Advanced Inventory Management System

Design an **Inventory Management System** for a warehouse using advanced Java Collections. The warehouse has the following requirements:

1. **Inventory Tracking:**
   - Each item in the inventory has a unique ID, a name, a category (e.g., electronics, furniture, groceries), and a quantity.
   - You need to store these items efficiently to support operations such as searching, adding, updating, and deleting items based on their unique ID.
2. **Category-wise Sorting and Retrieval:**
   - Items within a category should always be sorted in descending order of their quantity.
   - Efficient retrieval of all items belonging to a particular category is required.
3. **Restocking Notifications:**
   - If the quantity of any item drops below a specified threshold, it should trigger a notification (console output).
4. **Bulk Operations:**
   - Allow merging of two inventories while ensuring no duplicate item IDs exist. In case of duplicate IDs, retain the item with the higher quantity.
5. **Complex Query Support:**
   - Provide a function to retrieve the top `k` items with the highest quantity, regardless of category.

Constraints:

- The system should handle at least **100,000 items** efficiently.
- Your solution should make optimal use of Java's collection framework classes like `Map`, `Set`, `Queue`, or any others as needed.

Deliverables:

1. A well-documented Java class implementing the inventory system.
2. A test suite to demonstrate the functionality and edge cases of the implemented system.