(1) |
| Delete | O(1) |
| Search | O(1) |

### Optimizations:

* I will ensure unique product IDs to prevent collisions.
* For range-based filtering (e.g., price between ₹500–₹1000), I can combine dictionary with sorted lists or indexes.
* I can use caching for frequently accessed products.
* I may incorporate pagination when showing products in bulk.