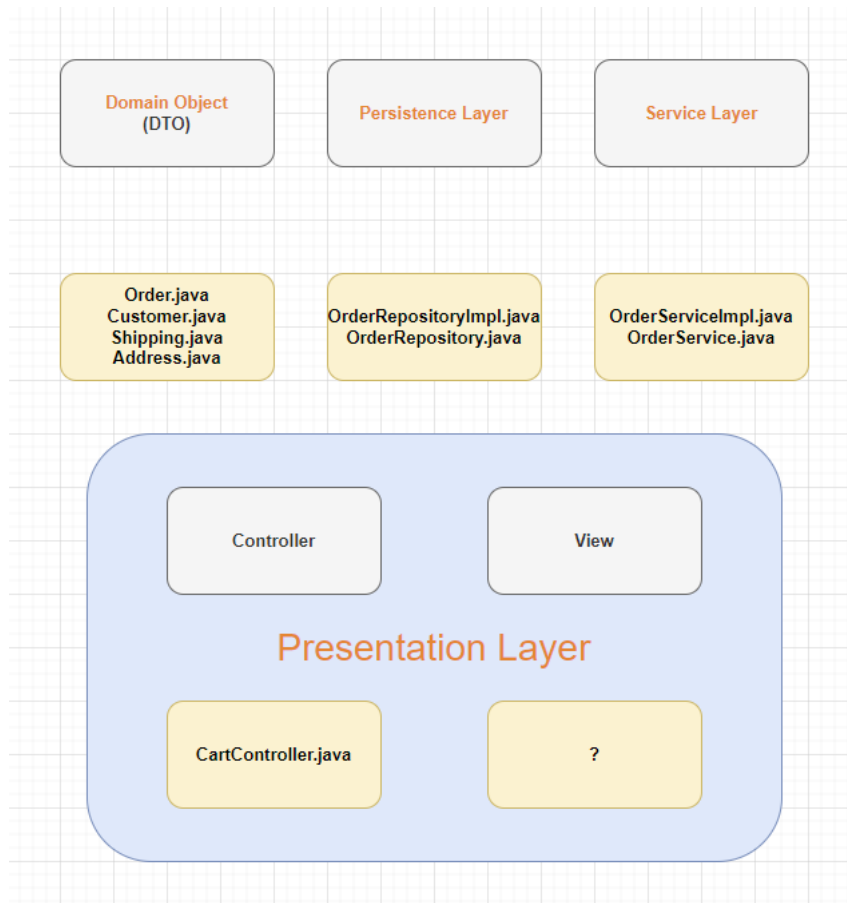


Spring Day 8

▼ Spring

Spring Web Flow(실습)



`CarDTO.java`, `Cart.java`, `CartItem.java` 직렬화(Serializable 인터페이스 구현)

```
@SuppressWarnings("serial")
public class CarDTO implements Serializable {
    //private static final long serialVersionUID = 3414129893883786050L;
```

```
@SuppressWarnings("serial")
public class Cart implements Serializable {
```

```
@SuppressWarnings("serial")
public class CartItem implements Serializable {
```

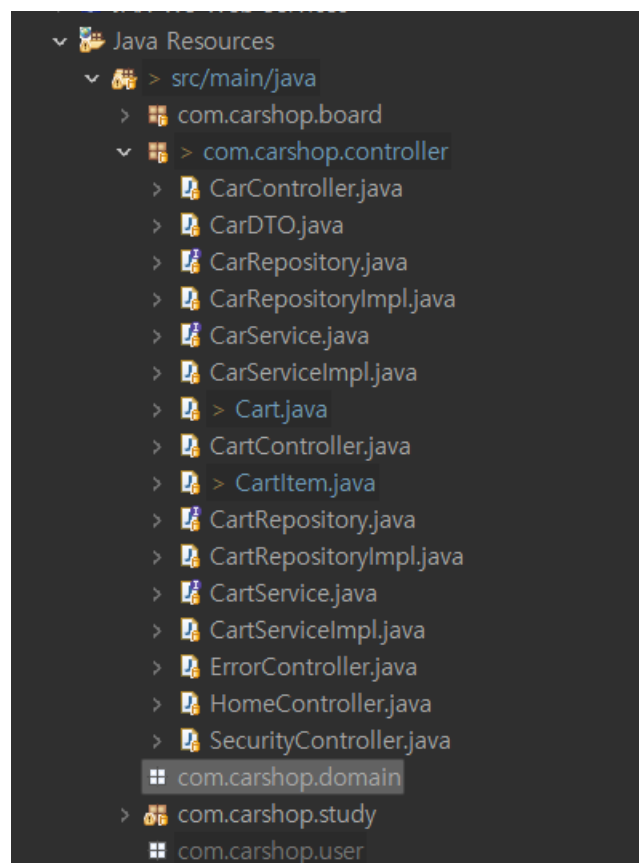
Serializable 인터페이스 구현(직렬화 버전의 고유값 부여)

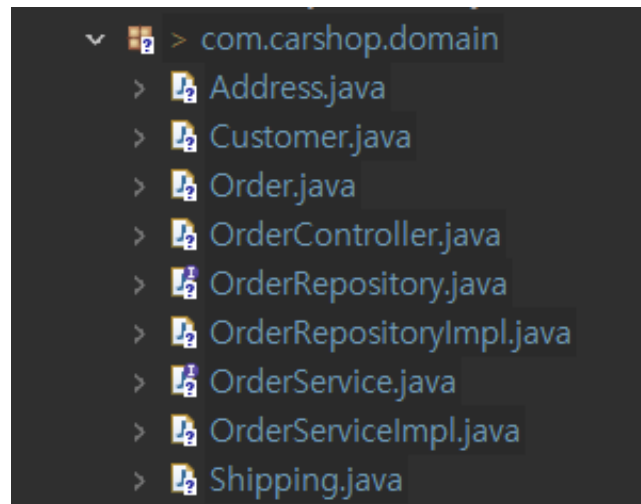
```
private static final long serialVersionUID = 3414129893883786050L;
```

직렬화 버전의 고유값을 선언하지 않을 경우(Default값을 만들어준다.)

```
@SuppressWarnings("serial")
```

지금은 편의상 domain 패키지를 만들어 플로우에 관련된 모든 클래스들을 위치 시키지만
원래는 Service, DTO, Repository 마다 각각 패키지를 만들어 사용하는 것을 권장한다.





Address.java(직렬화)

```
package com.carshop.domain;

import java.io.Serializable;

@SuppressWarnings("serial")
public class Address implements Serializable {

    private String detailName; //세부주소
    private String addressName; //주소
    private String country; //국가명
    private String zipCode; //우편번호

    public String getDetailName() {
        return detailName;
    }
    public void setDetailName(String detailName) {
        this.detailName = detailName;
    }
    public String getAddressName() {
        return addressName;
    }
    public void setAddressName(String addressName) {
        this.addressName = addressName;
    }
    public String getCountry() {
        return country;
    }
    public void setCountry(String country) {
        this.country = country;
    }
    public String getZipCode() {
        return zipCode;
    }
    public void setZipCode(String zipCode) {
        this.zipCode = zipCode;
    }
}

@Override
public int hashCode() {
    final int prime = 31;
    int result = 1;
    result = prime * result + ((addressName == null) ? 0 : addressName.hashCode());
    result = prime * result + ((country == null) ? 0 : country.hashCode());
    result = prime * result + ((detailName == null) ? 0 : detailName.hashCode());
    result = prime * result + ((zipCode == null) ? 0 : zipCode.hashCode());
    return result;
}

@Override
public boolean equals(Object obj) {
    if (this == obj)
        return true;
    if (obj == null)
        return false;
```

```

        if (getClass() != obj.getClass())
            return false;
        Address other = (Address) obj;
        if (addressName == null) {
            if (other.addressName != null)
                return false;
        } else if (!addressName.equals(other.addressName))
            return false;
        if (country == null) {
            if (other.country != null)
                return false;
        } else if (!country.equals(other.country))
            return false;
        if (detailName == null) {
            if (other.detailName != null)
                return false;
        } else if (!detailName.equals(other.detailName))
            return false;
        if (zipCode == null) {
            if (other.zipCode != null)
                return false;
        } else if (!zipCode.equals(other.zipCode))
            return false;
        return true;
    }

}

```

Customer.java(직렬화)

```

package com.carshop.domain;

import java.io.Serializable;

@SuppressWarnings("serial")
public class Customer implements Serializable {

    private String customerId; //고객id
    private String name; //고객이름
    private Address address; //고객주소 객체
    private String phone; //고객 전화번호

    public String getCustomerId() {
        return customerId;
    }
    public void setCustomerId(String customerId) {
        this.customerId = customerId;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public Address getAddress() {
        return address;
    }
    public void setAddress(Address address) {
        this.address = address;
    }
    public String getPhone() {
        return phone;
    }
    public void setPhone(String phone) {
        this.phone = phone;
    }

    @Override
    public int hashCode() {
        final int prime = 31;
        int result = 1;
        result = prime * result + ((customerId == null) ? 0 : customerId.hashCode());
        return result;
    }
    @Override
    public boolean equals(Object obj) {

```

```

    if (this == obj)
        return true;
    if (obj == null)
        return false;
    if (getClass() != obj.getClass())
        return false;
    Customer other = (Customer) obj;
    if (customerId == null) {
        if (other.customerId != null)
            return false;
    } else if (!customerId.equals(other.customerId))
        return false;
    return true;
}

public Customer() {
    this.address = new Address();
}

public Customer(String customerId, String name) {
    this();

    this.customerId = customerId;
    this.name = name;
}
}

```

Shipping.java(직렬화)

```

package com.carshop.domain;

import java.io.Serializable;
import java.util.Date;

import org.springframework.format.annotation.DateTimeFormat;

@SuppressWarnings("serial")
public class Shipping implements Serializable {

    private String name;           //배송받는 고객이름
    @DateTimeFormat(pattern = "yyyy/MM/dd") //날짜 패턴을 제한한다(이 패턴으로 변경할 수 없는 데이터의 경우에 오류 발생)
    private Date date;             //java.util.date
    private Address address;       //배송 주소 객체

    //기본 생성자
    public Shipping() {
        this.address = new Address();
    }

    public String getName() {
        return name;
    }

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

    public Date getDate() {
        return date;
    }

    public void setDate(Date date) {
        this.date = date;
    }

    public Address getAddress() {
        return address;
    }

    public void setAddress(Address address) {
        this.address = address;
    }
}

```

```
}
```

Order.java(직렬화)

```
package com.carshop.domain;

import java.io.Serializable;

@SuppressWarnings("serial")
public class Order implements Serializable {

    private Long orderId;    //주문 아이디
    private Cart cart;       //장바구니 객체
    private Customer customer; //고객 객체
    private Shipping shipping; //배송지 객체

    public Order() {
        this.customer = new Customer();
        this.shipping = new Shipping();
    }

    public Long getOrderId() {
        return orderId;
    }

    public void setOrderId(Long orderId) {
        this.orderId = orderId;
    }

    public Cart getCart() {
        return cart;
    }

    public void setCart(Cart cart) {
        this.cart = cart;
    }

    public Customer getCustomer() {
        return customer;
    }

    public void setCustomer(Customer customer) {
        this.customer = customer;
    }

    public Shipping getShipping() {
        return shipping;
    }

    public void setShipping(Shipping shipping) {
        this.shipping = shipping;
    }

    @Override
    public int hashCode() {
        final int prime = 31;
        int result = 1;
        result = prime * result + ((orderId == null) ? 0 : orderId.hashCode());
        return result;
    }

    @Override
    public boolean equals(Object obj) {
        if (this == obj)
            return true;
        if (obj == null)
            return false;
        if (getClass() != obj.getClass())
            return false;
        Order other = (Order) obj;
        if (orderId == null) {
            if (other.orderId != null)
                return false;
        } else if (!orderId.equals(other.orderId))
            return false;
        return true;
    }

}
```

OrderRepository.java

```
package com.carshop.domain;

public interface OrderRepository {

    Long saveOrder(Order order);

}
```

OrderRepositoryImpl.java

```
package com.carshop.domain;

import java.util.HashMap;
import java.util.Map;
import org.springframework.stereotype.Repository;

@Repository
public class OrderRepositoryImpl implements OrderRepository {

    private Map<Long, Order> listOfOrders;
    private long nextOrderId;

    public OrderRepositoryImpl() {
        listOfOrders = new HashMap<Long, Order>();
        nextOrderId = 20000;
    }

    @Override
    public Long saveOrder(Order order) {
        order.setOrderId(getNextOrderId());
        listOfOrders.put(order.getOrderId(), order);

        return order.getOrderId();
    }

    private synchronized long getNextOrderId() {
        return nextOrderId++;
    }

}
```

OrderService.java

```
package com.carshop.domain;

public interface OrderService {

    void confirmOrder(String cid, long quantity);
    Long saveOrder(Order order);

}
```

OrderServiceImpl.java

```
package com.carshop.domain;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.carshop.controller.CartService;
```

```

@Service
public class OrderServiceImpl implements OrderService {

    @Autowired
    private OrderRepository orderRepository;

    @Autowired
    private CartService cartService;

    @Override
    public void confirmOrder(String cid, long quantity) {
        //추후에는 판매가 이루어지면 재고량에서 1개를 빼주는 계산 처리
    }

    @Override
    public Long saveOrder(Order order) {
        Long orderId = orderRepository.saveOrder(order);
        cartService.delete(order.getCart().getCartId());
        return orderId;
        //주문이 처리가 되어 기존 장바구니는 비워준다.
    }
}

```

OrderController.java

```

package com.carshop.domain;

import org.springframework.stereotype.Controller;

@Controller
public class OrderController {

    //Order 컨트롤러는 실제 작동하지는 않는다. webflow가 주문처리를 담당한다.

}

```

▼ Web Flow 환경 설정

1. pom.xml에 의존성 라이브러리 주입.

Indexed Artifacts (32.6M)

Popular Categories

- Testing Frameworks & Tools
- Android Packages
- Logging Frameworks
- Java Specifications
- JSON Libraries
- Core Utilities
- JVM Languages
- Mocking

[Home](#) » [org.springframework.webflow](#) » [spring-webflow](#)

Spring Web Flow

Spring Web Flow builds on Spring MVC and allows implementing the "flows" of a web application. A flow encapsulates a sequence of steps that guide a user through the execution of some business task.

License	Apache 2.0
Categories	Web Frameworks
Tags	spring framework webflow web
Ranking	#706 in MvnRepository (See Top Artifacts) #6 in Web Frameworks
Used By	616 artifacts

[Central \(23\)](#)
[Spring Releases \(1\)](#)
[Spring Plugins \(13\)](#)
[Spring Milestones \(1\)](#)

	Version	Vulnerabilities	Repository	Usages	Date
2.5.x	2.5.1.RELEASE		Central	455	Sep 12, 2018
	2.5.0.RELEASE		Central	316	Mar 07, 2018



Spring Web Flow » 2.5.1.RELEASE

Spring Web Flow builds on Spring MVC and allows implementing the "flows" of a web application. A flow encapsulates a sequence of steps that guide a user through the execution of some business task.

License	Apache 2.0
Categories	Web Frameworks
Tags	spring framework webflow web
Organization	Spring IO
HomePage	https://github.com/spring-projects/spring-webflow
Date	Sep 12, 2018
Files	jar (517 KB) View All
Repositories	Central
Ranking	#706 in MvnRepository (See Top Artifacts) #6 in Web Frameworks
Used By	616 artifacts
Vulnerabilities	Vulnerabilities from dependencies: CVE-2022-22978 CVE-2022-22976 CVE-2022-22971 View 11 more ...

MavenGradleGradle (Short)Gradle (Kotlin)SBTLvyGrapeLeiningenBuildr

```
<!-- https://mvnrepository.com/artifact/org.springframework.webflow/spring-webflow -->
<dependency>
  <groupId>org.springframework.webflow</groupId>
  <artifactId>spring-webflow</artifactId>
  <version>2.5.1.RELEASE</version>
</dependency>
```

```
<!-- https://mvnrepository.com/artifact/org.springframework.webflow/spring-webflow -->
<dependency>
  <groupId>org.springframework.webflow</groupId>
  <artifactId>spring-webflow</artifactId>
  <version>2.5.1.RELEASE</version>
</dependency>
```

```
<!-- https://mvnrepository.com/artifact/org.springframework.webflow/spring-webflow -->
<dependency>
  <groupId>org.springframework.webflow</groupId>
  <artifactId>spring-webflow</artifactId>
  <version>2.5.1.RELEASE</version>
</dependency>

</dependencies>
<!-- https://mvnrepository.com/artifact/org.springframework.security/spring-security-web -->
```

2. servlet-context.xml 설정.

```
xmlns:webflow="http://www.springframework.org/schema/webflow-config"

http://www.springframework.org/schema/webflow-config http://www.springframework.org/schema/webflow-config/spring-webflow-config.xsd
```

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans:beans xmlns="http://www.springframework.org/schema/mvc"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xmlns:beans="http://www.springframework.org/schema/beans"
5   xmlns:context="http://www.springframework.org/schema/context"
6   xmlns:webflow="http://www.springframework.org/schema/webflow-config"
7   xsi:schemaLocation="http://www.springframework.org/schema/mvc https://www.springframework.org/schema/mvc/spring-mvc
8     http://www.springframework.org/schema/beans https://www.springframework.org/schema/beans/spring-beans.xsd
9     http://www.springframework.org/schema/webflow-config http://www.springframework.org/schema/webflow-config/spring-webflow-config.xsd
10    http://www.springframework.org/schema/context https://www.springframework.org/schema/context/spring-context.xsd"
11

```

필요한 여러가지 플로우를 설정하여 사용할 수 있다(위치와 아이디로 접근한다.)

```

<webflow:flow-registry id="flowRegistry">
  <webflow:flow-location
    path="/WEB-INF/flows/order/order-flow.xml" id="order" />
  <webflow:flow-location
    path="/WEB-INF/flows/ftest/ftest-flow.xml" id="ftest" />
</webflow:flow-registry>

```

```

<webflow:flow-executor id="flowExecutor"
  flow-registry="flowRegistry" />

<beans:bean id="flowHandlerMapping"
  class="org.springframework.webflow.mvc.servlet.FlowHandlerMapping">
  <beans:property name="flowRegistry" ref="flowRegistry" />
</beans:bean>

<beans:bean id="flowHandlerAdapter"
  class="org.springframework.webflow.mvc.servlet.FlowHandlerAdapter">
  <beans:property name="flowExecutor" ref="flowExecutor" />
</beans:bean>

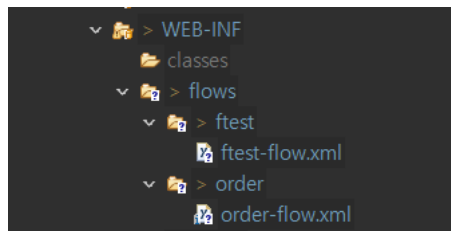
```

```

39 <!-- 일반 파일 업로드 경로 -->
40 <resources mapping="/images/**" location="/resources/images/" />
41
42
43 <webflow:flow-registry id="flowRegistry">
44   <webflow:flow-location
45     path="/WEB-INF/flows/order/order-flow.xml" id="order" />
46   <webflow:flow-location
47     path="/WEB-INF/flows/ftest/ftest-flow.xml" id="ftest" />
48 </webflow:flow-registry>
49
50 <webflow:flow-executor id="flowExecutor"
51   flow-registry="flowRegistry" />
52
53 <beans:bean id="flowHandlerMapping"
54   class="org.springframework.webflow.mvc.servlet.FlowHandlerMapping">
55   <beans:property name="flowRegistry" ref="flowRegistry" />
56 </beans:bean>
57
58 <beans:bean id="flowHandlerAdapter"
59   class="org.springframework.webflow.mvc.servlet.FlowHandlerAdapter">
60   <beans:property name="flowExecutor" ref="flowExecutor" />
61 </beans:bean>
62
63 </beans:beans>
64

```

폴더 설정



ftest-flow.xml

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

<!-- 모든 Web Flow는 <flow>로 시작하여 </flow>로 끝난다. -->
<flow xmlns="http://www.springframework.org/schema/webflow"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.springframework.org/schema/webflow
        http://www.springframework.org/schema/webflow/spring-webflow.xsd">

  <!-- name : 해당 플로우의 name값으로 호출이 되어 플로우가 진행된다. -->
  <!-- class : 매칭이 되는 클래스를 명목상 하나 연결한다. -->
  <var name="ftest" class="com.carshop.domain.Ftest"/>

  <!-- <view-state> : Web Flow에서 보여줄 View(jsp)를 정의한다(id값은 View(jsp)페이지와 매핑된다.) -->
  <view-state id="ftest1">
    <transition on="ftest2" to="ftest2" />
  </view-state>

  <!-- [view="ftest2.jsp"]처럼 View 파일의 이름을 직접 지정하면 해당 View(jsp)를 선택한다(생략하면 id값과 일치하는 View(jsp)로 자동선택 된다.) -->
  <!-- 만약 id 또는 view 에 지정한 파일이 폴더에 없는 경우에는 404 오류가 발생한다. -->
  <view-state id="ftest2" view="ftest2.jsp">
    <!-- <transition> : 페이지를 전환하는 역할을 한다. 따라서 to="ftest3"을 지정하면 <view-state id="ftest3">가 호출된다. -->
    <transition on="ftest3" to="ftest3" />
  </view-state>

  <view-state id="ftest3">
    <transition on="ftestCancelled" to="ftestCancelled" />
  </view-state>

  <!-- <end-state> : 페이지 흐름의 종료를 정의한다. -->
  <end-state id="ftestCancelled" />

  <!-- 어느 흐름에서도 사용가능 -->
  <global-transitions>
    <transition on="cancel" to="ftestCancelled"></transition>
  </global-transitions>

</flow>
```

Ftest.java

```
package com.carshop.domain;

import java.io.Serializable;

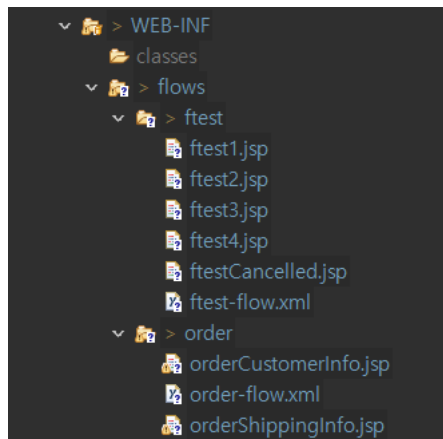
@SuppressWarnings("serial")
public class Ftest implements Serializable {


  //명목상 지정하는 클래스

}
```



Web Flow를 사용할 때는 xml 설정파일과 같은 위치에 View(jsp)를 생성해야 한다.



 View 이동은 컨트롤러가 우선하기 때문에
Web Flow를 사용할 때는 컨트롤러 매핑과 겹치지 않게 주의하자.

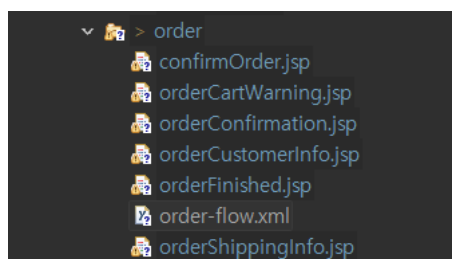
```

1 package com.carshop.controller;
2
3 import java.io.File;
4
19
20 @RequestMapping("/cars")
21 @Controller
22 public class CarController {
23
24     @Autowired
25     private CarService carService;
26
27     @RequestMapping
28     public String CarList(Model model) {
29         List<CarDTO> list = carService.getAllCarList();
30         model.addAttribute("carList", list);
31         return "cars";
32     }
33
34     @GetMapping("/{ccate}")
35     public String requestCarsByCategory(@PathVariable("ccate") String carCategory, Model model) {
36         List<CarDTO> carsByCategory = carService.getCarListByCategory(carCategory);
37         model.addAttribute("carList", carsByCategory);
38         return "cars";
39     }
40

```

Web Flow 실습(Order)

폴더 설정



cart.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Car Detail</title>
<script>
    function removeFromCart(action) {
        document.removeForm.action = action;
        document.removeForm.submit();
        window.location.reload();
    }

    function clearCart() {
        document.clearForm.submit();
        window.location.reload();
    }
</script>

</head>
<body>
    <%@ include file="header.jsp"%>

    <div class="my-5">
        <div class="alert alert-dark">
            <div class="container">
                <h1>장바구니</h1>
            </div>
        </div>
    </div>

    <div class="container">
        <div class="d-grid gap-2 d-md-flex justify-content-md-end">
            <form:form name="clearForm" method="delete">
                <a href="javascript:clearCart()" class="btn btn-danger pull-left">삭제하기</a>
                <a href="c:url value="/order?cartId=${cartId}" />" class="btn btn-success float-right"> 주문하기</a>
            </form:form>
        </div>
        <div style="padding-top: 50px">
            <table class="table table-hover">
                <tr>
                    <th>제품</th>
                    <th>가격</th>
                    <th>수량</th>
                    <th>소개</th>
                    <th>비고</th>
                </tr>
                <form:form name="removeForm" method="put">
                    <c:forEach items="${cart.cartItems}" var="item">
                        <tr>
                            <td>${item.value.car.cid}-${item.value.car.cname}</td>
                            <td>${item.value.car.cprice}</td>
                            <td>${item.value.quantity}</td>
                            <td>${item.value.totalPrice}</td>
                            <td><a
                                href="javascript:removeFromCart('/cart/remove/${item.value.car.cid}')"
                                class="btn btn-danger btn-sm">삭제</a></td>
                        </tr>
                    </c:forEach>
                </form:form>
                <tr>
                    <th></th>
                    <th></th>
                    <th>총액</th>
                    <th>${cart.grandTotal}</th>
                    <th></th>
                </tr>
            </table>

            <a href="c:url value="/cars" />" class="btn btn-primary">
                &laquo; 쇼핑 계속하기</a>
        </div>
    </div>
```

```

<%@ include file="footer.jsp"%>

</div>
</body>
</html>

```

order-flow.xml

```

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

<flow xmlns="http://www.springframework.org/schema/webflow"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.springframework.org/schema/webflow
        http://www.springframework.org/schema/webflow/spring-webflow.xsd">

    <var name="order" class="com.carshop.domain.Order"/>

    <action-state id="addCartToOrder">
<evaluate expression="cartServiceImpl.validateCart(requestParameters.cartId)" result="order.cart"></evaluate>
        <transition to="orderCartWarning" on-exception="com.carshop.exception.CartException" />
        <transition to="orderCustomerInfo"/>
    </action-state>

    <view-state id="orderCartWarning">
<transition to="endState"/>
    </view-state>

    <view-state id="orderCustomerInfo" model="order">
<transition on="customerInfo" to="orderShippingInfo"/>
    </view-state>

    <view-state id="orderShippingInfo" model="order">
<transition on="shippingInfo" to="orderConfirmation"/>
<transition on="backToCustomerInfo" to="orderCustomerInfo"/>
    </view-state>

    <view-state id="orderConfirmation">
<transition on="orderConfirm" to="confirmOrder"/>
<transition on="backToShippingInfo" to="orderShippingInfo"/>
    </view-state>

    <!-- <action-state> : 특정 시점에서 페이지의 흐름을 제어한다(결과(result)를 기반으로 다른 상태로 전환한다.) -->
    <action-state id="confirmOrder">
    <!-- <evaluate> : 매서드를 호출한 결과 값을 사용하여 분기한다. -->
    <!-- ex) 주문내용을 DB에 저장하는 매서드를 호출하는 경우[결과(result) : success] & [결과(result) : fail] -->
        <evaluate expression="orderServiceImpl.saveOrder(order)" result="order.orderId"/>
        <transition to="orderFinished"/>
    </action-state>

    <view-state id="orderFinished" model="order">
<transition to="endState"/>
    </view-state>

    <end-state id="endState"/>

    <global-transitions>
<transition on="cancel" to="endState"/>
</global-transitions>

</flow>

```

orderCustomerInfo.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
      pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
<%@ taglib uri="http://www.springframework.org/security/tags" prefix="sec" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">

```

```

<title>고객정보</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQh" integrity="sha384-kenU1KfD8Ie4zVF0s0G1M
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KfD8Ie4zVF0s0G1M

<style>
.bd-placeholder-img {
  font-size: 1.125rem;
  text-align: middle;
  -webkit-user-select: none;
  -moz-user-select: none;
  user-select: none;
}

@media ( min-width : 768px) {
  .bd-placeholder-img-lg {
    font-size: 3.5rem;
  }
}

.b-example-divider {
  height: 3rem;
  background-color: rgba(0, 0, 0, .1);
  border: solid rgba(0, 0, 0, .15);
  border-width: 1px 0;
  box-shadow: inset 0 .5em 1.5em rgba(0, 0, 0, .1), inset 0 .125em .5em
    rgba(0, 0, 0, .15);
}

.b-example-vr {
  flex-shrink: 0;
  width: 1.5rem;
  height: 100vh;
}

.bi {
  vertical-align: -.125em;
  fill: currentColor;
}

.nav-scroller {
  position: relative;
  z-index: 2;
  height: 2.75rem;
  overflow-y: hidden;
}

.nav-scroller .nav {
  display: flex;
  flex-wrap: nowrap;
  padding-bottom: 1rem;
  margin-top: -1px;
  overflow-x: auto;
  text-align: center;
  white-space: nowrap;
  -webkit-overflow-scrolling: touch;
}
}
</style>

</head>
<body>
<nav class="navbar navbar-expand navbar-dark bg-dark" aria-label="Second navbar example">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">CarShop</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarsExample02" aria-controls="navbars"
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarsExample02">
      <ul class="navbar-nav me-auto">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">홈</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/cars">차량보기</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/board">게시판</a>
        </li>
        <li class="nav-item dropdown">
          <button class="btn btn-dark dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
            회원관리
          </button>
          <ul class="dropdown-menu dropdown-menu-dark">
            <li><a class="dropdown-item" href="#">로그인</a></li>
            <li><a class="dropdown-item" href="#">로그아웃</a></li>

```

```

        </ul>
    </li>

    </ul>
    <sec:authorize access="isAnonymous()">
        <form action="/login" method="post">
            <input type="submit" class="btn btn-primary" value="login">
            <input type="hidden" name="{_csrf.parameterName}" value="{_csrf.token}"/>
        </form>
    </sec:authorize>

    <sec:authorize access="isAuthenticated()">
        <form action="/logout" method="post">
            <input type="submit" class="btn btn-success" value="logout">
            <input type="hidden" name="{_csrf.parameterName}" value="{_csrf.token}"/>
        </form>
    </sec:authorize>

    </div>
</div>
</nav>
<div class="my-30">
    <div class="alert alert-dark" role="alert">
        <div class="container">
            <h1>고객정보</h1>
        </div>
    </div>
</div>

<div class="container">
    <form:form modelAttribute="order.customer" class="form-holizo">
        <fieldset>
            <legend>고객 세부 사항</legend>

            <div class="form-group row">
                <label class="col-sm-2 control-label"> >고객 ID</label>
                <div class="col-sm-3">
                    <form:input path="customerId" class="form-control" />
                </div>
            </div>
            <div class="form-group row">
                <label class="col-sm-2 control-label"> >성명</label>
                <div class="col-sm-3">
                    <form:input path="name" class="form-control" />
                </div>
            </div>
            <div class="form-group row">
                <label class="col-sm-2 control-label"> >전화번호</label>
                <div class="col-sm-3">
                    <form:input path="phone" class="form-control" />
                </div>
            </div>
            <div class="form-group row">
                <label class="col-sm-2 control-label"> >국가명</label>
                <div class="col-sm-3">
                    <form:input path="address.country" class="form-control"/>
                </div>
            </div>
            <div class="form-group row">
                <label class="col-sm-2 control-label"> >우편번호</label>
                <div class="col-sm-3">
                    <form:input path="address.zipCode" class="form-control" />
                </div>
            </div>
            <div class="form-group row">
                <label class="col-sm-2 control-label"> >주소</label>
                <div class="col-sm-5">
                    <form:input path="address.addressName" class="form-control" />
                </div>
            </div>
            <div class="form-group row">
                <label class="col-sm-2 control-label"> >세부주소</label>
                <div class="col-sm-3">
                    <form:input path="address.detailName" class="form-control" />
                </div>
            </div>
            <input type="hidden" name="_flowExecutionKey" value="{flowExecutionKey }">

            <div class="form-group row">
                <div class="col-sm-offset-2 col-sm-10">
                    <input type="submit" class="btn btn-primary" value="등록" name="_eventId_customerInfo" />
                    <button class="btn btn-default" name="_eventId_cancel">취소</button>
                </div>
            </div>
        </fieldset>
    </form:form>
</div>

```



```

        </div>
    </div>

    </fieldset>

    </form:form>
</div>

<footer class="w3-center w3-black w3-padding-48 w3-xlarge" style="margin-top: 300px;">

    <a href="#" class="w3-button w3-light-grey"><i class="fa fa-arrow-up w3-margin-right"></i>To the top</a>
    <div class="w3-xlarge w3-section">
    </div>
     <a href="/" title="DaDaMall" class="w3-hover-text-light"> 2022 CarShop </a></p>
</footer>

</body>
</html>

```

orderShippingInfo.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
<%@ taglib uri="http://www.springframework.org/security/tags" prefix="sec" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>배송 세부 사항</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQh
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KfDbIe4zVF0s0G1M

<style>
    .bd-placeholder-img {
        font-size: 1.125rem;
        text-align: middle;
        -webkit-user-select: none;
        -moz-user-select: none;
        user-select: none;
    }

    @media ( min-width : 768px) {
        .bd-placeholder-img-lg {
            font-size: 3.5rem;
        }
    }

    .b-example-divider {
        height: 3rem;
        background-color: rgba(0, 0, 0, .1);
        border: solid rgba(0, 0, 0, .15);
        border-width: 1px 0;
        box-shadow: inset 0 .5em 1.5em rgba(0, 0, 0, .1), inset 0 .125em .5em
            rgba(0, 0, 0, .15);
    }

    .b-example-vr {
        flex-shrink: 0;
        width: 1.5rem;
        height: 100vh;
    }

    .bi {
        vertical-align: -.125em;
        fill: currentColor;
    }

    .nav-scroller {
        position: relative;
        z-index: 2;
        height: 2.75rem;
        overflow-y: hidden;

```

```

    }

    .nav-scroller .nav {
        display: flex;
        flex-wrap: nowrap;
        padding-bottom: 1rem;
        margin-top: -1px;
        overflow-x: auto;
        text-align: center;
        white-space: nowrap;
        -webkit-overflow-scrolling: touch;
    }
}
</style>

</head>
<body>
<nav class="navbar navbar-expand navbar-dark bg-dark" aria-label="Second navbar example">
    <div class="container-fluid">
        <a class="navbar-brand" href="#">CarShop</a>
        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarsExample02" aria-controls="navbars"
            <span class="navbar-toggler-icon"></span>
        </button>

        <div class="collapse navbar-collapse" id="navbarsExample02">
            <ul class="navbar-nav me-auto">
                <li class="nav-item">
                    <a class="nav-link active" aria-current="page" href="#">홈</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="/cars">차량보기</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="/board">게시판</a>
                </li>
                <li class="nav-item dropdown">
                    <button class="btn btn-dark dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
                        회원관리
                    </button>
                    <ul class="dropdown-menu dropdown-menu-dark">
                        <li><a class="dropdown-item" href="#">로그인</a></li>
                        <li><a class="dropdown-item" href="#">로그아웃</a></li>
                    </ul>
                </li>
            </ul>
            <sec:authorize access="isAnonymous()">
                <form action="/login" method="post">
                    <input type="submit" class="btn btn-primary" value="login">
                    <input type="hidden" name="{_csrf.parameterName}" value="{_csrf.token}"/>
                </form>
            </sec:authorize>

            <sec:authorize access="isAuthenticated()">
                <form action="/logout" method="post">
                    <input type="submit" class="btn btn-success" value="logout">
                    <input type="hidden" name="{_csrf.parameterName}" value="{_csrf.token}"/>
                </form>
            </sec:authorize>

        </div>
    </div>
</nav>
<div class="my-1">
    <div class="alert alert-dark" role="alert">
        <div class="container">
            <h1>배송 세부 사항</h1>
        </div>
    </div>
</div>
<div class="container">
    <form:form modelAttribute="order.shipping" class="form-horizontal">
        <fieldset>
            <legend>배송 세부 사항</legend>
            <div class="form-group row">
                <label class="col-sm-2 control-label">성명</label>
                <div class="col-sm-3">
                    <form:input path="name" class="form-control" />
                </div>
            </div>
            <div class="form-group row">
                <label class="col-sm-2 control-label">배송일</label>
                <div class="col-sm-3">

```

```

        <form:input path="date" class="form-control" /> (yyyy/mm/dd)
    </div>
    <div class="col-sm-6">
        <form:errors path="date" cssClass="text-danger"/>
    </div>
</div>
<div class="form-group row">
    <label class="col-sm-2 control-label">국가명</label>
    <div class="col-sm-3">
        <form:input path="address.country" class="form-control" />
    </div>
</div>
<div class="form-group row">
    <label class="col-sm-2 control-label">우편번호</label>
    <div class="col-sm-3">
        <form:input path="address.zipCode" class="form-control" />
    </div>
</div>
<div class="form-group row">
    <label class="col-sm-2 control-label">주소</label>
    <div class="col-sm-5">
        <form:input path="address.addressName" class="form-control" />
    </div>
</div>
<div class="form-group row">
    <label class="col-sm-2 control-label">세부주소</label>
    <div class="col-sm-3">
        <form:input path="address.detailName" class="form-control" />
    </div>
</div>
<input type="hidden" name="_flowExecutionKey" value="{flowExecutionKey}"/>
<div class="form-group row">
    <div class="col-sm-offset-2 col-sm-10">
        <button class="btn btn-default" name="_eventId_backToCustomerInfo">이전</button>
        <input type="submit" class="btn btn-primary" value="등록" name="_eventId_shippingInfo"/>
        <button class="btn btn-default" name="_eventId_cancel">취소</button>
    </div>
</div>
</fieldset>
</form:form>
</div>

<footer class="w3-center w3-black w3-padding-48 w3-xlarge" style="margin-top: 300px;">

    <a href="#" class="w3-button w3-light-grey"><i class="fa fa-arrow-up w3-margin-right"></i>To the top</a>
    <div class="w3-xlarge w3-section">
    </div>
     <a href="/" title="DaDaMall" class="w3-hover-text-light"> 2022 CarShop </a></p>
</footer>

</body>
</html>

```

orderConfirmation.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt"%>
<%@ taglib uri="http://www.springframework.org/security/tags" prefix="sec" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>영수증</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQh
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KFdBIe4zVF0s0G1M
    <style>
        .bd-placeholder-img {
            font-size: 1.125rem;
            text-align: middle;
            -webkit-user-select: none;
            -moz-user-select: none;
            user-select: none;
        }
    </style>

```

```

@media ( min-width : 768px) {
  .bd-placeholder-img-lg {
    font-size: 3.5rem;
  }
}

.b-example-divider {
  height: 3rem;
  background-color: rgba(0, 0, 0, .1);
  border: solid rgba(0, 0, 0, .15);
  border-width: 1px 0;
  box-shadow: inset 0 .5em 1.5em rgba(0, 0, 0, .1), inset 0 .125em .5em
    rgba(0, 0, 0, .15);
}

.b-example-vr {
  flex-shrink: 0;
  width: 1.5rem;
  height: 100vh;
}

.bi {
  vertical-align: -.125em;
  fill: currentColor;
}

.nav-scroller {
  position: relative;
  z-index: 2;
  height: 2.75rem;
  overflow-y: hidden;
}

.nav-scroller .nav {
  display: flex;
  flex-wrap: nowrap;
  padding-bottom: 1rem;
  margin-top: -1px;
  overflow-x: auto;
  text-align: center;
  white-space: nowrap;
  -webkit-overflow-scrolling: touch;
}
}
</style>

</head>
<body>
<nav class="navbar navbar-expand navbar-dark bg-dark" aria-label="Second navbar example">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">CarShop</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarsExample02" aria-controls="navbars"
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarsExample02">
      <ul class="navbar-nav me-auto">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="/">홈</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/cars">차량보기</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/board">게시판</a>
        </li>
        <li class="nav-item dropdown">
          <button class="btn btn-dark dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
            회원관리
          </button>
          <ul class="dropdown-menu dropdown-menu-dark">
            <li><a class="dropdown-item" href="#">로그인</a></li>
            <li><a class="dropdown-item" href="#">로그아웃</a></li>
          </ul>
        </li>
      </ul>
      <sec:authorize access="isAnonymous()">
        <form action="/login" method="post">
          <input type="submit" class="btn btn-primary" value="login">
          <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}"/>
        </form>
      </sec:authorize>

      <sec:authorize access="isAuthenticated()">
        <form action="/logout" method="post">

```

```

        <input type="submit" class="btn btn-success" value="logout">
        <input type="hidden" name="{_csrf.parameterName}" value="{_csrf.token}"/>
    </form>
</sec:authorize>

</div>
</div>
</nav>
<div class="my-30">
    <div class="alert alert-dark" role="alert">
        <div class="container">
            <h1>영수증</h1>
        </div>
    </div>
</div>
</div>

<div class="container" >

    <form:form modelAttribute="order" class="form-horizontal">
    <div class="well col-md-9 col-md-offset-2" style="background:#fafafe; padding: 20px">
        <div class="text-center">
            <h1>영수증</h1>
        </div>
        <div class="row">
            <div class="col-md-6">
                <address>
                    <strong>배송 주소</strong> <br>
                    성명 : ${order.shipping.name}<br>
                    우편번호 : ${order.shipping.address.zipCode}<br>
                    주소 : ${order.shipping.address.addressName} ${order.shipping.address.detailName} (${order.shipping.address.country}) <br>
                </address>
            </div>
            <div class="col-md-6 text-right">
                <p> <em>배송일: <fmt:formatDate type="date" value="{order.shipping.date}" /></em></p>
            </div>
        </div>
        <div class="row">
            <div class="col-md-9">
                <address>
                    <strong>청구주소</strong> <br>
                    성명 : ${order.customer.name}<br>
                    우편번호 : ${order.customer.address.zipCode}<br>
                    주소 : ${order.customer.address.addressName} ${order.customer.address.detailName} (${order.customer.address.country}) <br>
                    HP : ${order.customer.phone}<br>
                </address>
            </div>
        </div>
        <div class="row">
            <table class="table table-hover">
                <thead>
                    <tr><th>자동차</th>
                    <th>#</th>
                    <th class="text-center">가격</th>
                    <th class="text-center">소개</th>
                </tr>
                <tbody>
                    <c:forEach var="cartItem" items="{order.cart.cartItems}">
                        <tr>
                            <td> <em>${cartItem.value.car.cname}</em></td>
                            <td style="text-align: center">${cartItem.value.quantity}</td>
                            <td class="text-center">${cartItem.value.car.cprice}원</td>
                            <td class="text-center">${cartItem.value.totalPrice}원</td>
                        </tr>
                    </c:forEach>
                    <tr>
                        <td> </td>
                        <td> </td>
                        <td class="text-right"><h5> <strong>총액: </strong></h5></td>
                        <td class="text-center text-danger"><h4><strong>${order.cart.grandTotal}</strong></h4></td>
                    </tr>
                </tbody>
            </table>
            <input type="hidden" name="_flowExecutionKey" value="{flowExecutionKey}" />
            <button class="btn btn-default" name="_eventId_backToShippingInfo">이전</button>
            <button type="submit" class="btn btn-success" name="_eventId_orderConfirm"> 주문완료 </button>
            <button class="btn btn-default" name="_eventId_cancel">취소</button>
        </div>
    </div>
</form:form>
</div>

<footer class="w3-center w3-black w3-padding-48 w3-xlarge" style="margin-top: 300px;">

```

```

    <a href="#" class="w3-button w3-light-grey"><i class="fa fa-arrow-up w3-margin-right"></i>To the top</a>
  <div class="w3-xlarge w3-section">
  </div>
   <a href="/" title="DaDaMall" class="w3-hover-text-light"> 2022 CarShop </a></p>
</footer>

</body>
</html>

```

confirmOrder.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
<%@ taglib uri="http://www.springframework.org/security/tags" prefix="sec" %>
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Insert title here</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQh
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KFdBIe4zVF0s0G1M

<style>
  .bd-placeholder-img {
    font-size: 1.125rem;
    text-align: middle;
    -webkit-user-select: none;
    -moz-user-select: none;
    user-select: none;
  }

  @media ( min-width : 768px) {
    .bd-placeholder-img-lg {
      font-size: 3.5rem;
    }
  }

  .b-example-divider {
    height: 3rem;
    background-color: rgba(0, 0, 0, .1);
    border: solid rgba(0, 0, 0, .15);
    border-width: 1px 0;
    box-shadow: inset 0 .5em 1.5em rgba(0, 0, 0, .1), inset 0 .125em .5em
      rgba(0, 0, 0, .15);
  }

  .b-example-vr {
    flex-shrink: 0;
    width: 1.5rem;
    height: 100vh;
  }

  .bi {
    vertical-align: -.125em;
    fill: currentColor;
  }

  .nav-scroller {
    position: relative;
    z-index: 2;
    height: 2.75rem;
    overflow-y: hidden;
  }

  .nav-scroller .nav {
    display: flex;
    flex-wrap: nowrap;
    padding-bottom: 1rem;
    margin-top: -1px;
    overflow-x: auto;
    text-align: center;
    white-space: nowrap;
    -webkit-overflow-scrolling: touch;
  }
</style>

```

```

</head>
<body>
<nav class="navbar navbar-expand navbar-dark bg-dark" aria-label="Second navbar example">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">CarShop</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarsExample02" aria-controls="navbars"
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarsExample02">
      <ul class="navbar-nav me-auto">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">홈</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/cars">차량보기</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/board">게시판</a>
        </li>
        <li class="nav-item dropdown">
          <button class="btn btn-dark dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
            회원관리
          </button>
          <ul class="dropdown-menu dropdown-menu-dark">
            <li><a class="dropdown-item" href="#">로그인</a></li>
            <li><a class="dropdown-item" href="#">로그아웃</a></li>
          </ul>
        </li>
      </ul>
      <sec:authorize access="isAnonymous()">
        <form action="/login" method="post">
          <input type="submit" class="btn btn-primary" value="login">
          <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}"/>
        </form>
      </sec:authorize>

      <sec:authorize access="isAuthenticated()">
        <form action="/logout" method="post">
          <input type="submit" class="btn btn-success" value="logout">
          <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}"/>
        </form>
      </sec:authorize>

    </div>
  </div>
</nav>
<div class="my-30">
  <div class="alert alert-dark" role="alert">
    <div class="container">
      <h1>고객정보</h1>
    </div>
  </div>
</div>

<footer class="w3-center w3-black w3-padding-48 w3-xlarge" style="margin-top: 300px;">

  <a href="#" class="w3-button w3-light-grey"><i class="fa fa-arrow-up w3-margin-right"></i>To the top</a>
  <div class="w3-xlarge w3-section">
    <div>
       <a href="/" title="DaDaMall" class="w3-hover-text-light"> 2022 CarShop </a></p>
    </div>
  </div>

</body>
</html>

```

orderFinished.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>

```

```

<%@ taglib uri="http://www.springframework.org/security/tags" prefix="sec" %>
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>주문완료</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQh"
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KFdBIE4zVF0s0G1M

<style>
  .bd-placeholder-img {
    font-size: 1.125rem;
    text-align: middle;
    -webkit-user-select: none;
    -moz-user-select: none;
    user-select: none;
  }

  @media ( min-width : 768px) {
    .bd-placeholder-img-lg {
      font-size: 3.5rem;
    }
  }

  .b-example-divider {
    height: 3rem;
    background-color: rgba(0, 0, 0, .1);
    border: solid rgba(0, 0, 0, .15);
    border-width: 1px 0;
    box-shadow: inset 0 .5em 1.5em rgba(0, 0, 0, .1), inset 0 .125em .5em
      rgba(0, 0, 0, .15);
  }

  .b-example-vr {
    flex-shrink: 0;
    width: 1.5rem;
    height: 100vh;
  }

  .bi {
    vertical-align: -.125em;
    fill: currentColor;
  }

  .nav-scroller {
    position: relative;
    z-index: 2;
    height: 2.75rem;
    overflow-y: hidden;
  }

  .nav-scroller .nav {
    display: flex;
    flex-wrap: nowrap;
    padding-bottom: 1rem;
    margin-top: -1px;
    overflow-x: auto;
    text-align: center;
    white-space: nowrap;
    -webkit-overflow-scrolling: touch;
  }
</style>

</head>
<body>
<nav class="navbar navbar-expand navbar-dark bg-dark" aria-label="Second navbar example">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">CarShop</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarsExample02" aria-controls="navbars
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarsExample02">
      <ul class="navbar-nav me-auto">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="/">홈</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/cars">차량보기</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/board">게시판</a>
        </li>
        <li class="nav-item dropdown">
          <button class="btn btn-dark dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">

```



```

        회원관리
    </button>
    <ul class="dropdown-menu dropdown-menu-dark">
        <li><a class="dropdown-item" href="#">로그인</a></li>
        <li><a class="dropdown-item" href="#">로그아웃</a></li>
    </ul>
</li>

</ul>
<sec:authorize access="isAnonymous()">
    <form action="/login" method="post">
        <input type="submit" class="btn btn-primary" value="login">
        <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}"/>
    </form>
</sec:authorize>

<sec:authorize access="isAuthenticated()">
    <form action="/logout" method="post">
        <input type="submit" class="btn btn-success" value="logout">
        <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}"/>
    </form>
</sec:authorize>

    </div>
</div>
</nav>
<div class="my-30">
    <div class="alert alert-dark" role="alert">
        <div class="container">
            <h1>주문완료</h1>
        </div>
    </div>
</div>
</div>

<div class="container">
<h2 class="alert alert-danger">주문완료되었습니다. 감사합니다.</h2>
<p>주문 번호 : ${order.orderId}</p>
</div>

<div class="container">
<p><a href="/cars" class="btn btn-primary">제품목록</a></p>
</div>

<footer class="w3-center w3-black w3-padding-48 w3-xlarge" style="margin-top: 300px;">

    <a href="#" class="w3-button w3-light-grey"><i class="fa fa-arrow-up w3-margin-right"></i>To the top</a>
    <div class="w3-xlarge w3-section">
    </div>
     <a href="/" title="DaDaMall" class="w3-hover-text-light"> 2022 CarShop </a></p>
</footer>

</body>
</html>

```

orderCartWarning.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
<%@ taglib uri="http://www.springframework.org/security/tags" prefix="sec" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>장바구니 전송 에러</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQh
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KfDBe4zVF0s0G1M

<style>
    .bd-placeholder-img {
        font-size: 1.125rem;
        text-align: middle;
    }

```

```

-webkit-user-select: none;
-moz-user-select: none;
user-select: none;
}

@media ( min-width : 768px) {
  .bd-placeholder-img-lg {
    font-size: 3.5rem;
  }
}

.b-example-divider {
  height: 3rem;
  background-color: rgba(0, 0, 0, .1);
  border: solid rgba(0, 0, 0, .15);
  border-width: 1px 0;
  box-shadow: inset 0 .5em 1.5em rgba(0, 0, 0, .1), inset 0 .125em .5em
    rgba(0, 0, 0, .15);
}

.b-example-vr {
  flex-shrink: 0;
  width: 1.5rem;
  height: 100vh;
}

.bi {
  vertical-align: -.125em;
  fill: currentColor;
}

.nav-scroller {
  position: relative;
  z-index: 2;
  height: 2.75rem;
  overflow-y: hidden;
}

.nav-scroller .nav {
  display: flex;
  flex-wrap: nowrap;
  padding-bottom: 1rem;
  margin-top: -1px;
  overflow-x: auto;
  text-align: center;
  white-space: nowrap;
  -webkit-overflow-scrolling: touch;
}
}
</style>

</head>
<body>
<nav class="navbar navbar-expand navbar-dark bg-dark" aria-label="Second navbar example">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">CarShop</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarsExample02" aria-controls="navbars"
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarsExample02">
      <ul class="navbar-nav me-auto">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">홈</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/cars">차량보기</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/board">게시판</a>
        </li>
        <li class="nav-item dropdown">
          <button class="btn btn-dark dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
            회원관리
          </button>
          <ul class="dropdown-menu dropdown-menu-dark">
            <li><a class="dropdown-item" href="#">로그인</a></li>
            <li><a class="dropdown-item" href="#">로그아웃</a></li>
          </ul>
        </li>
      </ul>
      <sec:authorize access="isAnonymous()">
        <form action="/login" method="post">
          <input type="submit" class="btn btn-primary" value="login">
          <input type="hidden" name="{_csrf.parameterName}" value="{_csrf.token}"/>
        </form>
      </sec:authorize>
    </div>
  </div>
</nav>

```

```

        </sec:authorize>

        <sec:authorize access="isAuthenticated()">
            <form action="/logout" method="post">
                <input type="submit" class="btn btn-success" value="logout">
                <input type="hidden" name="${_csrf.parameterName}" value="${_csrf.token}"/>
            </form>
        </sec:authorize>

    </div>
</div>
</nav>
<div class="my-30">
    <div class="alert alert-dark" role="alert">
        <div class="container">
            <h1>장바구니 전송 에러</h1>
        </div>
    </div>
</div>

<div class="container">
    <h2>알수 없는 에러가 발생하였습니다.</h2>
</div>

<div class="container">
    <p><a href="/cars" class="btn btn-primary">제품목록</a>
</div>

<footer class="w3-center w3-black w3-padding-48 w3-xlarge" style="margin-top: 300px;">

    <a href="#" class="w3-button w3-light-grey"><i class="fa fa-arrow-up w3-margin-right"></i>To the top</a>
    <div class="w3-xlarge w3-section">
        </div>
     <a href="/" title="DaDaMall" class="w3-hover-text-light"> 2022 CarShop </a></p>
</footer>

</body>
</html>

```

CartException.java

```

package com.carshop.exception;

@SuppressWarnings("serial")
public class CartException extends RuntimeException {

    private String cartId;

    public CartException(String cartId) {
        this.cartId = cartId;
    }

    public String getCartId() {
        return cartId;
    }

}

```

CartServiceImpl.java

```

package com.carshop.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.carshop.exception.CartException;

@Service
public class CartServiceImpl implements CartService {

```

```

@Autowired
private CartRepository cartRepository;

@Override
public Cart create(Cart cart) {
    return cartRepository.create(cart);
}

@Override
public Cart read(String cartId) {
    return cartRepository.read(cartId);
}

@Override
public void update(String cartId, Cart cart) {

    cartRepository.update(cartId, cart);

}

@Override
public void delete(String cartId) {

    cartRepository.delete(cartId);

}

public Cart validateCart(String cartId) {

    Cart cart = cartRepository.read(cartId);
    if(cart == null || cart.getCartItems().size()==0) {
        throw new CartException(cartId);
    }

    return cart;
}
}

```

OrderServiceImpl.java

```

package com.carshop.domain;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.carshop.controller.CartService;

@Service
public class OrderServiceImpl implements OrderService {

    @Autowired
    private OrderRepository orderRepository;

    @Autowired
    private CartService cartService;

    @Override
    public void confirmOrder(String cid, long quantity) {
        //추후에는 판매가 이루어지면 재고량에서 1개를 빼주는 계산 처리
    }

    @Override
    public Long saveOrder(Order order) {
        Long orderId = orderRepository.saveOrder(order);
        cartService.delete(order.getCart().getCartId());
        return orderId;
        //주문이 처리가 되어 기존 장바구니는 비워준다.
    }

}

```

▼ etc.

equals() & hashCode()

[Java] equals와 hashCode 함수

1. equals와 hashCode란? equals와 hashCode는 모든 Java 객체의 부모 객체인 Object 클래스에 정의되어 있다. 그렇기 때문에 Java의 모든 객체는 Object 클래스에 정의된 equals와 hashCode 함수를 상속받고 있다. [equals()란?] boolean equals(Object obj)로 정의된 equals 메소드는 기본적으로 2개의 객체가 동일한지 검사

👉 <https://mangkyu.tistory.com/101>



Synchronized

Java의 동기화 Synchronized 개념 정리#1

Java의 동기화 -Synchronized 키워드의 사용Java를 프로그래밍 하다면 multi-thread로 인하여 동기화를 제어해야 하는 경우가 생깁니다. 그래서 흔히 Synchronized 키워드를 사용하는데요. 그냥 multi-thread로 동시접근되는 것을 막는다! 라는 개념보다는 여러 case를 살펴보면서 좀더 디테일하게 보려 합니다. Thread는 class의 멤버변수

👉 <https://tourspace.tistory.com/54>

