

[주요 파일]

메인 페이지

>index.jsp

사용자가 처음 보게 되는 메인 페이지로, login 페이지로 넘어갈지 help 페이지로 넘어갈지를 결정하게 된다. Home 서블릿을 호출하는 url 에 action 값을 넣어 전달한다.

Model

>Card.java

cardType 과 cardNumber 를 필드로 갖는 Card 클래스이다. Lombok 을 사용해 어노테이션으로 Getter, Setter, AllArgsConstructor 를 구성하였다.

>Customer.java

Id 와 password 를 필드로 갖는 Card 클래스이다. Lombok 을 사용해 어노테이션으로 Getter, Setter, AllArgsConstructor 를 구성하였다.

>Order.java

cardNum, cardType, price, midInitial, itemNum, description, address, firstName, lastName 를 필드로 갖는 Order 클래스이다. Lombok 을 사용해 어노테이션으로 Getter, Setter, AllArgsConstructor 를 구성하였다.

View

>loginform.jsp

사용자가 ID 와 password 를 입력하여 로그인할 수 있도록 한 로그인 페이지이다.

>help.jsp

사용자가 메인 페이지에서 go to help page 항목 클릭 시 넘어가는 페이지이다.

>form.jsp

사용자가 로그인 성공 시 넘어오게 되는 주문서 작성 페이지이다. 상품 정보, 주문자 이름, 주소, 카드 번호 등을 입력하게 된다. 이 때 카드 종류(Visa, mastercard 등)는 사용자 편의를 위해 라디오 버튼으로 구현하였다.

>cardError.jsp

사용자가 form.jsp 에서 카드 번호 입력 시 발생하는 다음의 경우

1. 카드 타입과 카드 번호가 일치하지 않는 경우
2. 최초 입력 카드 번호와 재입력 카드 번호가 일치하지 않는 경우
3. 카드 번호를 입력하지 않은 경우

넘어오게 되는 페이지이다. try again 클릭으로 form.jsp 페이지로 넘어가 주문서를 재입력하도록 하였다.

>order.jsp

form.jsp 에서 입력한 정보를 확인하는 페이지이다. 이 때, 객체의 모든 property 는 expression language 로 접근하였다.

Controller

>Home.java

HttpServlet 을 상속받는 Home 서블릿은 doGet 함수 내에서 request 객체로 action 값을 받아 action 이 login 일 경우 loginform.jsp 를, 아니라면 help.jsp 로 포워딩한다.

>DoLogin.java

HttpServlet 을 상속받는 DoLogin 서블릿은 doGet 함수 내에서 request 객체로 loginform.jsp 에서 사용자가 입력한 customerId 와 password

값을 받는다. 이 customerId 와 password 를 CustomerService 의 findCustomer 함수의 파라미터로 주어 customerMap 에 저장된 Customer 객체를 리턴받아 customer 객체에 담아 request.setAttribute()으로 customer 속성의 값에 지정한다.

CustomerService 에서 리턴받은 customer 객체가 null 이 아니며-사용자가 입력한 customerId 가 존재하는 ID 이며-, 사용자가 입력한 password 가 customerId 와 일치할 경우 form.jsp 페이지로, 위의 조건을 하나라도 만족하지 못 할 경우 error.jsp 로 포워딩한다.

>DoOrder.java

HttpServlet 을 상속받는 DoOrder 서블릿은 doGet 함수에서 request 객체로 form.jsp 에서 사용자가 입력한 input 값들을 받는다. 이 값들을 order 객체를 만들어 저장하고, request.setAttribute()으로 order 속성의 값에 지정한다 cardType 과 cardNumber 값을 CardService 의 findCar()의 파라미터로 주어 cardMap 에 저장된 Card 객체를 리턴받아 함수 내에 생성한 card 객체에 담아 request.setAttribute()으로 card 속성의 값에 지정한다.

CardService에서 리턴 받은 card 객체가 null 이 아니며-사용자가 입력한 cardNumber 가 존재하는 카드번호이며-, 사용자가 최초 입력한 cardNumber 와 재입력한 cardNumber 가 일치함과 동시에, 사용자가 선택한 cardType 이 cardNumber 와 조합이 일치하는 cardType 일 경우 order.jsp 로 포워딩하여 주문 결과 확인을, 위의 조건을 하나라도 만족하지 못 할 경우 cardError.jsp 로 포워딩하여 주문서를 재입력 하도록 하였다.

Service

>CardService.java

생성자에서 card 객체를 생성해 addCard()로 cardMap 에 저장한다. addCard()는 cardNumber 와 card 객체를 cardMap.put()에 파라미터로 주어 맵에 저장되도록 구현하였다. findCard()는 파라미터로 받은 cardType 과 cardNumber 가 null 이 아닌 경우 cardMap.get(cardNumber)로 card 객체를, 그 외에는 null 을 리턴한다.

> CustomerService.java

생성자에서 customer 객체를 생성해 addCustomer()로 customerMap에 저장한다. addCustomer()는 id 와 customer 객체를 customerMap.put()에 파라미터로 주어 맵에 저장되도록 구현하였다. findCustomer()는 파라미터로 받은 id 와 password 가 null 이 아닌 경우 customerMap.get(id.toLowerCase())로 customer 객체를, 그 외에는 null 을 리턴한다. 이 때, customerMap.get()의 파라미터로 준 id 는 사용자가 id 를 대문자로 입력할 경우를 고려하여 id.toLowerCase()로 소문자화 해주었다.

[주요코드]

메인 페이지

> index.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html> <head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body bgcolor="#f7ffe1"> <div align="center">
<h1>This is a homepage for shopping</h1> <br> <hr> <br>
<p> [<a href="/helloMVC/home?action=login"> go to login
page</a>]</p>
<p> [<a href="/helloMVC/home?action=help"> go to help
page</a> </p>
</div> </body>

</html>
```

Model

> Card.java

```
package model;
import lombok.AllArgsConstructor;
import lombok.Getter;
import lombok.Setter;

@Getter
@Setter
@AllArgsConstructor
public class Card {
    private String cardType;
    private String cardNumber;
}
```

> Customer.java

```
package model;
import lombok.AllArgsConstructor;
import lombok.Getter;
import lombok.Setter;

@Getter
@Setter
@AllArgsConstructor
public class Customer {
    private String id;
    private String password;
}
```

> Order.java

```
package model;

import lombok.AllArgsConstructor;
import lombok.Getter;
import lombok.Setter;

@Getter
@Setter
@AllArgsConstructor

public class Order {
    private String cardNum;
    private String cardType;
    private String price;
    private String midInitial;
    private String itemNum;
    private String description;
    private String address;
    private String firstName;
    private String lastName;
}
```

View

>loginform.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html> <head>
<link rel="stylesheet" href="onlineShop.css" type="text/css" media="screen"
/>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body bgcolor="#f7ffe1"> <div align="center">

Customer Id (id001, id002, id003, id004, id005) <br/>
Customer PW ( 0000, 1111, aaaa, bbbb, aa00) <br/> <br/> <br/>

<form action="/helloMVC/doLogin" method="get">
<table border="1">
    <tr> <th colspan="2">로그인 </th> </tr>
    <tr> <td>I D</td> <td><input type="text"
name="customerId"> </td> </tr>
    <tr> <td>Password</td> <td><input type="password"
name="password"> </td> </tr>
    <tr> <td colspan="2"><input type="submit"
value="Press"> </td> </tr>
</table>
</form>
</div> </body>
</html>
```

> fomr.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body bgcolor="#f7ffe1">
<center><h1> An Order Form </h1></center>
    <form action="/helloMVC/doOrder" method="get">
        Item Number : <input type="text"
name="itemNum"> <br/> <br/>
        Description : <input type="text"
name="description"> <br/> <br/>
        Price Each : <input type="text" name="eachPrice"> <br/> <br/>
        <hr> <br/>
        First Name : <input type="text" name="firstName"> <br/> <br/>
        Last Name : <input type="text" name="lastName"> <br/> <br/>
        Middle Initial : <input type="text" name="middleInitial"
placeholder="No Value"> <br/> <br/>
        <table>
            <tr>
                <td>Shipping Address : </td>
                <td><textarea name="shippingAddr" rows="4"
cols="25"> </textarea> </td>
            </tr>
```



```

        </table>
        <br/>
        Credit Card : <br/>
        <input type= "radio" name= "cardType" value= "Visa"
checked= "checked"> Visa <br/>
        <input type= "radio" name= "cardType" value= "MasterCard">
MasterCard <br/>
        <input type= "radio" name= "cardType" value= "American
Express"> American Express <br/>
        <input type= "radio" name= "cardType" value= "Discover">
Discover <br/>
        <input type= "radio" name= "cardType" value= "Java
SmartCard"> Java SmartCard <br/> <br/>
        Credit Card Number : <input type= "password"
name= "cardNum"> <br/> <br/>
        Repeat Credit Card Number : <input type= "password"
name= "reCardNum"> <br/> <br/> <br/>
        <table width= 500dp>
        <center>
        <tr> <td>
        <input type= "submit" value= "Submit Order" style="float:
right;">
        </td> </tr>
        </table>
    </form>

</body>
</html>

```

> order.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html>
<html> <head>
<link rel="stylesheet" href="onlineShop.css" type="text/css" media="screen" />
<meta charset="UTF-8">
<title>Insert title here</title>
</head>

<body bgcolor="#f7ffe1"> <div align="center">
<h1>The Order Sheet</h1> <br> <hr> <br> <br>
<table border="1">
    <tr> <th>Parameter Name</th> <th>Parameter Value(s)</th> </tr>
    <tr> <td>Card Number</td> <td>${order.cardNum}</td> </tr>
    <tr> <td>Card Type</td> <td>${order.cardType}</td> </tr>
    <tr> <td>Price</td> <td>${order.price}</td> </tr>
    <tr> <td>Item Number</td> <td>${order.itemNum}</td> </tr>
    <tr> <td>Description</td> <td>${order.description}</td> </tr>
    <tr> <td>Address</td> <td>${order.address}</td> </tr>
    <tr> <td>First Name</td> <td>${order.firstName}</td> </tr>
    <tr> <td>Last Name</td> <td>${order.lastName}</td> </tr>
    <tr> <td>Middle Initial</td> <td>${order.midInitial}</td> </tr>
</table> <br> <br>
<p> [<a href="/helloMVC/view/form.jsp"> go to order page again</a>]</p>
</div> </body>
</html>
```

> help.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body bgcolor="#f7ffe1"> <div align="center">

<h1>This is a help page.</h1> <br> <hr> <br>
<p> [<a href="/helloMVC/index.jsp"> go to main page</a>] </p>
</div> </body>
</html>
```

> error.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body bgcolor="#f7ffe1"> <div align="center">

    <h1>This is an error page</h1> <br> <hr> <br>
    <p> [<a href="/helloMVC/index.jsp"> try again</a>] </p>
</div> </body> </html>
```

> cardError.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body bgcolor="#f7ffe1"> <div align="center">

    <h1>Card Number Error</h1><br><hr><br>
    <p> [<a href="/helloMVC/view/form.jsp"> try again</a>] </p>

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

Controller

>DoLogin.java

```
package controller;

import java.io.IOException;
import java.util.List;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import model.Customer;
import service.CustomerService;

/**
 * Servlet implementation class DoLogin
 */
@WebServlet("/doLogin")
public class DoLogin extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public DoLogin() {
        super();
        // TODO Auto-generated constructor stub
    }
}
```

```

    /**
     * @see HttpServlet#doGet(HttpServletRequest request,
    HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {

        String customerId = request.getParameter("customerId");
        String customerPW = request.getParameter("password");

        // Perform business logic. Return a bean as a result.
        CustomerService service = new CustomerService();
        Customer customer = service.findCustomer(customerId,
customerPW);

        request.setAttribute("customer", customer);

        String page;

        if((customer != null) &&
customer.getPassword().equals(customerPW))
            page="/view/form.jsp";
        else
            page = "/view/error.jsp";

        RequestDispatcher dispatcher =
request.getRequestDispatcher(page);
        dispatcher.forward(request, response);
    }
}

```

>DoOrder.java

```
package controller;

import java.io.IOException;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import model.Card;
import model.Order;
import service.CardService;

/**
 * Servlet implementation class DoOrder
 */
@WebServlet("/doOrder")
public class DoOrder extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public DoOrder() {
        super();
        // TODO Auto-generated constructor stub
    }
}
```

```

/**
 * @see HttpServlet#doGet(HttpServletRequest request,
HttpServletResponse response)
 */
protected void doGet(HttpServletRequest request,
HttpServletResponse response) throws ServletException, IOException {

    String itemNum = request.getParameter("itemNum");
    String Description = request.getParameter("description");
    String price = request.getParameter("eachPrice");
    String firstName = request.getParameter("firstName");
    String lastName = request.getParameter("lastName");
    String midInitial = request.getParameter("middleInitial");
    String address = request.getParameter("shippingAddr");
    String cardType = request.getParameter("cardType");
    String cardNum = request.getParameter("cardNum");
    String reCardNum = request.getParameter("reCardNum");

    // Perform business logic. Return a bean as a result.
    CardService cardService = new CardService();
    Card card = cardService.findCard(cardType, cardNum);

    Order order =
new Order(cardNum, cardType, price, midInitial, itemNum, Description,
address, firstName, lastName);

    request.setAttribute("order", order);

```



```

        String page;
        if (card != null && cardNum.equals(reCardNum)
&& card.getCardNumber().equals(cardNum)
&& card.getCardType().equals(cardType)) {
            page="/view/order.jsp";
        }
        else
            page = "/view/cardError.jsp";

        RequestDispatcher dispatcher
= request.getRequestDispatcher(page);
        dispatcher.forward(request, response);
    }
}

```

> Home.java

```

package controller;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

```

```

/**
 * Servlet implementation class Home
 */
@WebServlet("/home")
public class Home extends HttpServlet {
    private static final long serialVersionUID = 1L;
    /** @see HttpServlet#HttpServlet() */
    public Home() {
        super();
        // TODO Auto-generated constructor stub
    }
    /**
     * @see HttpServlet#doGet(HttpServletRequest request,
    HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        String action = request.getParameter("action");
        String page = null;
        if(action.equals("login"))
            page = "/view/loginform.jsp";
        else
            page="/view/help.jsp";
        RequestDispatcher dispatcher
    = request.getRequestDispatcher(page);
        dispatcher.forward(request, response);
    }
}

```

Service

> CardService.java

```
package service;

import java.util.HashMap;
import java.util.Map;
import model.Card;

public class CardService {
    private Map<String, Card> cardMap; //key, value 로 쉽게 서치 가능한 자료구조

    public CardService() {

        cardMap = new HashMap<String, Card> ();

        addCard(new Card("Visa", "12345678") );
        addCard(new Card("MasterCard", "23456789") );
        addCard(new Card("American Express", "98765432") );
        addCard(new Card("Discover", "87654321") );
        addCard(new Card("Java SmartCard", "34567891") );
    }

    private void addCard(Card card) {

        cardMap.put(card.getCardNumber(), card);

    }
}
```

```
public Card findCard(String cardType, String cardNumber) {  
  
    if( cardType != null && cardNumber != null )  
        return (cardMap.get(cardNumber));  
    else  
        return null;  
  
}  
}
```

>CustomerService.java

```
package service;  
  
import java.util.ArrayList;  
import java.util.HashMap;  
import java.util.List;  
import java.util.Map;  
  
import model.Customer;  
  
public class CustomerService {  
  
    private Map<String, Customer> customerMap; //key, value 로 쉽게 서치  
    가능한 자료구조  
  
    public CustomerService() {  
  
        customerMap = new HashMap<String, Customer> ();  

```

```
        addCustomer(new Customer("id001", "0000") );
        addCustomer(new Customer("id002", "1111") );
        addCustomer(new Customer("id003", "aaaa") );
        addCustomer(new Customer("id004", "bbbb") );
        addCustomer(new Customer("id005", "aa00") );
    }

    private void addCustomer(Customer customer) {

        customerMap.put(customer.getId(), customer);

    }

    public Customer findCustomer(String id, String pw) {

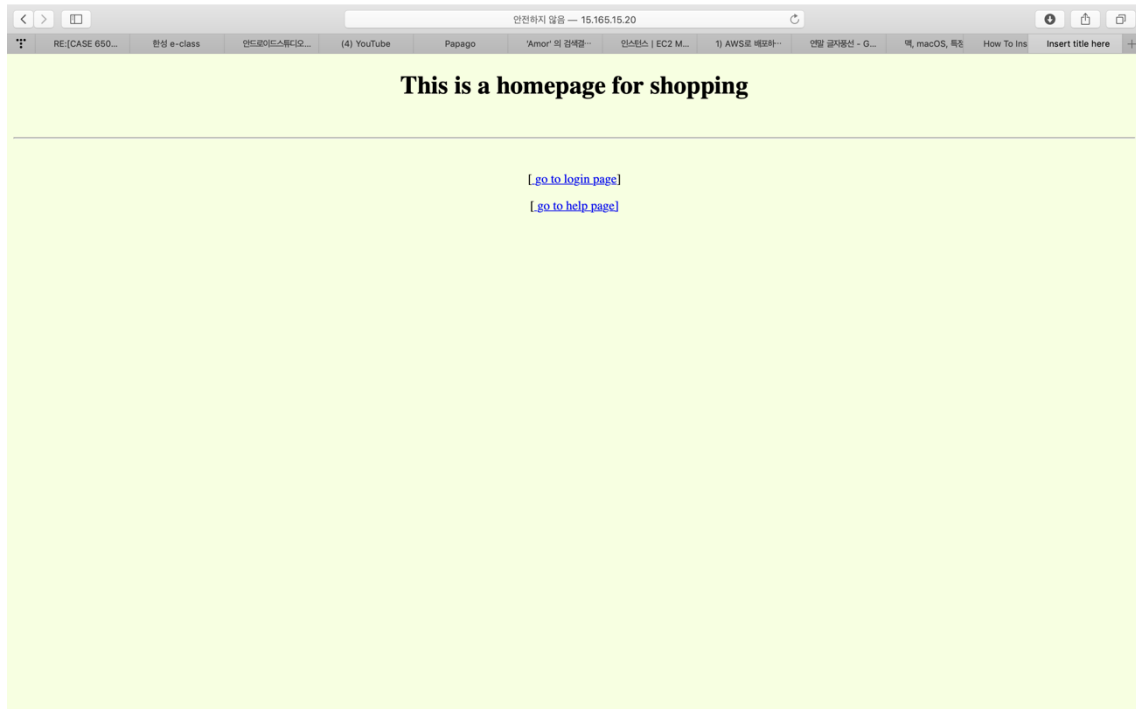
        if( id != null && pw != null )
            return (customerMap.get(id.toLowerCase()));
        else
            return null;

    }

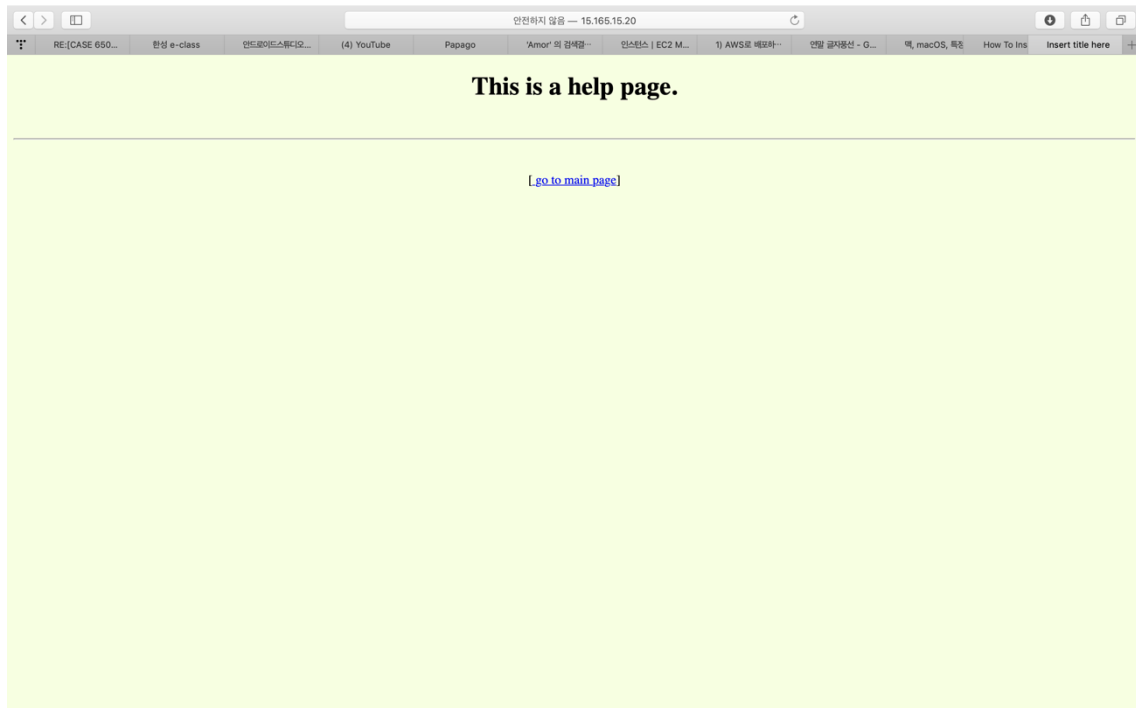
}
```

[결과 화면]

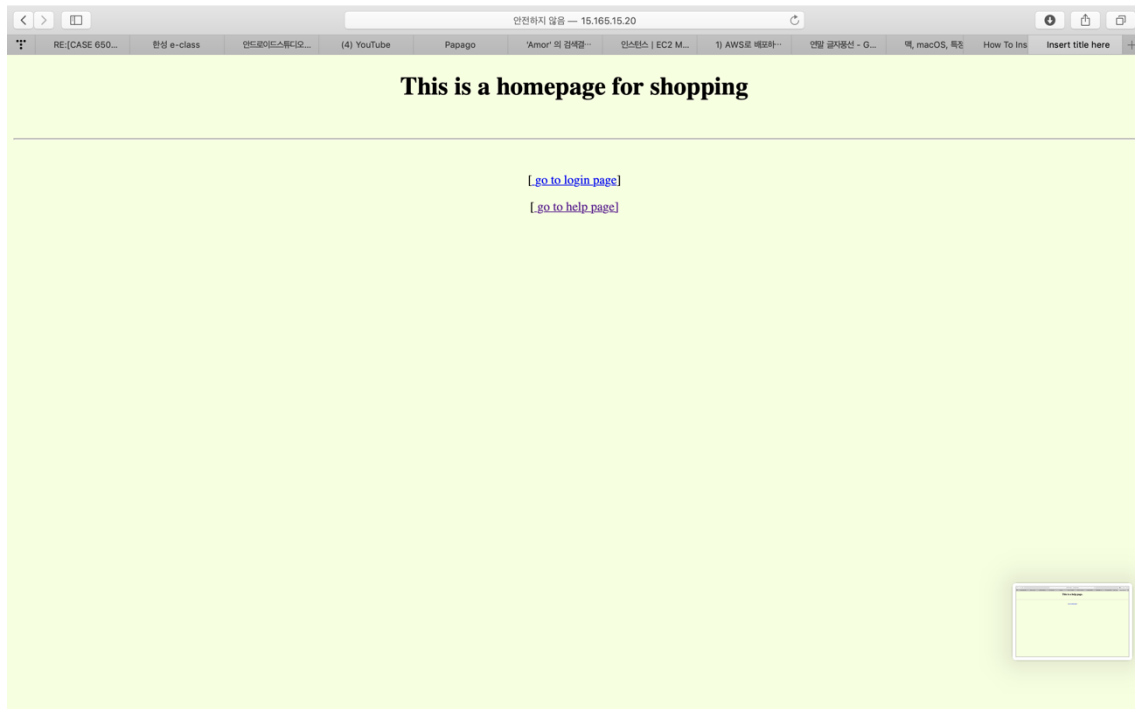
> 메인 화면



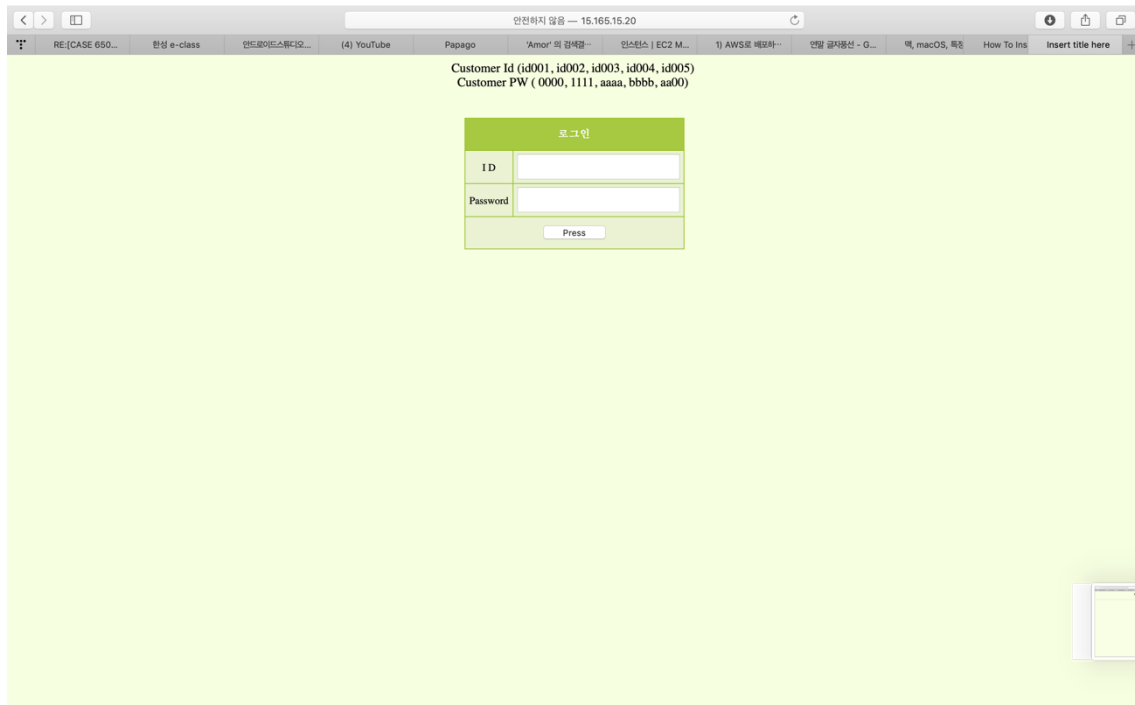
> 메인 화면에서 [go to help page] 클릭



>help page 에서 [go to main page] 클릭

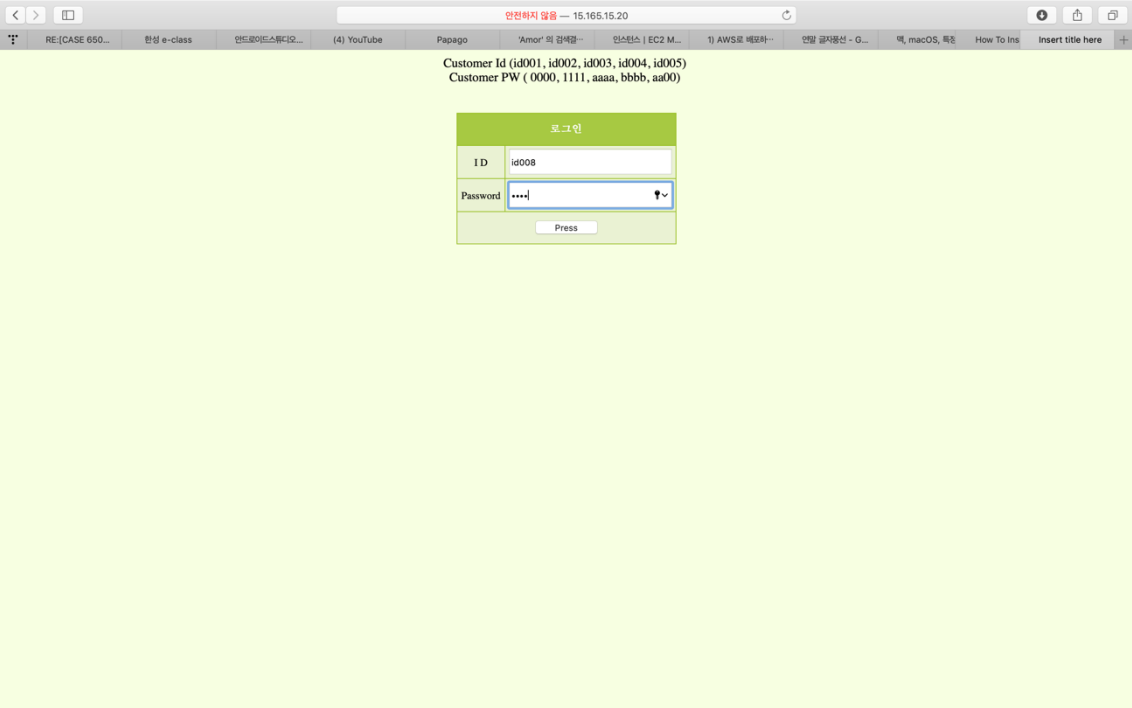


>메인 화면에서 [go to login page] 클릭

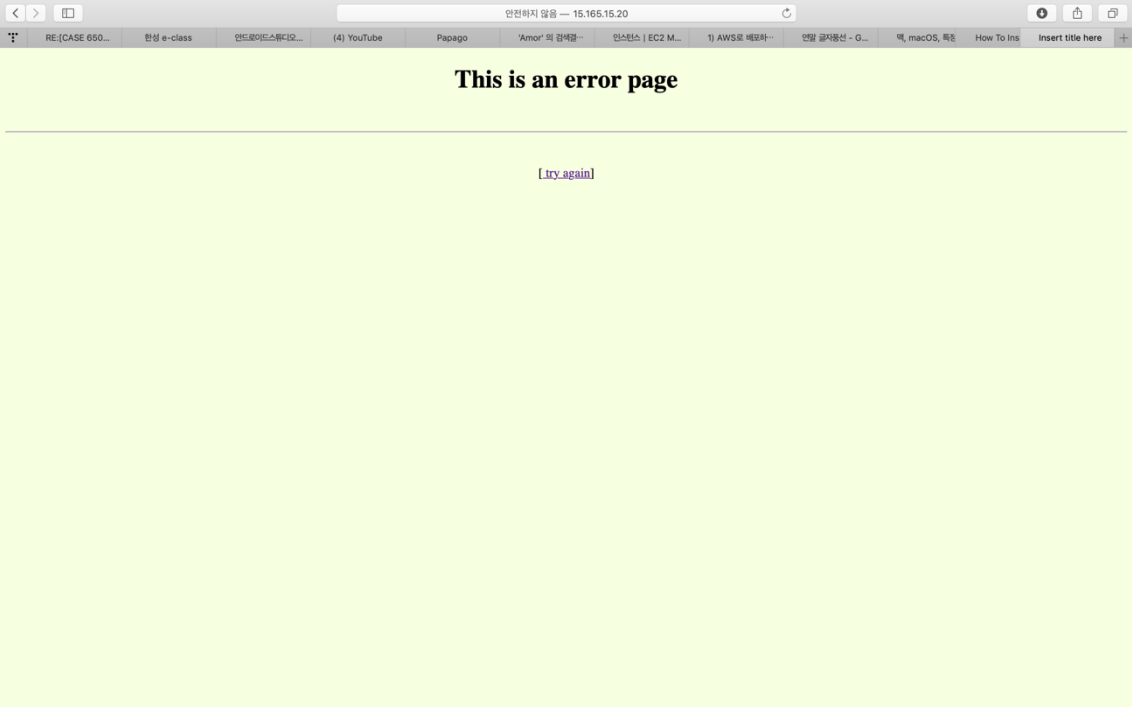


상단의 id,pw 목록은 테스트 편의를 위해 삽입한 것입니다.

>ID, Password 입력



>로그인 실패



>주문 페이지(로그인 성공)

안전하지 않음 — 15.165.15.20

RE:CASE 650... 한성 e-class 안드로이드스튜디오... (4) YouTube Papago 'Amor'의 감격감 인스턴스 | EC2 M... 1) AWS로 배포하... 연말 결재용선 - G... 맥, macOS, 특강 How To Ins Insert title here +

id003's Order Form

Item Number :

Description :

Price Each :

First Name :

Last Name :


Middle Initial :

Shipping Address :

Credit Card :
☒ Visa
☐ MasterCard
☐ American Express
☐ Discover
☐ Java SmartCard

Credit Card Number :

Repeat Credit Card Number :



>주문서 작성

안전하지 않음 — 15.165.15.20

RE:CASE 650... 한성 e-class 안드로이드스튜디오... (4) YouTube Papago 'Amor'의 감격감 인스턴스 | EC2 M... 1) AWS로 배포하... 연말 결재용선 - G... 맥, macOS, 특강 How To Ins Insert title here +

id003's Order Form

Item Number :

Description :

Price Each :

First Name :

Last Name :

Middle Initial :

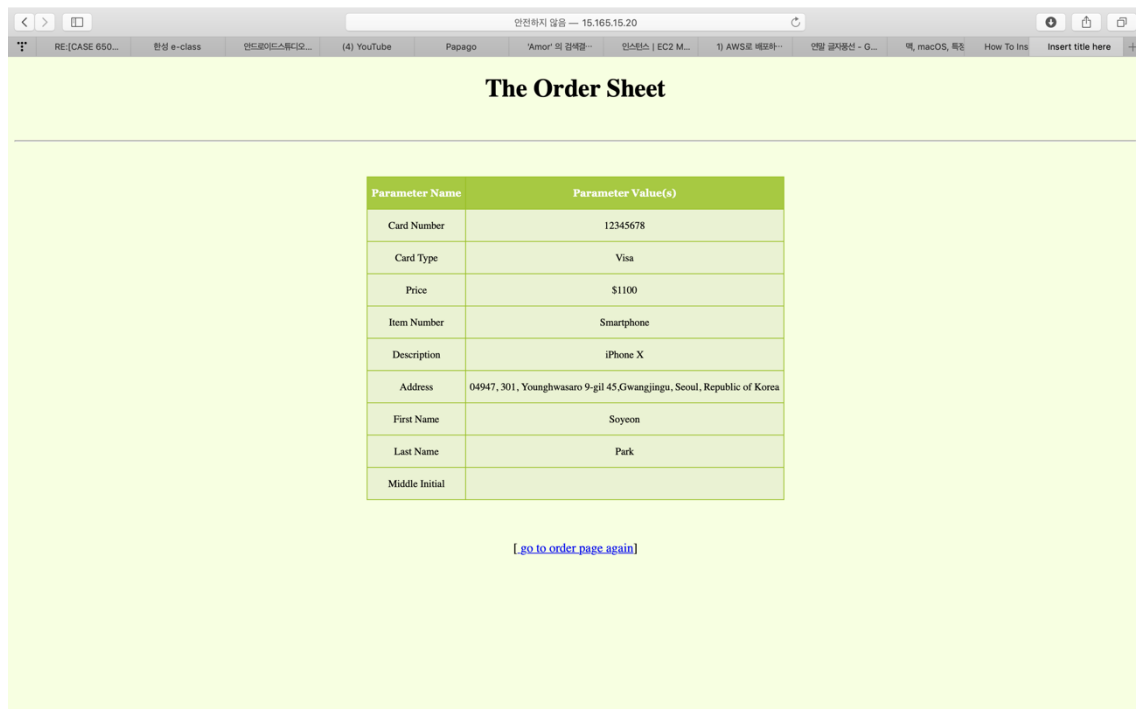
Shipping Address :

Credit Card :
☒ Visa
☐ MasterCard
☐ American Express
☐ Discover
☐ Java SmartCard

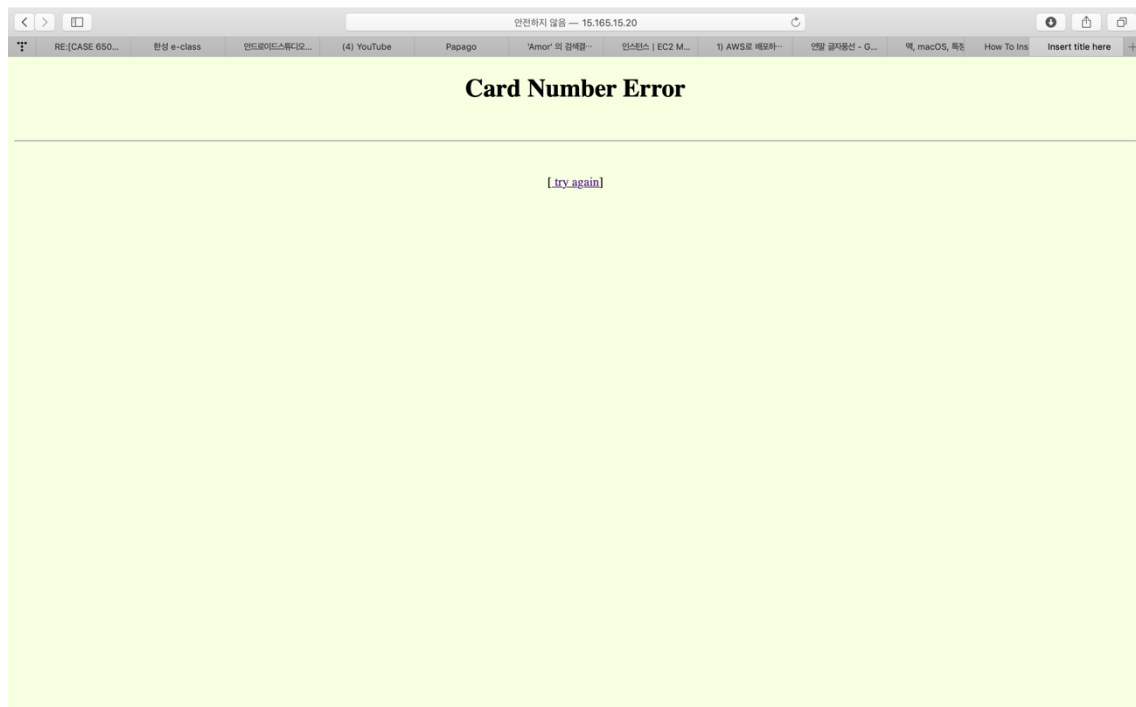
Credit Card Number :

Repeat Credit Card Number :

>주문확인 페이지(옳은 카드 번호 입력)



>카드 번호 최초 입력과 재입력 불일치 경우



-[\[try again\]](#) 클릭시 주문 페이지로 돌아감