

[0706]RE:답변글

글번호 그룹번호 그룹순서 제목탭(공백칸)

1 1 1 test

4 1 2 Ltest

5 1 3 Ltest

2 1 4 Ltest

3 1 5 Ltest

답변글들은 한칸씩 떨어져있다(계층글인것을 나타내기 위해서)

- answer.sql

```
DROP SEQUENCE BOARDNOSEQ;
DROP SEQUENCE GROUPNOSEQ;

CREATE SEQUENCE BOARDNOSEQ;
CREATE SEQUENCE GROUPNOSEQ;
DROP TABLE ANSWERBOARD;

--글번호 그룹번호 그룹순서 제목탭(공백칸) 제목 내용 작성자 작성일
CREATE TABLE ANSWERBOARD(
  BOARDNO NUMBER PRIMARY KEY,
  GROUPNO NUMBER NOT NULL,
  GROUPSEQ NUMBER NOT NULL,
  TITLETAB NUMBER NOT NULL,
  TITLE VARCHAR2(1000) NOT NULL,
  CONTENT VARCHAR2(4000) NOT NULL,
  WRITER VARCHAR2(500) NOT NULL,
  REGDATE DATE NOT NULL
);

SELECT BOARDNO, GROUPNO, GROUPSEQ, TITLETAB,
       TITLE, CONTENT, WRITER, REGDATE
FROM ANSWERBOARD
ORDER BY GROUPNO DESC, GROUPSEQ;
-- GROUPNO : DESC (내림차순), GROUPSEQ : SEC (오름차순)

--1번 글 작성
INSERT INTO ANSWERBOARD
VALUES (BOARDNOSEQ.NEXTVAL, GROUPNOSEQ.NEXTVAL, 1, 0, '첫번째 글 입니다', '첫번째 내용 입니다', '관리자', SYSDATE);

INSERT INTO ANSWERBOARD
VALUES (BOARDNOSEQ.NEXTVAL, GROUPNOSEQ.NEXTVAL, 1, 0, '두번째 글 입니다.', '두번째 내용 입니다', '아무나', SYSDATE);

--1번 글에 답변글 작성
-- 부모의 글과 같은 그룹에서 부모의 순서 다음순서에 있는 글들을 찾아서 순서+1 먼저 해줘야 한다.
UPDATE ANSWERBOARD
SET GROUPSEQ = GROUPSEQ +1
WHERE GROUPNO = (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO =1)
AND GROUPSEQ > (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO =1);

-- 답변글 추가 ( 부모의 글과 같은 그룹, 부모의 순서 +1, 부모의 제목 공백 +1 )
INSERT INTO ANSWERBOARD
VALUES (BOARDNOSEQ.NEXTVAL,
       (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO = 1),
       (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO = 1) +1 ,
       (SELECT TITLETAB +1 FROM ANSWERBOARD WHERE BOARDNO = 1),
       'LRE:첫번째 글 입니다.',
       '답변글 어렵다.',
       '홍길동',
       SYSDATE
);

-- 답변글 (3번글)에 답변 달자
-- UPDATE (부모와 같은 그룹, 부모 순서 +1)인 글들의 순서를 +1 하자
UPDATE ANSWERBOARD
SET GROUPSEQ = GROUPSEQ +1
WHERE GROUPNO = (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO = 3)
AND GROUPSEQ > (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO = 3);

-- INSERT : 부모와 같은 그룹, 부모 순서 +1, 부모 제목 공백 +1
INSERT INTO ANSWERBOARD
VALUES (BOARDNOSEQ.NEXTVAL,
       (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO = 3 ),
       (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO = 3 ) +1,
```

```

        (SELECT TITLETAB + 1 FROM ANSWERBOARD WHERE BOARDNO = 3 ),
        'ㄴRE:두번째 글 입니다',
        'ㅎㅎㅎㅎㅎ',
        '강여림',
        SYSDATE
    );

-- 첫번째 글에 ㄴ RE 달기
UPDATE ANSWERBOARD
SET GROUPSEQ = GROUPSEQ +1
WHERE GROUPNO = (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO = 1)
AND GROUPSEQ > (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO = 1);

INSERT INTO ANSWERBOARD
VALUES (
    BOARDNOSEQ.NEXTVAL,
    (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO = 1),
    (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO = 1) +1,
    (SELECT TITLETAB FROM ANSWERBOARD WHERE BOARDNO = 1) +1,
    'ㄴRE:이제 알겠다',
    '리얼?', '강여림', SYSDATE
);

```

- AnswerDto.java

```

package com.answer.dto;

import java.util.Date;

public class AnswerDto {
    private int boardno;
    private int groupno;
    private int groupseq;
    private int titletab;
    private String title;
    private String content;
    private String writer;
    private Date regdate;

    public AnswerDto() {
    }

    // selectone
    public AnswerDto(int titletab, String title, String content, String writer, Date regdate) {
        super();
        this.titletab = titletab;
        this.title = title;
        this.content = content;
        this.writer = writer;
        this.regdate = regdate;
    }

    // insert
    public AnswerDto(String title, String content, String writer) {
        super();
        this.title = title;
        this.content = content;
        this.writer = writer;
    }

    // update
    public AnswerDto(int boardno, String title, String content, String writer) {
        super();
        this.boardno = boardno;
        this.title = title;
        this.content = content;
        this.writer = writer;
    }

    public AnswerDto(int boardno, int groupno, int groupseq, int titletab, String title, String content, String writer,
        Date regdate) {
        super();
        this.boardno = boardno;
        this.groupno = groupno;
        this.groupseq = groupseq;
        this.titletab = titletab;
        this.title = title;
        this.content = content;
    }
}

```

```

        this.writer = writer;
        this.regdate = regdate;
    }

    public int getboardno() {
        return boardno;
    }

    public void setboardno(int boardno) {
        this.boardno = boardno;
    }

    public int getgroupno() {
        return groupno;
    }

    public void setgroupno(int groupno) {
        this.groupno = groupno;
    }

    public int getGroupSeq() {
        return groupseq;
    }

    public void setGroupSeq(int groupseq) {
        this.groupseq = groupseq;
    }

    public int gettitletab() {
        return titletab;
    }

    public void settitletab(int titletab) {
        this.titletab = titletab;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getContent() {
        return content;
    }

    public void setContent(String content) {
        this.content = content;
    }

    public String getWriter() {
        return writer;
    }

    public void setWriter(String writer) {
        this.writer = writer;
    }

    public Date getRegdate() {
        return regdate;
    }

    public void setRegdate(Date regdate) {
        this.regdate = regdate;
    }

    @Override
    public String toString() {
        return "AnswerDto [boardno=" + boardno + ", groupno=" + groupno + ", groupseq=" + groupseq + ", titletab="
            + titletab + ", title=" + title + ", content=" + content + ", writer=" + writer + ", regdate=" + regdate
            + "]\n";
    }
}

```

- AnswerDao.java

```
package com.answer.dao;
```

```

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;

import javax.naming.ldap.PagedResultsControl;

import com.answer.db.JDBCTemplate;
import com.answer.dto.AnswerDto;

public class AnswerDao extends JDBCTemplate{
    public List<AnswerDto> selectList(){
        Connection con = getConnection();
        PreparedStatement pstmt = null;
        ResultSet rs = null;
        List<AnswerDto> list = new ArrayList<AnswerDto>();
        String sql = " SELECT BOARDNO, GROUPNO, GROUPSEQ, TITLETAB, TITLE, CONTENT, WRITER, REGDATE FROM ANSWERBOARD ORDER BY GROUPNO";
        try {
            pstmt = con.prepareStatement(sql);
            rs = pstmt.executeQuery();
            while(rs.next()) {
                AnswerDto dto = new AnswerDto (
                    rs.getInt(1), rs.getInt(2), rs.getInt(3), rs.getInt(4), rs.getString(5), rs.getString(6), rs.getString(7), rs.getString(8)
                );

                list.add(dto);
            }
        } catch (SQLException e) {
            e.printStackTrace();
        } finally {
            close(rs);
            close(pstmt);
            close(con);
        }

        return list;
    }

    public AnswerDto selectOne(int boardno) {
        Connection con = getConnection();
        PreparedStatement pstmt = null;
        ResultSet rs = null;
        String sql = " SELECT BOARDNO, TITLETAB, TITLE, CONTENT, WRITER, REGDATE FROM ANSWERBOARD WHERE BOARDNO=? ";
        AnswerDto dto = new AnswerDto();
        try {
            pstmt = con.prepareStatement(sql);
            pstmt.setInt(1, boardno);
            rs = pstmt.executeQuery();
            while(rs.next()) {
                dto.setboardno(rs.getInt(1));
                dto.settitletab(rs.getInt(2));
                dto.setTitle(rs.getString(3));
                dto.setContent(rs.getString(4));
                dto.setWriter(rs.getString(5));
                dto.setRegdate(rs.getDate(6));
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }

        return dto;
    }

    public int insert(AnswerDto dto) {
        Connection con = getConnection();
        PreparedStatement pstmt = null;
        int res = 0;
        String sql = " INSERT INTO ANSWERBOARD VALUES (BOARDNOSEQ.NEXTVAL, GROUPNOSEQ.NEXTVAL, 1, 0, ?, ?, ?, SYSDATE ) ";
        try {
            pstmt = con.prepareStatement(sql);
            pstmt.setString(1, dto.getTitle());
            pstmt.setString(2, dto.getContent());
            pstmt.setString(3, dto.getWriter());
            res = pstmt.executeUpdate();
            if(res > 0) {
                commit(con);
            }
        } catch (SQLException e) {
            e.printStackTrace();
        } finally {
            close(pstmt);
            close(con);
        }

        return res;
    }
}

```

```

}

public int update(AnswerDto dto) {
    Connection con = getConnection();
    PreparedStatement pstmt = null;
    int res = 0;
    String sql = " UPDATE ANSWERBOARD SET TITLE=?, CONTENT=?, WRITER=? WHERE BOARDNO=? ";
    try {
        pstmt = con.prepareStatement(sql);
        pstmt.setString(1, dto.getTitle());
        pstmt.setString(2, dto.getContent());
        pstmt.setString(3, dto.getWriter());
        pstmt.setInt(4, dto.getboardno());
        res = pstmt.executeUpdate();
        if(res > 0) {
            commit(con);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        close(pstmt);
        close(con);
    }
    return res;
}

public int delete(int boardno) {
    Connection con = getConnection();
    PreparedStatement pstmt = null;
    int res = 0;
    // String sql = " DELETE FROM ANSWERBOARD WHERE BOARDNO =? ";
    String sql = " UPDATE ANSWERBOARD SET BOARDNO = 0 WHERE BOARDNO = ? ";

    try {
        pstmt = con.prepareStatement(sql);
        pstmt.setInt(1, boardno);
        res = pstmt.executeUpdate();
        if(res > 0) {
            commit(con);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        close(pstmt);
        close(con);
    }
    return res;
}

public int answerUpdate(int parentboardno) {
    Connection con = getConnection();
    PreparedStatement pstmt = null;
    int res = 0;
    String sql = " UPDATE ANSWERBOARD "
        + " SET GROUPSEQ = GROUPSEQ +1 "
        + " WHERE GROUPNO = (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO =? ) "
        + " AND GROUPSEQ > (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO =? ) ";
    try {
        pstmt = con.prepareStatement(sql);
        pstmt.setInt(1, parentboardno);
        pstmt.setInt(2, parentboardno);
        res = pstmt.executeUpdate();
        if(res > 0) {
            commit(con);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        close(pstmt);
        close(con);
    }

    return res;
}

public int answerInsert(AnswerDto dto) {
    Connection con = getConnection();
    PreparedStatement pstmt = null;
    int res = 0;
    String sql = " INSERT INTO ANSWERBOARD "
        + " VALUES ( BOARDNOSEQ.NEXTVAL, "
        + " (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO =? ),"
        + " (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO =? )+1,"
        + " (SELECT TITLETAB FROM ANSWERBOARD WHERE BOARDNO =? )+1,"
        + " ?, ?, ?, SYSDATE ) ";

```

```

try {
    pstmt = con.prepareStatement(sql);
    pstmt.setInt(1, dto.getboardno());
    pstmt.setInt(2, dto.getboardno());
    pstmt.setInt(3, dto.getboardno());
    pstmt.setString(4, dto.getTitle());
    pstmt.setString(5, dto.getContent());
    pstmt.setString(6, dto.getWriter());
    res = pstmt.executeUpdate();
    if(res > 0) {
        commit(con);
    }
} catch (SQLException e) {
    e.printStackTrace();
}finally {
    close(pstmt);
    close(con);
}
return res;
}
}

/*
 * 메소드 하나당 기능 1개 이기 때문에 이렇게 하면 안됨.
 */
public int insertAnswer(AnswerDto dto) {
    Connection con = getConnection();
    PreparedStatement pstmt1 = null;
    PreparedStatement pstmt2 = null;
    int res1 = 0;
    int res2 = 0;
    String sql1 = " UPDATE ANSWERBOARD "
        + " SET GROUPSEQ = GROUPSEQ+1 "
        + " WHERE GROUPNO = (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO =? "
        + " AND GROUPSEQ > (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO =? ";
    String sql2= " INSERT INTO ANSWERBOARD "
        + " VALUES ( BOARDNOSEQ.NEXTVAL, "
        + " (SELECT GROUPNO FROM ANSWERBOARD WHERE BOARDNO =? ),"
        + " (SELECT GROUPSEQ FROM ANSWERBOARD WHERE BOARDNO =? )+1,"
        + " (SELECT TITLETAB FROM ANSWERBOARD WHERE BOARDNO =? )+1,"
        + " TITLE =? , CONTENT =?, WRITER=?, SYSDATE ) ";
    try {
        pstmt1 = con.prepareStatement(sql1);
        pstmt1.setInt(1, dto.getboardno());
        pstmt1.setInt(2, dto.getboardno());
        res1 = pstmt1.executeUpdate();

        pstmt2 = con.prepareStatement(sql2);
        pstmt2.setInt(1, dto.getboardno());
        pstmt2.setInt(2, dto.getboardno());
        pstmt2.setInt(3, dto.getboardno());
        pstmt2.setString(4, dto.getTitle());
        pstmt2.setString(5, dto.getContent());
        pstmt2.setString(6, dto.getWriter());
        res2 = pstmt2.executeUpdate();

        if((res1 & res2) > 1) {
            commit(con);
        }

    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        close(pstmt2);
        close(pstmt1);
        close(con);
    }

    return res1+res2;
}
*/

```

→ 처음에 강사님과 같이하기전에 혼자 insertAnswer()를 해보았는데 한 메소드안에는 하나의 기능만 처리 하는 것이 원칙이고 지금은 하고 있지 않지만 보통 트랜잭션 처리를 해주어야하기 때문에 기능을 나눠서 구현해야 한다.

→ delete 구현시 sql문을 제대로 delete로 쓰면 구현이 잘 되는데 '삭제된 글 입니다' 라고 나오게 하고 싶어서 삭제를 원하는 boardno를 받아와서 boardno를 0으로 update 시켰다.

그래서 boardlist.jsp에서 boardno = 0 이면 '삭제된 글 입니다' 라고 띄우게 하고 싶었는데

문제 발생



java.sql.SQLIntegrityConstraintViolationException: ORA-00001: unique constraint (KH.SYS_C007358) violated

에러가 발생하며 무결성 제약조건에 위배된다고 나왔다.
sql문을 cascade하며 수정 할 필요가 있을 것 같다.

- AnswerController.java

```
package com.answer.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 com.answer.biz.AnswerBiz;
import com.answer.dto.AnswerDto;

@WebServlet("/answer.do")
public class AnswerController extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        response.setContentType("text/html; charset=utf-8;");

        String command = request.getParameter("command");
        System.out.println("[ "+command+" ]");

        AnswerBiz biz = new AnswerBiz();

        if(command.equals("list")) {
            List<AnswerDto> list = biz.selectList();
            request.setAttribute("list", list);
            dispatch("answerlist.jsp", response, request);
        } else if(command.equals("detail")) {
            int boardno = Integer.parseInt(request.getParameter("boardno"));
            AnswerDto dto = biz.selectOne(boardno);
            request.setAttribute("dto", dto);
            dispatch("detail.jsp", response, request);
        } else if(command.equals("insertform")) {
            response.sendRedirect("insertform.jsp");
        } else if(command.equals("insert")) {
            String title = request.getParameter("title");
            String content = request.getParameter("content");
            String writer = request.getParameter("writer");
            AnswerDto dto = new AnswerDto(title, content, writer);
            int res = biz.insert(dto);
            if(res > 0) {
                jsResponse("작성되었습니다", "answer.do?command=list", response);
            } else {
                jsResponse("작성 실패", "answer.do?command=list", response);
            }
        } else if(command.equals("updateform")) {
            int boardno = Integer.parseInt(request.getParameter("boardno"));
            AnswerDto dto = biz.selectOne(boardno);
            request.setAttribute("dto", dto);
            dispatch("updateform.jsp", response, request);
        } else if(command.equals("update")) {
            int boardno = Integer.parseInt(request.getParameter("boardno"));
            String title = request.getParameter("title");
            String content = request.getParameter("content");
            String writer = request.getParameter("writer");
            AnswerDto dto = new AnswerDto(boardno, title, content, writer);
            int res = biz.update(dto);
            if(res > 0){
                jsResponse("수정되었습니다", "answer.do?command=list", response);
            } else {
                jsResponse("수정 실패", "answer.do?command=updateform&boardno="+boardno, response);
            }
        } else if(command.equals("delete")) {

```

```

        int boardno = Integer.parseInt(request.getParameter("boardno"));
        int res = biz.delete(boardno);
        if(res > 0) {
            jsResponse("삭제되었습니다", "answer.do?command=list", response);
        }else {
            jsResponse("삭제 실패", "answer.do?command=detail&boardno="+boardno, response);
        }
    }else if(command.equals("answerform")) {
        int boardno = Integer.parseInt(request.getParameter("boardno"));
        AnswerDto dto = biz.selectOne(boardno);
        request.setAttribute("dto", dto);
        dispatch("answerform.jsp", response, request);
    }else if(command.equals("answer")) {
        int parentboardno = Integer.parseInt(request.getParameter("parentboardno"));
        String title = request.getParameter("title");
        String content = request.getParameter("content");
        String writer = request.getParameter("writer");
        AnswerDto dto = new AnswerDto(parentboardno, title, content, writer);
        int res = biz.answerProc(dto);
        if(res > 0) {
            jsResponse("답변작성 성공", "answer.do?command=list", response);
        }else {
            jsResponse("답변작성 실패", "answer.do?command=answerform%boardno="+parentboardno, response);
        }
    }
}

response.getWriter().append("<h1>잘못왔네 아이구<h1>");
}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
    doGet(request, response);
}

public void dispatch(String url, HttpServletResponse response, HttpServletRequest request) throws ServletException, IOException {
    RequestDispatcher dispatch = request.getRequestDispatcher(url);
    dispatch.forward(request, response);
}

public void jsResponse(String msg, String url, HttpServletResponse response) throws IOException {
    String s = "<script type='text/javascript'>"+
        "alert('"+msg+"');"+
        "location.href='"+url+"';"+
        "</script>";

    response.getWriter().append(s);
}
}
}

```

- boardbiz.java

```

package com.answer.biz;

import java.util.List;

import com.answer.dao.AnswerDao;
import com.answer.dto.AnswerDto;

public class AnswerBiz {

    AnswerDao dao = new AnswerDao();

    public List<AnswerDto> selectList(){
        return dao.selectList();
    }

    public AnswerDto selectOne(int boardno) {
        return dao.selectOne(boardno);
    }

    public int insert(AnswerDto dto) {
        return dao.insert(dto);
    }

    public int update(AnswerDto dto) {
        return dao.update(dto);
    }

    public int delete(int boardno) {
        return dao.delete(boardno);
    }
}

```



```

public int answerProc(AnswerDto dto) {
    // 원래는 트랜잭션(DB에서 최소한의 작업단위) 처리해줘야함
    // update할 것수가 있으면 update를 해라. 먼저 계산해주는게 맞다
    int update = dao.answerUpdate(dto.getboardno());
    int insert = dao.answerInsert(dto);

    return update+insert;
}
}

```

- anwerlist.jsp

```

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

<%
    request.setCharacterEncoding("UTF-8");
    response.setContentType("text/html; charset=UTF-8");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>게시판 목록</h1>
<table border="1" style="border-collapse:collapse">
    <colgroup>
        <col width="50">
        <col width="300">
        <col width="150">
        <col width="100">
    </colgroup>
    <tr>
        <th>BoardNo</th>
        <th>Title</th>
        <th>Writer</th>
        <th>Regdate</th>
    </tr>
    <c:choose>
        <c:when test="${empty list }">
            <tr>
                <td colspan="4" align="center">작성된 글이 없습니다</td>
            </tr>
        </c:when>
        <c:otherwise>
            <c:forEach items="${list }" var="dto">
                <tr>
                    <c:choose>
                        <c:when test="${dto.boardno eq 0}">
                            <td colspan="4" align="center">삭제된 글입니다.</td>
                        </c:when>
                        <c:otherwise>
                            <td>${dto.boardno }</td>
                            <td>
                                <c:forEach begin="1" end="${dto.titletab }">
                                    &nbsp;
                                </c:forEach>
                                <a href="answer.do?command=detail&boardno=${dto.boardno }">${dto.title }</a>
                            </td>
                            <td>${dto.writer }</td>
                            <td>${dto.regdate }</td>
                        </c:otherwise>
                    </c:choose>
                </tr>
            </c:forEach>
            <tr>
                <td colspan="4" align="right">
                    <input type="button" value="글작성" onclick="location.href='answer.do?command=insertform'"/>
                </td>
            </tr>
        </c:otherwise>
    </c:choose>
</table>
</body>
</html>

```

★원글은 제대로 삭제가 되는데 연결되어있는 RE 글들은 삭제실패! (무결성제약 조건 때문에)
→SQL문 수정할 필요 있음 잊지 말것

- insertform.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    request.setCharacterEncoding("UTF-8");
    response.setContentType("text/html; charset=UTF-8");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>INSERT</h1>
<form action="answer.do" method="post">
<input type="hidden" name="command" value="insert">
<table border="1" style="border-collapse: collapse">
<tr>
<th>writer</th>
<td><input type="text" name="writer"></td>
</tr>
<tr>
<th>title</th>
<td><input type="text" name="title"></td>
</tr>
<tr>
<th>content</th>
<td><textarea cols="60" rows="10" name="content" style="resize: none"></textarea></td>
</tr>
<tr>
<td colspan="3" align="right">
<input type="button" value="목록" onclick="location.href='answer.do?command=list'">
<input type="submit" value="전송">
</td>
</tr>
</table>
</form>
</body>
</html>
```

← → ↻ ⓘ localhost:8787/Jsp09_AnswerBoard/insertform.jsp

INSERT

writer	<input type="text" value="테스트10"/>
title	<input type="text" value="테스트제목10"/>
content	<div>테스트내용10</div>
<input type="button" value="목록"/> <input type="button" value="전송"/>	

- detail.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
```

```

<%
    request.setCharacterEncoding("UTF-8");
    response.setContentType("text/html; charset=UTF-8");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>DETAIL</h1>
<table border="1" style="border-collapse: collapse">
    <tr>
        <th>writer</th>
        <td>${dto.writer }</td>
    </tr>
    <tr>
        <th>title</th>
        <td>${dto.title }</td>
    </tr>
    <tr>
        <th>content</th>
        <td><textarea cols="60" rows="10" readonly="readonly">${dto.content }</textarea></td>
    </tr>
    <tr>
        <td colspan="3" align="right">
            <input type="button" value="답변" onclick="location.href='answer.do?command=answerform&boardno=${dto.boardno}'">
            <input type="button" value="삭제" onclick="location.href='answer.do?command=delete&boardno=${dto.boardno}'"/>
            <input type="button" value="목록" onclick="location.href='answer.do?command=list'"/>
            <input type="button" value="수정" onclick="location.href='answer.do?command=updateform&boardno=${dto.boardno}'">
        </td>
    </tr>
</table>
</body>
</html>

```

← → ↺ ⓘ localhost:8787/Jsp09_AnswerBoard/answer.do?command=detail&boardno=25

DETAIL

writer	강여림
title	테스트02
content	테스트02
<div> <input type="button" value="답변"/> <input type="button" value="삭제"/> <input type="button" value="목록"/> <input type="button" value="수정"/> </div>	

- updateform.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    request.setCharacterEncoding("UTF-8");
    response.setContentType("text/html; charset=UTF-8");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>UPDATE</h1>

```

```

<form action="answer.do" method="post">
<input type="hidden" name="command" value="update">
<input type="hidden" name="boardno" value="${dto.boardno }">
<table border="1" style="border-collapse:collapse">
  <tr>
    <th>writer</th>
    <td><input type="text" name="writer" value="${dto.writer }"></td>
  </tr>
  <tr>
    <th>title</th>
    <td><input type="text" name="title" value="${dto.title }"></td>
  </tr>
  <tr>
    <th>content</th>
    <td><textarea cols="60" rows="10" name="content" style="resize:none">${dto.content }</textarea></td>
  </tr>
  <tr>
    <td colspan="3" align="right">
      <input type="button" value="목록" onclick="location.href='answer.do?command=list'">
      <input type="submit" value="수정">
    </td>
  </tr>
</table>
</form>
</body>
</html>

```

← → ↻ ⓘ localhost:8787/Jsp09_AnswerBoard/answer.do?command=updateform&boardno=25

UPDATE

writer	<input type="text" value="강여림"/>
title	<input type="text" value="테스트02_수정"/>
content	<div>테스트02_수정</div>
<input type="button" value="목록"/> <input type="button" value="수정"/>	

- answerform.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    request.setCharacterEncoding("UTF-8");
    response.setContentType("text/html; charset=UTF-8");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1>REPLY</h1>
<form action="answer.do" method="post">
<input type="hidden" name="command" value="answer">
<input type="hidden" name="parentboardno" value="${dto.boardno }">
<table border="1" style="border-collapse:collapse">
  <tr>
    <th>writer</th>
    <td><input type="text" name="writer" value="${dto.writer }"></td>
  </tr>

```

```

<tr>
  <th>title</th>
  <td><input type="text" name="title" value="RE:${dto.title }"></td>
</tr>
<tr>
  <th>content</th>
  <td><textarea cols="60" rows="10" name="content" style="resize:none">${dto.content }</textarea></td>
</tr>
<tr>
  <td colspan="2" align="right">
    <input type="button" value="취소" onclick="location.href='answer.do?command=list'">
    <input type="submit" value="등록">
  </td>
</tr>
</table>
</form>
</body>
</html>

```

← → ↻ ⓘ localhost:8787/Jsp09_AnswerBoard/answer.do?command=updateform&boardno=28

UPDATE

writer	강여림
title	RE:테스트02_3-수정
content	테스트02_3-수정
<div> <div>목록</div> <div>수정</div> </div>	

게시판 목록

BoardNo	Title	Writer	Regdate
29	테스트제목10	테스트10	2020-07-06
25	테스트02_수정	강여림	2020-07-06
28	RE:테스트02_3	강여림	2020-07-06
26	RE:테스트02_2	강여림	2020-07-06
27	RE:RE:테스트02_2	강여림	2020-07-06
19	테스트01	강여림	2020-07-06
22	RE:테스트01_2	강여림	2020-07-06
24	RE:테스트01_4	강여림	2020-07-06
23	RE:테스트01_3	강여림	2020-07-06
20	RE:테스트01_1	강여림	2020-07-06
21	RE:RE:테스트01_2	강여림	2020-07-06
6	테스트 글01_수정ㅍㅍ	작성자01_수정	2020-07-06
18	RE:테스트 글01_수정ㄷㄷ=ㅌㅌㄷ	작성자01_수정	2020-07-06
17	RE:테스트 글01_수정ㅅㅅ	작성자01_수정	2020-07-06
16	RE:테스트 글01_수정ㅋ	작성자01_수정	2020-07-06
7	RE:테스트 글01_수정	작성자01_수정	2020-07-06
13	RE:테스트 글01_수정ㅋㅋ	작성자01_수정ㅋㅋ	2020-07-06
12	RE:RE:진짜달려요	작성자01_수정	2020-07-06
삭제된 글입니다.			
2	두번째 글 입니다.	아무나	2020-07-06
14	RE:두번째 글 입니다.ㅎ	아무나	2020-07-06
1	첫번째 글 입니다	관리자	2020-07-06
5	ㄴRE:이제 알겠다	강여림	2020-07-06
3	ㄴRE:첫번째 글 입니다.	홍길동	2020-07-06
4	ㄴRE:두번째 글 입니다	강여림	2020-07-06
			글작성