Task: 10 # Spring MVC: Online Grocery Store Application

Date:

Algorithm

1. **Start**
2. **Initialize Spring MVC Project**
    - Create a new Spring MVC project (e.g., using Spring Tool Suite or manually in Eclipse).
    - Configure DispatcherServlet in web.xml.
    - Set up spring-servlet.xml for component scanning and view resolver.
3. **Create Model Class**
    - Define a Product class with fields like id, name, price, and category.
4. **Create Controller**
    - Create GroceryController class.
    - Handle /home request to show the homepage.
    - Handle /products request to display a list of grocery items.
5. **Create Views**
    - Create home.jsp for homepage.
    - Create products.jsp to display all grocery items.
6. **Run the Application**
    - Deploy the project on Apache Tomcat.
    - Access through browser → http://localhost:8080/OnlineGroceryStore/home
7. **Display Output**
    - Homepage with navigation link to grocery list.
    - Product list displayed in table form.

Program:

web.xml

```
<web-app>
  <servlet>
    <servlet-name>spring</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
    <load-on-startup>1</load-on-startup>
  </servlet>

  <servlet-mapping>
    <servlet-name>spring</servlet-name>
    <url-pattern>/</url-pattern>
  </servlet-mapping>
</web-app>
```

spring-servlet.xml (Spring Configuration)

```
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:context="http://www.springframework.org/schema/context"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="
    http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans.xsd
    http://www.springframework.org/schema/context
    http://www.springframework.org/schema/context/spring-context.xsd">

  <!-- Enable annotation-based configuration -->
  <context:component-scan base-package="com.grocerystore" />
```

```
    <!-- View Resolver -->
    <bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <property name="prefix" value="/WEB-INF/views/" />
        <property name="suffix" value=".jsp" />
    </bean>
</beans>
```

Product.java (Model)

```
package com.grocerystore.model;

public class Product {
    private int id;
    private String name;
    private double price;
    private String category;

    public Product(int id, String name, double price, String category) {
        this.id = id;
        this.name = name;
        this.price = price;
        this.category = category;
    }

    // Getters and Setters
    public int getId() { return id; }
    public String getName() { return name; }
    public double getPrice() { return price; }
    public String getCategory() { return category; }
}
```

GroceryController.java (Controller)

```
package com.grocerystore.controller;

import java.util.*;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
import com.grocerystore.model.Product;

@Controller
public class GroceryController {

    @RequestMapping("/home")
    public String home() {
        return "home";
    }

    @RequestMapping("/products")
    public String showProducts(Model model) {
        List<Product> productList = new ArrayList<>();
        productList.add(new Product(1, "Apple", 120.0, "Fruits"));
        productList.add(new Product(2, "Rice", 60.0, "Grains"));
        productList.add(new Product(3, "Milk", 50.0, "Dairy"));
        productList.add(new Product(4, "Bread", 40.0, "Bakery"));

        model.addAttribute("products", productList);
        return "products";
    }
}
```

home.jsp (View)

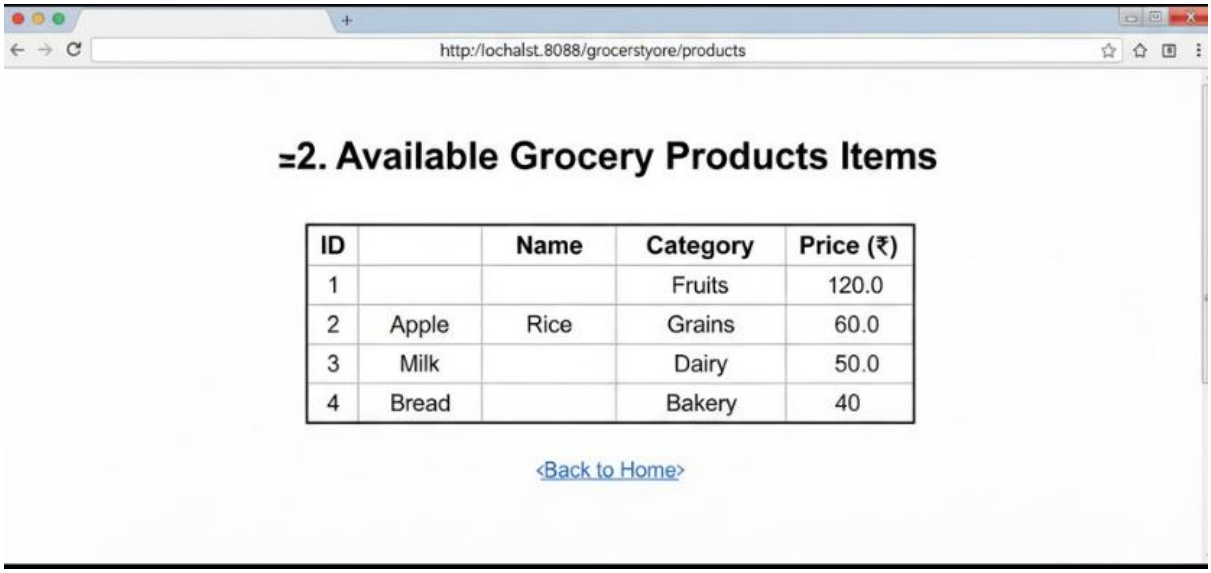
```
<!DOCTYPE html>
<html>
<head>
    <title>Online Grocery Store</title>
</head>
<body style="text-align:center;">
    <h1>Welcome to the Online Grocery Store</h1>
    <p>Click below to view available grocery items</p>
    <a href="products">View Products</a>
</body>
</html>
```

products.jsp (View)

```
<!DOCTYPE html>
<html>
<head>
    <title>Grocery Products</title>
</head>
<body style="text-align:center;">
    <h2>Available Grocery Items</h2>
    <table border="1" align="center" cellpadding="10">
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Category</th>
            <th>Price (₹)</th>
        </tr>
        <c:forEach var="item" items="${products}">
```

```
            <tr>
                <td>${item.id}</td>
                <td>${item.name}</td>
                <td>${item.category}</td>
                <td>${item.price}</td>
            </tr>
        </c:forEach>
    </table>
    <br>
    <a href="home">Back to Home</a>
</body>
</html>
```

OUTPUT



Result:

The Online Grocery Store was successfully developed using Spring MVC. It displays grocery items dynamically on the products page and demonstrates proper MVC architecture with smooth navigation and data handling.