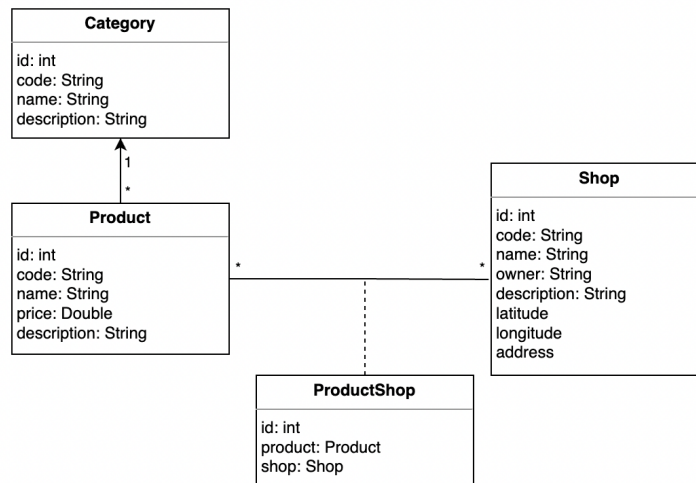


## LAB-09-2: Spring Framework MVC + Hibernate

ความสัมพันธ์ระหว่างคลาส

1. ให้นักศึกษาใช้โปรเจกต์เดิมจากปฏิบัติการก่อนมาทำต่อ (Shop)
2. แก้ไขความสัมพันธ์ระหว่างคลาส



- 2.1) สร้าง Category entity model ใน package “org.itsci.shop.model” โดยมีรายละเอียดดังต่อไปนี้

```

@Entity
@Table(name="categories")
public class Category {
    @Id
    @Column(name = "id", nullable = false)
    @GeneratedValue(strategy= GenerationType.IDENTITY)
    @GenericGenerator(name = "increment", strategy = "increment")
    int id;

    @Column(name = "code", nullable = false)
    private String code;

    @Column(name = "name", nullable = false)
    private String name;

    @Column(name = "description", columnDefinition="TEXT")
    private String description;
    // getter/setter
}
  
```

## 2.2) เพิ่มความสัมพันธ์ระหว่างคลาสให้กับ Product

```

@Entity
@Table(name = "products")
public class Product {
    ...
    @NotNull
    @ManyToOne
    @JoinColumn(name = "category_id")
    private Category category;

    @ManyToMany(fetch = FetchType.EAGER, cascade = {CascadeType.ALL})
    @JoinTable(name = "product_shop",
        joinColumns= { @JoinColumn(name = "product_id") },
        inverseJoinColumns= { @JoinColumn(name = "shop_id") })
    private List<Shop> shops;

    // getter/setter
}

```

## 2.3) เพิ่มความสัมพันธ์ระหว่างคลาสให้กับ Shop

```

@ManyToMany(fetch = FetchType.EAGER, cascade = CascadeType.ALL, mappedBy =
"shops")
private List<Product> products;

```

## 3. การจัดการกับความสัมพันธ์ระหว่าง Product และ Category

### 3.1) เพิ่มข้อมูลให้กับ categories ในฐานข้อมูลด้วยคำสั่ง SQL ต่อไปนี้

```

insert into `categories` (`id`, `code`, `name`, `description`)
values
    (1, 'CT001', 'PHP', 'PHP category'),
    (2, 'CT002', 'JavaScript', 'JavaScript category'),
    (3, 'CT003', 'Python', 'Python category');

```

### 3.2) คัดลอกไฟล์ CategoryDao.java, CategoryDaoImpl.java, CategoryService.java, และ CategoryServiceImpl.java ไปไว้ใน package “org.itsci.shop.dao” และ “org.itsci.shop.service” ตามลำดับ

### 3.3) แก้ไขคลาส ProductController เพิ่มโค้ดดึงค่า categories ดังแสดงต่อไปนี้

```

...
@Autowired
private CategoryService categoryService;
...
@GetMapping("/create")
public String showFormForAdd(Model model) {
    model.addAttribute("title", "เพิ่ม" + title);
    model.addAttribute("categories", categoryService.getCategories());
    model.addAttribute("product", new Product());
    return "product/product-form";
}

@GetMapping("/{id}/update")
public String showFormForUpdate(@PathVariable("id") int id, Model model) {
    Product product = productService.getProduct(Integer.valueOf(id));
    model.addAttribute("title", "แก้ไข" + title);
}

```

---

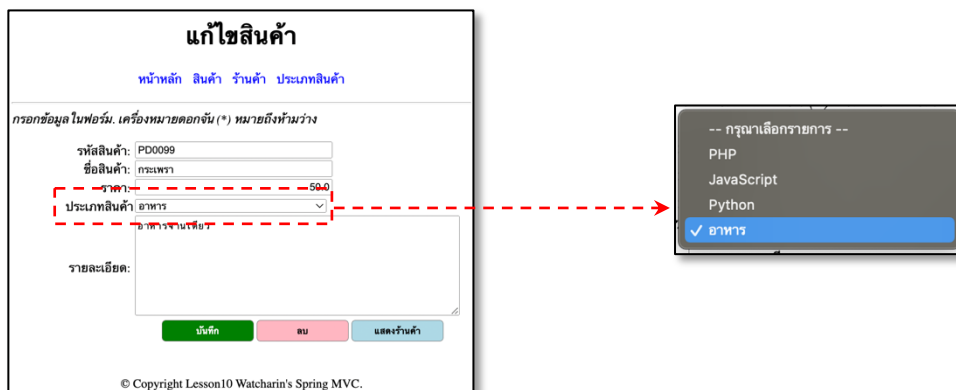
```
model.addAttribute("categories", categoryService.getCategories());  
model.addAttribute("product", product);  
return "product/product-form";  
}
```

## 3.4) แก้ไขไฟล์ JSP “product-form.jsp” เพิ่ม select list ของประเภทสินค้าดังแสดงต่อไปนี้

```

...
<tr>
  <td><label>ราคา:</label></td>
  <td>
    <form:input path="price" cssClass="number"/>
    <form:errors path="price" cssClass="error"/>
  </td>
</tr>
<tr>
  <td><label>ประเภทสินค้า</label></td>
  <td>
    <form:select path="category.code">
      <form:option value="" label="-- กรุณาเลือกรายการ --"/>
      <form:options items="${categories}" itemLabel="name"
itemValue="code"/>
    </form:select>
    <form:errors path="category.code" cssClass="error"/>
  </td>
</tr>
...

```



## 3.5) แก้ไขเมธอด “saveProduct” ใน ProductController ดังแสดงต่อไปนี้

```

@RequestMapping(path="/save", method = RequestMethod.POST)
public String saveProduct(@Valid @ModelAttribute("product") Product product,
                          BindingResult bindingResult,
                          Model model) {
    if (bindingResult.hasErrors()) {
        model.addAttribute("title", "มีข้อผิดพลาดในการบันทึก" + title);
        model.addAttribute("categories", categoryService.getCategories());
        return "product/product-form";
    }
    else {
        Product dbProduct = productService.getProduct(product.getId());
        if (dbProduct != null) {
            productService.updateProduct(dbProduct, product);
        } else {
            productService.saveProduct(product);
        }
        return "redirect:/product/list";
    }
}

```

## 3.6) เพิ่มเมธอด “fill()” ใน Product ดังแสดงต่อไปนี้

```
public void fill(Product product) {
    this.code = product.getCode();
    this.name = product.getName();
    this.category = product.getCategory();
    this.description = product.getDescription();
}
```

## 3.7) แก้ไขเมธอด “updateProduct()” ใน ProductServiceImpl ดังแสดงต่อไปนี้

```
@Override
@Transactional
public void updateProduct(Product productEntity, Product product) {
    productEntity.fill(product);
    saveProduct(productEntity);
}
```

## 3.8) แก้ไขเมธอด “saveProduct()” ใน ProductServiceImpl ดังแสดงต่อไปนี้

```
@Service
public class ProductServiceImpl implements ProductService {
    ...
    @Override
    @Transactional
    public void saveProduct(Product product) {
        Category category =
            categoryDao.getCategoryByCode(product.getCategory().getCode());
        product.setCategory(category);
        productDao.saveProduct(productEntity);
    }
}
```

## 4. การจัดการกับความสัมพันธ์ระหว่าง Product และ Shop

- 4.1) แสดงความสัมพันธ์ของ Product และ Shop โดยแสดงจำนวนร้านค้า (Shop) ที่สินค้า (Product) วางจำหน่าย  
 ดังแสดงในรูปต่อไปนี้

รายการสินค้า			
หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า			
เพิ่มสินค้า			
รหัสสินค้า	ชื่อสินค้า	ราคา	จำนวนร้านค้า
PD0001	Programming PHP	345.00	3
PD0002	JavaScript: The Definitive Guide	250.00	2
PD0003	Learning PHP, MySQL & JavaScript	450.00	3
PD0004	Python Crash Course, 2nd Edition	560.00	2
PD0009	มาฆ่า	7.00	1
PD1233	ทดสอบ	5.00	2
PD0099	กระเพรา	50.00	0

© Copyright Lesson10 Watcharin's Spring MVC.

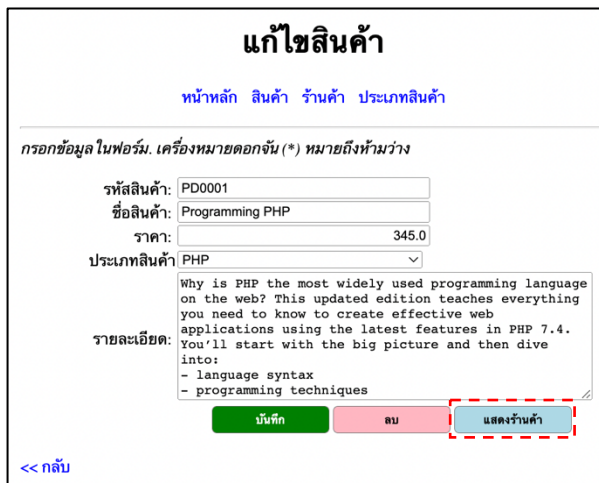
- 4.2) แก้ไขไฟล์ “WEB-INF/view/product/list.jsp” โดยเพิ่มส่วนของโค้ดที่เป็นสีแดง ดังแสดงต่อไปนี้

```

...
<%@ taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions"
%>
<!DOCTYPE html>
...
<table class="table-bordered">
  <thead>
    <tr>
      <th>รหัสสินค้า</th><th>ชื่อสินค้า</th><th>ราคา</th><th>จำนวนร้านค้า</th></tr>
    </thead>
    <tbody>
      <c:forEach var="product" items="{products}">
        <tr>
          ...
          <td class="center">${fn:length(product.shops)}</td>
        </tr>
      </c:forEach>
    </tbody>
  </table>
...

```

- 4.3) เพิ่มร้านค้าที่สินค้าวางขาย โดยเพิ่มปุ่ม “แสดงร้านค้า” ดังแสดงในรูปต่อไปนี้ (นักศึกษาลบปุ่ม “ยกเลิก” ออกเราจะทำเพิ่มในภายหลัง)



เพิ่มโค้ดปุ่มที่เป็นสีแดงในไฟล์ “WEB-INF/view/product/product-form.jsp” ดังแสดงต่อไปนี้

```

...
<tr>
  <td><label></label></td>
  <td class="center">
    <input type="submit" value="บันทึก" class="save-button"/>
    ...
    <c:if test="${product.id > 0}">
      <input type="button" value="แสดงร้านค้า"
        onclick="window.location.href=
        '${pageContext.request.contextPath}/product/${product.id}/view-shops';
      return false;"
      class="add-button"
    </c:if>
  </td>
</tr>
...

```

ทำให้ไม่แสดงปุ่มตอนสร้าง  
Product ใหม่  
“product.id” จะมีค่า  
เป็น 0

`${pageContext.request.contextPath}/product/${product.id}/view-shops`

- 4.4) เพิ่มเมธอด “productViewShop()” ในคลาส ProductController

```

@Controller
@RequestMapping("/product")
public class ProductController {
  ...
  @GetMapping("/{id}/view-shops")
  public String productViewShop(@PathVariable("id") int id, Model model) {
    Product product = productService.getProduct(id);
    model.addAttribute("title", title + " - รายการร้านค้า");
    model.addAttribute("product", product);
    model.addAttribute("shops", product.getShops());
    return "product/product-view-shops";
  }
  ...
}

```

## 4.5) สร้างหน้าเว็บ “WEB-INF/view/product/product-view-shops.jsp” ดังแสดงต่อไปนี้

สินค้า - รายการร้านค้า		
หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า		
รหัสร้านค้า	ชื่อร้านค้า	ชื่อเจ้าของ
SH101	Adam Shop	Adam
SH111	ทดสอบ	กชค
SH999	ddd	ssf

© Copyright Lesson10 Watcharin's Spring MVC.

```

<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<title>${title}</title>
<link href="${pageContext.request.contextPath}/assets/css/style.css"
rel="stylesheet">
</head>
<body>
<div id="header"><h1>${title}</h1></div>
<div class="container">
<jsp:include page="/WEB-INF/view/layouts/nav.jsp"/>
<table class="table-bordered">
<thead>
<tr>
<th>รหัสร้านค้า</th><th>ชื่อร้านค้า</th><th>ชื่อเจ้าของ</th><th>Action</th>
</tr>
</thead>
<tbody>
<c:forEach var="shop" items="${shops}">
<tr>
<td class="center"><a href=
"${pageContext.request.contextPath}/shop/${shop.id}/update">${shop.code}</a>
</td>
<td>${shop.name}</td>
<td>${shop.owner}</td>
</tr>
</c:forEach>
</tbody>
</table>
</div>
<jsp:include page="/WEB-INF/view/layouts/footer.jsp"/>
</body>
</html>

```



## 4.6) เพิ่มโค้ดที่เป็นสีแดงสำหรับปุ่มสำหรับเพิ่มร้านค้าไปในไฟล์

“WEB-INF/view/product/product-view-shop.jsp” ดังแสดงในรูป

### สินค้า - รายการร้านค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

เพิ่มร้านค้า

รหัสร้านค้า	ชื่อร้านค้า	ชื่อเจ้าของ
SH101	Adam Shop	Adam
SH111	ทดสอบ	กชค
SH999	ddd	ssf

© Copyright Lesson10 Watcharin's Spring MVC.

ต้องไม่มีใน  
รายการให้เลือก

### รายการร้านค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

รหัสร้านค้า	ชื่อร้านค้า	ชื่อเจ้าของ	Action
SH102	Bob Shop	Bob	เพิ่ม
SH104	Cindy Shop	Cindy	เพิ่ม
SH105	Dian Shop	Dian	เพิ่ม
SH106	Evan Shop	Evan	เพิ่ม
SH107	Frank Shop	Frank	เพิ่ม
SH108	Gorge Shop	Gorge	เพิ่ม
SH109	Henry Shop	Henry	เพิ่ม
SH110	Ivy Shop	Ivy	เพิ่ม
SH888	ddff	dsfsdf	เพิ่ม

© Copyright Lesson10 Watcharin's Spring MVC.

```
...
<jsp:include page="/WEB-INF/view/layouts/nav.jsp"/>
```

```
<input type="button" value="เพิ่มร้านค้า"
      onclick="window.location.href=
'${pageContext.request.contextPath}/product/${product.id}/shop/add';
return false;"
      class="add-button"
/>
...
```

## 4.7) เพิ่มเมธอด “showShopForAdd” ในคลาส ProductController ดังแสดงต่อไปนี้

```
...
@GetMapping("/{id}/shop/add")
public String showShopForAdd(@PathVariable("id") int id, Model model) {
    Product product = productService.getProduct(id);
    List<Shop> shops = productService.getShopDoesNotHaveProduct(id);
    model.addAttribute("title", "เพิ่มร้านค้า");
    model.addAttribute("product", product);
    model.addAttribute("shops", shops);
    return "product/shop-list";
}
...
```

ค้นหาร้านค้าเฉพาะที่  
ไม่ได้ขายสินค้าเท่านั้น

- 4.8) เพิ่มเมธอด “getShopDoesNotHaveProduct()” ใน interface “ProductService” และคลาส “ProductServiceImpl” ดังแสดงต่อไปนี้

```
public interface ProductService {
    ...
    List<Shop> getShopDoesNotHaveProduct(int id);
}

@Service
public class ProductServiceImpl implements ProductService {
    ...
    @Override
    @Transactional
    public List<Shop> getShopDoesNotHaveProduct(int id) {
        return productDao.getShopDoesNotHaveProduct(id);
    }
}
```

- 4.9) เพิ่มเมธอด “getShopDoesNotHaveProduct()” ใน interface “ProductDao” และคลาส “ProductDaoImpl” ดังแสดงต่อไปนี้

```
public interface ProductDao {
    ...
    List<Shop> getShopDoesNotHaveProduct(int id);
}

@Service
public class ProductDaoImpl implements ProductDao {
    ...
    @Override
    public List<Shop> getShopDoesNotHaveProduct(int id) {
        Session session = sessionFactory.getCurrentSession();
        Query query = session.createQuery("select d.shops from Product d where
d.id=:id");
        query.setParameter("id", id);
        List<Shop> shops1 = query.getResultList();
        query = session.createQuery("from Shop");
        List<Shop> shops2 = query.getResultList();
        shops2.removeAll(shops1);
        return shops2;
    }
}
```

ค้นหาเฉพาะ Shop ที่  
สัมพันธ์กับ Product

ค้นหาเฉพาะ Shop  
ทั้งหมดที่มี

นำ Shop ที่สัมพันธ์มาลบออก  
จาก Shop ทั้งหมด

Query เฉพาะ Shop ที่ไม่มีความสัมพันธ์กับ Product เท่านั้น

## 4.10) สร้างหน้าเว็บ “WEB-INF/view/product/shop-list.jsp” ดังแสดงต่อไปนี้

```

<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<title>${title}</title>
<link href="${pageContext.request.contextPath}/assets/css/style.css"
rel="stylesheet">
</head>
<body>
<div id="header"><h1>${title}</h1></div>
<div class="container">
<jsp:include page="/WEB-INF/view/layouts/nav.jsp"/>
<form action
="${pageContext.request.contextPath}/product/${product.id}/shop/add"
method="post">
<input type="hidden"
name="${_csrf.parameterName}"
value="${_csrf.token}"/>
<table class="table-bordered">
<thead>
<tr>
<th>รหัสร้านค้า</th><th>ชื่อร้านค้า</th><th>ชื่อเจ้าของ</th><th>Action</th>
</tr>
</thead>
<tbody>
<c:forEach var="shop" items="${shops}">
<tr>
<td class="center"><a href=
"${pageContext.request.contextPath}/shop/${shop.id}/update">${shop.code}</a></td>
<td>${shop.name}</td>
<td>${shop.owner}</td>
<td class="center">
<button type="submit" name="shop" value="${shop.id}">
</button>
</td>
</tr>
</c:forEach>
</tbody>
</table>
</form>
</div>
<jsp:include page="/WEB-INF/view/layouts/footer.jsp"/>
</body>
</html>

```

เขียน <form> ครอบ <table>

shop id พารามิเตอร์ <table>

เพิ่ม

## 4.11) เพิ่มเมธอด “addShop()” ในคลาส “ProductController” ดังแสดงต่อไปนี้

```

...
<code>@PostMapping("/{id}/shop/add")
public String addShop(@PathVariable("id") int productId,
    @RequestParam("shop") int shopId) {
    productService.addShopToProduct(productId, shopId);
    return "redirect:/product/" + productId + "/shop/add";
}
...

```

Post Method

กลับไปหน้าเดิม

- 4.12) เพิ่มเมธอด “addShopToProduct()” ใน interface “ProductService” และคลาส “ProductServiceImpl” ดังแสดงต่อไปนี้

```
public interface ProductService {
    ...
    void addShopToProduct(int productId, int shopId);
}

@Service
public class ProductServiceImpl implements ProductService {
    ...
    @Override
    @Transactional
    public void addShopToProduct(int productId, int shopId) {
        Shop shop = shopDao.getShop(shopId);
        Product product = productDao.getProduct(productId);
        product.getShops().add(shop);
        productDao.saveProduct(product);
    }
}
```

สร้างความสัมพันธ์

- 4.13) จากหน้าเว็บ “WEB-INF/view/product/product-view-shop.jsp” การลบร้านค้าออกจากสินค้า โดยการเพิ่มปุ่ม “ลบ” ดังแสดงรูปต่อไปนี้

สินค้า - รายการร้านค้า			
หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า			
เพิ่มร้านค้า			
รหัสร้านค้า	ชื่อร้านค้า	ชื่อเจ้าของ	Action
SH999	ddd	ssf	ลบ
SH888	ddff	dsfsdf	ลบ
SH104	Cindy Shop	Cindy	ลบ

© Copyright Lesson10 Watcharin's Spring MVC.

และได้ดัดแปลงต่อไปนี้

```
<table class="table-bordered">
  <thead>
    <tr>
      <th>รหัสร้านค้า</th><th>ชื่อร้านค้า</th><th>ชื่อเจ้าของ</th><th>Action</th>
    </tr>
  </thead>
  <tbody>
    <c:forEach var="shop" items="${shops}">
      <tr>
        ...
        <td class="center"><a
href="${pageContext.request.contextPath}/product/${product.id}/shop/${shop.id}
/remove">ลบ</a></td>
      </tr>
    </c:forEach>
  </tbody>
</table>
```

## 4.14) เพิ่มเมธอด “productRemoveShop()” ในคลาส ProductController ดังแสดงต่อไปนี้

```

...
@GetMapping("/{id}/shop/{shop}/remove")
public String productRemoveShop(@PathVariable("id") int productId,
                                @PathVariable("shop") int shopId) {
    productService.removeShopFromProduct(productId, shopId);
    return "redirect:/product/" + productId + "/view-shops";
}
...

```

## 4.15) เพิ่มเมธอด “removeShopFromProduct()” ใน interface “ProductService” และคลาส “ProductServiceImpl” ดังแสดงต่อไปนี้

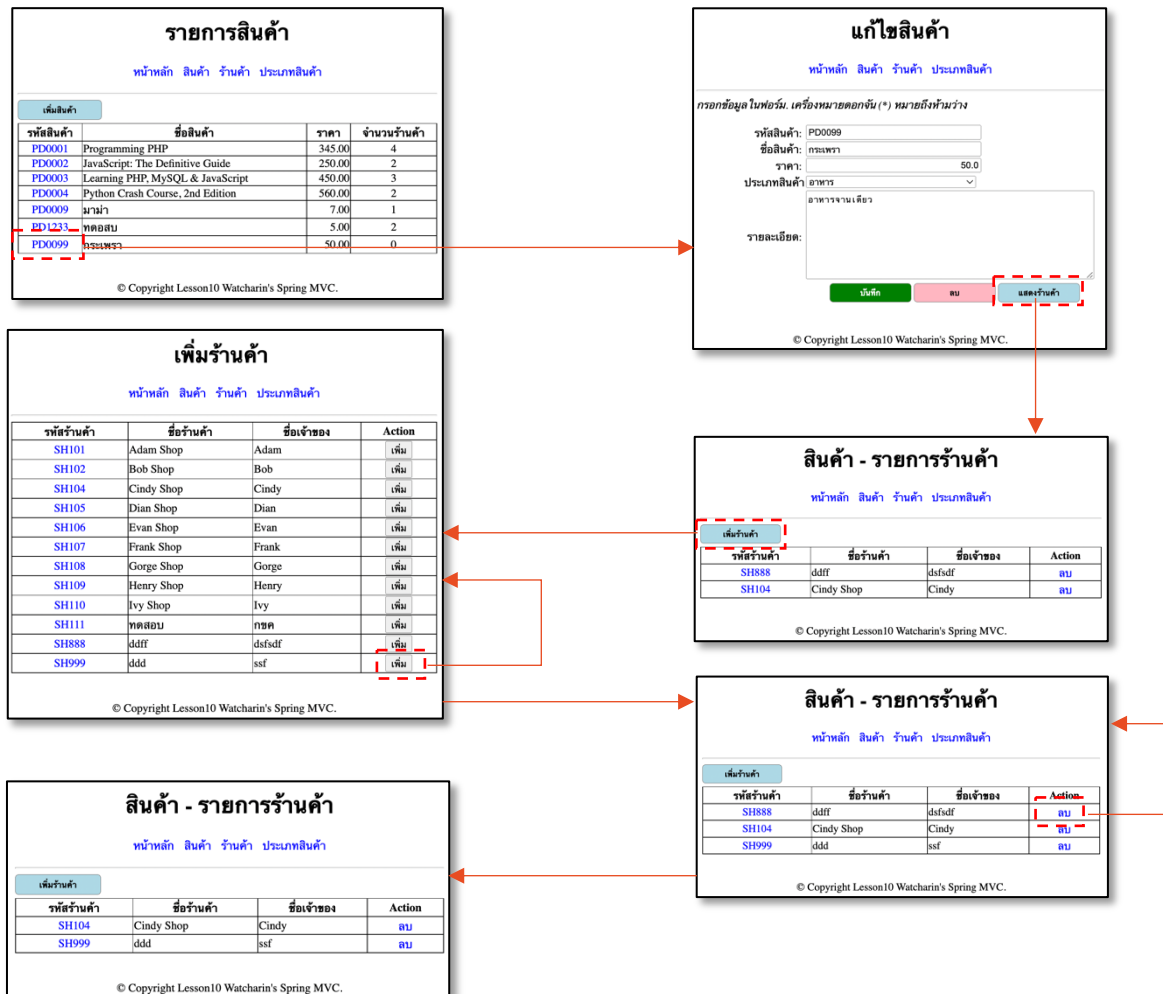
```

public interface ProductService {
    ...
    void removeShopFromProduct(int productId, int shopId);
}

@Service
public class ProductServiceImpl implements ProductService {
    ...
    @Override
    @Transactional
    public void removeShopFromProduct(int productId, int shopId) {
        Shop shop = shopDao.getShop(shopId);
        Product product = productDao.getProduct(productId);
        product.getShops().remove(shop);
        productDao.saveProduct(product);
    }
}

```

#### 4.16) ทดสอบการทำงาน



## 5. แก๊ไขโมเดลของ Shop และ ShopController ที่มีการทำงานเหมือนกับโมเดล Product และ ProductController

### รายการร้านค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

เพิ่มร้านค้า

รหัสสินค้า	ชื่อสินค้า	เจ้าของร้าน	Latitude	Longitude	จำนวนสินค้า
SH101	Adam Shop	Adam	19.93633	25.54209	3
SH102	Bob Shop	Bob	22.41123	-90.5768	2
SH104	Cindy Shop	Cindy	24.99135	80.35458	0
SH105	Dian Shop	Dian	54.11798	40.51694	0
SH106	Evan Shop	Evan	29.64557	114.1839	0
SH107	Frank Shop	Frank	38.74781	-125.22089	0
SH108	Gorge Shop	Gorge	3.58527	24.9636	0
SH109	Henry Shop	Henry	-15.14895	-66.80432	0
SH110	Ivy Shop	Ivy	-28.30137	120.25675	1
SH111	ทดสอบ	กชด	111.0	111.0	2
SH888	ddff	dsfsdf	4444.0	444.0	3
SH999	ddd	ssf	222.0	3444.0	5

© Copyright Lesson10 Watcharin's Spring MVC.

### แก้ไขร้านค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

กรอกข้อมูลในฟอร์ม. เครื่องหมายดอกจัน(\*) หมายถึงห้ามว่าง

รหัสร้านค้า: SH104  
ชื่อร้านค้า: Cindy Shop  
ชื่อเจ้าของร้าน: Cindy  
Latitude: -24.99135  
Longitude: 80.35458  
ที่อยู่: Address 1  
Address 2

บันทึก ลบ แสดงสินค้า

© Copyright Lesson10 Watcharin's Spring MVC.

### เพิ่มสินค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

รหัสสินค้า	ชื่อสินค้า	ราคา	Action
PD0001	Programming PHP	345.00	เพิ่ม
PD0002	JavaScript: The Definitive Guide	250.00	เพิ่ม
PD0003	Learning PHP, MySQL & JavaScript	450.00	เพิ่ม
PD0004	Python Crash Course, 2nd Edition	560.00	เพิ่ม
PD0009	มาว่า	7.00	เพิ่ม
PD1233	ทดสอบ	5.00	เพิ่ม
PD0099	กระเพรา	50.00	เพิ่ม

© Copyright Lesson10 Watcharin's Spring MVC.

### ร้านค้า - รายการสินค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

เพิ่มสินค้า

รหัสสินค้า	ชื่อสินค้า	ราคา	Action
------------	------------	------	--------

© Copyright Lesson10 Watcharin's Spring MVC.

### ร้านค้า - รายการสินค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

เพิ่มสินค้า

รหัสสินค้า	ชื่อสินค้า	ราคา	Action
PD1233	ทดสอบ	5.0	ลบ

© Copyright Lesson10 Watcharin's Spring MVC.

### ร้านค้า - รายการสินค้า

หน้าหลัก สินค้า ร้านค้า ประเภทสินค้า

เพิ่มสินค้า

รหัสสินค้า	ชื่อสินค้า	ราคา	Action
PD0099	กระเพรา	50.0	ลบ
PD1233	ทดสอบ	5.0	ลบ

© Copyright Lesson10 Watcharin's Spring MVC.