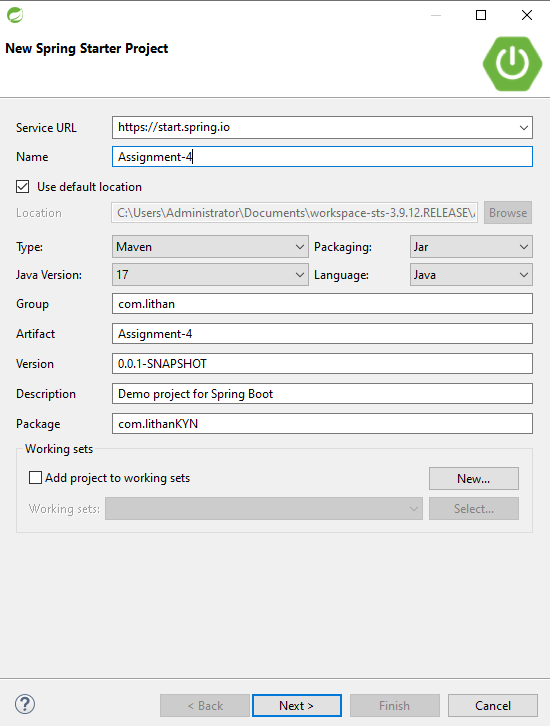
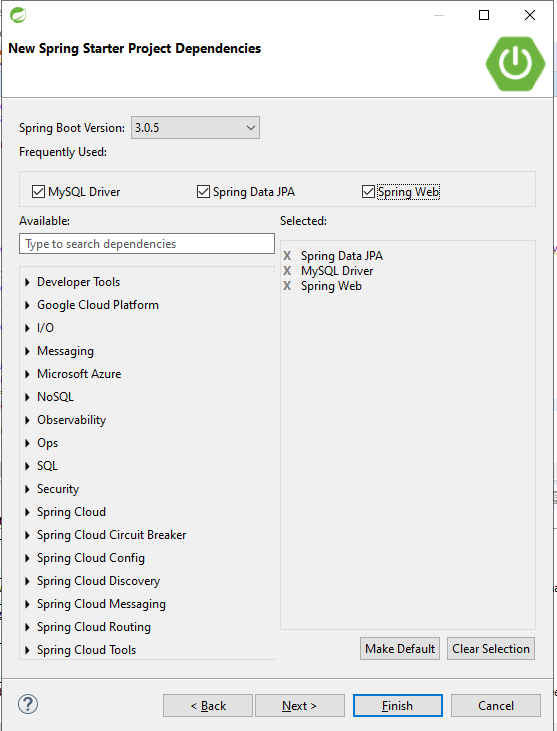
Assignment -4

|  |  |
| --- | --- |
| **Student Name/ID Number:** | Agung Yuda Pratama / 0922-076 |
| **Academic Year:** | 2022 |
| **Unit Assessor:** | Archana Sakpal |
| **Project Title:** | Assignment 4 - SPRING BOOT + JPA |
| **Issue Date:** |  |
| **Submission Date:** |  |
| **Internal Verifier Name:** |  |
| **Date:** | 14-04-2023 |

|  |
| --- |
| **Learner declaration** |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.    14/04/2023  Student signature: Date: |

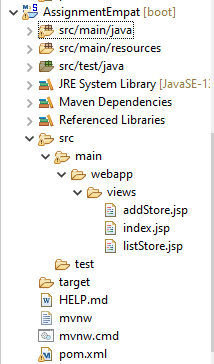
Let’s build on the previous assignment. (Develop “Know-Your-Neighbourhood”) application. The goal of this application is to provide details on all stores in the user’s neighbourhood.

1. Create a Spring Boot application for “Know-Your-Neighbourhood”.

Figure 1 – Start.spring.io (Spring MVC + JPA)

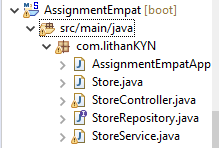
1. Add support for JSP views and create required folder structure.

Figure 2 – KYN Application Jsp file location



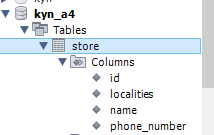
1. Move already developed classes into this project.

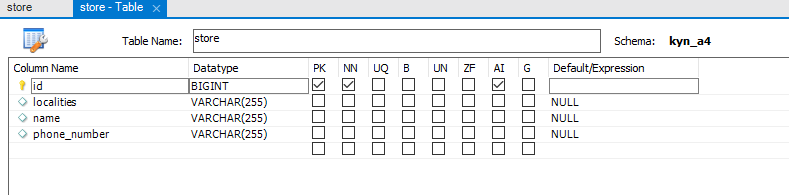
Figure 3 – KYN application project created classes



1. Create store table in the database

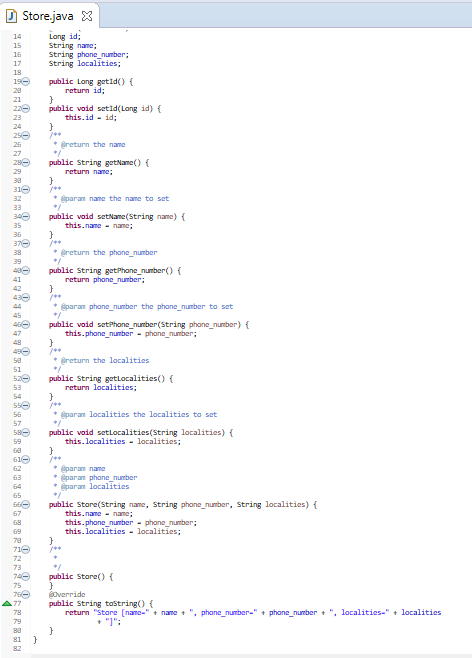
Figure 4 – Db of KYN application





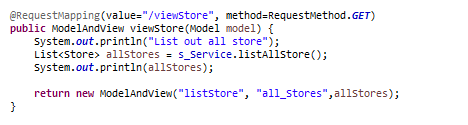
1. Annotate all the fields of Store object appropriately.

Figure 5 – Store.java



1. Develop all components required to view the stores
2. Add method to existing Controller class to receive the request to fetch stores

Figure 6 – StoreController.java (fetch request mapping)



1. Enhance Store service and Store Controller objects to support CRUD operations on store object.

Figure 6.1 – StoreController.java (addStore request mapping)

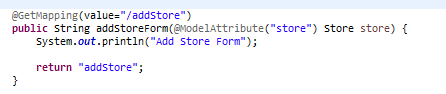


Figure 6.2 – StoreController.java (saveStore request mapping)

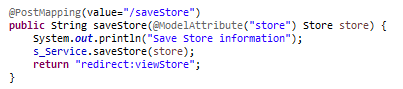


Figure 7 – StoreController.java (deleteStore request mapping)

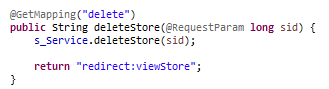


Figure 7.1 – StoreController.java (updateStore request mapping)

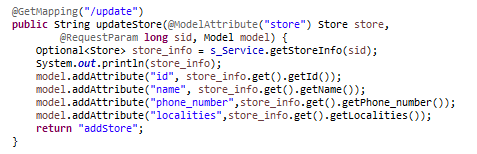
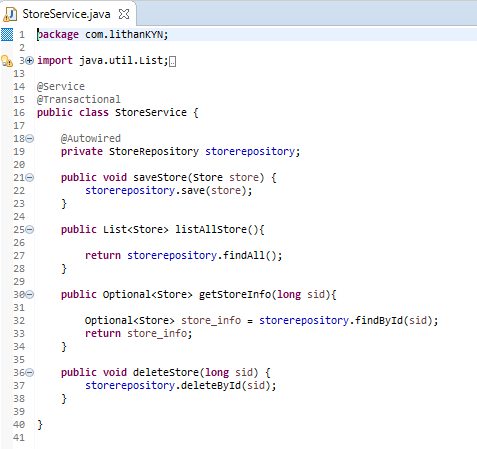
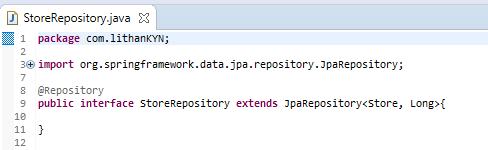


Figure 8 – StoreService.java (CRUD)



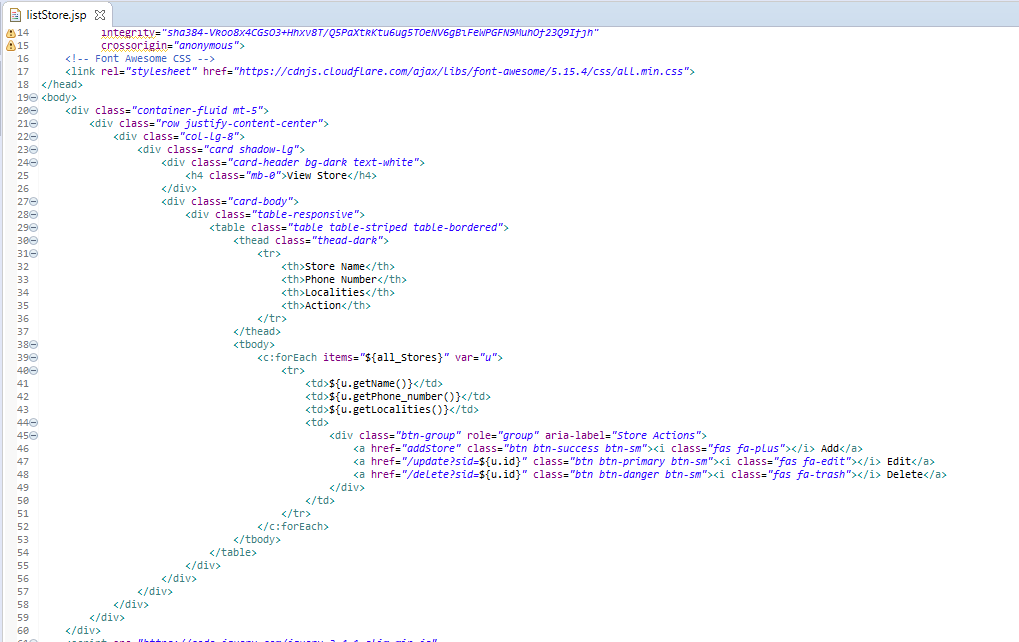
1. Add method existing repository class to perform crud operations on Store object.

Figure 9 – StoreRepository.java



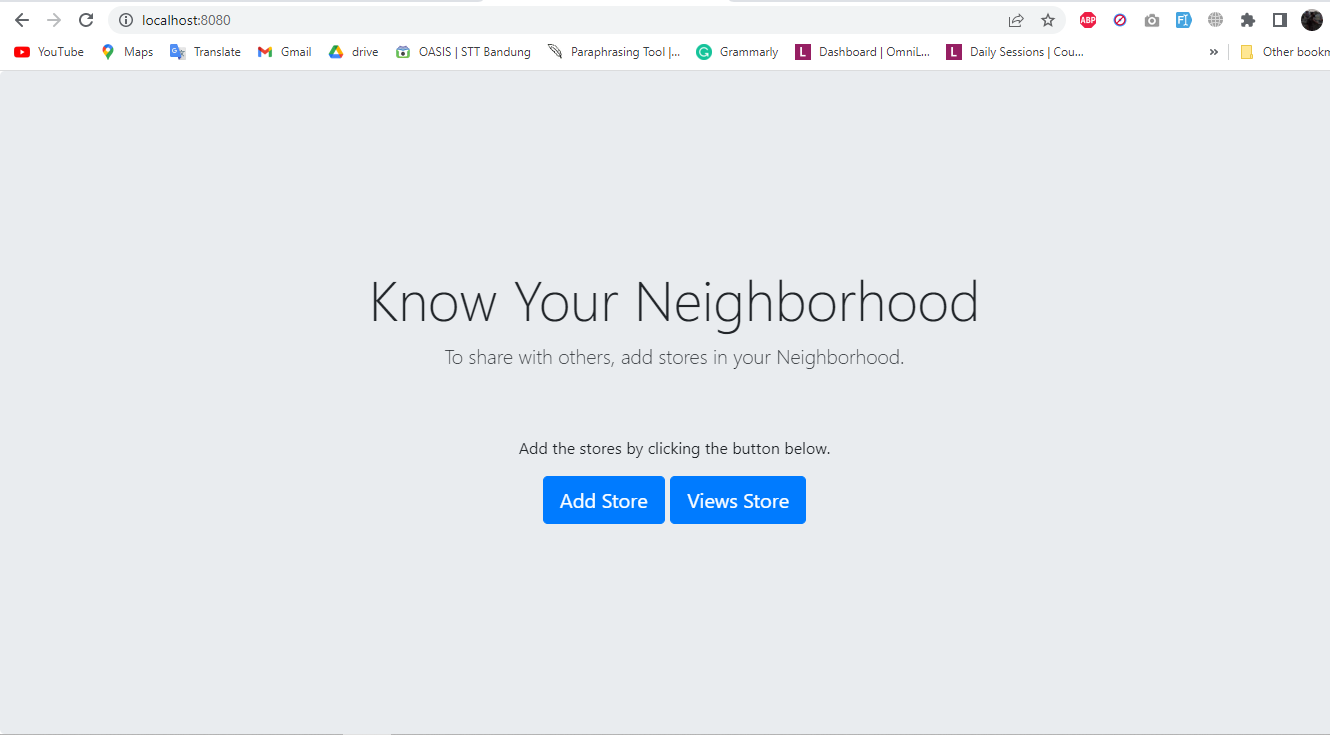
1. Create HTML to view the stores. Show name, phone number and localities it serves for each store.

Figure 10 – listStore.java



1. Ensure that view stores request works end-to-end. (i.e., should be able to submit request to view the stores in the browser and get the page back with all stores).

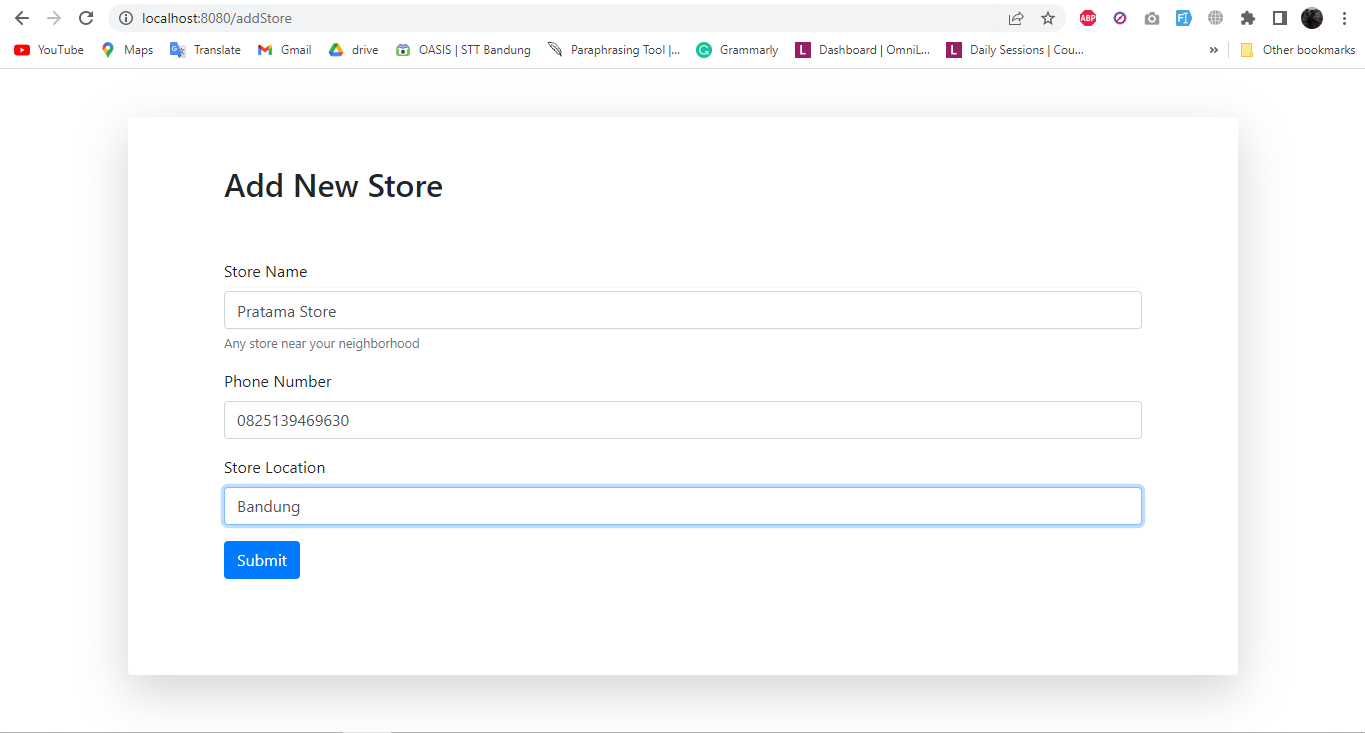
Figure 11 – index.jsp(result)



Click Add Store to add store information

Click View Store to view store available

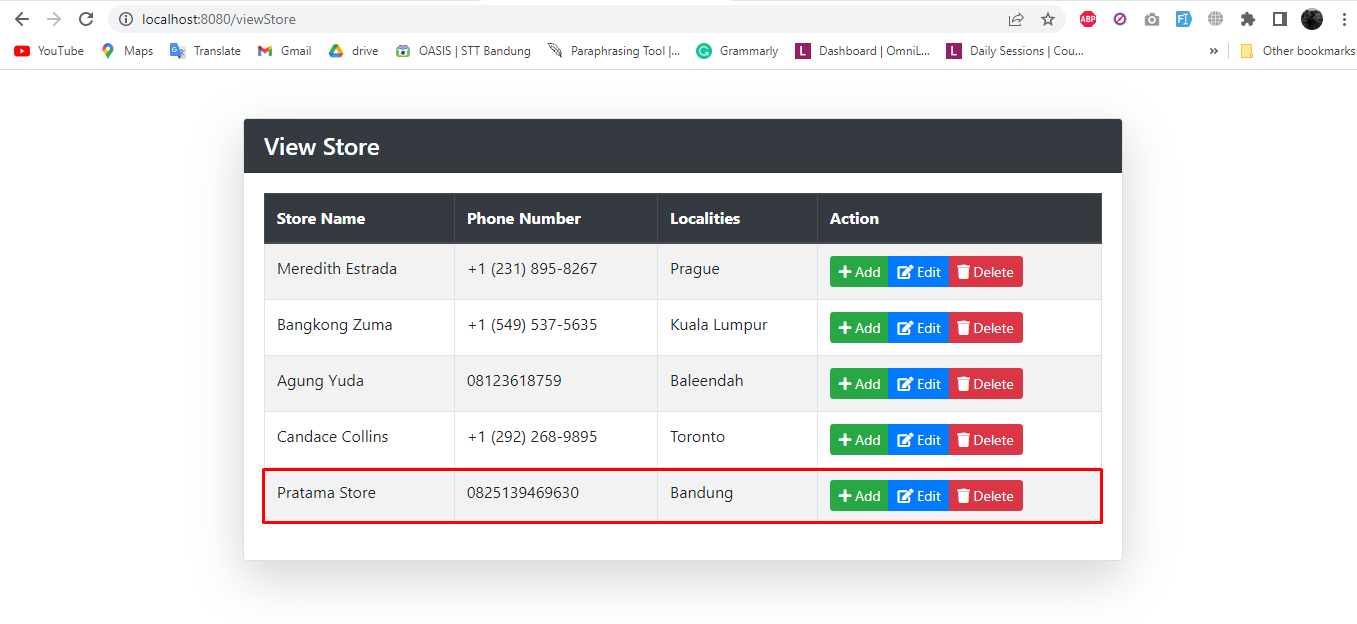
Figure 12 – addStore.jsp(result)



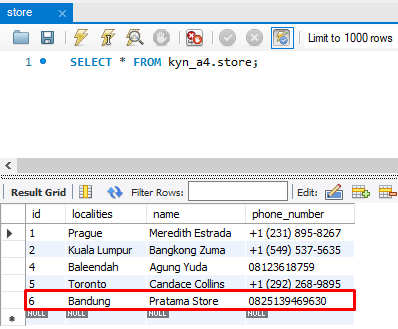
Enter New Store data (name, phone number, localities)

Submit the store form after completing the data

Figure 13 – storeList.jsp(result)



Added Store appears in listing page



Added Store appears in kyn\_a4 database

1. Create an HTML page to perform Add store, Fetch stores, Update store, & Delete store operations from the UI.

Figure 14 – CRUD created in jsp

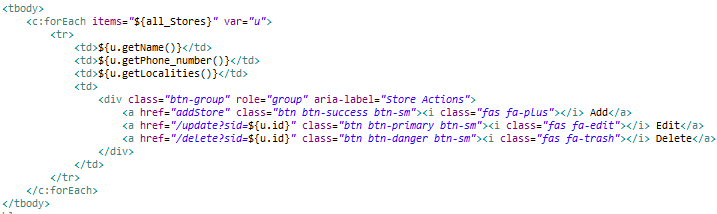
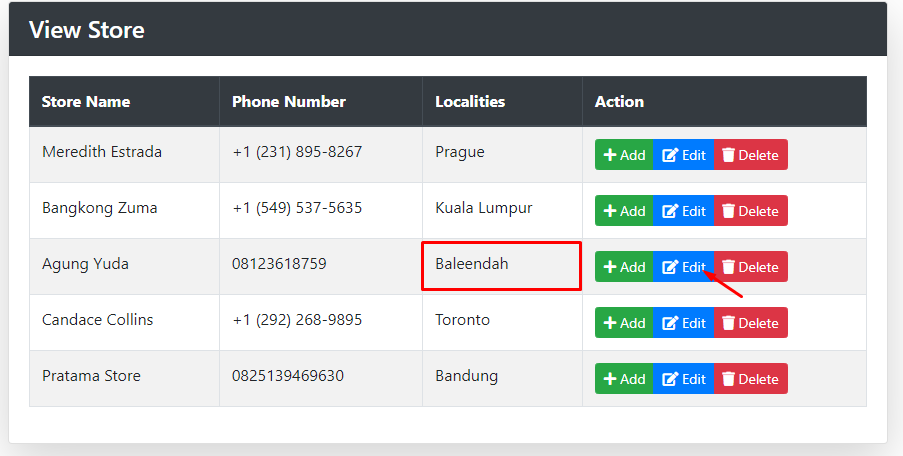
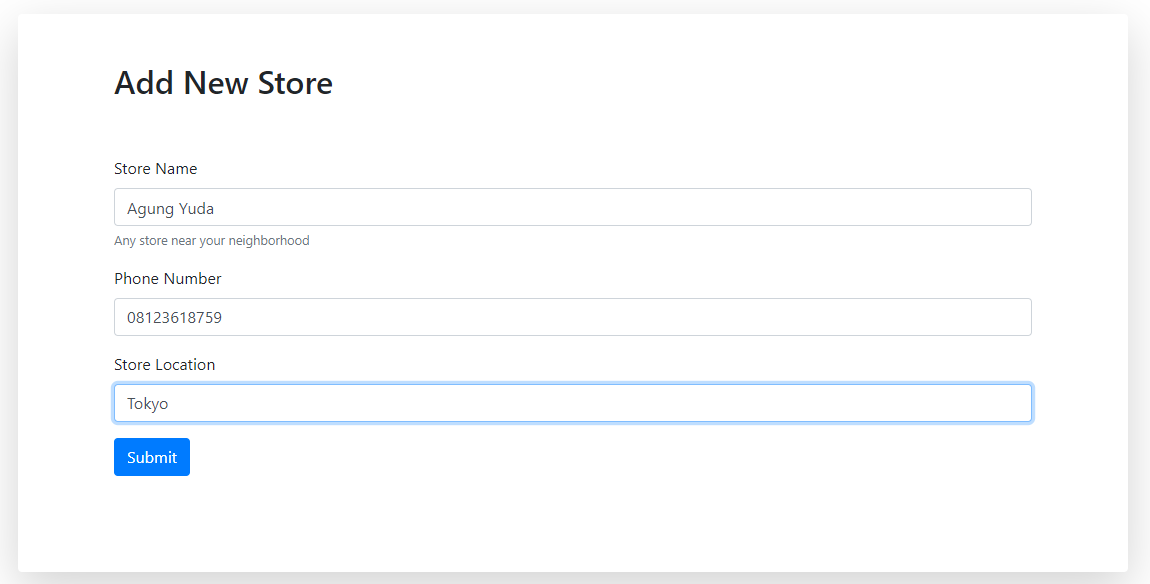


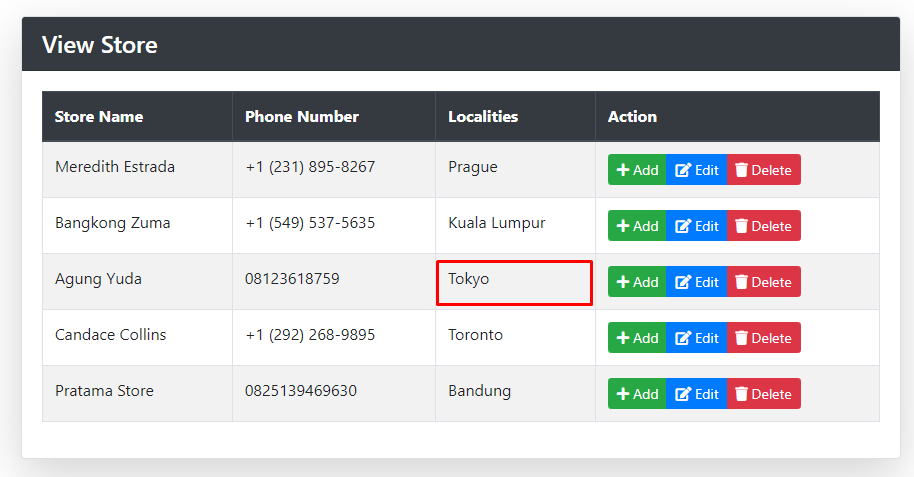
Figure 15 – Update store data (Change Family Mart localities to KL)



Click Update to make the changes



Click Submit for the new changes



Location is changed from Baleendah to Tokyo

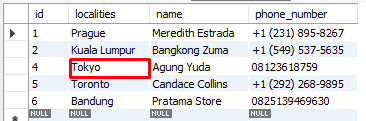
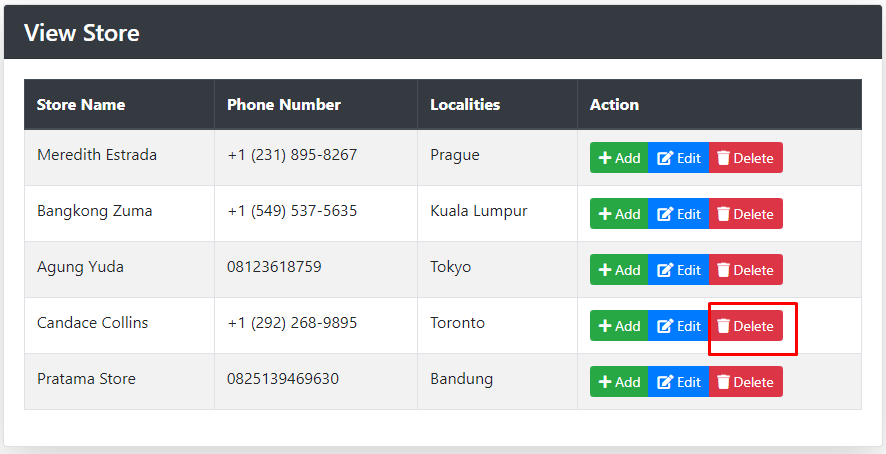
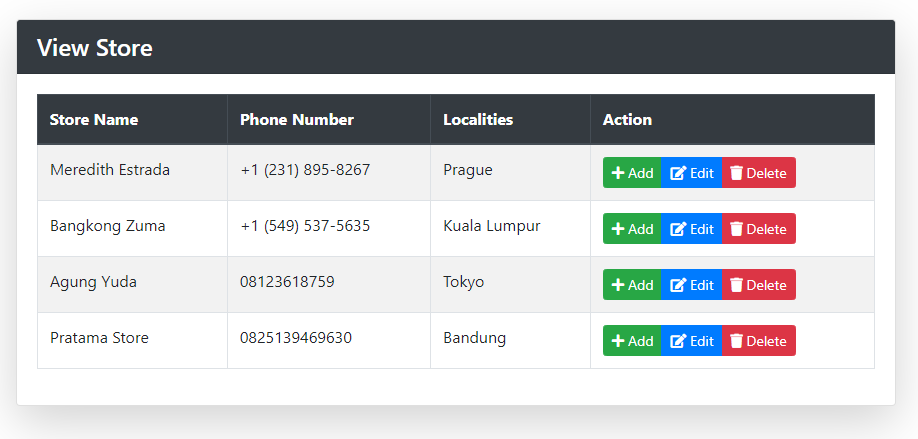


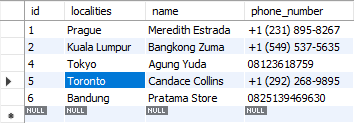
Figure 16 – Delete store data (Delete Bakery Deluxe store)



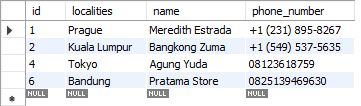
Click delete to delete selected store



Candace Collins is deleted

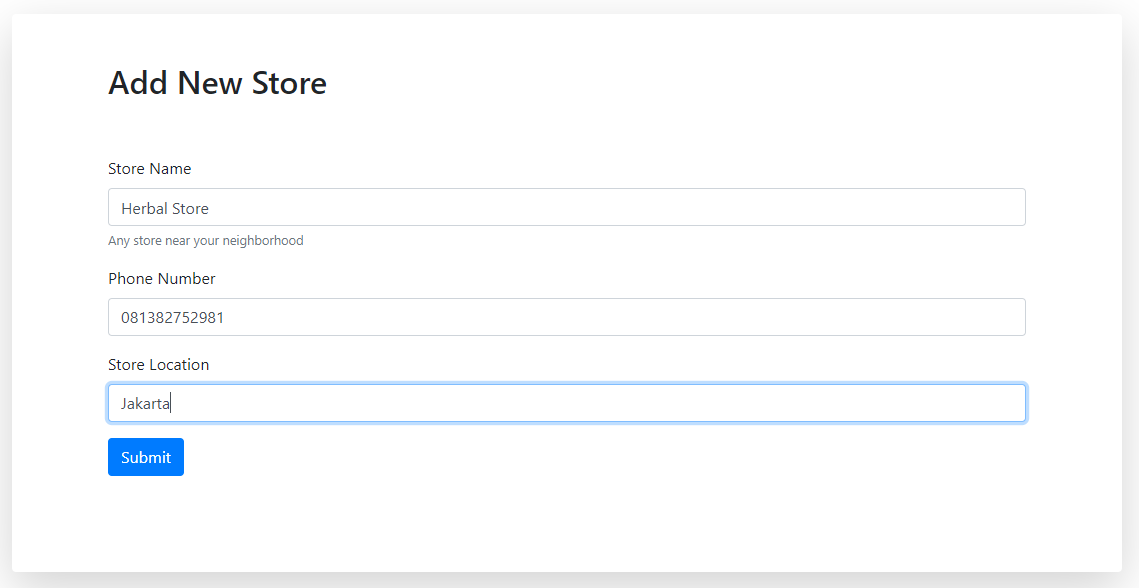


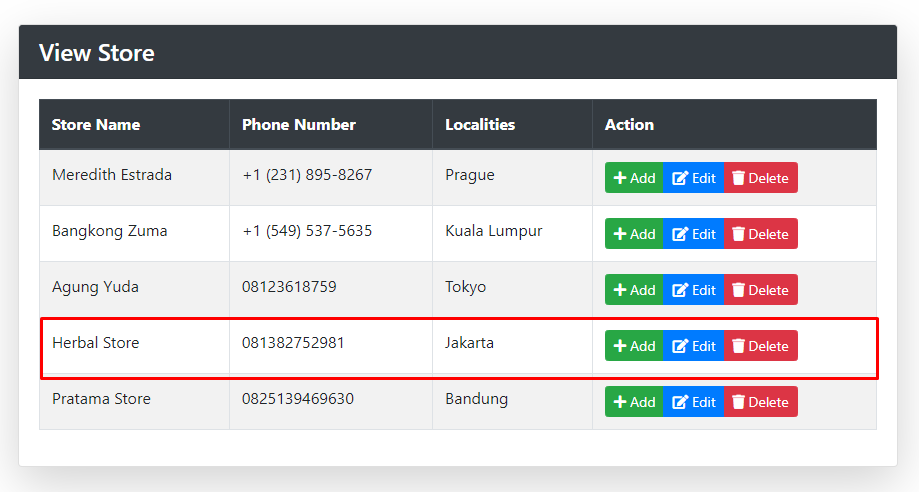
Before

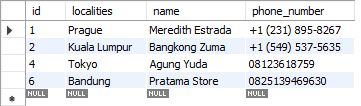


After

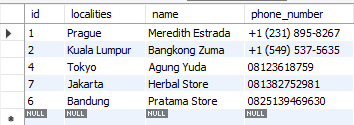
Figure 17 – Add store data (store=Herbal Store, phone number = 081382752981, localities=Jakarta)







Before



After