**SCD Lab 11**

**Name:** Hafsa Salman

**Roll no.** 22K-5161

**Task 01**

Code:

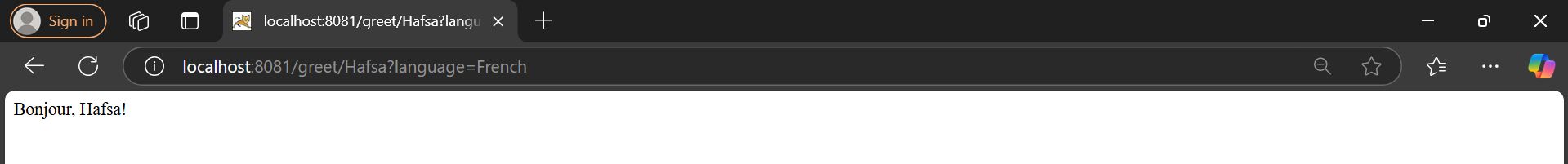
//GreetingController.java

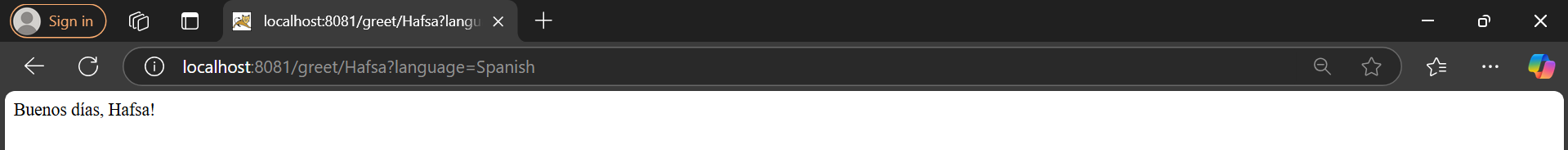
//Hafsa Salman  
//22K-5161  
//Task 11  
  
package com.lab\_11.lab\_11;  
  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.web.bind.annotation.\*;  
  
@RestController  
@RequestMapping("/greet")  
public class GreetingController  
{  
 private final GreetingService greetingService;  
  
 @Autowired  
 public GreetingController(GreetingService greetingService)  
 {  
 this.greetingService = greetingService;  
 }  
  
 @GetMapping("/{name}")  
 public String greetUser(  
 @PathVariable String name,  
 @RequestParam(defaultValue = "english") String language)  
 {  
 try  
 {  
 String greeting = greetingService.getGreeting(language);  
 return greeting + ", " + name + "!";  
 }  
  
 catch (Exception e)  
 {  
 return "Error: Unsupported language requested. Please use 'english', 'french', or 'spanish'.";  
 }  
 }  
}

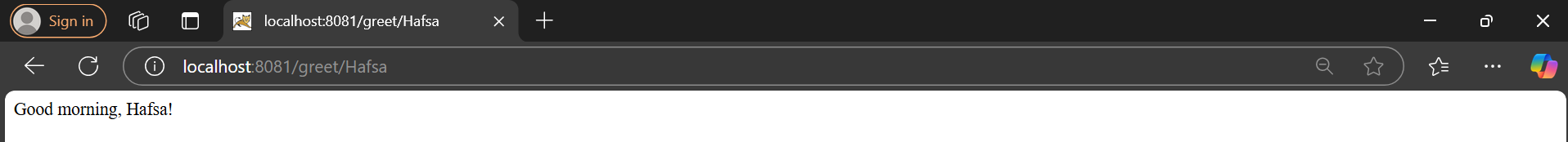
//GreetingService.java

//Hafsa Salman  
//22K-5161  
//Task 11  
  
package com.lab\_11.lab\_11;  
  
  
import org.springframework.stereotype.Service;  
import java.time.LocalTime;  
import java.util.Map;  
  
@Service  
public class GreetingService  
{  
 private static final Map<String, String> *GREETINGS\_ENGLISH* = Map.*of*(  
 "morning", "Good morning",  
 "afternoon", "Good afternoon",  
 "evening", "Good evening"  
 );  
  
 private static final Map<String, String> *GREETINGS\_FRENCH* = Map.*of*(  
 "morning", "Bonjour",  
 "afternoon", "Bon après-midi",  
 "evening", "Bonsoir"  
 );  
  
 private static final Map<String, String> *GREETINGS\_SPANISH* = Map.*of*(  
 "morning", "Buenos días",  
 "afternoon", "Buenas tardes",  
 "evening", "Buenas noches"  
 );  
  
 public String getGreeting(String language)  
 {  
 Map<String, String> greetingsMap = switch (language.toLowerCase())  
 {  
 case "french" -> *GREETINGS\_FRENCH*;  
 case "spanish" -> *GREETINGS\_SPANISH*;  
 default -> *GREETINGS\_ENGLISH*;  
 };  
  
 String timeOfDay = getTimeOfDay();  
 return greetingsMap.getOrDefault(timeOfDay, "Hello");  
 }  
  
 private String getTimeOfDay()  
 {  
 int hour = LocalTime.*now*().getHour();  
 if (hour >= 5 && hour < 12)  
 {  
 return "morning";  
 }  
  
 else if (hour >= 12 && hour < 17)  
 {  
 return "afternoon";  
 }  
  
 else  
 {  
 return "evening";  
 }  
 }  
}

Output:







**Task 02**

Code:

//UserController.java

//Hafsa Salman  
//22K-5161  
//Task no. 02  
  
package com.lab\_11.lab\_11;  
  
import org.springframework.stereotype.Controller;  
import org.springframework.web.bind.annotation.\*;  
  
import java.util.HashSet;  
import java.util.Set;  
  
@Controller  
public class UserController  
{  
 private final Set<String> registeredUsers = new HashSet<>(Set.*of*("Hafsa", "12345"));  
  
 @GetMapping("/menu")  
 public String showMenu()  
 {  
 return "menu";  
 }  
  
 @GetMapping("/loginPage")  
 public String showLoginPage()  
 {  
 return "login";  
 }  
  
 @GetMapping("/registerPage")  
 public String showRegisterPage()  
 {  
 return "Register";  
 }  
  
 @PostMapping("/register")  
 @ResponseBody  
 public String registerUser(@RequestParam String username, @RequestParam String password)  
 {  
 if (registeredUsers.contains(username))  
 {  
 return "Username is already in use, please choose diffrent one.";  
 }  
  
 else  
 {  
 registeredUsers.add(username);  
 return "Registration successful for user: " + username;  
 }  
 }  
  
 @PostMapping("/login")  
 @ResponseBody  
 public String loginUser(@RequestParam String username, @RequestParam String password) {  
 if (registeredUsers.contains(username))  
 {  
 return "Login successful for user: " + username;  
 }  
  
 else  
 {  
 return "Login failed: Username not found or incorrect password.";  
 }  
 }  
}

//menu.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 02  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 02</title>  
</head>  
<body>  
<h2>22K-5161: Task no. 02</h2>  
<button onclick="*location*.href='/loginPage'">Login</button>  
<button onclick="*location*.href='/registerPage'">Register</button>  
</body>

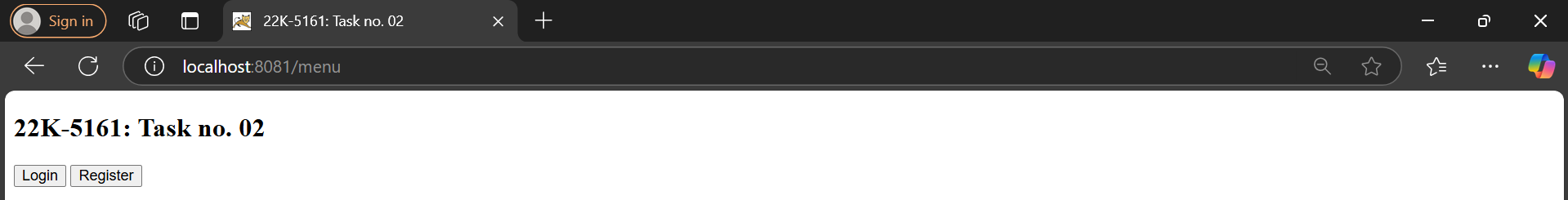
//login.html

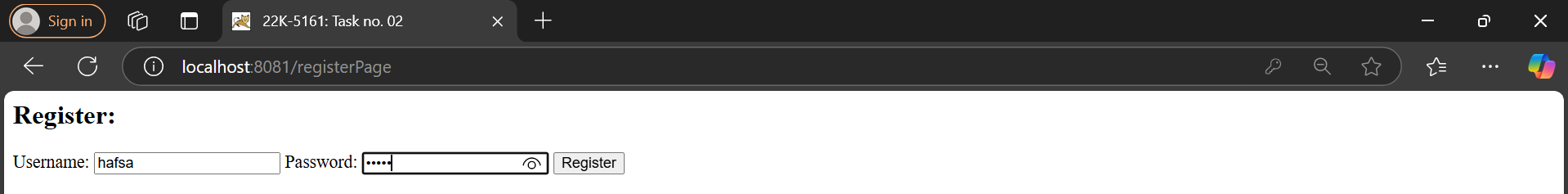
<!--  
Hafsa Salman  
22K-5161  
Task no. 02  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 02</title>  
</head>  
<body>  
<h2>Login:</h2>  
<form action="/login" method="post">  
 <label for="username">Username:</label>  
 <input type="text" id="username" name="username" required>  
  
 <label for="password">Password:</label>  
 <input type="password" id="password" name="password" required>  
  
 <button type="submit">Login</button>  
</form>  
</body>  
</html>

//Register.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 02  
-->  
  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 02</title>  
</head>  
<body>  
<h2>Register:</h2>  
<form action="/register" method="post">  
 <label for="username">Username:</label>  
 <input type="text" id="username" name="username" required>  
  
 <label for="password">Password:</label>  
 <input type="password" id="password" name="password" required>  
  
 <button type="submit">Register</button>  
</form>  
</body>  
</html>

Output:

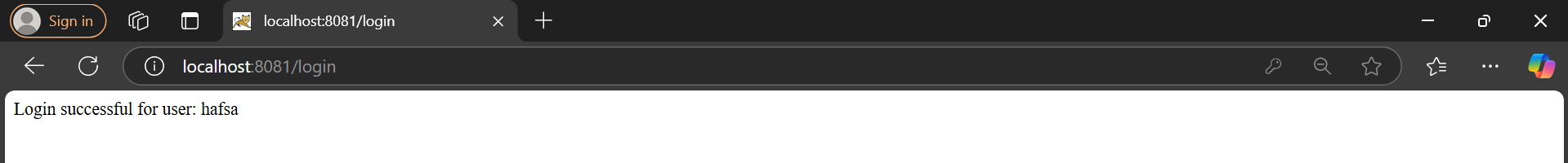






A black and white screen

Description automatically generated with medium confidence



**Task 03**

Code:

//Item.java

//Hafsa Salman  
//22K-5161  
//Task no. 03  
  
package com.lab\_11.lab\_11;  
  
public class Item  
{  
 private int Id;  
 private String name;  
 private int quantity;  
  
 public Item (int Id, String name, int quantity)  
 {  
 this.Id = Id;  
 this.name = name;  
 this.quantity = this.quantity;  
 }  
  
 public int getId()  
 {  
 return Id;  
 }  
  
 public void setId(int Id)  
 {  
 this.Id = Id;  
 }  
  
 public String getName()  
 {  
 return name;  
 }  
  
 public void setName(String name)  
 {  
 this.name = name;  
 }  
  
 public int getQuantity() {  
 return quantity;  
 }  
  
 public void setQuantity(int quantity) {  
 this.quantity = quantity;  
 }  
}

//ItemController.java

//Hafsa Salman  
//22K-5161  
//Task no. 03  
  
package com.lab\_11.lab\_11;  
  
import org.springframework.stereotype.Controller;  
import org.springframework.ui.Model;  
import org.springframework.web.bind.annotation.\*;  
  
import java.util.ArrayList;  
import java.util.List;  
  
@Controller  
public class ItemController  
{  
 private final List<Item> itemList = new ArrayList<>();  
  
 @GetMapping("/itemMenu")  
 public String itemMenu()  
 {  
 return "itemMenu";  
 }  
  
 @GetMapping("/addItemForm")  
 public String showAddItemForm()  
 {  
 return "addItemForm";  
 }  
  
 @PostMapping("/addItemForm")  
 public String addItem(@RequestParam int id, @RequestParam String name, @RequestParam int quantity, Model model) {  
 itemList.add(new Item(id, name, quantity));  
 model.addAttribute("message", "Item added successfully.");  
 return "itemMenu";  
 }  
  
 @GetMapping("/getItemForm")  
 public String showGetItemForm()  
 {  
 return "getItemForm";  
 }  
  
 @PostMapping("/getItem")  
 public String getItemById(@RequestParam int id, Model model)  
 {  
 Item foundItem = null;  
 for (Item item : itemList)  
 {  
 if (item.getId() == id)  
 {  
 foundItem = item;  
 break;  
 }  
 }  
  
 if (foundItem != null)  
 {  
 model.addAttribute("item", foundItem);  
 }  
  
 else  
 {  
 model.addAttribute("message", "Item not found.");  
 }  
 return "getItemForm";  
 }  
  
 @GetMapping("/updateItemForm")  
 public String showUpdateItemForm()  
 {  
 return "updateItemForm";  
 }  
  
 @PostMapping("/updateItem")  
 public String updateItem(@RequestParam int id, @RequestParam String name, @RequestParam int quantity, Model model)  
 {  
 boolean itemUpdated = false;  
  
 for (Item item : itemList)  
 {  
 if (item.getId() == id)  
 {  
 item.setName(name);  
 item.setQuantity(quantity);  
 itemUpdated = true;  
 break;  
 }  
 }  
  
 if (itemUpdated)  
 {  
 model.addAttribute("message", "Item updated successfully.");  
 }  
  
 else  
 {  
 model.addAttribute("message", "Item not found.");  
 }  
  
 return "updateItemForm";  
 }  
  
 @GetMapping("/deleteItemForm")  
 public String showDeleteItemForm()  
 {  
 return "deleteItemForm";  
 }  
  
 @PostMapping("/deleteItem")  
 public String deleteItem(@RequestParam int id, Model model)  
 {  
 boolean itemFoundAndRemoved = false;  
  
 for (Item item : itemList)  
 {  
 if (item.getId() == id)  
 {  
 itemList.remove(item);  
 itemFoundAndRemoved = true;  
  
 break;  
 }  
 }  
  
 model.addAttribute("message", itemFoundAndRemoved ? "Item deleted successfully." : "Item not found.");  
 return "deleteItemForm";  
 }  
  
 @GetMapping("/viewAllItems")  
 public String viewAllItems(Model model)  
 {  
 model.addAttribute("items", itemList);  
  
 return "viewAllItems";  
 }  
}

//addItemForm.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 03  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 03</title>  
</head>  
<body>  
<h1>22K-5161: Task no. 03</h1>  
<form action="/addItemForm" method="post">  
 <label>ID: <input type="number" name="id" required></label><br>  
 <label>Name: <input type="text" name="name" required></label><br>  
 <label>Quantity: <input type="number" name="quantity"  
 required></label><br>  
 <button type="submit">Add Item</button>  
</form>  
<p th:text="${message}"></p>  
</body>  
</html>

//deleteItemForm.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 03  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 03</title>  
</head>  
<body>  
<h1>22K-5161: Task no. 03</h1>  
<form action="/deleteItem" method="post">  
 <label>ID: <input type="number" name="id" required></label><br>  
 <button type="submit">Delete Item</button>  
</form>  
<p th:text="${message}"></p>  
</body>  
</html>

//getItemForm.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 03  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 03</title>  
</head>  
<body>  
<h1>22K-5161: Task no. 03</h1>  
<form action="/getItem" method="post">  
 <label>ID: <input type="number" name="id" required></label><br>  
 <button type="submit">Get Item</button>  
</form>  
<p th:text="${message}"></p>  
<div th:if="${item != null}">  
 <p>ID: <span th:text="${item.id}"></span></p>  
 <p>Name: <span th:text="${item.name}"></span></p>  
 <p>Quantity: <span th:text="${item.quantity}"></span></p>  
</div>  
</body>  
</html>

//itemMenu.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 03  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 03</title>  
</head>  
<body>  
<h1>22K-5161: Task no. 03</h1>  
<ul>  
 <li><a href="/addItemForm">Add Item</a></li>  
 <li><a href="/getItemForm">Search Item by ID</a></li>  
 <li><a href="/updateItemForm">Update Item</a></li>  
 <li><a href="/deleteItemForm">Delete Item</a></li>  
 <li><a href="/viewAllItems">View All Items</a></li>  
</ul>  
</body>  
</html>

//updateItemForm.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 03  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 03</title>  
</head>  
<body>  
<h1>K22K-5161: Task no. 03</h1>  
<form action="/updateItem" method="post">  
 <label>ID: <input type="number" name="id" required></label><br>  
 <label>New Name: <input type="text" name="name" required></label><br>  
 <label>New Quantity: <input type="number" name="quantity" required></label><br>  
 <button type="submit">Update Item</button>  
</form>  
<p th:text="${message}"></p>  
</body>  
</html>

//viewAllItems.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 03  
-->  
  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>22K-5161: Task no. 03</title>  
</head>  
<body>  
<h1>22K-5161: Task no. 03</h1>  
<table border="1">  
 <tr>  
 <th>ID</th>  
 <th>Name</th>  
 <th>Quantity</th>  
 </tr>  
 <tr th:each="item : ${items}">  
 <td th:text="${item.id}"></td>  
 <td th:text="${item.name}"></td>  
 <td th:text="${item.quantity}"></td>  
 </tr>  
</table>  
</body>  
</html>

Output:

**A black and white screen

Description automatically generated**

A black and white screen

Description automatically generated

A black and white screen

Description automatically generated

A black and white screen

Description automatically generated

A black and white rectangular object

Description automatically generated

A black and white screen

Description automatically generated

A black and white screen

Description automatically generated

A black and white rectangular object

Description automatically generated

A black and white rectangular object

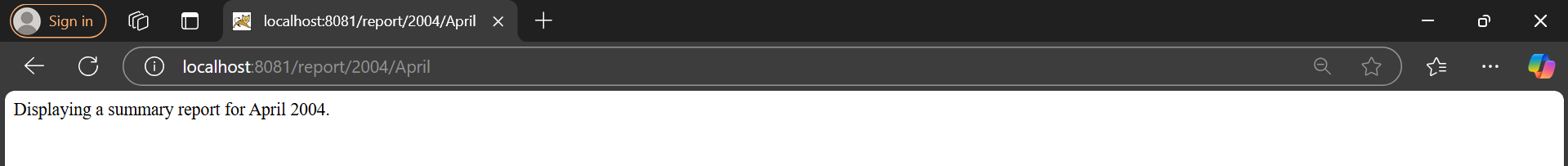
Description automatically generated

**Task 04**

Code:

//Hafsa Salman  
//22K-5161  
//Task no. 04  
  
package com.lab\_11.lab\_11;  
  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.PathVariable;  
import org.springframework.web.bind.annotation.RequestParam;  
import org.springframework.web.bind.annotation.RestController;  
  
@RestController  
public class ReportController  
{  
 @GetMapping("/report/{year}/{month}")  
 public String getReport(  
 @PathVariable int year,  
 @PathVariable String month,  
 @RequestParam(name = "type", defaultValue = "summary") String type)  
 {  
 String reportType = type.equalsIgnoreCase("detailed") ? "a detailed" : "a summary";  
 return "Displaying " + reportType + " report for " + month + " " + year + ".";  
 }  
}

Output:

****

**Task 05**

Code:

//ProductController.java

//Hafsa Salman  
//22K-5161  
//Task no. 05  
  
package com.lab\_11.lab\_11;  
  
import org.springframework.stereotype.Controller;  
import org.springframework.ui.Model;  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.RequestParam;  
  
import java.util.ArrayList;  
import java.util.List;  
  
@Controller  
public class ProductController  
{  
 private List<Product> getProducts()  
 {  
 List<Product> products = new ArrayList<>();  
  
 products.add(new Product("Chocolate", "Food"));  
 products.add(new Product("Pizza", "Food"));  
 products.add(new Product("Sandwich", "Food"));  
 products.add(new Product("Pasta", "Food"));  
 products.add(new Product("Stapler", "Office Supplies"));  
 products.add(new Product("Highlighter", "Office Supplies"));  
 products.add(new Product("Calculator", "Office Supplies"));  
 products.add(new Product("Soda", "Drinks"));  
 products.add(new Product("Milkshake", "Drinks"));  
 products.add(new Product("Smoothie", "Drinks"));  
  
 return products;  
 }  
  
 @GetMapping("/products")  
 public String showProducts(@RequestParam(name = "category", required = false) String category, Model model)  
 {  
 List<Product> products = getProducts();  
  
 List<Product> filteredProducts = new ArrayList<>();  
  
 if (category != null && !category.isEmpty())  
 {  
 for (Product product : products)  
 {  
 if (product.getCategory().equalsIgnoreCase(category))  
 {  
 filteredProducts.add(product);  
 }  
 }  
 }  
  
 else  
 {  
 filteredProducts = products;  
 }  
  
 model.addAttribute("products", filteredProducts);  
 model.addAttribute("selectedCategory", category);  
  
 return "products";  
 }  
}

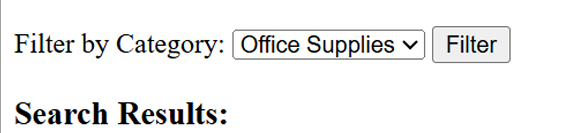
//Product.java

//Hafsa Salman  
//22K-5161  
//Task no. 05  
  
package com.lab\_11.lab\_11;  
  
public class Product  
{  
  
 private String productName;  
 private String category;  
  
 public Product(String productName, String category)  
 {  
 this.productName = productName;  
 this.category = category;  
 }  
  
 public String getProductName()  
 {  
 return productName;  
 }  
  
 public void setProductName(String productName)  
 {  
 this.productName = productName;  
 }  
  
 public String getCategory()  
 {  
 return category;  
 }  
  
 public void setCategory(String category)  
 {  
 this.category = category;  
 }  
}

//product.html

<!--  
Hafsa Salman  
22K-5161  
Task no. 05  
-->  
  
<!DOCTYPE html>  
<html xmlns:th="http://www.thymeleaf.org">  
<head>  
 <title>22K-5161: Task no. 05</title>  
</head>  
<body>  
  
<h1>22K-5161: Task no. 05</h1>  
  
<form action="#" th:action="@{/products}" method="get">  
 <label for="category">Filter by Category:</label>  
 <select name="category" id="category">  
 <option value="" th:selected="${selectedCategory == null}">All</option>  
 <option value="Food" th:selected="${selectedCategory == 'Food'}">Food</option>  
 <option value="Office Supplies" th:selected="${selectedCategory == 'Office Supplies'}">Office Supplies</option>  
 <option value="Drinks" th:selected="${selectedCategory == 'Drinks'}">Drinks</option>  
 </select>  
 <button type="submit">Filter</button>  
</form>  
  
<h3>Search Results:</h3>  
<table>  
 <thead>  
 <tr>  
 <th>Product Name</th>  
 <th>Category</th>  
 </tr>  
 </thead>  
 <tbody>  
 <tr th:each="product : ${products}">  
 <td th:text="${product.productName}">Product Name</td>  
 <td th:text="${product.category}">Category</td>  
 </tr>  
 </tbody>  
</table>  
  
</body>  
</html>

Output:

****