통합 구현

2022/11/18 임민지

.]	u로젝	E	생성	및	구성
	.]	. 프로젝	. 프로젝트	. 프로젝트 생성	. 프로젝트 생성 및

• 2. 화면 구연

• 3. 기능 구현

• 4. 실행

목

ナト

문제 | . 프로젝트 생성 및 구성

```
▼ ② > Bookstore2 [JSP main]

  > 🛅 Deployment Descriptor: Bookstore2
  JAX-WS Web Services

✓ 四 > src/main/java

→ 

⊕ > controller.book

        > 🚂 ListController.java
        ModifyController.java
        > 📭 RegisterController.java
    > 🛂 ListController.java
      > ModifyController.java
      > 📭 RegisterController.java

✓ # > dao

      > 📭 BookDAO.java
      CustomerDAO.java

✓ ∰ > VO

      > 🛂 BookVO.java
      > 📭 CustomerVO.java
    > Maries
  > 📂 build

✓ ②

→ Src

    🕶 🔄 > main
      > 🔄 > java
      🗸 🔄 > book
             📑 list.jsp
             modify.jsp
             register.jsp
        📑 list.jsp
             modify.jsp
             register.jsp
        > 🚁 > META-INF

✓ ②

→ > WEB-INF

            🗁 lib
             index.jsp
```

문제2. 화면 구현

• 시작 페이지

```
← → C ① localhost:8080/Bookstore2/index.jsp
```

BookStore

도서목록 고객목록

문제2-1. 화면 구현

• 도서 목록

```
c%@ page contentType="text/html;charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
 chtml>
                              cmeta charset="UTF-8">
                              <title>Insert title here</title>
                </head>
            <body>
<h3>도서報목</h3>
                              ca href="#">村备으로</a>
                              <a href="#">도서毎異</a>
                              ctable border="1">
                                            도서변호
                                                           출판사
                                                          관리
                                             ctd>1c/td>
ctd>축구의 역사c/td>
                                                          くtd>テム포츠
                                                          7,000
                                                                        ca href="#">수정</a>
                                                                        ca href="#">公제</a>
                                                          ctro.
                                                          2
                                                          ctd>축구의이해c/td>
                                                          General Control Contro
                                                          22,000
                                                           <a href="#">수정</a>
                                                                        <a href="#">삭제</a>
                                                          (/tr>
                                                          3
                                                          >골프바이블
                                                          대한미디어

>35,000

                                                                       <a href="#">수정</a>
                                                                        <a href="#">삭제</a>
                                                           c/tr>
                              </body>
c/html>
```

도서목록

처음으로 도서등록

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7,000	<u>수정 삭제</u>
2	축구의이해	대한미디어	22,000	<u>수정 삭제</u>
3	골프바이블	대한미디어	35,000	<u>수정 삭제</u>

도서 목록

ListController

```
package controller.book;
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;
@WebServlet("/book/list.do")
public class ListController extends HttpServlet {
  private static final long serialVersionUID = 1L;
    public void init() throws ServletException {
    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
       RequestDispatcher dispatcher = req.getRequestDispatcher("/book/list.jsp");
       dispatcher.forward(req, resp);
    protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
```

← → C (i) localhost:8080/Bookstore2/book/list.do

도서목록

처음으로 도서등록

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7,000	<u>수정 삭제</u>
2	축구의이해	대한미디어	22,000	<u>수정 삭제</u>
3	골프바이블	대한미디어	35,000	<u>수정 삭제</u>

문제2-2. 화면 구현

• 도서 등록

```
:%@ page contentType="text/html;charset=UTF-8" pageEncoding="UTF-8"%>
: IDOCTYPE html>
chtml>
  <head>
     <meta charset="UTF-8">
     <title>Insert title here</title>
   chody>
     <h3>도서등록</h3>
     <a href="#">처음으로</a>
     <a href="#">도서목록</a>
     <form action="./registerProc.jsp">
        도서변호
              <input type="text" name="bookId" />
             도서명
              <input type="text" name="bookName" />
           (tr)
              >출판사
              <input type="text" name="publisher" />
           (tr>
              가격
              <input type="text" name="price" />
              <input type="submit" value="5#"/>
        </form>
  </body>
:/html>
```

도서등록				
처음으로 .	<u>도서목록</u>			
도서번호				
도서명				
출판사				
가격				
	등록			

도서 등록

RegisterController

package controller.book;
import java.io.IOException;
<pre>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.HttpServletRequest;</pre>
@WebServlet("/book/register.do")
public class RegisterController extends HttpServlet {
private static final long serialVersionUID = 11; (Override public void init() throws ServletException {
} @Override protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException { RequestDispatcher dispatcher = req.getRequestDispatcher("/book/register.jsp"); dispatcher.forward(req, resp); }
(Override protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
}

← → G	① localhost:8080/Bookstore2/book/register.do		
도서등록			
<u>처음으로</u>	<u>도서목록</u>		
도서번호			
도서명			
출판사			
가격			
	등록		

```
// 데이터 수신
String bookId = req.getParameter("bookId");
String bookName= req.getParameter("bookName");
String publisher = req.getParameter("publisher");
String price = req.getParameter("price");

BookVO vo = new BookVO();
vo.setBookId(bookId);
vo.setBookName(bookName);
vo.setPublisher(publisher);
vo.setPrice(price);

// DAO - insertBook() 호출
BookDAO.getInstance().insertBook(vo);

// 리다이렉트(도서목록으로...)
resp.sendRedirect("/Bookstore2/book/list.do");
}
```

도서 등록

도서등록				
<u>처음으로</u>				
도서번호	123			
도서명	sdf			
출판사	sgetryhty			
가격	777777			
	등록			

bookId	7	bookName	publisher	price
	1	축구의 역사	굿스포츠	7,000
	2	축구아는 여자	나무수	13,000
	3	축구의 이해	대한미디어	22,000
	4	골프 바이블	대한미디어	35,000
	5	피겨교본	굿스포츠	8,000
	6	역도 단계별기술	굿스포츠	6,000
	7	야구의 추억	이상미디어	20,000
	8	야구를 부탁해	이상미디어	13,000
	9	올림픽 이야기	삼성당	7,500
	10	Olympic Champions	Pearson	13,000
	11	iii	jijiji	1,200
	23	i⊣ê³¼ë°ëë	iai "	8,000
	34	sdfgdfsg	dsfgs	3,333
	123	sdf	sgetryhty	777,777
	555	i⊣ê³1⁄4	i-nê³1/4ì§	5,000
	777	sdfs	sdf	1,000
	999	etrtwr	wertrt	888,999

문제2-3. 화면 구현

• 도서 수정





ModifyController

pac	kage controller.book;
inp	ort java.io.IOException;
imp	ort javax.servlet.RequestDispatcher;
inp	ort javax.servlet.ServletException;
inp	ort javax.servlet.annotation.WebServlet;
inp	ort javax.servlet.http.HttpServlet;
inp	ort javax.servlet.http.HttpServletRequest;
inp	ort javax.servlet.http.HttpServletResponse;
01	2.70
	Servlet("/book/modify.do")
oub.	lic class ModifyController extends HttpServlet {
	private static final long serialVersionVID = 1L;
	60verride
	public void init() throws ServletException {
	}
	, @Override
	<pre>protected void doSet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IDException { RequestDispatcher dispatcher = req.getRequestDispatcher("/book/modify.jsp"); dispatcher.forward(req, resp);</pre>
	}
	@Override
	protected woid doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
	}
1	,

← → G	i localhost:8080/Bookstore2/book/modify.d
도서수정	
처음으로	<u>도서등록</u>
도서번호	
도서명	
출판사	
가격	
	수정

BOOKDAO & BOOKVO

```
import java.atil.Arraytist;
import java.atil.List;
Import vo.Benist;
public class SockS40 extends SSMelper (
      private static SockSAD (nature + new SockSAD();
     public static SockGAD getInstance() (
            return (natonce:
      private ScokDAD() ()
       public wold insertSock(Bock+D +c) {
            try (
com * getConnection();
                  pont * com.preper@istement("incert into "book" values (7,7,2,7)");
pont.setInt(1, vs.pstBookIn());
                 post.setitring(2, vo.getBookWame());
post.setitring(3, vo.getPublisher());
post.setlet(4, vo.getPrice());
                    past.esecuteOpdate();
                  close();
            )catch (Esception a) {
                  *.printStackfrace();
      public SouthO selectfook(String book): {
        try {
    see = petiterection();
    post = zee, preparefatament("select * from bank" share book)d = ["];
    post = zee(petite(), look)d)
    rs = post_sec(stiplary());
                pr(rs.cent()) {
    ve = res funkt();
    ve.extfunkt(rs.get2re(1));
    ve.extfunkt(re.get2re(1));
    ve.extfunkt(rs.get2re(rg(2));
    ve.extfunkt(rs.get2re(rg(2));
    ve.extfunkt(rs.get2re(rs);
                  elevedly.
       making the consistent and action and it
```

```
pwblse List(BookVUP selectBooks() {
              List(BookWD) books a new ArreyList(3();
              try {
    con = get(enection[];
    stat = con.createStatement();
    rs = stat.executeQuery["select * from "book"];
}
                             while(rs.next()) {
   BookVO = new BookVO();
   vs.setDockId(rs.getLnt(1));
   vs.setDockId(rs.getLnt(1));
   vs.setDockId(rs.getString(2));
   vs.setPolisher(rs.getString(3));
   vs.setPrice(rs.getInt(4));
   books add(sp);
                               close();
              |catch (Exception =) {
    o.printStackTrace();
              return books;
public void updateBook(BookVO vo) (
         white vote updration([notaty vo) {

two {

ground production(] }

con {

ground | norm_properfortered("quide "look" set "look(less"=), "publisher"=), "prije"=) where "look(less"=))

put._estSring(), vo._estProdistan());

put._estSring(), vo._estProdistan());

put._estTrin(4, vo._estProdist());

put._estTrin(4, vo._estProdist());

put._estTrin(4, vo._estProdist());

close(),

close(),

put._estTrin(4, vo._estProdist());

close(),

clos
                               e.printStackTrace();
 public void deleteBook(String bookId) (
                                 com = getConnection();
                                 psmt = conn.prepareStatement("delete from 'book' where 'bookId'=?");
                                 psat.setString(1, bookId);
                                 psmt.executeUpdate();
                                   close();
                    }catch (Exception e) {
                                 e.printStackTrace();
```

package vo; public class BookVO { private int bookId; private String bookName; private String publisher; private int price; public int getBookId() { return bookId; public void setBookId(int bookId) { this.bookId = bookId; public void setBookId(String bookId) { this.bookId = Integer.parseInt(bookId); public String getBookName() { return bookName; public void setBookName(String bookName) { this.bookName = bookName; public String getPublisher() { return publisher; public void setPublisher(String publisher) { this.publisher = publisher; public int getPrice() { return price; public void setPrice(int price) { this.price = price; public void setprice(String price) { this.price = Integer.parseInt(price);

문제2-4. 화면 구현

• 고객 목록

```
(No page contentType="text/html;chorset=UTF-8" pagetncoding="UTF-8"%> (IDCCTYPE html)
        chete charact="UTF-8">
        <title>Insert title here(/title>
       ca href="#">科格巴里</a>
ca href="#">之特别等</a>
        ctable bonder="1">
               (th) 2 年世立(/th)
(th) 2 年世立(/th)
(th) 2 年世(/th)
(th) 中立(/th)
(th) 存储を(/th)
                 ethorals/ths
                (td)010-1234-1001(/td)
                    (a href="#">中形c/a>
                     ca hrefe"#">NRc/a>
                (10)2(/10)
                ctd>집영약
                ctd)대한민국 서울
                (td)010-1234-1002(/td)
                   ca href="#">中間c/a>
                    ca href="#">48c/a>
             c/tra
                さtd>を可否く/td>
                대한민국 광범도
                010-1234-1003
                    ca heef="#">+Bc/a>
                    ca href="#">4/4c/a>
        c/table>
4/html)
```

고객목록

<u>처음으로 고객등록</u>

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스터	010-1234-1001	<u>수정 삭제</u>
2	김연아	대한민국 서울	010-1234-1002	<u>수정 삭제</u>
3	장미란	대한민국 강원도	010-1234-1003	<u>수정 삭제</u>

ListController

```
package controller.customer;
import java.io.IDException;

@webServlet("/customer/list.do")
public class listController extends HttpServlet {
    private static final long serialVersionUID = 1L;

@Override
    public void init() throws ServletException {
    }

@Override
protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IDException {
        RequestDispatcher dispatcher = req.getRequestDispatcher("/customer/list.jsp");
        dispatcher.forward(req, resp);
    }

@Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IDException {
    }
}
```

문제2-5. 화면 구현

• 고객 등록

<%@ page contentType="text/html;charset=UTF-8" pageEncoding="UTF-8"%> <!DOCTYPE html> <html> <meta charset="UTF-8"> <title>고객등록</title> <body> <h3>고객등록</h3> 처음으로 고객목록 <input type="text" name="custId" /> > 주소 <input type="text" name="name" /> 휴대폰 <input type="text" name="address" /> </form> </body> </html> </html>

RegisterController

```
// 데이터 수신
String custId = req.getParameter("custId");
String name= req.getParameter("name");
String address = req.getParameter("addrress");
String phone = req.getParameter("phone");

CustomerVO vo = new CustomerVO();
vo.setCustId(custId);
vo.setName(name);
vo.setAddress(address);
vo.setPhone(phone);

// DAO - insertBook() 호출
CustomerDAO.getInstance().insertCustomer(vo);

// 리타이렉트(도서목록으로...)
resp.sendRedirect("/Bookstore2/customer/list.do");
}
```

문제2-6. 화면 구현

• 고객 수정

</html>



고객수정	고객수정				
<u>처음으로</u>	처음으로 고객등록				
고객번호					
고객명					
주소					
휴대폰					
	수정				

ModifyController package controller.customer;

```
import java.io.IOException;[]
@WebServlet("/customer/modify.do")
public class ModifyController extends HttpServlet{
    private static final long serialVersionUID = 1L;
    @Override
    public void init() throws ServletException {
    }
    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
        RequestDispatcher dispatcher = req.getRequestDispatcher("/customer/modify.jsp");
        dispatcher.forward(req, resp);
    }
    @Override
    protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    }
}
```

CUSTOMERDAO&CUSTOMERVO

```
import java.util.ArrayList;
public class CustomerOAO extends D8Helper (
    private static CustomerDAO instance - new CustomerDAO();
public static CustomerDAO getInstance() {
    return instance;
    private CustomerDAO() ()
    // Pie CRUD
public void insertCustomer(CustomerVO vo) {
       close();
        )catch (Exception e) (
    e.printStackTrace();
    public CustomerVO selectCustomer(String custId) {
        CustomerVO vo - mull;
           cosm - get(onnection();
           psat = conn.prepareStatement("select * from 'customer' where 'custid'-?");
           psat.setStrine(1, custId);
           rs - psmt.executeQuery();
           if(rs.next()) {
               vo - new CustomerVO();
              vo.set(ustld(rs.getlnt(1));
               vo.setName(rs.getString(2));
              vo.setAddress(rs.getString(3));
vo.setPhone(rs.getString(4));
           close();
       }catch (Exception e) {
           e.printStackTrace();
       return vo:
    public List(CustomerVO> selectcustomers() (
       ListoCustomerWO> customers = new ArrayListo();
       try (
conn = get(onnection();
           stat = conn.createStatement():
           rs = stat_executeQuery("select " from 'customer'");
           while(rs.next()) (
CustomerVO vo = new CustomerVO();
```

```
vo.setName(rs.getString(2));
vo.setAddress(rs.getMString(3));
vo.setPhone(rs.getString(4));
customers.add(vo);
     return customers;
public void updateCustomer(CustomerVO vo) {
     try {
    conn = getConnection();
         com = getConnectLon();
pust = com.specificate("update 'customen' set "Name"-?, 'address'-?, 'phone'-? where 'custif'-?");
pust.setString(1, vo.getAndress());
pust.setString(1, vo.getAndress());
pust.setString(1, vo.getAndress());
pust.setString(1, vo.getAndress());
          pswt.executeUpdate();
     close();
]catch (Exception e) {
e.printStackTrace();
public void deleteCustomer(String custId) {
         psmt = conn.prepareStatement("d
psmt.setString(1, custId);
psmt.executeUpdate();
       ]catch (Exception e) {
           e.printStackTrace();
public void deleteCustomer(String custId) {
           conn = getConnection();
           psmt = conn.prepareStatement("delete from 'customer' where 'custId'=?");
           psmt.setString(1, custId);
            psmt.executeUpdate();
           close();
       ]catch (Exception e) {
           e.printStackTrace();
```

```
package vo;
public class CustomerVO {
    private int custId;
    private String name;
    private String address;
    private String phone;
    public int getCustId() {
       return custId;
    public void setCustId(int custId) {
       this.custId = custId;
    public void setCustId(String custId) {
       this.custId = Integer.parseInt(custId);
    public String getName() {
       return name;
    public void setName(String name) {
       this.name = name;
    public String getAddress() {
       return address;
    public void setAddress(String address) {
       this.address = address;
    public String getPhone() {
       return phone;
    public void setPhone(String phone) {
       this.phone = phone;
```