

1.Insert

Controller

```
@WebServlet(name = "OfficeInsertServlet", value = "/insert-office")
public class OfficeInsertServlet extends HttpServlet {
    2 usages
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        getServletContext().getRequestDispatcher("/office_insert.jsp").forward(request, response);
    }
}
```

doGet

getServletContext().getRequestDispatcher("/office_insert.jsp").forward(request, response);

ส่งต่อไปที่ office_list.jsp (office-list)

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

    if (officeCode.isEmpty() || city.isEmpty() || phone.isEmpty() || addressLine1.isEmpty() || country.isEmpty() || postalCode.isEmpty() || territory.isEmpty()) {
        request.setAttribute(Ⓢ "errorMessage", Ⓢ "!! Insert failed !! ");
        doGet(request, response);
    }

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

    OfficeRepository officeRepository = new OfficeRepository();

    if (officeRepository.insert(office)) {
        request.setAttribute(Ⓢ "successMessage", Ⓢ "!! Insert successfully !!");
    } else {
        request.setAttribute(Ⓢ "errorMessage", Ⓢ "!! Insert failed !! ");
    }
    officeRepository.close();
    doGet(request, response);
}
```

doPost

สร้าง Object **"Office"** จาก Office และดึงค่าของพารามิเตอร์ officeCode , city , phone , addressLine1 , country , postalCode , territory ที่ User ส่งมาเก็บไว้ใน String

ทำการเช็คเงื่อนไขถ้าพารามิเตอร์ไม่มีค่าส่งมา ให้ set attribute **"errorMessage"** ข้อความแสดง **"!! Insert failed !! "** ใน request จากนั้น set ค่าที่รับมาจากพารามิเตอร์ลง **"Office"**

ทำการ instance OfficeRepository เพื่อใช้คำสั่งในการ insert ข้อมูลในฐานข้อมูล และเช็คเงื่อนไข officeRepository.insert (office) จะถูกเรียกเพื่อเพิ่มข้อมูล office แล้ว set attribute **"successMessage"** ข้อความแสดง **"!! Insert successfully !! "** ใน request ถ้าไม่ตรงเงื่อนไข จะ set attribute ข้อความแสดง **"!! Insert failed !! "** และส่งต่อไปที่ doGet

```

</div>
<div class="row">
  <a href="insert-office">Insert</a>
</div>

```

View

Insert : ปุ่ม link ไปที่ office_list.jsp (office-list) และไปที่ method doGet

```

<form method="post" action="insert-office">
  <div class="form-group">
    <label for="officeCode">officeCode:</label>
    <input type="text" class="form-control" id="officeCode" name="officeCode">
  </div>
  <div class="form-group">
    <label for="city">City:</label>
    <input type="text" class="form-control" id="city" name="city">
  </div>
  <div class="form-group">
    <label for="phone">Phone:</label>
    <input type="text" class="form-control" id="phone" name="phone">
  </div>
  <div class="form-group">
    <label for="addressLine1">addressLine1:</label>
    <input type="text" class="form-control" id="addressLine1" name="addressLine1">
  </div>
  <div class="form-group">
    <label for="country">country:</label>
    <input type="text" class="form-control" id="country" name="country">
  </div>
  <div class="form-group">
    <label for="city">postalCode:</label>
    <input type="text" class="form-control" id="postalCode" name="postalCode">
  </div>
  <div class="form-group">
    <label for="territory">territory:</label>
    <input type="text" class="form-control" id="territory" name="territory">
  </div>
  <input type="submit" class="btn">
  <a href="office-list">Back</a>
</form>

```

```

<div class="form-group">
  <label for="city">postalCode:</label>
  <input type="text" class="form-control" id="postalCode" name="postalCode">
</div>
<div class="form-group">
  <label for="territory">territory:</label>
  <input type="text" class="form-control" id="territory" name="territory">
</div>
<input type="submit" class="btn">
<a href="office-list">Back</a>
</form>

</div>

<c:if test="${successMessage != null}">
  <h5 style="...">
    ${successMessage}
  </h5>
</c:if>

<c:if test="${errorMessage != null}">
  <h5 style="...">
    ${errorMessage}
  </h5>
</c:if>

```

ทำ form รับค่าที่ User กรอกเข้ามาเพื่อทำการ Insert ข้อมูล

ใช้ JSTL ในการเช็คเงื่อนไข ถ้า “successMessage” ไม่เท่ากับ null จะแสดงข้อความ “!! Insert successfully !!” และถ้า “errorMessage” ไม่เท่ากับ null จะแสดงข้อความ “!! Insert failed!!”

ผลลัพธ์

Insert

officeCode: 100
 City: JASD
 Phone: 09123456
 addressLine1: ASE
 country: USA
 postalCode: 12300
 territory: sdwds33

Submit Back

!! insert successfully !!



Classic Model Offices

Office Code	City	Country	Phone	update	delete
1	San Francisco	Thail	+1 650 219 4782	update	delete
100	JASD	USA	09123456	update	delete
2	Boston	USA	+1 215 887 1825	update	delete
3	NYC	USA	+1 212 555 3000	update	delete
4	Paris	France	+33 14 723 4404	update	delete
5	Tokyo	Japan	+81 33 234 5000	update	delete
6	Sydney	Australia	+61 2 9264 2401	update	delete
7	London	UK	+44 20 7897 2041	update	delete
8	Bangkok	Thailand	+61 33 234 5000	update	delete

Insert

Insert

officeCode:
 City:
 Phone:
 addressLine1:
 country:
 postalCode:
 territory:

Submit Back

!! Insert failed !!

2.Delete

Controller

```
1 package sit.int202.classicmodels.controllers;
2
3 import ...
4
5 @WebServlet(name = "OfficeDeleteServlet", value = "/office-delete")
6 public class OfficeDeleteServlet extends HttpServlet {
7     1 usage
8     @Override
9     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
10     String officeCode = request.getParameter("officeCode");
11     OfficeRepository officeRepository = new OfficeRepository();
12     if (officeRepository.delete(officeCode)){
13         request.setAttribute("successMessage", "Delete successfully!!");
14     }else {
15         request.setAttribute("errorMessage", "Delete failed!!");
16     }
17     request.getRequestDispatcher("office-list").forward(request, response);
18 }
19
20 no usages
21 @Override
22 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException
23 }
```

doGet

ดึงค่าของพารามิเตอร์ officeCode จากคำขอ HTTP ที่ User กรอกเข้ามา จากนั้นทำการ instance OfficeRepository เพื่อใช้คำสั่งในการ delete ข้อมูลในฐานข้อมูล ซึ่งการ delete จะใช้ officeCode ในการไปลบข้อมูล ส่วนการเช็คเงื่อนไข officeRepository.delete(officeCode) จะถูกเรียกเพื่อลบ office โดยใช้รหัส officeCode ที่ได้มาแล้ว แล้ว set attribute “successMessage” ข้อความแสดง “Delete successfully!!” ใน request ถ้าไม่ตรงเงื่อนไข จะ set attribute “errorMessage” ข้อความแสดง “Delete failed!!”

```
request.getRequestDispatcher("office-list").forward(request, response);
```

จากนั้น ส่งต่อไปที่ office_list.jsp (office-list)

View

```
<td><a href="office-delete?officeCode=${office.officeCode}">delete</a></td>
```

delete : ทำ hyperlink ในการลบ office และจะส่ง officeCode ไปที่ servlet doGet

```
<c:if test="${successMessage != null}">
    <h5 style="...">
        ${successMessage}
    </h5>
</c:if>
<c:if test="${errorMessage != null}">
    <h5 style="...">
        ${errorMessage}
    </h5>
</c:if>
```

ใช้ JSTL ในการเช็คเงื่อนไข ถ้า “successMessage” ไม่เท่ากับ null จะแสดงข้อความ “Delete successfully!!” และถ้า “errorMessage” ไม่เท่ากับ null จะแสดงข้อความ “Delete failed!!”

ผลลัพธ์

Office Code	City	Country	Phone	update	delete
1	San Francisco	Texas	+1 800 219 4782	update	delete
100	DIST	USA	0902386	update	delete
2	Boston	USA	+1 201 837 0825	update	delete
3	NYC	USA	+1 212 555 3000	update	delete
4	Paris	France	+33 14 723 4404	update	delete
5	Tokyo	Japan	+81 33 224 5000	update	delete
6	Sydney	Australia	+61 2 9264 2401	update	delete
7	London	UK	+44 20 7897 2041	update	delete
8	Bangkok	Thailand	+66 33 224 5000	update	delete



Office Code	City	Country	Phone	update	delete
1	San Francisco	Texas	+1 800 219 4782	update	delete
2	Boston	USA	+1 201 837 0825	update	delete
3	NYC	USA	+1 212 555 3000	update	delete
4	Paris	France	+33 14 723 4404	update	delete
5	Tokyo	Japan	+81 33 224 5000	update	delete
6	Sydney	Australia	+61 2 9264 2401	update	delete
7	London	UK	+44 20 7897 2041	update	delete
8	Bangkok	Thailand	+66 33 224 5000	update	delete

Office Code	City	Country	Phone	update	delete
1	San Francisco	Texas	+1 800 219 4782	update	delete
100	DIST	USA	0902386	update	delete
2	Boston	USA	+1 201 837 0825	update	delete
3	NYC	USA	+1 212 555 3000	update	delete
4	Paris	France	+33 14 723 4404	update	delete
5	Tokyo	Japan	+81 33 224 5000	update	delete
6	Sydney	Australia	+61 2 9264 2401	update	delete
7	London	UK	+44 20 7897 2041	update	delete
8	Bangkok	Thailand	+66 33 224 5000	update	delete

3.Update

Controller

```
package sit.int202.classicmodels.controllers;

import ...

@WebServlet(name = "OfficeUpdateServlet", value = "/office-update")
public class OfficeUpdateServlet extends HttpServlet {
    1 usage
    @Override
    2 protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        String officeCode = request.getParameter( "officeCode");
        OfficeRepository officeRepository = new OfficeRepository();
        request.setAttribute( "selected", officeRepository.find(officeCode));
        3 getServletContext().getRequestDispatcher( "office_update.jsp").forward(request, response);
    }
}
```

doGet

ดึงค่าของพารามิเตอร์ officeCode จากคำขอ HTTP ที่ User กรอกเข้ามา จากนั้นทำการ instance OfficeRepository เพื่อใช้คำสั่งในการ find ข้อมูลในฐานข้อมูล แล้วนำไป set attribute ว่า "selected" จากนั้น ส่งต่อไปที่ office_update.jsp (office-update)

```
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    Office newOffice = new Office();
    String officeCode = request.getParameter( "officeCode");
    String city = request.getParameter( "city");
    String phone = request.getParameter( "phone");
    String addressLine1 = request.getParameter( "addressLine1");
    String addressLine2 = request.getParameter( "addressLine2");
    String state = request.getParameter( "state");
    String country = request.getParameter( "country");
    String postalCode = request.getParameter( "postalCode");
    String territory = request.getParameter( "territory");

    newOffice.setOfficeCode(officeCode);
    newOffice.setCity(city);
    newOffice.setPhone(phone);
    newOffice.setAddressLine1(addressLine1);
    newOffice.setAddressLine2(addressLine2);
    newOffice.setState(state);
    newOffice.setCountry(country);
    newOffice.setPostalCode(postalCode);
    newOffice.setTerritory(territory);

    OfficeRepository officeRepository = new OfficeRepository();
    officeRepository.update(newOffice);

    getServletContext().getRequestDispatcher( "office_update.jsp").forward(request, response);
}
```

doPost

สร้าง Object "newOffice" จาก Office และดึงค่าของพารามิเตอร์ officeCode , city , phone , addressLine1 , addressLine2 , state , country , postalCode , territory ที่ User ส่งมาเก็บไว้ใน String แล้ว set ค่าที่รับมาจากพารามิเตอร์ของ "newOffice" จากนั้นทำการ instance OfficeRepository เพื่อใช้คำสั่งในการ update ข้อมูลในฐานข้อมูล และส่งต่อไปที่ office_update.jsp (office-update)

View

```
<div class="form-group">
  <div class="row1">
    <h2>Update</h2>
  </div>
  <div class="row">
    <form method="post" action="office-update">
      <div class="form-group">
        <label for="city">City:</label>
        <input type="hidden" name="officeCode" value="${selected.officeCode}">
        <input type="text" class="form-control" id="city" name="city" value="${selected.city}">
      </div>
      <div class="form-group">
        <label for="phone">Phone:</label>
        <input type="text" class="form-control" id="phone" name="phone" value="${selected.phone}">
      </div>
      <div class="form-group">
        <label for="addressLine1">addressLine1:</label>
        <input type="text" class="form-control" id="addressLine1" name="addressLine1" value="${selected.addressLine1}">
      </div>
      <div class="form-group">
        <label for="addressLine2">addressLine2:</label>
        <input type="text" class="form-control" id="addressLine2" name="addressLine2" value="${selected.addressLine2}">
      </div>
      <div class="form-group">
        <label for="state">state:</label>
        <input type="text" class="form-control" id="state" name="state" value="${selected.state}">
      </div>
      <div class="form-group">
        <label for="country">country:</label>
        <input type="text" class="form-control" id="country" name="country" value="${selected.country}">
      </div>
      <div class="form-group">
        <label for="postalCode">postalCode:</label>
        <input type="text" class="form-control" id="postalCode" name="postalCode" value="${selected.postalCode}">
      </div>
      <div class="form-group">
        <label for="territory">territory:</label>
        <input type="text" class="form-control" id="territory" name="territory" value="${selected.territory}">
      </div>
      <input type="submit" class="btn">
      <a href="office-list">Back</a>
    </form>
  </div>
</div>
</body>
```

ทำ form เพื่อรับค่าพารามิเตอร์ที่ User กรอกเข้ามาเพื่อจะทำการเปลี่ยนแปลงข้อมูล

`<input type="hidden" name="officeCode" value="${selected.officeCode}">` : ดึงค่าของ **"officeCode"** จาก attribute **"selected"** ใช้เพื่อให้เห็นข้อมูลเดิมก่อนจะทำการเปลี่ยนแปลงข้อมูล (type hidden ซ่อนป้อนข้อมูลจะไม่ขึ้นหน้าเว็บแต่ยังคงเป็นส่วนหนึ่งของแบบฟอร์มและสามารถส่งพร้อมกับข้อมูลแบบฟอร์มอื่นๆ)

`<td>update</td>` : ทำ hyperlink ในการ update office และจะส่ง officeCode ไปที่ servlet doGet

ผลลัพธ์

Classic Model Offices

Office Code	City	Country	Phone		
1	San Francisco	THAI	+1 650 219 4782	<input type="button" value="update"/>	<input type="button" value="delete"/>
2	Boston	USA	+1 215 837 0825	<input type="button" value="update"/>	<input type="button" value="delete"/>
3	NYC	USA	+1 212 555 3000	<input type="button" value="update"/>	<input type="button" value="delete"/>
4	Paris	France	+33 14 723 4404	<input type="button" value="update"/>	<input type="button" value="delete"/>
5	Tokyo	Japan	+81 33 224 5000	<input type="button" value="update"/>	<input type="button" value="delete"/>
6	Sydney	Australia	+61 2 9264 2451	<input type="button" value="update"/>	<input type="button" value="delete"/>
7	London	UK	+44 20 7877 2041	<input type="button" value="update"/>	<input type="button" value="delete"/>
8	BangMod	Thailand	+81 33 224 5000	<input type="button" value="update"/>	<input type="button" value="delete"/>

City:

BangLumPho

Phone:

+81 33 224 5000

addressLine1:

4-1 Kioicho

addressLine2:

state:

Chiyoda-Ku

country:

Thailand

postalCode:

102-6578

territory:

Japan

Classic Model Offices

Office Code	City	Country	Phone		
1	San Francisco	THAI	+1 650 219 4782	<input type="button" value="update"/>	<input type="button" value="delete"/>
2	Boston	USA	+1 215 837 0825	<input type="button" value="update"/>	<input type="button" value="delete"/>
3	NYC	USA	+1 212 555 3000	<input type="button" value="update"/>	<input type="button" value="delete"/>
4	Paris	France	+33 14 723 4404	<input type="button" value="update"/>	<input type="button" value="delete"/>
5	Tokyo	Japan	+81 33 224 5000	<input type="button" value="update"/>	<input type="button" value="delete"/>
6	Sydney	Australia	+61 2 9264 2451	<input type="button" value="update"/>	<input type="button" value="delete"/>
7	London	UK	+44 20 7877 2041	<input type="button" value="update"/>	<input type="button" value="delete"/>
8	BangLumPho	Thailand	+81 33 224 5000	<input type="button" value="update"/>	<input type="button" value="delete"/>

4. ค้นหาตาม city , country

```

no usages
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    OfficeRepository officeRepository = new OfficeRepository();
    String find = request.getParameter("find");
    request.setAttribute("offices", officeRepository.findByCityOrCountry(find));

    getServletContext().getRequestDispatcher("/office_list.jsp").forward(request, response);
}

```

Controller

doPost

ทำการ instance OfficeRepository เพื่อใช้คำสั่งในการ find ข้อมูลในฐานข้อมูล แล้วนำไป set attribute ว่า "offices" และดึงค่าของพารามิเตอร์ที่ User ต้องการหาลงใน "find" จากนั้นส่งต่อไปที่ office_list.jsp (office-list)

View

```

<div class="form-group">
<div class="row1">
<h2>Classic Model Offices</h2>
<span>
<form action="office-search" method="post">
<input type="text" name="find">
<input type="submit" value="search" class="btn">
</form>
</span>
</div>

```

ทำ form รับค่าพารามิเตอร์ที่ User กรอกเข้ามา เพื่อทำการ find city หรือ contry กด search จะส่งไปที่ doPost servlet

Classic Model Offices

Office Code	City	Country	Phone		
1	San Francisco	THAI	+1 650 219 4782	<input type="button" value="update"/>	<input type="button" value="delete"/>
100	JAST	USA	091238ue	<input type="button" value="update"/>	<input type="button" value="delete"/>
2	Boston	USA	+1 215 837 0825	<input type="button" value="update"/>	<input type="button" value="delete"/>
3	NYC	USA	+1 212 555 3000	<input type="button" value="update"/>	<input type="button" value="delete"/>
4	Paris	France	+33 14 723 4404	<input type="button" value="update"/>	<input type="button" value="delete"/>
5	Tokyo	Japan	+81 33 224 5000	<input type="button" value="update"/>	<input type="button" value="delete"/>
6	Sydney	Australia	+61 2 9264 2451	<input type="button" value="update"/>	<input type="button" value="delete"/>
7	London	UK	+44 20 7877 2041	<input type="button" value="update"/>	<input type="button" value="delete"/>
8	BangMod	Thailand	+81 33 224 5000	<input type="button" value="update"/>	<input type="button" value="delete"/>

Classic Model Offices

Office Code	City	Country	Phone		
100	JAST	USA	091238ue	<input type="button" value="update"/>	<input type="button" value="delete"/>
2	Boston	USA	+1 215 837 0825	<input type="button" value="update"/>	<input type="button" value="delete"/>
3	NYC	USA	+1 212 555 3000	<input type="button" value="update"/>	<input type="button" value="delete"/>

ผลลัพธ์