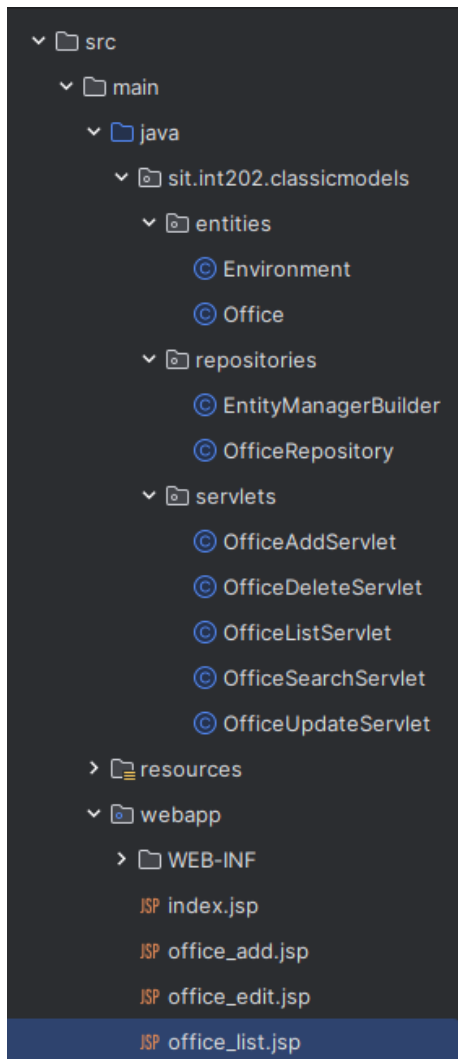


# Office CRUD Assignment 1

Project Structure :



## 1. Package entities

1.1 Office เอาไว้เก็บ attribute และ query ของ office ที่เราจะเรียกไปใช้ใน EntityManager

1.2 Environment เก็บ persistence unit เพื่อนำไปเรียกใช้ได้ง่ายและเป็นระบบ

## 2. Package repositories

2.1 EntityManagerBuilder เอาไว้ createEntityManager เพื่อให้ class ใน repo เรียกใช้

2.2 เป็น model เอาไว้ควบคุม method เพื่อนำไปเรียกใช้

## 3. Package Servlets

เป็น controller เอาไว้เก็บ servlet เพื่อจะทำไปใช้ใน view

เดี๋ยวผมจะอธิบาย หน้าที่ของแต่ละตัวในหน้าต่อไปครับ

## Servlets

### 1. OfficeListServlet

```
@WebServlet(name = "OfficeListServlet", value = "/office-list")
public class OfficeListServlet extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        OfficeRepository repository = new OfficeRepository();
        request.setAttribute("offices", repository.findAll());
        getServletContext().getRequestDispatcher("/office_list.jsp").forward(request, response);
    }
}
```

เขียนอยู่ใน method `doGet()` เพื่อเรียกใช้ class `OfficeRepository` จากนั้นเรียกใช้ method `findAll()` จากนั้น `setAttribute` เพื่อไปเรียกใช้ใน `office_list.jsp` ในส่วนของ `loop forEach()` เพื่อแสดงออกมาเป็น table

### 2. OfficeSearchServlet

```
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    String searchCityCountry = request.getParameter("searchByCityOrCountry");
    if(searchCityCountry != null){
        OfficeRepository repository = new OfficeRepository();
        List<Office> offices = repository.findByCityOrCountry(searchCityCountry);
        request.setAttribute("offices", offices);
    }
    request.getServletContext().getRequestDispatcher("/office_list.jsp").forward(request, response);
}
}
```

ในส่วนนี้จะสร้าง String เพื่อ `getParameter()` แล้วตั้งชื่อเพื่อนำไปใช้ใน view แล้วเช็คเงื่อนไขเพื่อเรียกใช้ method `findByCityOrCountry()` ที่อยู่ใน `officeRepository` class และก็เก็บอยู่ในรูปแบบ `List<>` จากนั้นก็ `setAttribute` แล้วก็ map ไปที่ `office_list.jsp`

### 3.OfficeAddServlet

```
@WebServlet(name = "OfficeAddServlet", value = "/office-add")
public class OfficeAddServlet extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        request.getRequestDispatcher(s: "/office_add.jsp").forward(request, response);
    }
}
```

method doGet() ไม่มีอะไรมากแค่ส่ง request ไปที่ office\_add.jsp

```
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    OfficeRepository repository = new OfficeRepository();
    String officeCode = request.getParameter(s: "officeCode");
    String city = request.getParameter(s: "city");
    String phone = request.getParameter(s: "phone");
    String addressLine1 = request.getParameter(s: "addressLine1");
    String country = request.getParameter(s: "country");
    String postalCode = request.getParameter(s: "postalCode");
    String territory = request.getParameter(s: "territory");
```

```
    if (inputNotEmpty(officeCode, city, phone, addressLine1, country, postalCode, territory)) {
        Office newOffice = new Office();
        newOffice.setOfficeCode(officeCode);
        newOffice.setCity(city);
        newOffice.setPhone(phone);
        newOffice.setAddressLine1(addressLine1);
        newOffice.setCountry(country);
        newOffice.setPostalCode(postalCode);
        newOffice.setTerritory(territory);

        boolean statusAdd = repository.insert(newOffice);

        if (statusAdd) {
            request.setAttribute(s: "messageAdd", o: "Add Successfully");
            request.setAttribute(s: "statusAdd", o: true);
            request.setAttribute(s: "offices", repository.findAll());
            request.getRequestDispatcher(s: "/office_list.jsp").forward(request, response);
        } else {
            request.setAttribute(s: "messageAdd", o: "Add not Successful. Please check your input.");
            request.setAttribute(s: "statusAdd", o: false);
            request.setAttribute(s: "offices", repository.findAll());
            request.getRequestDispatcher(s: "/office_add.jsp").forward(request, response);
        }
    } else {
        request.setAttribute(s: "messageAdd", o: "Add not Successful. Please check your input.");
        request.setAttribute(s: "statusAdd", o: false);
        request.setAttribute(s: "offices", repository.findAll());
        request.getRequestDispatcher(s: "/office_list.jsp").forward(request, response);
    }
}
```

ส่วน method doPost() นี่เป็นส่วนสำคัญ ทำงานโดยการ new object เพื่อเรียก

OfficeRepository class จากนั้นก็ request.getParameter() เพื่อนำไปเรียกใช้งาน ใน jsp

ส่วนของ form เพื่อที่จะ add ข้อมูล จากนั้นก็จะทำการเช็คเงื่อนไขว่า attribute ทุกตัวจะต้อง

ไม่เป็น null และเป็นค่าว่าง โดยการเรียกใช้ method `inputNotEmpty()` ที่ได้สร้างไว้ข้างล่าง

จากนั้นก็สร้างเงื่อนไขเช็ค **status** ว่าสามารถเพิ่มได้สำเร็จมั้ยโดยจะมี ข้อความที่กำหนดโชว์ขึ้นมาถ้าตรงตามเงื่อนไขที่ตั้งไว้

## Method `inputNotEmpty`

```
public boolean inputNotEmpty(String officeCode, String city, String phone, String addressLine1, String country, String postalCode, String territory) {
    if (officeCode != null && city != null && phone != null && addressLine1 != null && country != null && postalCode != null && territory != null &&
        !officeCode.isEmpty() && !city.isEmpty() && !phone.isEmpty() && !addressLine1.isEmpty() && !country.isEmpty() && !postalCode.isEmpty() && !territory.isEmpty()) {
        return true;
    }
    return false;
}
```

## 4. OfficeUpdateServlet

```
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    OfficeRepository repository = new OfficeRepository();
    String officeCode = request.getParameter("officeCode");
    if (officeCode != null) {
        Office office = repository.find(officeCode);
        request.setAttribute("office", office);
    }
    request.getRequestDispatcher("/office_edit.jsp").forward(request, response);
}
```

ในส่วน `doGet()` ของ `Update` จะทำงานโดยการ new object `OfficeRepository` แล้วก็ `request getParam` เพื่อนำไปเรียกใช้ใน `jsp` จากนั้นเข้าเงื่อนไข ถ้าไม่เท่ากับ `null` จะเรียกใช้ `method find()` ใน `OfficeRepositories` สุดท้ายทำการส่งไปยัง `office_edit.jsp` แล้วเรียกดูผลลัพธ์ที่ add ผ่าน `method findAll()`

```

if (inputNotEmpty(officeCode, city, phone, addressLine1, country, postalCode, territory)) {
    Office newOffice = new Office();

    newOffice.setOfficeCode(officeCode);
    newOffice.setCity(city);
    newOffice.setPhone(phone);
    newOffice.setAddressLine1(addressLine1);
    newOffice.setCountry(country);
    newOffice.setPostalCode(postalCode);
    newOffice.setTerritory(territory);

    boolean statusUpdate = repository.update(newOffice);

    if (statusUpdate) {
        request.setAttribute("messageUpdate", "Update Successfully");
        request.setAttribute("statusUpdate", true);
        request.setAttribute("office", repository.findAll());
        getServletContext().getRequestDispatcher("/office_list.jsp").forward(request, response);
    } else {
        request.setAttribute("messageUpdate", "Update not Successful. Please check your input.");
        request.setAttribute("statusUpdate", false);
        request.setAttribute("office", repository.findAll());
        getServletContext().getRequestDispatcher("/office_edit.jsp").forward(request, response);
    }
} else {
    request.setAttribute("messageUpdate", "Update not Successful. Please check your input.");
    request.setAttribute("statusUpdate", false);
    request.setAttribute("office", repository.findAll());
    getServletContext().getRequestDispatcher("/office_list.jsp").forward(request, response);
}
}

```

การทำงานอันนี้จะทำงานคล้ายๆกับอันข้างบน ทำงานโดยการ **new object** เพื่อเรียก **OfficeRepository** class จากนั้นก็ **request.getParameter()** เพื่อนำไปเรียกใช้งาน ใน **jsp** ส่วนของ **form** เพื่อที่จะ **Update** ข้อมูล จากนั้นก็จะทำการเช็คเงื่อนไขว่า **attribute** ทุกตัวจะต้องไม่เป็น **null** และเป็นค่าว่าง โดยการเรียกใช้ **method inputNotEmpty()** ที่ได้สร้างไว้ข้างล่าง

ส่วนสุดท้ายจะเป็นการเช็คดูว่าเงื่อนไขไหนตรงกับ **status** ที่ได้เขียนไว้ หลังจากนั้นก็จะส่งข้อมูลกลับไปผ่านทาง **jsp** แล้วเรียกดูผลลัพธ์ที่ **update** ผ่าน **method findAll()**

## 5.OfficeDeleteServlet

```

@WebServlet(name = "OfficeDeleteServlet", value = "/office-delete")
public class OfficeDeleteServlet extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        OfficeRepository repository = new OfficeRepository();
        String officeCode = request.getParameter("officeCode");
        if (officeCode != null) {
            boolean statusDelete = repository.delete(officeCode);
            if (statusDelete) {
                request.setAttribute("messageDelete", "Delete Successfully");
                request.setAttribute("StatusDelete", true);
            } else {
                request.setAttribute("messageDelete", "Delete not Successful");
                request.setAttribute("StatusDelete", false);
            }
        }
        request.setAttribute("offices", repository.findAll());
        request.getRequestDispatcher("/office_list.jsp").forward(request, response);
    }
}

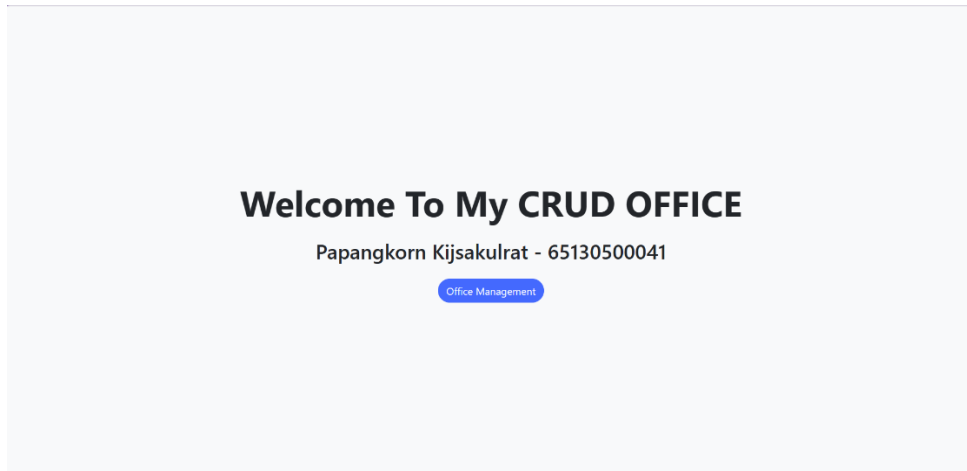
```

จะเป็น controller ที่ควบคุมในการ delete ข้อมูล การทำงานเริ่มจาก new object OfficeRepository จากนั้นก็จะ request getParameter() แล้วก็จะเข้าเงื่อนไข เพื่อเรียกใช้ method delete() จาก OfficeRepository จากนั้นเข้าเงื่อนไข เช็คค่าลบสำเร็จมั้ย ถ้าสำเร็จก็จะโชว์ status ที่ได้ตั้งไว้ แล้วทำการเรียกใช้ method findAll() เพื่อแสดงผลลัพธ์จากการ delete แล้วก็ส่งหน้า office\_list.jsp เพื่อแสดงในส่วน of view

## Webapp/jsp/view

jsp ทุกหน้าตกแต่งด้วย bootstrap

จะเริ่มเข้ามาในหน้า index



```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html lang="en">
<head>
  <title>OFFICE_CRUD_65130500041</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body class="d-flex align-items-center justify-content-center min-vh-100 bg-light">
  <div class="text-center">
    <h1 class="display-4 fw-bold mb-4">Welcome To My CRUD OFFICE</h1>
    <h2 class="h2 mb-4">Papangkorn Kijsakulrat - 65130500041</h2>
    <a href="office-list" class="btn btn-primary rounded-pill">Office Management</a>
  </div>
</body>
</html>
```

เป็นหน้าต้อนรับโดยมีปุ่มให้กดเข้าไปใน office-list อ้างอิงจากการ urlMapping

Classic Model Offices Management

Office Code	City	Phone	AddressLine1	Country	Postal Code	Territory	
1	San Jose	+1 650 219 4782	100 Market Street	USA	94080	NA	<a href="#">Edit</a>
2	Boston	+1 215 837 0825	1550 Court Place	USA	02107	NA	<a href="#">Edit</a>
3	NYC	+1 212 555 3000	523 East 53rd Street	USA	10022	NA	<a href="#">Edit</a>
4	Paris	+33 14 723 4404	43 Rue Joffroy D'abbans	France	75017	EMEA	<a href="#">Edit</a>
5	Tokyo	+81 33 224 5000	4-1 Kioicho	Japan	102-8578	Japan	<a href="#">Edit</a>
6	Sydney	+61 2 9264 2451	5-11 Wentworth Avenue	Australia	NSW 2010	APAC	<a href="#">Edit</a>
7	London	+44 20 7877 2041	25 Old Broad Street	UK	EC2N 1HN	EMEA	<a href="#">Edit</a>
8	Nonthaburi	+6612343445	156 Ladpraduk	Thailand	11110	APAC	<a href="#">Edit</a>

จากนั้นก็จะเข้ามาในหน้า office\_list.jsp

```

<!--
  Created by IntelliJ IDEA.
  User: butte
  Date: 15/11/2023
  Time: 22:49
  To change this template use File | Settings | File Templates.
-->
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<html>
<head>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet">
  <title>OFFICE_CRUD_65130500041</title>
</head>
<body class="bg-light">
<div class="container mt-5">

  <div class="bg-primary rounded p-4 mb-4">
    <a href="office-list" class="text-decoration-none">
      <h1 class="display-4 mb-10 text-white">Classic Model Offices Management</h1>
    </a>
  </div>

```

```

<!-- Search Function -->
<form action="office-search" class="mb-4" method="POST">
  <div class="input-group">
    <input name="searchByCityOrCountry" type="text" class="form-control"
      placeholder="Search By City or Country">
    <button type="submit" class="btn btn-primary">Search</button>
  </div>
</form>

```

ก็จะเจอกับ search function ที่มีหน้าที่เอาไว้ค้นหาข้อมูลโดย city or country

Search By City or Country

Search

Office Code	City	Phone	AddressLine1	Country	Postal Code	Territory	
-------------	------	-------	--------------	---------	-------------	-----------	--



```

<!-- Table office list -->
<table class="table table-bordered">
  <thead>
    <tr>
      <th>Office Code</th>
      <th>City</th>
      <th>Phone</th>
      <th>AddressLine1</th>
      <th>Country</th>
      <th>Postal Code</th>
      <th>Territory</th>
      <th></th>
    </tr>
  </thead>
  <tbody>

<!-- loop for listing all attribute values -->
<c:forEach items="{offices}" var="office">
  <tr>
    <td>{office.officeCode}</td>
    <td>{office.city}</td>
    <td>{office.phone}</td>
    <td>{office.addressLine1}</td>
    <td>{office.country}</td>
    <td>{office.postalCode}</td>
    <td>{office.territory}</td>
    <!-- edit/update function -->
    <td>
      <a href="office-update?officeCode={office.officeCode}">Edit</a>
    </td>
  </tr>
</c:forEach>
</tbody>
</table>

```

จากนั้นก็จะเป็นในส่วนของ **table** เพื่อทำการโชว์ข้อมูลทั้งหมดของ **office** จะทำการ **loop** โดยการใช้คำสั่ง **forEach()** เพื่อทำการ **loop value** ทั้งหมดออกมา

Office Code	City	Phone	AddressLine1	Country	Postal Code	Territory	
1	San Jose	+1 650 219 4782	100 Market Street	USA	94080	NA	<a href="#">Edit</a>
2	Boston	+1 215 837 0825	1550 Court Place	USA	02107	NA	<a href="#">Edit</a>
3	NYC	+1 212 555 3000	523 East 53rd Street	USA	10022	NA	<a href="#">Edit</a>
4	Paris	+33 14 723 4404	43 Rue Jouffroy D'abbans	France	75017	EMEA	<a href="#">Edit</a>
5	Tokyo	+81 33 224 5000	4-1 Kioicho	Japan	102-8578	Japan	<a href="#">Edit</a>
6	Sydney	+61 2 9264 2451	5-11 Wentworth Avenue	Australia	NSW 2010	APAC	<a href="#">Edit</a>
7	London	+44 20 7877 2041	25 Old Broad Street	UK	EC2N 1HN	EMEA	<a href="#">Edit</a>
8	Nonthaburi	+6612343445	156 Ladpraduk	Thailand	11110	APAC	<a href="#">Edit</a>

```

<!-- Status of Insert,Update,Delete -->
<div class="container mt-4">
  <!--Insert office Status-->
  <c:choose>
    <c:when test="${statusAdd}">
      <h4 class="text-success">${messageAdd}</h4>
    </c:when>
    <c:otherwise>
      <h4 class="text-danger">${messageAdd}</h4>
    </c:otherwise>
  </c:choose>

  <!--Delete Office Status-->
  <c:choose>
    <c:when test="${StatusDelete}">
      <h4 class="text-success">${messageDelete}</h4>
    </c:when>
    <c:otherwise>
      <h4 class="text-danger">${messageDelete}</h4>
    </c:otherwise>
  </c:choose>

  <!--Update Office Status-->
  <c:choose>
    <c:when test="${statusUpdate}">
      <h4 class="text-success">${messageUpdate}</h4>
    </c:when>
    <c:otherwise>
      <h4 class="text-danger">${messageUpdate}</h4>
    </c:otherwise>
  </c:choose>
</div>

```

จากนั้นจะเป็นการเขียน choos when เพื่อเช็คเงื่อนไข status ที่อยู่ใน servlet

insert, update, delete

Add Successfully

Update Successfully

Delete Successfully

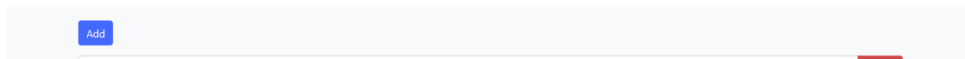
```

<!--Button fuction-->
<div class="container mt-4">
  <!--      Insert button      -->
  <a href="office-add" class="btn btn-primary mr-2">Add</a>

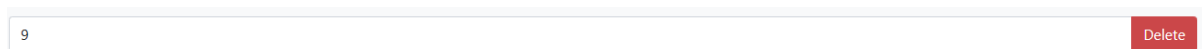
  <!--      Delete button      -->
  <form action="office-delete" method="GET" class="mt-3">
    <div class="input-group mb-3">
      <input name="officeCode" type="text" class="form-control" placeholder="Enter Office Code to Delete"
        aria-label="Enter office Code" aria-describedby="button-addon2">
      <button class="btn btn-danger" type="submit" id="button-addon2">Delete</button>
    </div>
  </form>
</div>

```

จากนั้นก็จะเป็นปุ่มต่างๆ อันแรกเป็นปุ่ม Add ใช้เพื่อ insert ข้อมูลใหม่ แล้วก็ urlMapping ไปที่ office-add



ในส่วนของปุ่ม delete จะทำการสร้างเป็น form เพื่อกด officeCode เพื่อทำการลบข้อมูล



จากนั้นก็จะเป็นหน้า add ข้อมูล



```

<!--
  Created by IntelliJ IDEA.
  User: butte
  Date: 15/11/2023
  Time: 23:49
  To change this template use File | Settings | File Templates.
-->
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<html>
<head>
  <title>OFFICE_CRUD_65130500041</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body class="container mt-5">

<div class="bg-primary rounded p-4 mb-4">
  <a href="office-list" class="text-decoration-none">
    <h1 class="display-4 mb-10 text-white">Office Management</h1>
  </a>
</div>

```

```

<!-- Form to receive information to insert into attributes -->
<div class="mt-4">
  <h3>Add Office Management</h3>
  <form action="office-add" method="POST">
    <div class="mb-3">
      <label for="officeCode" class="form-label">Office Code:</label>
      <input type="text" class="form-control" id="officeCode" name="officeCode">
    </div>
    <div class="mb-3">
      <label for="city" class="form-label">City:</label>
      <input type="text" class="form-control" id="city" name="city">
    </div>
    <div class="mb-3">
      <label for="phone" class="form-label">Phone:</label>
      <input type="text" class="form-control" id="phone" name="phone">
    </div>
    <div class="mb-3">
      <label for="addressLine1" class="form-label">Address Line 1:</label>
      <input type="text" class="form-control" id="addressLine1" name="addressLine1">
    </div>
    <div class="mb-3">
      <label for="country" class="form-label">Country:</label>
      <input type="text" class="form-control" id="country" name="country">
    </div>
    <div class="mb-3">
      <label for="postalCode" class="form-label">Postal Code:</label>
      <input type="text" class="form-control" id="postalCode" name="postalCode">
    </div>
    <div class="mb-3">
      <label for="territory" class="form-label">Territory:</label>
      <input type="text" class="form-control" id="territory" name="territory">
    </div>

    <button type="submit" class="btn btn-primary">Add</button>
  </form>
</div>
</body>
</html>

```

ทำการสร้าง form เพื่อรับข้อมูล ผ่าน method post และ action ไปยัง office-add หรือ OfficeAddServlet

ในส่วนสุดท้ายคือหน้า edit หรือ update

## Office Management

### Update Office Management

Office Code:

8

City:

Rayong

Phone:

+6612343445

Address Line 1:

156 aaa bbb

Country:

Thailand

Postal Code:

112342

Territory:

APAC

Update

Reset

```
<!--
  Created by IntelliJ IDEA.
  User: butte
  Date: 15/11/2023
  Time: 23:50
  To change this template use File | Settings | File Templates.
-->
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<html>
<head>
  <title>OFFICE_CRUD_65130500041</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body class="container mt-5">

  <div class="bg-primary rounded p-4 mb-4">
    <a href="office-list" class="text-decoration-none">
      <h1 class="display-4 mb-10 text-white">Office Management</h1>
    </a>
  </div>
```

```

22 <!-- Form to receive information to update into attributes -->
23 <div class="mt-4">
24   <h3>Update Office Management</h3>
25   <form action="office-update?officeCode=${office.officeCode}" method="POST">
26     <div class="mb-3">
27       <label for="officeCode" class="form-label">Office Code:</label>
28       <!-- office code for read-only -->
29       <input type="text" class="form-control" id="officeCode" name="officeCode" value="${office.officeCode}" readonly>
30     </div>
31     <div class="mb-3">
32       <label for="city" class="form-label">City:</label>
33       <input type="text" id="city" class="form-control" name="city" value="${office.city}">
34     </div>
35     <div class="mb-3">
36       <label for="phone" class="form-label">Phone:</label>
37       <input type="text" id="phone" class="form-control" name="phone" value="${office.phone}">
38     </div>
39     <div class="mb-3">
40       <label for="addressLine1" class="form-label">Address Line 1:</label>
41       <input type="text" id="addressLine1" class="form-control" name="addressLine1" value="${office.addressLine1}">
42     </div>
43     <div class="mb-3">
44       <label for="country" class="form-label">Country:</label>
45       <input type="text" id="country" class="form-control" name="country" value="${office.country}">
46     </div>
47     <div class="mb-3">
48       <label for="postalCode" class="form-label">Postal Code:</label>
49       <input type="text" id="postalCode" class="form-control" name="postalCode" value="${office.postalCode}">
50     </div>
51     <div class="mb-3">
52       <label for="territory" class="form-label">Territory:</label>
53       <input type="text" id="territory" class="form-control" name="territory" value="${office.territory}">
54     </div>
55     <button type="submit" class="btn btn-primary">Update</button>
56     <button type="reset" class="btn btn-secondary">Reset</button>
57   </form>
58 </div>
59 </div>
60 </body>
61 </html>

```

html > body.container.mt-5 > div.mt-4

ในส่วนนี้ได้จะคล้ายกับการ jsp ของ update จะทำการสร้าง form เพื่อรับข้อมูลแล้วนำข้อมูลมาแสดงโดยอ้างอิงจาก officeCode