Lab7: SPRING BOOT & JPA CRUD

PhamVuHaiDang_ITITDK23016

Time spent: 8 hours

Screenshots:

EXERCISE 5: ADVANCED SEARCH

ID	Code	Name	Price	Quantity	Category	Actions
2	P002	iPhone 15 Pro	$999.99	25	Electronics	
3	P003	Office Chair	$199.99	50	Furniture	
4	P004	Office Chair Ergonomic	$199.99	50	Furniture	
8	P007	iphone 14	$40.00	11	Electronics	
9	P008	iphone 14 pro	$100.00	19	Electronics	
10	P009	iphone 13	$30.00	4	Electronics	
11	P0010	iphone 13 pr	$55.00	23	Electronics	
14	P0011	iphone 12 pro	$45.00	12	Electronics	

ID	Code	Name	Price	Quantity	Category	Actions
1	P001	Laptop Dell XPS 13	$1300.00	10	Electronics	✏️ 🗑️
2	P002	iPhone 15 Pro	$999.99	25	Electronics	✏️ 🗑️
3	P003	Office Chair	$199.99	50	Furniture	✏️ 🗑️
4	P004	Office Chair Ergonomic	$199.99	50	Furniture	✏️ 🗑️
6	P006	RTX 5090	$400.00	6	Other	✏️ 🗑️
7	P005	RTX 4090	$200.00	12	Other	✏️ 🗑️
8	P007	iphone 14	$40.00	11	Electronics	✏️ 🗑️
9	P008	iphone 14 pro	$100.00	19	Electronics	✏️ 🗑️
10	P009	iphone 13	$30.00	4	Electronics	✏️ 🗑️
11	P0010	iphone 13 pr	$55.00	23	Electronics	✏️ 🗑️

1 2

Product Management

View Dashboard   Add Product   Search...

Name contains...   All Categories   0   999999   Advanced Search

All Categories

ID	Code	Name	Price	Quantity	Category	Actions
14	P0011	iphone 12 pro	$45.00	12	Electronics	✏️ 🗑️
20	P0014	BookA	$32.00	12	Books	✏️ 🗑️
21	P0015	foodA	$32.00	12	Food	✏️ 🗑️

1 2

EXERCISE 6: VALIDATION

➕ **Add New Product**

Product Code *

Enter product code (e.g., P001)

Product code must be 3-20 characters
Product code must start with P followed by numbers
Product code is required

Product Name *

FoodB

Price ($) *

12

Quantity *

433

Category *

Food ▾

Description

Enter product description (optional)

💾 Save Product   ✖ Cancel

➕ **Add New Product**

Product Code *

P0017

Product Name *

Enter product name

Name must be 3-100 characters
Product name is required

Price ($) *

12

Quantity *

433

Category *

Food ▾

Description

Enter product description (optional)

💾 Save Product   ✖ Cancel

➕ Add New Product

Product Code *

P0017

Product Name *

FoodB

Price ($) *

0.00

Price is required

Quantity *

433

Category *

Food

Description

Enter product description (optional)

[💾 Save Product] [✖ Cancel]

➕ Add New Product

Product Code *

P0017

Product Name *

FoodB

Price ($) *

12

Quantity *

0

Quantity is required

Category *

Food

Description

Enter product description (optional)

[💾 Save Product] [✖ Cancel]

EXERCISE 7: SORTING & FILTERING

EXERCISE 8: STATISTICS DASHBOARD

Report:

Ex5:

Advanced Search Workflow:

- User fills filters (name, category, minPrice, maxPrice).
- Controller routes to productService.advancedSearch().
- Repository runs JPQL with dynamic conditions.
- Returns Page with filtered results.
- View renders advanced search results with pagination.

Ex6:

1. User submits product form to /products/save.
2. Controller receives @Valid Product + BindingResult.
3. If validation fails (result.hasErrors()):

a. Controller returns product-form.html.

b. Invalid fields show red border and error messages.

4. If validation passes:

a. Service saves product to DB.

b. Redirects to /products with success message

Ex7:

1. Sorting Workflow

a. User clicks sortable column →
/products?sortBy=price&sortDir;=asc.

b. Controller builds Sort sort =
Sort.by("price").ascending().

c. Calls productService.getAllProducts(sort).

d. Repository executes ORDER BY query.

e. Sorted results returned and displayed.

2. Category Filter Workflow

a. User selects category filter →
/products?category=Furniture.

b. Controller calls getProductsByCategory("Furniture").

c. View displays filtered list.

3. Combined Sorting + Filtering Workflow

a. URL example:
/products?category=Electronics&sortBy;=name&sort
Dir;=desc.• Controller builds Sort object and calls
getProductsByCategory(category, sort).

b. Repository runs SELECT * WHERE
category='Electronics' ORDER BY name DESC.

Ex8:

1. User visits /dashboard.
2. DashboardController collects statistics via ProductService:
   a. countByCategory(category)
   b. calculateTotalValue()
   c. calculateAveragePrice()