

## Day14

### Casestudy2: Order Service

#### Pom.xml:

```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">

    <modelVersion>4.0.0</modelVersion>

    <parent>

        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>3.5.4</version>
        <relativePath/> <!-- lookup parent from repository -->
    </parent>

    <groupId>com.example</groupId>
    <artifactId>resturant</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>resturant</name>
    <description>Demo project for Spring Boot</description>
    <url/>
    <licenses>
        <license/>
    </licenses>
    <developers>
        <developer/>
```

```
</developers>

<scm>
    <connection/>
    <developerConnection/>
    <tag/>
    <url/>
</scm>

<properties>
    <java.version>17</java.version>
</properties>

<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-data-jpa</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
    <!-- Swagger / OpenAPI 3 for Spring Boot 3 -->
<dependency>
    <groupId>org.springdoc</groupId>
    <artifactId>springdoc-openapi-starter-webmvc-ui</artifactId>
    <version>2.2.0</version>
</dependency>
```

```
<!-- https://mvnrepository.com/artifact/io.springfox/springfox-swagger-ui -->
```

```
<dependency>
```

```
    <groupId>io.springfox</groupId>
```

```
    <artifactId>springfox-swagger-ui</artifactId>
```

```
    <version>3.0.0</version>
```

```
</dependency>
```

```
    <dependency>
```

```
        <groupId>org.springframework.boot</groupId>
```

```
        <artifactId>spring-boot-devtools</artifactId>
```

```
        <scope>runtime</scope>
```

```
        <optional>true</optional>
```

```
    </dependency>
```

```
    <dependency>
```

```
        <groupId>com.h2database</groupId>
```

```
        <artifactId>h2</artifactId>
```

```
        <scope>runtime</scope>
```

```
    </dependency>
```

```
    <dependency>
```

```
        <groupId>org.springframework.boot</groupId>
```

```
        <artifactId>spring-boot-starter-test</artifactId>
```

```
        <scope>test</scope>
```

```
    </dependency>
```

```
</dependencies>
```

```
<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>

</project>
```

Entity:

Menu.java:

```
package com.example.resturant.entity;
```

```
import jakarta.persistence.*;
```

```
@Entity
```

```
public class Menu{
```

```
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    private String name;
```

```
    private String description;
```

```
    private double price;
```

@ManyToOne

@JoinColumn(name = "restaurant\_id")

private Restaurant restaurant;

public Menu() {}

public Menu(Long id, String name, String description, double price,  
Restaurant restaurant) {

    this.id = id;

    this.name = name;

    this.description = description;

    this.price = price;

    this.restaurant = restaurant;

}

public Long getId() {

    return id;

}

public void setId(Long id) {

    this.id = id;

}

public String getName() {

```
        return name;
    }
```

```
public void setName(String name) {
    this.name = name;
}
```

```
public String getDescription() {
    return description;
}
```

```
public void setDescription(String description) {
    this.description = description;
}
```

```
public double getPrice() {
    return price;
}
```

```
public void setPrice(double price) {
    this.price = price;
}
```

```
public Restaurant getRestaurant() {
    return restaurant;
}
```

```
}
```

```
public void setRestaurant(Restaurant restaurant) {  
    this.restaurant = restaurant;  
}
```

```
}
```

Restaurerent.java:

```
package com.example.restaurant.entity;
```

```
import jakarta.persistence.*;
```

```
@Entity
```

```
public class Restaurant {
```

```
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    private String name;
```

```
    private String location;
```

```
    private String contactNumber;
```

```
    public Restaurant(Long id, String name, String location, String  
contactNumber) {
```

```
        this.id = id;
```

```
        this.name = name;

        this.location = location;

        this.contactNumber = contactNumber;
    }

    public Restaurant() {}

    public Long getId() {
        return id;
    }

    public void setId(Long id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getLocation() {
        return location;
    }

    public void setLocation(String location) {
        this.location = location;
    }

    public String getContactNumber() {
        return contactNumber;
    }
}
```



```
    }  
    public void setContactNumber(String contactNumber) {  
        this.contactNumber = contactNumber;  
    }  
  
}
```

Repository:

MenuRepository:

```
package com.example.resturant.repository;
```

```
import com.example.resturant.entity.Menu;
```

```
import org.springframework.data.jpa.repository.JpaRepository;
```

```
import java.util.List;
```

```
public interface MenuItemRepository extends JpaRepository<MenuItem, Long>  
{
```

```
    List<MenuItem> findByRestaurantId(Long restaurantId);
```

```
}
```

RestaurantRepository:

```
package com.example.resturant.repository;
```

```
import com.example.resturant.entity.Restaurant;
```

```
import org.springframework.data.jpa.repository.JpaRepository;
```

```
public interface RestaurantRepository extends JpaRepository<Restaurant,  
Long> {  
  
}
```

Controller:

MenuController:

```
package com.example.resturant.controller;
```

```
import com.example.resturant.entity.MenuItem;
```

```
import com.example.resturant.service.MenuItemService;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.http.ResponseEntity;
```

```
import org.springframework.web.bind.annotation.*;
```

```
import java.util.List;
```

```
@RestController
```

```
@RequestMapping("/menu-items")
```

```
public class MenuController {
```

```
    @Autowired
```

```
    private MenuItemService menuItemService;
```

```

    @PostMapping("/restaurant/{restaurantId}")

    public ResponseEntity<Menu> addMenu (@PathVariable Long restaurantId,
    @RequestBody Menu menu) {

        return ResponseEntity.ok(menuService.addMenu (restaurantId, menu));

    }


    @GetMapping("/restaurant/{restaurantId}")

    public List<Menu> getMenusByRestaurant(@PathVariable Long restaurantId)
    {

        return menuService.getMenusByRestaurant(restaurantId);

    }


    @PutMapping("/{id}")

    public Menu updateMenu(@PathVariable Long id, @RequestBody Menu
    menu) {

        return menuService.updateMen(id, menu);

    }


    @DeleteMapping("/{id}")

    public void deleteMenu(@PathVariable Long id) {

        menuService.deleteMenu(id);

    }

}

```

RestaurantController:

```
package com.example.resturant.controller;
```

```
import com.example.resturant.entity.Menuitem;
import com.example.resturant.entity.Restaurant;
import com.example.resturant.service.MenuitemService;
import com.example.resturant.service.RestaurantService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.*;
```

```
import java.util.List;
```

```
@RestController
```

```
@RequestMapping("/restaurants")
```

```
public class RestaurantController {
```

```
    @Autowired
```

```
    private RestaurantService restaurantService;
```

```
    @Autowired
```

```
    private MenuitemService menuitemService;
```

```
    // --- Restaurant Endpoints ---
```

```
    @PostMapping
```

```
public ResponseEntity<Restaurant> addRestaurant(@RequestBody
Restaurant restaurant) {
    return ResponseEntity.ok(restaurantService.addRestaurant(restaurant));
}
```

@GetMapping

```
public List<Restaurant> getAllRestaurants() {
    return restaurantService.getAllRestaurants();
}
```

@GetMapping("/{id}")

```
public Restaurant getRestaurantById(@PathVariable Long id) {
    return restaurantService.getRestaurantById(id);
}
```

@PutMapping("/{id}")

```
public Restaurant updateRestaurant(@PathVariable Long id, @RequestBody
Restaurant restaurant) {
    return restaurantService.updateRestaurant(id, restaurant);
}
```

@DeleteMapping("/{id}")

```
public void deleteRestaurant(@PathVariable Long id) {
    restaurantService.deleteRestaurant(id);
}
```

// --- Menu Item Endpoints ---

@PostMapping("/{id}/menu-items")

```
public ResponseEntity<MenuItem> addMenuItem(@PathVariable Long id,
@RequestBody MenuItem menuItem) {
    return ResponseEntity.ok(menuItemService.addMenuItem(id, menuItem));
}
```

@GetMapping("/{id}/menu-items")

```
public List<MenuItem> getMenuItems(@PathVariable Long id) {
    return menuItemService.getMenuItemsByRestaurant(id);
}
```

@PutMapping("/menu-items/{menuItemId}")

```
public MenuItem updateMenuItem(@PathVariable Long menuItemId,
@RequestBody MenuItem menuItem) {
    return menuItemService.updateMenuItem(menuItemId, menuItem);
}
```

@DeleteMapping("/menu-items/{menuItemId}")

```
public void deleteMenuItem(@PathVariable Long menuItemId) {
    menuItemService.deleteMenuItem(menuItemId);
}
}
```

Service:

MenuService.java:

```
package com.example.resturant.service;
```

```
import com.example.resturant.entity.MenuItem;
```

```
import com.example.resturant.entity.Restaurant;
```

```
import com.example.resturant.repository.MenuItemRepository;
```

```
import com.example.resturant.repository.RestaurantRepository;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Service;
```

```
import java.util.List;
```

```
@Service
```

```
public class MenuService {
```

```
    @Autowired
```

```
    private MenuItemRepository menuItemRepository;
```

```
    @Autowired
```

```
    private RestaurantRepository restaurantRepository;
```

```
    public Menu addMenu(Long restaurantId, Menu menu) {
```

```
        Restaurant restaurant = restaurantRepository.findById(restaurantId)
```

```
            .orElseThrow(() -> new RuntimeException("Restaurant not found"));
```

```
    menu.setRestaurant(restaurant);  
    return menuRepository.save(menu);  
}
```

```
public List<Menu> getMenusByRestaurant(Long restaurantId) {  
    return menuRepository.findByRestaurantId(restaurantId);  
}
```

```
public Menu updateMenu(Long id, Menu updatedItem) {  
    Menu menu = menuRepository.findById(id)  
        .orElseThrow(() -> new RuntimeException("Menu item not found"));
```

```
    menu.setName(updatedItem.getName());  
    menu.setDescription(updatedItem.getDescription());  
    menu.setPrice(updatedItem.getPrice());  
    return menuRepository.save(menu);  
}
```

```
public void deleteMenu(Long id) {  
    menuRepository.deleteById(id);  
}  
}
```

RestaurantService:

```
package com.example.resturant.service;
```



```
import com.example.resturant.entity.Restaurant;
import com.example.resturant.repository.RestaurantRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
```

```
import java.util.List;
```

```
@Service
```

```
public class RestaurantService {
```

```
    @Autowired
```

```
    private RestaurantRepository restaurantRepository;
```

```
    public Restaurant addRestaurant(Restaurant restaurant) {
```

```
        return restaurantRepository.save(restaurant);
```

```
    }
```

```
    public List<Restaurant> getAllRestaurants() {
```

```
        return restaurantRepository.findAll();
```

```
    }
```

```
    public Restaurant getRestaurantById(Long id) {
```

```
        return restaurantRepository.findById(id)
```

```
            .orElseThrow(() -> new RuntimeException("Restaurant not found"));
```

```
}
```

```
public Restaurant updateRestaurant(Long id, Restaurant updated) {  
    Restaurant restaurant = getRestaurantById(id);  
    restaurant.setName(updated.getName());  
    restaurant.setLocation(updated.getLocation());  
    restaurant.setContactNumber(updated.getContactNumber());  
    return restaurantRepository.save(restaurant);  
}
```

```
public void deleteRestaurant(Long id) {  
    restaurantRepository.deleteById(id);  
}  
}
```

Application.properties:

server.port=8081

spring.datasource.url=jdbc:h2:mem:restaurantdb

spring.datasource.driverClassName=org.h2.Driver

spring.datasource.username=sa

spring.datasource.password=

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

springdoc.swagger-ui.path=/swagger-ui.html

springdoc.api-docs.path=/api-docs

spring.h2.console.enabled=true