Inventory Management System

(Hibernate 5 – Criteria API Based Mini Project)

Product Entity Details

Each product in the inventory system must have the following attributes:

Id (int)
Name (String)
Category (String)
Price (double)
Manufacturing Date (String, format: dd-MM-yyyy)
Expiry Date (String, format: dd-MM-yyyy)
Warranty (int, in months)
Quantity (int)

Important Notes for Students:
• Dates are stored as String in dd-MM-yyyy format
• Date comparison should be handled using string comparison logic
• Only Hibernate Criteria API should be used
• HQL, JPQL, and native SQL are not allowed

LEVEL 1: BEGINNER (MANDATORY – 20 TASKS)

1. Add a new product to the inventory

   Method: int addProduct(Product product)

   Input: Product object

   Output: Generated product id

2. Update product details using product id

   Method: int updateProductById(int id, Product newProduct)

   Input: Product id, new Product object

   Output: Updated product id

3. Delete a product using product id

   Method: int deleteProductById(int id)

Input: Product id

Output: Deleted product id

4. Fetch a single product by product id

   Method: Product getProductById(int id)

   Input: Product id

   Output: Product object or null

5. Fetch all products from inventory

   Method: List<Product> getAllProducts()

   Input: None

   Output: List of products

6. Fetch products belonging to a specific category

   Method: List<Product> getProductsByCategory(String category)

   Input: Category name

   Output: List of products

7. Fetch products with an exact product name

   Method: List<Product> getProductsByName(String name)

   Input: Product name

   Output: List of products

8. Fetch products with price greater than a given value

   Method: List<Product> getProductsPriceGreaterThan(double price)

   Input: Price value

   Output: List of products

9. Fetch products with price less than a given value

   Method: List<Product> getProductsPriceLessThan(double price)

Input: Price value

Output: List of products

10. Fetch products within a given price range

Method: List<Product> getProductsBetweenPrice(double min, double max)

Input: Minimum price, Maximum price

Output: List of products

11. Fetch all products ordered by price in ascending order

Method: List<Product> getProductsOrderByPriceAsc()

Input: None

Output: Sorted list of products

12. Fetch all products ordered by price in descending order

Method: List<Product> getProductsOrderByPriceDesc()

Input: None

Output: Sorted list of products

13. Fetch products with warranty greater than a given number of months

Method: List<Product> getProductsWithWarrantyGreaterThan(int months)

Input: Warranty months

Output: List of products

14. Fetch all available products (products in stock)

Method: List<Product> getAvailableProducts()

Condition: Quantity greater than zero

Output: List of products

15. Fetch all out-of-stock products

Method: List<Product> getOutOfStockProducts()

Condition: Quantity equals zero

Output: List of products

16. Fetch products manufactured after a given date

Method: List<Product> getProductsManufacturedAfter(String mfgDate)

Input: Manufacturing date (dd-MM-yyyy)

Output: List of products

17. Fetch products expiring before a given date

Method: List<Product> getProductsExpiringBefore(String expDate)

Input: Expiry date (dd-MM-yyyy)

Output: List of products

18. Fetch products by category and price condition

Method: List<Product> getProductsByCategoryAndPrice(String category, double price)

Input: Category name, Price limit

Output: List of products

19. Search products using name keyword

Method: List<Product> searchProductsByName(String keyword)

Input: Keyword

Output: List of products whose name contains the keyword

20. Count total number of products in inventory

Method: Long countTotalProducts()

Input: None

Output: Total product count

LEVEL 2: INTERMEDIATE (10 TASKS)

21. Fetch product(s) with the maximum price

Method: List<Product> getMaxPricedProducts()

Output: List of highest-priced products

22. Fetch product(s) with the minimum price

Method: List<Product> getMinPricedProducts()

Output: List of lowest-priced products

23. Find the average price of all products

Method: Double getAverageProductPrice()

Output: Average price value

24. Find the total quantity of all products in inventory

Method: Long getTotalStockCount()

Output: Total stock count

25. Fetch distinct product categories

Method: List<String> getDistinctCategories()

Output: List of unique categories

26. Fetch product count for each category

Method: List<Object[]> getProductCountByCategory()

Output: Category and count

27. Fetch average product price for each category

Method: List<Object[]> getCategoryWiseAveragePrice()

Output: Category and average price

28. Fetch products expiring before a given date

Method: List<Product> getProductsExpiringBefore(String date)

Output: List of products

29. Fetch product count grouped by warranty period

Method: List<Object[]> getWarrantyWiseProductCount()

Output: Warranty and count

30. Calculate total inventory value

Method: Double getTotalInventoryValue()

Output: Total inventory value

LEVEL 3: EXPERT (10 TASKS)

31. Fetch product(s) having the highest stock quantity

Method: List<Product> getHighestStockProducts()

32. Fetch product(s) having the lowest stock quantity

Method: List<Product> getLowestStockProducts()

33. Fetch products priced above the average product price

Method: List<Product> getProductsAboveAveragePrice()

34. Fetch products priced below the average product price

Method: List<Product> getProductsBelowAveragePrice()

35. Fetch the most expensive product from each category

Method: List<Product> getMostExpensiveProductPerCategory()

36. Fetch all expired products based on today’s date

Method: List<Product> getExpiredProducts(String todayDate)

37. Fetch products with zero warranty

Method: List<Product> getProductsWithZeroWarranty()

38. Fetch low-stock products for inventory alert

Method: List<Product> getLowStockProducts(int threshold)

39. Generate inventory health report

Method: Object[] getInventoryHealthReport()

40. Fetch products that do not belong to given categories

Method: List<Product> getProductsNotInCategories(List<String> categories)

Submission Details: By Mail
Subject: Inventory Management System by Tejas Sir

Instructions:

- Attach the file to the email.
- If you're submitting a project, please export it as a zip file.
- If you're including screenshots, please combine them into a single PDF file before submission.

Email Body:
Include your message body here with any necessary details and attachments.

Recipient: batchnumber@thekiranacademy.com (e.g., 7030@thekiranacademy.com)
CC: tejas@thekiranacademy.com

Note:
Please replace **batchnumber** with your respective batch number (e.g., 7030 if your batch number is 7030). Do not use "batchnum" as it is; it is a placeholder.

Thanks & All the Best,
Tejas.