[ 11 ] MVC 게시판

BDeleteService

BWriteService

BListService

BmodifyReplyViewService

BReplyService

BContentService

BModifyService

1. 프로젝트 설계

로직실행

Dispatcher

BController

클라이언트

요청 지시

DB

DAO

DTO

content.jsp

list.jsp

w.jsp

reply.jsp

write.jsp

w.jsp

modify.jsp

**적절한 응답을 뿌려준다**

1. DB(DataBase) 생성

**-- TABLE & SEQUENCE DROP**

**DROP TABLE MVC\_BOARD;**

**DROP SEQUENCE MVC\_BOARD\_SEQ;**

**-- TABLE & SEQUENCE CREATE**

**CREATE TABLE MVC\_BOARD(**

**bID NUMBER(6),**

**bNAME VARCHAR2(50) NOT NULL,**

**bTITLE VARCHAR2(100) NOT NULL,**

**bCONTENT VARCHAR2(1000),**

**bDATE DATE DEFAULT SYSDATE NOT NULL, -- 작성일**

**bHIT NUMBER(6) DEFAULT 0 NOT NULL, -- 조회수**

**bGROUP NUMBER(6) NOT NULL, -- 원글이면 BID와 같고, 답변글의 경우 원글의 BGROUP과 같다**

**bSTEP NUMBER(3) NOT NULL, -- 같은 그룹내 출력 순서**

**bINDENT NUMBER(3) NOT NULL, -- 들여쓰기 정도**

**bIP VARCHAR2(20),**

**PRIMARY KEY(BID)**

**);**

**CREATE SEQUENCE MVC\_BOARD\_SEQ MAXVALUE 999999 NOCACHE NOCYCLE;**

**-- DUMMY DATA**

**SELECT \* FROM MVC\_BOARD;**

**INSERT INTO MVC\_BOARD (BID, BNAME, BTITLE, BCONTENT, BGROUP, BSTEP, BINDENT, BIP)**

**VALUES (MVC\_BOARD\_SEQ.NEXTVAL, '홍','제목', NULL, MVC\_BOARD\_SEQ.CURRVAL, 0,0,'192.1.1.1');**

**INSERT INTO MVC\_BOARD (BID, BNAME, BTITLE, BCONTENT, BGROUP, BSTEP, BINDENT, BIP)**

**VALUES (MVC\_BOARD\_SEQ.NEXTVAL, '김','비와', NULL, MVC\_BOARD\_SEQ.CURRVAL, 0,0,'192.1.1.2');**

**SELECT \* FROM MVC\_BOARD;**

**-- 2번글의 답글**

**INSERT INTO MVC\_BOARD (BID, BNAME, BTITLE, BCONTENT, BGROUP, BSTEP, BINDENT, BIP)**

**VALUES (MVC\_BOARD\_SEQ.NEXTVAL, '이','답','답', 2, 1, 1, '127.0.0.1');**

**-- DAO에 들어갈 QUERY**

**-- 1. 글목록(startRow ~ endRow까지)**

**SELECT \* FROM MVC\_BOARD ORDER BY BGROUP DESC, BSTEP;**

**SELECT \***

**FROM (SELECT ROWNUM RN, A.\***

**FROM (SELECT \* FROM MVC\_BOARD ORDER BY BGROUP DESC, BSTEP) A)**

**WHERE RN BETWEEN 2 AND 3;**

**-- 2. 전체 글 갯수**

**SELECT COUNT(\*) FROM MVC\_BOARD;**

**-- 3. 원글 쓰기**

**INSERT INTO MVC\_BOARD (BID, BNAME, BTITLE, BCONTENT, BGROUP, BSTEP, BINDENT, BIP)**

**VALUES (MVC\_BOARD\_SEQ.NEXTVAL, '홍','TITLE', NULL, MVC\_BOARD\_SEQ.CURRVAL, 0,0,'192.1.1.1');**

**-- 4. bID로 조회수 1 올리기**

**UPDATE MVC\_BOARD SET bHIT = bHIT + 1 WHERE bID=1;**

**SELECT \* FROM MVC\_BOARD;**

**-- 5. bID로 DTO가져오기 (글상세보기, 글수정VIEW, 답변글VIEW)**

**SELECT \* FROM MVC\_BOARD WHERE bID=1;**

**-- 6. 글 수정**

**UPDATE MVC\_BOARD**

**SET BNAME = '홍수정',**

**BTITLE = '제목바꿈',**

**BCONTENT = '본문바꿈',**

**BIP = '163.1.1.1'**

**WHERE BID=1;**

**-- 7. 글 삭제**

**DELETE FROM MVC\_BOARD WHERE bID=1;**

**SELECT \* FROM MVC\_BOARD;**

**-- 8. 답변글 저장전 작업(STEP ⓐ : 예시에서는 2번글의 답변)**

**UPDATE MVC\_BOARD SET BSTEP = BSTEP + 1**

**WHERE BGROUP = 2 AND BSTEP > 0;**

**-- 9. 답변글 쓰기 (예시는 2번의 답변)**

**SELECT \* FROM MVC\_BOARD WHERE bID=2;**

**INSERT INTO MVC\_BOARD (BID, BNAME, BTITLE, BCONTENT, BGROUP, BSTEP, BINDENT, BIP)**

**VALUES (MVC\_BOARD\_SEQ.NEXTVAL, '진','T', NULL, 2, 1, 1, '126.1.1.1');**

**COMMIT;**

**SELECT \* FROM MVC\_BOARD ORDER BY BGROUP DESC, BSTEP;**

1. DTO & DAO

**public** **class** BoardDto {

**private** **int** bid;

**private** String bname;

**private** String btitle;

**private** String bcontent;

**private** Timestamp bdate;

**private** **int** bhit;

**private** **int** bgroup;

**private** **int** bstep;

**private** **int** bindent;

**private** String bip;

}

**import** java.sql.Connection;

**import** java.sql.Date;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Timestamp;

**import** java.util.ArrayList;

**import** javax.naming.Context;

**import** javax.naming.InitialContext;

**import** javax.naming.NamingException;

**import** javax.sql.DataSource;

**import** com.lec.ch10.dto.BoardDto;

**public** **class** BoardDao {

**public** **static** **final** **int** ***FAIL*** = 0;

**public** **static** **final** **int** ***SUCCESS*** = 1;

**private** **static** BoardDao *instance* = **new** BoardDao();

**public** **static** BoardDao getInstance() {

**return** *instance*;

}

**private** BoardDao() {}

**private** Connection getConnection() **throws** SQLException {

Connection conn = **null**;

**try** {

Context ctx = **new** InitialContext();

DataSource ds = (DataSource)ctx.lookup("java:comp/env/jdbc/Oracle11g");

conn = ds.getConnection();

} **catch** (NamingException e) {

System.***out***.println(e.getMessage());

}

**return** conn;

}

// 글목록

**public** ArrayList<BoardDto> boardList(**int** startRow, **int** endRow){

ArrayList<BoardDto> dtos = **new** ArrayList<BoardDto>();

Connection conn = **null**;

PreparedStatement pstmt = **null**;

ResultSet rs = **null**;

String sql = "SELECT \* FROM (SELECT ROWNUM RN, A.\* " +

" FROM (SELECT \* FROM MVC\_BOARD ORDER BY BGROUP DESC, BSTEP) A)" +

" WHERE RN BETWEEN ? AND ?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setInt(1, startRow);

pstmt.setInt(2, endRow);

rs = pstmt.executeQuery();

**while**(rs.next()) {

**int** bid = rs.getInt("bid");

String bname = rs.getString("bname");

String btitle = rs.getString("btitle");

String bcontent = rs.getString("bcontent");

Timestamp bdate = rs.getTimestamp("bdate");

Date date = rs.getDate("bdate");

**int** bhit = rs.getInt("bhit");

**int** bgroup = rs.getInt("bgroup");

**int** bstep = rs.getInt("bstep");

**int** bindent = rs.getInt("bindent");

String bip = rs.getString("bip");

dtos.add(**new** BoardDto(bid, bname, btitle, bcontent, bdate, date, bhit, bgroup, bstep, bindent, bip));

}

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}**finally** {

**try** {

**if**(rs!=**null**) rs.close();

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** dtos;

}

// 글 갯수

**public** **int** getBoardTotCnt() {

**int** totCnt = 0;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

ResultSet rs = **null**;

String sql = "SELECT COUNT(\*) FROM MVC\_BOARD";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

rs = pstmt.executeQuery();

**if**(rs.next()) {

totCnt = rs.getInt(1);

}

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}**finally** {

**try** {

**if**(rs!=**null**) rs.close();

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** totCnt;

}

//원글쓰기

**public** **int** write(BoardDto bDto) {

**int** result = ***FAIL***;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

String sql = "INSERT INTO MVC\_BOARD "

+ "(bID, bNAME, bTITLE, bCONTENT, bGROUP, bSTEP, bINDENT, bIP)" +

" VALUES (MVC\_BOARD\_SEQ.NEXTVAL, ?, ?, ?, " +

" MVC\_BOARD\_SEQ.CURRVAL, 0,0, ?)";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setString(1, bDto.getBname());

pstmt.setString(2, bDto.getBtitle());

pstmt.setString(3, bDto.getBcontent());

pstmt.setString(4, bDto.getBip());

result = pstmt.executeUpdate();

System.***out***.println(result==***SUCCESS*** ? "글쓰기 성공":"글쓰기 실패 : " + bDto);

} **catch** (SQLException e) {

System.***out***.println("글쓰기 실패 예외 : "+e.getMessage());

}**finally** {

**try** {

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** result;

}

// hit수 올리기

**private** **void** hitUp(**int** bid) {

Connection conn = **null**;

PreparedStatement pstmt = **null**;

String sql = "UPDATE MVC\_BOARD SET BHIT = BHIT+1 WHERE BID=?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setInt(1, bid);

pstmt.executeUpdate();

} **catch** (SQLException e) {

System.***out***.println("조회수 올리기 예외 : "+e.getMessage());

}**finally** {

**try** {

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

}

// 글 상세보기(글 상세볼 때 조회수도 올림)

**public** BoardDto contentView(**int** bid) {

hitUp(bid);

BoardDto bDto = **null**;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

ResultSet rs = **null**;

String sql = "SELECT \* FROM MVC\_BOARD WHERE BID=?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setInt(1, bid);

rs = pstmt.executeQuery();

**if**(rs.next()) {

String bname = rs.getString("bname");

String btitle = rs.getString("btitle");

String bcontent = rs.getString("bcontent");

Timestamp bdate = rs.getTimestamp("bdate");

Date date = rs.getDate("bdate");

**int** bhit = rs.getInt("bhit");

**int** bgroup = rs.getInt("bgroup");

**int** bstep = rs.getInt("bstep");

**int** bindent = rs.getInt("bindent");

String bip = rs.getString("bip");

bDto = **new** BoardDto(bid, bname, btitle, bcontent, bdate, date, bhit, bgroup, bstep, bindent, bip);

}

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}**finally** {

**try** {

**if**(rs!=**null**) rs.close();

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** bDto;

}

// bid로 dto 가져오기 (답글쓰기 + 수정)

**public** BoardDto modifyView\_replyView(**int** bid) {

BoardDto bDto = **null**;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

ResultSet rs = **null**;

String sql = "SELECT \* FROM MVC\_BOARD WHERE BID=?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setInt(1, bid);

rs = pstmt.executeQuery();

**if**(rs.next()) {

String bname = rs.getString("bname");

String btitle = rs.getString("btitle");

String bcontent = rs.getString("bcontent");

Timestamp bdate= rs.getTimestamp("bdate");

Date date = rs.getDate("bdate");

**int** bhit = rs.getInt("bhit");

**int** bgroup = rs.getInt("bgroup");

**int** bstep = rs.getInt("bstep");

**int** bindent = rs.getInt("bindent");

String bip = rs.getString("bip");

bDto = **new** BoardDto(bid, bname, btitle, bcontent, bdate, date, bhit, bgroup, bstep, bindent, bip);

}

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}**finally** {

**try** {

**if**(rs!=**null**) rs.close();

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** bDto;

}

// 글 수정하기

**public** **int** modify(BoardDto bDto) {

**int** result = ***FAIL***;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

String sql = "UPDATE MVC\_BOARD SET BNAME=?, " +

" BTITLE = ?, " +

" BCONTENT = ?, " +

" BIP = ?" +

" WHERE BID=?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setString(1, bDto.getBname());

pstmt.setString(2, bDto.getBtitle());

pstmt.setString(3, bDto.getBcontent());

pstmt.setString(4, bDto.getBip());

pstmt.setInt(5, bDto.getBid());

result = pstmt.executeUpdate();

System.***out***.println(result==***SUCCESS*** ? "글수정 성공":"글수정 실패");

} **catch** (SQLException e) {

System.***out***.println("글수정 실패 예외 : "+e.getMessage());

}**finally** {

**try** {

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** result;

}

// 글 삭제

**public** **int** delete(**int** bid) {

**int** result = ***FAIL***;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

String sql = "DELETE FROM MVC\_BOARD WHERE BID=?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setInt(1, bid);

result = pstmt.executeUpdate();

System.***out***.println(result==***SUCCESS***? bid+"번 글삭제 성공":bid+"번글 삭제실패");

} **catch** (SQLException e) {

System.***out***.println("글삭제 예외 : "+e.getMessage());

}**finally** {

**try** {

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** result;

}

// 답변글 쓰기 전 작업 : stepA

**private** **void** preReplyStepA(**int** bgroup, **int** bstep) {

Connection conn = **null**;

PreparedStatement pstmt = **null**;

String sql = "UPDATE MVC\_BOARD SET BSTEP = BSTEP+1 WHERE BGROUP=? AND BSTEP>?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setInt(1, bgroup);

pstmt.setInt(2, bstep);

pstmt.executeUpdate();

} **catch** (SQLException e) {

System.***out***.println("답변글 쓰기 전 작업 예외 : "+e.getMessage());

}**finally** {

**try** {

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

}

// 답변글 쓰기

**public** **int** reply(BoardDto bDto) { // bname, btitle, bcontent, bip : 답변글에 대한 정보

// bgroup, bstep, bindent : 원글에 대한 정보

preReplyStepA(bDto.getBgroup(), bDto.getBstep());

**int** result = ***FAIL***;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

String sql = "INSERT INTO MVC\_BOARD" +

" (BID, BNAME, BTITLE, BCONTENT, BGROUP, BSTEP, BINDENT, BIP) " +

" VALUES (MVC\_BOARD\_SEQ.NEXTVAL, ?,?,?, ?, ?, ?, ?)";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setString(1, bDto.getBname());

pstmt.setString(2, bDto.getBtitle());

pstmt.setString(3, bDto.getBcontent());

pstmt.setInt(4, bDto.getBgroup());

pstmt.setInt(5, bDto.getBstep()+1);

pstmt.setInt(6, bDto.getBindent()+1);

pstmt.setString(7, bDto.getBip());

result = pstmt.executeUpdate();

System.***out***.println(result==***SUCCESS*** ? "답변쓰기 성공":"답변쓰기 실패");

} **catch** (SQLException e) {

System.***out***.println("답변쓰기 실패 예외 : "+e.getMessage());

}**finally** {

**try** {

**if**(pstmt!=**null**) pstmt.close();

**if**(conn!=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** result;

}

}

1. 프로젝트 생성
   1. MVC 프로젝트 생성

<!-- https://mvnrepository.com/artifact/com.oracle.database.jdbc/ojdbc6 -->

<dependency>

<groupId>com.oracle.database.jdbc</groupId>

<artifactId>ojdbc6</artifactId>

<version>11.2.0.4</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.projectlombok/lombok -->

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<version>1.18.26</version>

<scope>provided</scope>

</dependency>

1. 한글처리

<!-- Processes application requests -->

<servlet>

<servlet-name>appServlet</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

<init-param>

<param-name>contextConfigLocation</param-name>

<param-value>/WEB-INF/spring/appServlet/servlet-context.xml</param-value>

</init-param>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>appServlet</servlet-name>

<url-pattern>\*.do</url-pattern>

</servlet-mapping>

<!-- 한글처리 filter -->

<filter>

<filter-name>encodingFilter</filter-name>

<filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>

<init-param>

<param-name>encoding</param-name>

<param-value>UTF-8</param-value>

</init-param>

<init-param>

<param-name>forceEncoding</param-name>

<param-value>true</param-value>

</init-param>

</filter>

<filter-mapping>

<filter-name>encodingFilter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

1. 패키지, 인터페이스, 클래스 제작
   1. com.lec.board.service
      1. BService.java

**public** **interface** BService {

**public void** execute(Model model);

}

* + 1. BContentService.java
    2. BDeleteService.java
    3. BListService.java
    4. BModifyService.java
    5. BModifyReplyViewService.java
    6. BReplyService.java
    7. BWRiteService.java
  1. com.lec.board.controller
     1. BoardController.java
  2. com.lec.board.dao
     1. BoardDao.java
  3. com.lec.board.dto
     1. BoardDto.java

1. Controller 제작
   1. 클라이언트의 요청에 따른 전체적인 작업을 지휘하는 Controller를 만들어 본다.
   2. JSP의 MVC model2 방식을 기억하면서 작업을 진행한다.(Client → DispatcherServlet → Controller)
2. 리스트 페이지 만들기

private BService service;

@RequestMapping(value="list", method=RequestMethod.***GET***)

**public** String list(String pageNum, Model model) {

// mvcboard/list.do

model.addAttribute("pageNum", pageNum);

service = **new** BListService();

service.execute(model);

**return** "mvcboard/list";

}

@RequestMapping(value="list", method=RequestMethod.***POST***)

**public** String list\_post(String pageNum, Model model) {

// mvcboard/list.do?pageNum=2

model.addAttribute("pageNum", pageNum);

service = **new** BListService();

service.execute(model);

**return** "mvcboard/list";

} ↓

**public** **class** BListService **implements** BService {

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap(); // model을 Map화

// model.addAttribute("pageNum","1");

String pageNum = (String)map.get("pageNum");

**if**(pageNum == **null**) {

pageNum = "1";

}

**final** **int** PAGESIZE = 10, BLOCKSIZE = 10;

**int** currentPage = Integer.*parseInt*(pageNum);

**int** startRow = (currentPage-1) \* PAGESIZE + 1;

**int** endRow = startRow + PAGESIZE - 1;

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("boardList", boardDao.boardList(startRow, endRow));

**int** orderNum = startRow; // 출력될 순번호(1,2,3 순)

**int** totCnt = boardDao.getBoardTotCnt(); // 전체 글 갯수

**int** inverseNum = totCnt - startRow + 1; // 출력될 역순번호

**int** pageCnt = (**int**)Math.*ceil*((**double**)totCnt/PAGESIZE);

**int** startPage = ((currentPage - 1) / BLOCKSIZE) \* BLOCKSIZE + 1;

startPage = currentPage - (currentPage-1)%BLOCKSIZE;

**int** endPage = startPage + BLOCKSIZE - 1;

**if**(endPage > pageCnt) {

endPage = pageCnt;

}

model.addAttribute("orderNum", orderNum);

model.addAttribute("totCnt", totCnt);

model.addAttribute("inverseNum", inverseNum);

model.addAttribute("pageCnt", pageCnt);

model.addAttribute("startPage", startPage);

model.addAttribute("endPage", endPage);

model.addAttribute("BLOCKSIZE", BLOCKSIZE);

model.addAttribute("pageNum", currentPage);

}

}

↓

(list.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=*"fmt"* uri=*"http://java.sun.com/jsp/jstl/fmt"* %>

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"*/>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<link href=*"*${conPath }*/css/style.css"* rel=*"stylesheet"*>

<script src=*"https://code.jquery.com/jquery-3.6.0.js"*></script>

<script>

$(document).ready(**function**(){

/\* $('tr').click(function(){

var bid = Number($(this).children().eq(2).text());

if(!isNaN(bid)){

//alert(bid);

location.href = '${conPath}/mvcboard/content.do?bid='+bid+'&pageNum=${pageNum}';

}

}); \*/

});

**function** trClicked(bid){

location.href = '${conPath}/mvcboard/content.do?bid='+bid+'&pageNum=${pageNum}';

}

</script>

</head>

<body>

<c:set var=*"SUCCESS"* value=*"1"*/>

<c:set var=*"FAIL"* value=*"0"*/>

<c:if test=*"*${writeResult==SUCCESS }*"*>

<script>alert('글쓰기 성공');</script>

</c:if>

<c:if test=*"*${writeResult==FAIL }*"*>

<script>alert('글쓰기 실패');</script>

</c:if>

<c:if test=*"*${modifyResult==SUCCESS }*"*>

<script>alert('글수정 성공');</script>

</c:if>

<c:if test=*"*${modifyResult==FAIL }*"*>

<script>alert('글수정 실패');</script>

</c:if>

<c:if test=*"*${deleteResult==SUCCESS }*"*>

<script>alert('글삭제 성공');</script>

</c:if>

<c:if test=*"*${deleteResult==FAIL }*"*>

<script>alert('글삭제 실패');</script>

</c:if>

<c:if test=*"*${replyResult==SUCCESS }*"*>

<script>alert('답변달기 성공');</script>

</c:if>

<c:if test=*"*${replyResult==FAIL }*"*>

<script>alert('답변달기 실패');</script>

</c:if>

<table><tr><td><a href=*"*${conPath }*/mvcboard/write.do"*>글쓰기</a></td></tr></table>

<c:set var=*"ordNum"* value=*"*${orderNum }*"*/>

<c:set var=*"invNum"* value=*"*${inverseNum }*"*/>

<table>

<tr><th>순번</th><th>역순</th><th>글번호</th><th>글쓴이</th><th>글제목</th><th>작성일</th><th>조회수</th></tr>

<c:if test=*"*${totCnt == 0 }*"*>

<tr><td colspan=*"7"*>등록된 글이 없습니다</td></tr>

</c:if>

<c:if test=*"*${totCnt!=0 }*"*>

<c:forEach items=*"*${boardList }*"* var=*"bDto"*>

<tr onclick=*"trClicked(*${bDto.bid}*)"*>

<td>${ordNum }</td>

<td>${invNum }</td>

<td>${bDto.bid }</td>

<td>${bDto.bname }</td>

<td class=*"left"*>

<c:forEach var=*"i"* begin=*"1"* end=*"*${bDto.bindent }*"*>

<c:if test=*"*${i==bDto.bindent }*"*>

└─

</c:if>

<c:if test=*"*${i!=bDto.bindent }*"*>

&nbsp; &nbsp; &nbsp;

</c:if>

</c:forEach>

${bDto.btitle }

</td>

<td>

<fmt:formatDate value=*"*${bDto.bdate }*"* pattern=*"yy/MM/dd(E) hh:mm:ss(a)"*/><%-- <br>

<fmt:formatDate value="${bDto.date }" pattern="yy/MM/dd(E) hh:mm:ss(a)"/> --%>

</td>

<td><fmt:formatNumber value=*"*${bDto.bhit }*"* groupingUsed=*"true"*/></td>

</tr>

<c:set var=*"ordNum"* value=*"*${ordNum+1 }*"*/>

<c:set var=*"invNum"* value=*"*${invNum-1 }*"*/>

</c:forEach>

</c:if>

</table>

<div class=*"paging"*>

<c:if test=*"*${startPage > BLOCKSIZE}*"*>

[ <a href=*"*${conPath }*/mvcboard/list.do?pageNum=*${startPage-1}*"*> 이 전 </a>]

</c:if>

<c:forEach var=*"i"* begin=*"*${startPage }*"* end=*"*${endPage }*"*>

<c:if test=*"*${i == pageNum }*"*>

[ <b>${i }</b> ]

</c:if>

<c:if test=*"*${i != pageNum }*"*>

[ <a href=*"*${conPath }*/mvcboard/list.do?pageNum=*${i}*"*>${i }</a> ]

</c:if>

</c:forEach>

<c:if test=*"*${endPage<pageCnt }*"*>

[ <a href=*"*${conPath }*/mvcboard/list.do?pageNum=*${endPage+1}*"*> 다 음 </a>]

</c:if>

</div>

</body>

</html>

1. 글 작성 페이지 만들기(스프링의 JDBC 기법을 사용하지 않은 내용)

@RequestMapping(value="write", method=RequestMethod.***GET***)

**public** String write\_view() {

writable = true;

**return** "mvcboard/write";

}

(write.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=*"fmt"* uri=*"http://java.sun.com/jsp/jstl/fmt"* %>

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"*/>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<link href=*"*${conPath }*/css/style.css"* rel=*"stylesheet"*>

<script src=*"https://code.jquery.com/jquery-3.6.0.js"*></script>

<script>

$(document).ready(**function**(){

});

</script>

</head>

<body>

<form action=*"*${conPath }*/mvcboard/write.do"* method=*"post"*>

<table>

<caption>원글 쓰기</caption>

<tr><th>작성자</th>

<td><input type=*"text"* name=*"bname"* required=*"required"*></td>

</tr>

<tr><th>제목</th>

<td><input type=*"text"* name=*"btitle"* required=*"required"*></td>

</tr>

<tr><th>본문</th>

<td><textarea rows=*"5"* cols=*"20"* name=*"bcontent"*></textarea></td>

</tr>

<tr><td colspan=*"2"*>

<input type=*"submit"* value=*"글쓰기"* class=*"btn"*>

<input type=*"reset"* value=*"취소"* class=*"btn"*>

<input type=*"button"* value=*"목록"* class=*"btn"*

onclick="history.back()">

</td>

</tr>

</table>

</form>

</body>

</html>

**private** **boolean** writable;

@RequestMapping(value="write", method=RequestMethod.***POST***)

**public** String write(@ModelAttribute("bDto") BoardDto boardDto,

HttpServletRequest request, Model model) {

**if**(writable) {

model.addAttribute("request", request);

service = **new** BWriteService();

service.execute(model);

writable = false;

}

**return** "forward:list.do";

}

↓

**public** **class** BWriteService **implements** BService {

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap(); // model안의 attribute값을 추출하기 위해 map화

BoardDto bDto = (BoardDto)map.get("bDto"); // bname, btitle, bcontent

HttpServletRequest request = (HttpServletRequest)map.get("request");

bDto.setBip(request.getRemoteAddr());

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("writeResult", boardDao.write(bDto));

}

}

1. 글 내용 페이지 만들기

@RequestMapping(value="content", method=RequestMethod.***GET***)

**public** String content\_view(**int** bid, Model model) {

model.addAttribute("bid", bid);

service = **new** BContentService();

service.execute(model);

**return** "mvcboard/content";

} ↓

**public** **class** BContentService **implements** BService {

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap();

**int** bid = (Integer)map.get("bid");

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("bDto", boardDao.contentView(bid));

}

} ↓

(content.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=*"fmt"* uri=*"http://java.sun.com/jsp/jstl/fmt"* %>

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"*/>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<link href=*"*${conPath }*/css/style.css"* rel=*"stylesheet"*>

<script src=*"https://code.jquery.com/jquery-3.6.0.js"*></script>

<script>

$(document).ready(**function**(){

});

</script>

</head>

<body>

<table>

<caption>${bDto.bid }번 글 상세보기</caption>

<tr><th>작성자</th><td>${bDto.bname }</td></tr>

<tr><th>제 목</th><td>${bDto.btitle }</td></tr>

<tr><th>본 문</th><td><pre>${bDto.bcontent }</pre></td></tr>

<tr><th>작성일</th><td><fmt:formatDate value=*"*${bDto.bdate }*"* pattern=*"yy/MM/dd(E) hh:mm:ss(a)"*/>

<tr><th>조회수</th><td>${bDto.bhit }</td></tr>

<tr><th>ip</th><td>${bDto.bip }</td></tr>

<tr><td colspan=*"2"*>

<button onclick=*"location='*${conPath}*/mvcboard/modify.do?bid=*${bid }*&pageNum=*${param.pageNum }*'"*>수정</button>

<button onclick=*"location='*${conPath}*/mvcboard/delete.do?bid=*${bid }*&pageNum=*${param.pageNum }*'"*>삭제</button>

<button onclick=*"location='*${conPath}*/mvcboard/reply.do?bid=*${bid }*&pageNum=*${param.pageNum }*'"*>답변</button>

<button onclick=*"location='*${conPath}*/mvcboard/list.do?pageNum=*${param.pageNum }*'"*>목록</button>

</td></tr>

</table>

</body>

</html>

1. 글 수정 페이지 만들기

@RequestMapping(value="modify", method = RequestMethod.***GET***)

**public** String modify\_view(**int** bid, Model model) {

model.addAttribute("bid", bid);

service = **new** BModifyReplyViewService();

service.execute(model);

**return** "mvcboard/modify";

}

↓

**public** **class** BModifyReplyViewService **implements** BService {

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap();

**int** bid = (Integer)map.get("bid");

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("bDto", boardDao.modifyView\_replyView(bid));

}

} ↓

(modify.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=*"fmt"* uri=*"http://java.sun.com/jsp/jstl/fmt"* %>

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"*/>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<link href=*"*${conPath }*/css/style.css"* rel=*"stylesheet"*>

<script src=*"https://code.jquery.com/jquery-3.6.0.js"*></script>

<script>

$(document).ready(**function**(){

});

</script>

</head>

<body>

<form action=*"*${conPath }*/mvcboard/modify.do"* method=*"post"*>

<input type=*"hidden"* name=*"pageNum"* value=*"*${param.pageNum }*"*>

<table>

<tr><th>글번호</th>

<td><input type=*"text"* name=*"bid"* value=*"*${bDto.bid }*"* readonly=*"readonly"*></td>

</tr>

<tr><th>작성자</th>

<td><input type=*"text"* name=*"bname"* value=*"*${bDto.bname }*"* required=*"required"*></td>

</tr>

<tr><th>글제목</th>

<td><input type=*"text"* name=*"btitle"* value=*"*${bDto.btitle }*"* required=*"required"*></td>

</tr>

<tr><th>본문</th>

<td><textarea rows=*"5"* cols=*"32"* name=*"bcontent"*>${bDto.bcontent }</textarea></td>

</tr>

<tr><td colspan=*"2"*>

<input type=*"submit"* value=*"수정"* class=*"btn"*>

<input type=*"reset"* value=*"초기화"* class=*"btn"*>

<input type=*"button"* value=*"목록"* class=*"btn"*

onclick=*"location.href='*${conPath}*/mvcboard/list.do?pageNum=*${param.pageNum }*'"*>

</td></tr>

</table>

</form>

</body>

</html>

@RequestMapping(value="modify", method = RequestMethod.***POST***)

**public** String modify(@ModelAttribute("bDto") BoardDto bDto, HttpServletRequest request,

Model model) {

model.addAttribute("request", request);

service = **new** BModifyService();

service.execute(model);

**return** "forward:list.do";

} ↓

**public** **class** BModifyService **implements** BService{

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap();

HttpServletRequest request = (HttpServletRequest)map.get("request");

BoardDto bDto = (BoardDto)map.get("bDto");

bDto.setBip(request.getRemoteAddr());

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("modifyResult", boardDao.modify(bDto));

}

}

1. 글 삭제 페이지 만들기

@RequestMapping(value="delete", method = RequestMethod.***GET***)

**public** String delete(**int** bid, Model model) {

model.addAttribute("bid", bid);

service = **new** BDeleteService();

service.execute(model);

**return** "forward:list.do";

}

↓

**public** **class** BDeleteService **implements** BService {

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap();

**int** bid = (Integer)map.get("bid");

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("deleteResult", boardDao.delete(bid));

}

}

1. 글 답변 페이지 만들기

@RequestMapping(value="reply", method = RequestMethod.***GET***)

**public** String reply\_view(**int** bid, Model model) {

model.addAttribute("bid", bid);

service = **new** BModifyReplyViewService();

service.execute(model);

**return** "mvcboard/reply";

}

↓

**public** **class** BModifyReplyViewService **implements** BService {

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap();

**int** bid = (Integer)map.get("bid");

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("bDto", boardDao.modifyView\_replyView(bid));

}

}

↓

(reply.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=*"fmt"* uri=*"http://java.sun.com/jsp/jstl/fmt"* %>

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"*/>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<link href=*"*${conPath }*/css/style.css"* rel=*"stylesheet"*>

<script src=*"https://code.jquery.com/jquery-3.6.0.js"*></script>

<script>

$(document).ready(**function**(){

});

</script>

</head>

<body>

<!-- 넘어온 파라미터 : bid, pageNum -->

<!-- model에 add된 것 : dto(원글에 대한 dto), bid -->

<form action=*"*${conPath }*/mvvboard/reply.do"* method=*"post"*>

<input type=*"hidden"* name=*"pageNum"* value=*"*${param.pageNum }*"*>

<input type=*"hidden"* name=*"bgroup"* value=*"*${bDto.bgroup }*"*>

<input type=*"hidden"* name=*"bstep"* value=*"*${bDto.bstep }*"*>

<input type=*"hidden"* name=*"bindent"* value=*"*${bDto.bindent }*"*>

<table>

<tr><th>작성자</th><td><input type=*"text"* name=*"bname"* required=*"required"*></td></tr>

<tr><th>제 목</th><td><input type=*"text"* name=*"btitle"* value=*"[답]*${bDto.btitle.substring(0,1) }*"*

required=*"required"*></td></tr>

<tr><th>본 문</th><td><textarea rows=*"5"* cols=*"32"* name=*"bcontent"*></textarea></td></tr>

<tr><td colspan=*"2"*>

<input type=*"submit"* value=*"답글저장"* class=*"btn"*>

<input type=*"reset"* value=*"취소"* class=*"btn"*>

<input type=*"button"* value=*"이전"* onclick="history.go(-1);" class=*"btn"*>

<input type=*"button"* value=*"목록"* class=*"btn"*

onclick=*"location.href='*${conPath}*/mvvboard/list.do?pageNum=*${param.pageNum }*'"*>

</td></tr>

</table>

</form>

</body>

</html>

@RequestMapping(value="reply", method = RequestMethod.***POST***)

**public** String reply(BoardDto boardDto, HttpServletRequest request, Model model) {

model.addAttribute("request", request);

service = **new** BReplyService();

service.execute(model);

**return** "forward:list.do";

} ↓

**public** **class** BReplyService **implements** BService {

@Override

**public** **void** execute(Model model) {

Map<String, Object> map = model.asMap();

BoardDto boardDto = (BoardDto)map.get("boardDto");

HttpServletRequest request = (HttpServletRequest) map.get("request");

boardDto.setBip(request.getRemoteAddr());

BoardDao boardDao = BoardDao.*getInstance*();

model.addAttribute("replyResult", boardDao.reply(boardDto));

}

}

(index.jsp)

<body>

<jsp:forward page=*"mvcboard/list.do"*/>

</body>

(insert100.jsp)

<%@page import=*"com.lec.ch10.dto.BoardDto"*%>

<%@page import=*"com.lec.ch10.dao.BoardDao"*%>

<%@ 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=*"fmt"* uri=*"http://java.sun.com/jsp/jstl/fmt"* %>

<c:set var=*"conPath"* value=*"*${pageContext.request.contextPath }*"*/>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<link href=*"*${conPath }*/css/style.css"* rel=*"stylesheet"*>

<script src=*"https://code.jquery.com/jquery-3.6.0.js"*></script>

<script>

$(document).ready(**function**(){

});

</script>

</head>

<body>

<%

BoardDao boardDao = BoardDao.getInstance();

BoardDto dto = **new** BoardDto();

**for**(**int** i=0 ; i<110 ; i++){

**if**(i%3==1){

bDto.setBname("홍길동");

}**else** **if**(i%3==1){

bDto.setBname("이춘추");

}**else**{

bDto.setBname("마리오");

}

dto.setBtitle("제목 " + i);

dto.setBcontent("본문 내용 " + i);

dto.setBip("127.0.0.1");

**int** result = boardDao.write(dto);

System.out.println(i+"번째 : " + (result==1? "성공":"실패"));

}

response.sendRedirect("mvcboard/list.do");

%>

</body>

</html>

(style.css)

@CHARSET *"UTF-8"*;

**body** {

background-color: *#fff0B5*;

font-size: *12px*;

}

**table** {border: *2px* *solid* *pink*; width:*95%*; margin: *10px* *auto*;}

**table** **tr** { background-color: *#FFE271*; }

**table** **tr***:hover* { background-color: *orange*;

cursor: *pointer*;

}

**td,** **th** {text-align: *center*; padding:*5px*; }

**caption** {font-size: *25px*; padding:*10px*;}

**h2,** **b**{text-align: *center*; color:*red*;}

**a** { text-decoration: *none*; color:*black*}

*.left*{text-align: *left*;}

*.paging* {text-align: *center*;}

**input***:not*(*.btn*)**,** **textarea** {

width: *90%*;

}