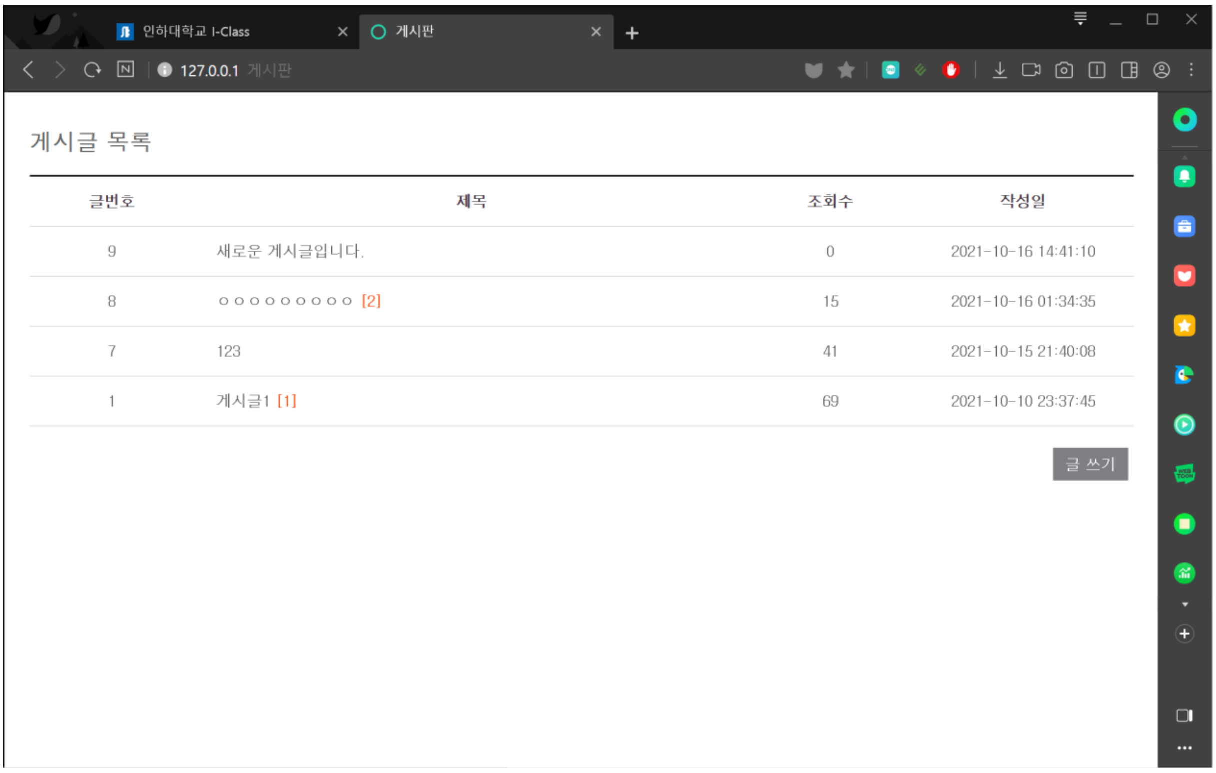
웹서버프로그래밍 댓글작성과제

12204172 최현식

1. 동작화면 캡처

A. 게시글 목록

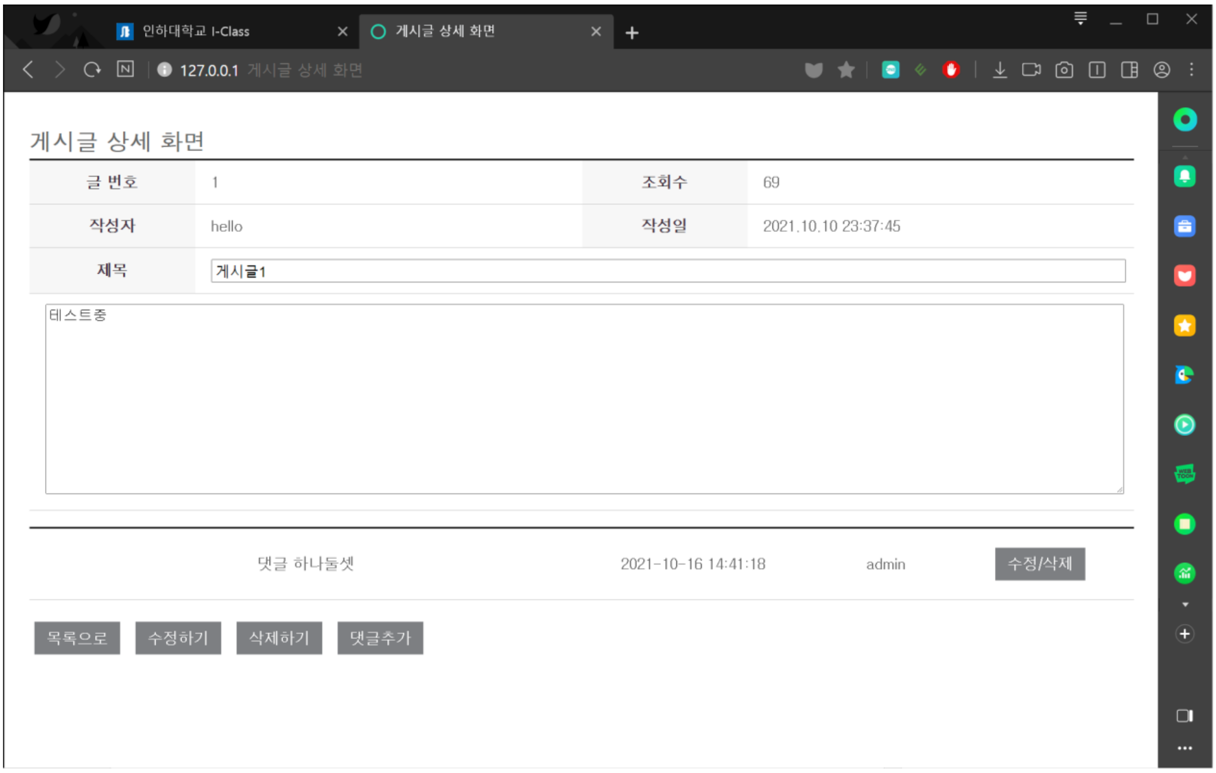


게시글 목록

글번호	제목	조회수	작성일
9	새로운 게시글입니다.	0	2021-10-16 14:41:10
8	ㅇㅇㅇㅇㅇㅇㅇㅇㅇ [2]	15	2021-10-16 01:34:35
7	123	41	2021-10-15 21:40:08
1	게시글1 [1]	69	2021-10-10 23:37:45

글 쓰기

B. 게시글 상세



게시글 상세 화면

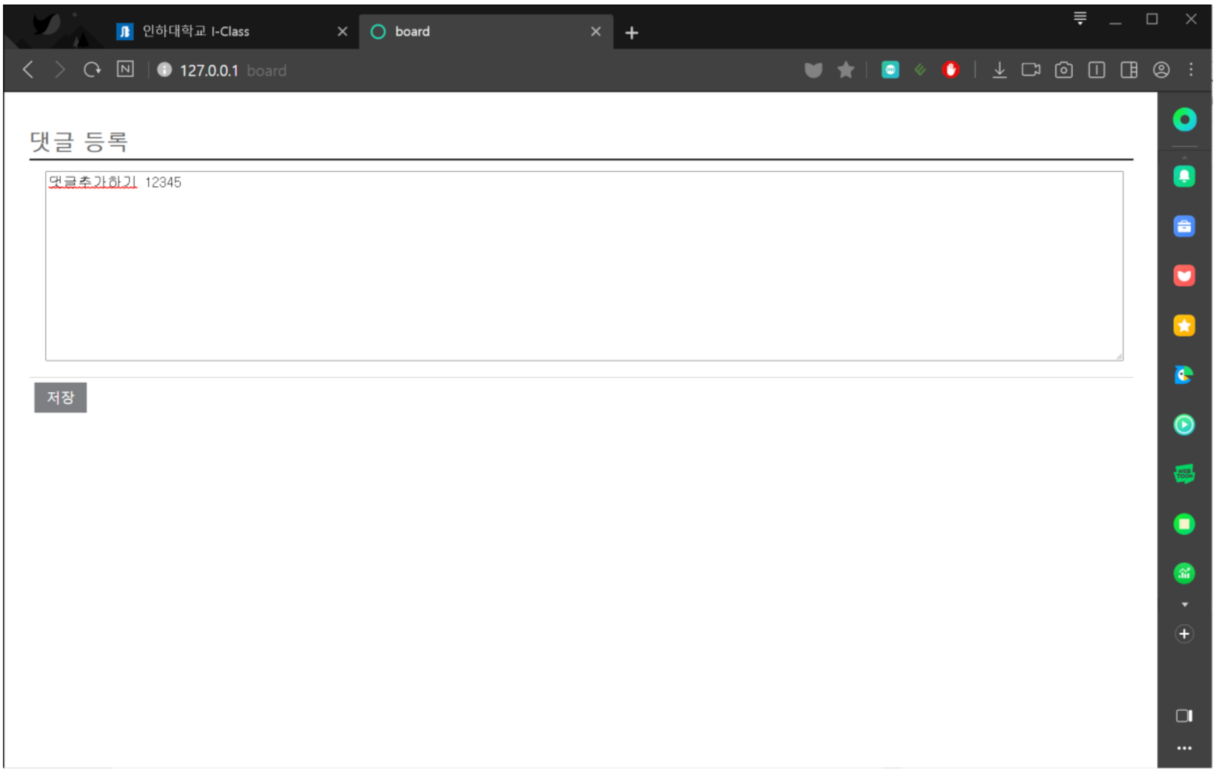
글 번호	1	조회수	69
작성자	hello	작성일	2021.10.10 23:37:45
제목	게시글1		

테스트중

댓글 하나둘셋	2021-10-16 14:41:18	admin	수정/삭제

목록으로 수정하기 삭제하기 댓글추가

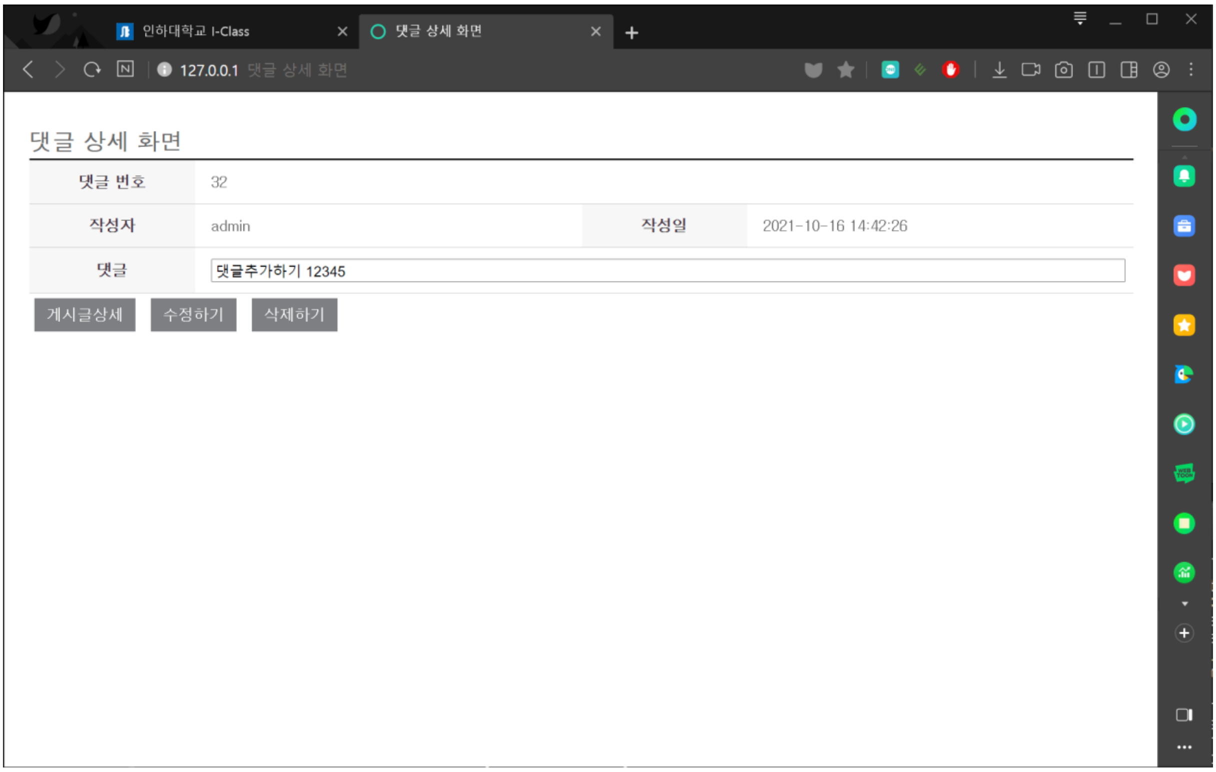
C. 댓글 추가



댓글 등록

댓글추가하기 12345

D. 댓글 수정/삭제

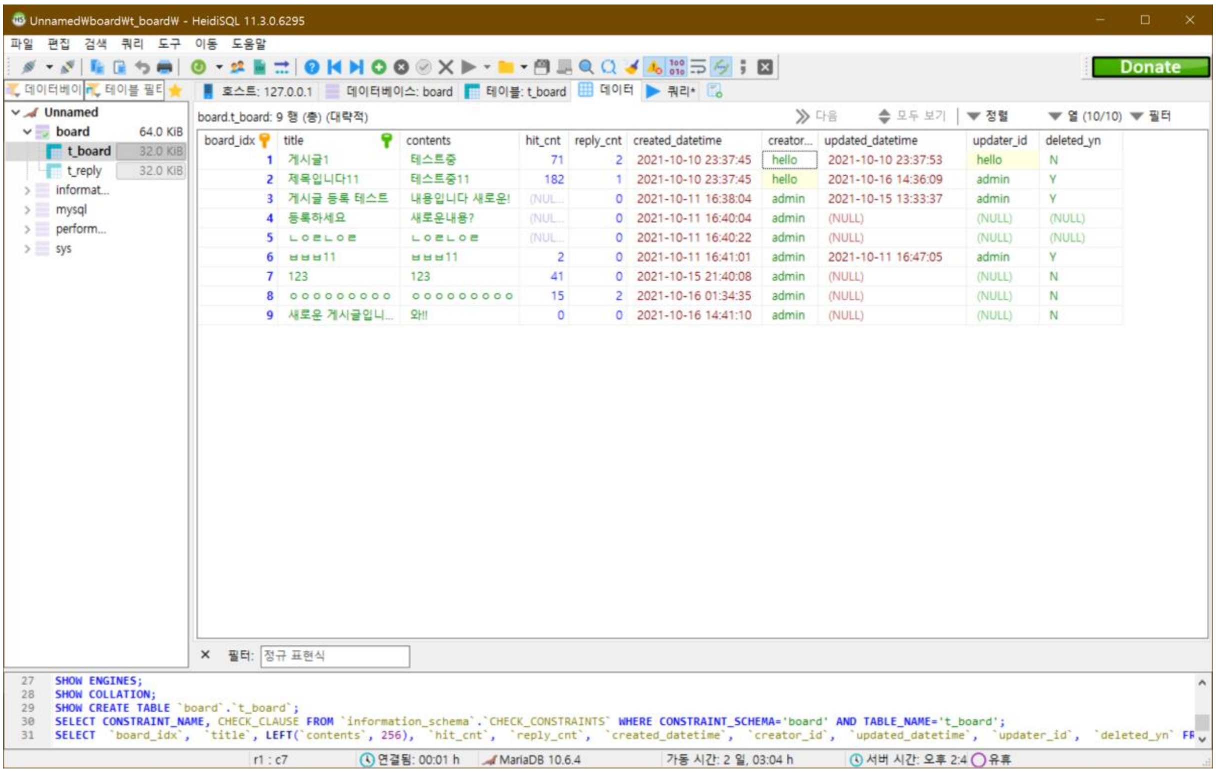


댓글 상세 화면

댓글 번호	32		
작성자	admin	작성일	2021-10-16 14:42:26
댓글	댓글추가하기 12345		

2. 테이블 캡처

A. T_board

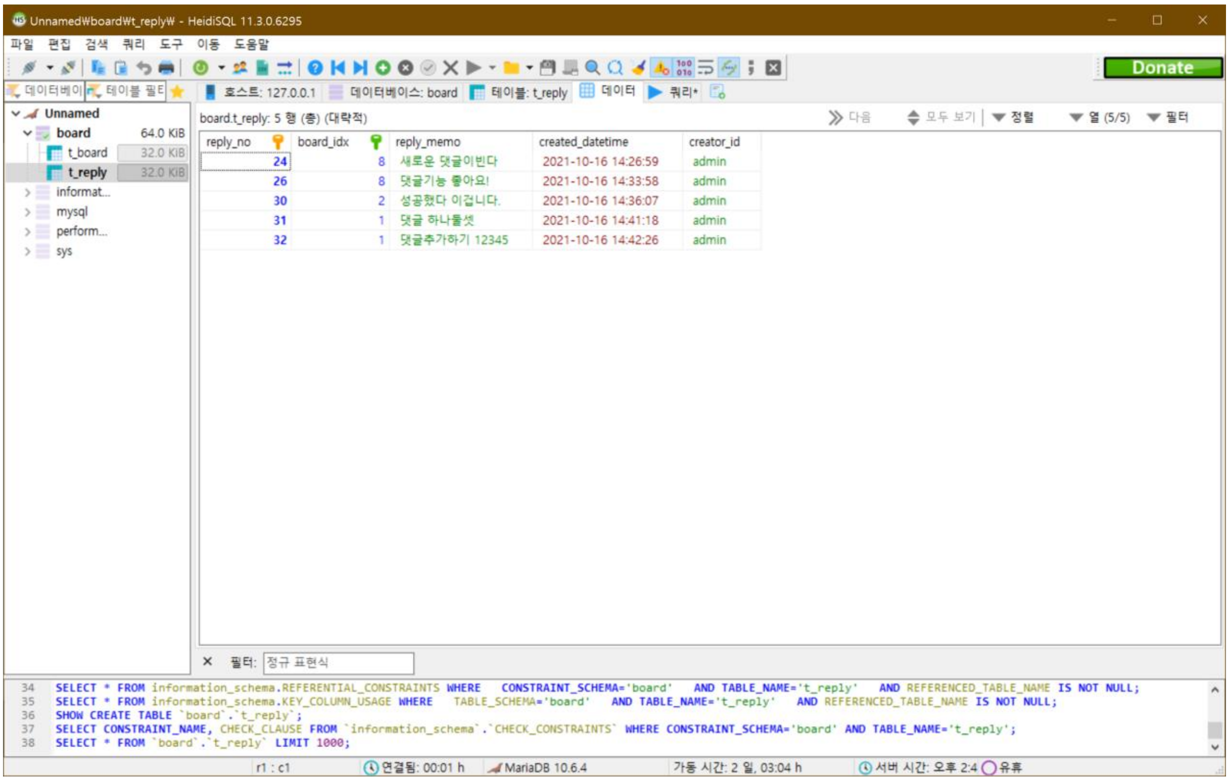


board.t_board: 9 행 (총) (대략적)

board_idx	title	contents	hit_cnt	reply_cnt	created_datetime	creator...	updated_datetime	updater_id	deleted_yn
1	게시글1	테스트중	71	2	2021-10-10 23:37:45	hello	2021-10-10 23:37:53	hello	N
2	제목입니다11	테스트중11	182	1	2021-10-10 23:37:45	hello	2021-10-16 14:36:09	admin	Y
3	게시글 등록 테스트	내용입니다 새로운!	(NUL...	0	2021-10-11 16:38:04	admin	2021-10-15 13:33:37	admin	Y
4	등록하세요	새로운내용?	(NUL...	0	2021-10-11 16:40:04	admin	(NULL)	(NULL)	(NULL)
5	ㄴㅇㄹㄴㅇㄹ	ㄴㅇㄹㄴㅇㄹ	(NUL...	0	2021-10-11 16:40:22	admin	(NULL)	(NULL)	(NULL)
6	ㅂㅂㅂ11	ㅂㅂㅂ11	2	0	2021-10-11 16:41:01	admin	2021-10-11 16:47:05	admin	Y
7	123	123	41	0	2021-10-15 21:40:08	admin	(NULL)	(NULL)	N
8	ㅇㅇㅇㅇㅇㅇㅇㅇㅇㅇ	ㅇㅇㅇㅇㅇㅇㅇㅇㅇㅇ	15	2	2021-10-16 01:34:35	admin	(NULL)	(NULL)	N
9	새로운 게시글입니...	와!!	0	0	2021-10-16 14:41:10	admin	(NULL)	(NULL)	N

```
SHOW ENGINES;
SHOW COLLATION;
SHOW CREATE TABLE `board`.`t_board`;
SELECT CONSTRAINT_NAME, CHECK_CLAUSE FROM `information_schema`.`CHECK_CONSTRAINTS` WHERE CONSTRAINT_SCHEMA='board' AND TABLE_NAME='t_board';
SELECT  `board_idx`,  `title`, LEFT(`contents`, 256),  `hit_cnt`,  `reply_cnt`,  `created_datetime`,  `creator_id`,  `updated_datetime`,  `updater_id`,  `deleted_yn` FR
```

B. T_Reply



board.t_reply: 5 행 (총) (대략적)

reply_no	board_idx	reply_memo	created_datetime	creator_id
24	8	새로운 댓글이빈다	2021-10-16 14:26:59	admin
26	8	댓글기능 좋아요!	2021-10-16 14:33:58	admin
30	2	성공했다 이겁니다.	2021-10-16 14:36:07	admin
31	1	댓글 하나둘셋	2021-10-16 14:41:18	admin
32	1	댓글추가하기 12345	2021-10-16 14:42:26	admin

```
SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE   CONSTRAINT_SCHEMA='board'   AND TABLE_NAME='t_reply'   AND REFERENCED_TABLE_NAME IS NOT NULL;
SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE   TABLE_SCHEMA='board'   AND TABLE_NAME='t_reply'   AND REFERENCED_TABLE_NAME IS NOT NULL;
SHOW CREATE TABLE `board`.`t_reply`;
SELECT CONSTRAINT_NAME, CHECK_CLAUSE FROM `information_schema`.`CHECK_CONSTRAINTS` WHERE CONSTRAINT_SCHEMA='board' AND TABLE_NAME='t_reply';
SELECT * FROM `board`.`t_reply` LIMIT 1000;
```

3. 소스코드

Controller (BoardController.java)

```java
package kr.ac.inha.board.board.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;

import kr.ac.inha.board.board.dto.BoardDto;
import kr.ac.inha.board.board.dto.ReplyDto;
import kr.ac.inha.board.board.service.BoardService;

@Controller
public class BoardController {
        @Autowired
        private BoardService boardService;

        // 게시물 관련
        @RequestMapping("/board/openBoardList.do")
        public ModelAndView openBoardList() throws Exception {
                ModelAndView mv = new ModelAndView("/board/boardList");

                List<BoardDto> list = boardService.selectBoardList();
                mv.addObject("list", list);

                return mv;
        }

        @RequestMapping("/board/openBoardWrite.do")
        public String openBoardWrite() throws Exception {
                return "/board/boardWrite";
        }

        @RequestMapping("/board/insertBoard.do")
        public String insertBoard(BoardDto board) throws Exception {
                boardService.insertBoard(board);
                return "redirect:/board/openBoardList.do";
        }

        @RequestMapping("/board/openBoardDetail.do")
        public ModelAndView openBoardDetail(@RequestParam int boardIdx) throws Exception {
                ModelAndView mv = new ModelAndView("/board/boardDetail");

                BoardDto board = boardService.selectBoardDetail(boardIdx);
                List<ReplyDto> reply = boardService.selectReplyList(boardIdx);

                mv.addObject("board", board);
                mv.addObject("reply", reply);
                return mv;
        }

        @RequestMapping("/board/updateBoard.do")
        public String updateBoard(BoardDto board) throws Exception {
                boardService.updateBoard(board);
                return "redirect:/board/openBoardList.do";
        }

        @RequestMapping("/board/deleteBoard.do")
        public String deleteBoard(int boardIdx) throws Exception {
                boardService.deleteBoard(boardIdx);
                return "redirect:/board/openBoardList.do";
        }



        // 댓글 관련
        @RequestMapping(value = "/board/openReplyWrite.do", method = RequestMethod.GET)
        public ModelAndView openReplyWrite(int boardIdx) throws Exception {
                System.out.println(boardIdx);
                ModelAndView mv = new ModelAndView("/board/commentWrite");
                mv.addObject("idx",boardIdx);
                return mv;
        }
        @RequestMapping("/board/insertReply.do")
        public String insertReply(ReplyDto reply) throws Exception {
                boardService.insertReply(reply);
                return "redirect:/board/openBoardDetail.do?boardIdx="+reply.getBoardIdx();
        }

        @RequestMapping("/board/openReplyDetail.do")
        public ModelAndView openReplyDetail(@RequestParam int replyNo) throws Exception {
                ModelAndView mv = new ModelAndView("/board/replyDetail");
                ReplyDto reply = boardService.selectReplyDetail(replyNo);
                mv.addObject("reply", reply);
                return mv;
        }

        @RequestMapping("/board/updateReply.do")
        public String updateReply(ReplyDto reply) throws Exception {
                boardService.updateReply(reply);
                return "redirect:/board/openBoardDetail.do?boardIdx="+reply.getBoardIdx();
        }

        @RequestMapping("/board/deleteReply.do")
        public String deleteReply(@RequestParam int replyNo) throws Exception {
                ReplyDto reply = boardService.selectReplyDetail(replyNo);
                boardService.deleteReply(replyNo);
                return "redirect:/board/openBoardDetail.do?boardIdx="+reply.getBoardIdx();

        }
}
```

Service (BoardService.java)

```
package kr.ac.inha.board.board.service;

import java.util.List;

import kr.ac.inha.board.board.dto.BoardDto;
import kr.ac.inha.board.board.dto.ReplyDto;

public interface BoardService {
        List<BoardDto> selectBoardList() throws Exception;

        void insertBoard(BoardDto board) throws Exception;

        BoardDto selectBoardDetail(int boardIdx) throws Exception;
        void updateBoard(BoardDto board) throws Exception;
        void deleteBoard(int boardIdx) throws Exception;

        // 댓글 관련
        List<ReplyDto> selectReplyList(int boardIdx) throws Exception;
        ReplyDto selectReplyDetail(int replyNo) throws Exception;
        void insertReply(ReplyDto board) throws Exception;
        void updateReply(ReplyDto reply) throws Exception;
        void deleteReply(int replyNo) throws Exception;
}
```

Service (BoardServiceImpl.java)

```
package kr.ac.inha.board.board.service;

import java.util.List;

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

import kr.ac.inha.board.board.dto.BoardDto;
import kr.ac.inha.board.board.dto.ReplyDto;
import kr.ac.inha.board.board.mapper.BoardMapper;

@Service
public class BoardServiceImpl implements BoardService {
        @Autowired
        private BoardMapper boardMapper;

        @Override
        public List<BoardDto> selectBoardList() throws Exception {
                return boardMapper.selectBoardList();
        }

        @Override
        public void insertBoard(BoardDto board) throws Exception {
                boardMapper.insertBoard(board);
        }

        @Override
        public BoardDto selectBoardDetail(int boardIdx) throws Exception {
                BoardDto board = boardMapper.selectBoardDetail(boardIdx);
                boardMapper.updateHitCount(boardIdx);
                return board;
        }

        @Override
        public void updateBoard(BoardDto board) throws Exception {
                boardMapper.updateBoard(board);
        }

        @Override
        public void deleteBoard(int boardIdx) throws Exception {
                boardMapper.deleteBoard(boardIdx);
        }

        // 댓글 기능
        @Override
        public List<ReplyDto> selectReplyList(int boardIdx) throws Exception {
                return boardMapper.selectReplyList(boardIdx);
        }

        @Override
        public ReplyDto selectReplyDetail(int replyNo) throws Exception{
                ReplyDto reply = boardMapper.selectReplyDetail(replyNo);
                //boardMapper.updateReplyCount(reply.getBoardIdx(), replyNo);
                return reply;
        }

        @Override
        public void insertReply(ReplyDto board) throws Exception{
                boardMapper.insertReply(board);
                boardMapper.updateReplyCount(board.getBoardIdx(), 1);
        }

        @Override
        public void updateReply(ReplyDto reply) throws Exception {
                boardMapper.updateReply(reply);
        }

        @Override
        public void deleteReply(int replyNo) throws Exception {
                ReplyDto reply = boardMapper.selectReplyDetail(replyNo);
                boardMapper.updateReplyCount(reply.getBoardIdx(), -1);
                boardMapper.deleteReply(replyNo);
        }

}
```

Mapper (BoardMapper.java)

```
package kr.ac.inha.board.board.mapper;

import java.util.List;

import org.apache.ibatis.annotations.Mapper;
import org.apache.ibatis.annotations.Param;

import kr.ac.inha.board.board.dto.BoardDto;
import kr.ac.inha.board.board.dto.ReplyDto;

@Mapper
public interface BoardMapper {
        // 게시글 관련
        List<BoardDto> selectBoardList() throws Exception;

        void insertBoard(BoardDto board) throws Exception;

        BoardDto selectBoardDetail(int boardIdx) throws Exception;
        void updateHitCount(int boardIdx) throws Exception;
        void updateBoard(BoardDto board) throws Exception;
        void deleteBoard(int boardIdx) throws Exception;

        // 댓글 관련
        List<ReplyDto> selectReplyList(int boardIdx) throws Exception;
        ReplyDto selectReplyDetail(int replyNo) throws Exception;
        void updateReplyCount(@Param("boardIdx") int boardIdx, @Param("interval") int interval)
throws Exception;
        void insertReply(ReplyDto board) throws Exception;
        void updateReply(ReplyDto reply) throws Exception;
        void deleteReply(int replyNo) throws Exception;
}
```

Template (boardDetail.html)

```
<!DOCTYPE html><!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org">
        <head>
                <meta charset="UTF-8">
                <title>게시글 상세 화면</title>
                <link rel="stylesheet" th:href="@{/css/style.css}"/>
        </head>
        <body>
                <div class="container">
                <h2>게시글 상세 화면</h2>
                <form id="frm" method="post">
                        <table class="board_detail">
                                <colgroup>
                                        <col width="15%"/>
                                        <col width="35%"/>
                                        <col width="15%"/>
                                        <col width="35%"/>
                                </colgroup>
                                <caption>게시글 상세내용</caption>
                                <tbody>
                                        <tr>
                                                <th scope="row">글 번호</th>
                                                <td th:text="${board.boardIdx}"></td>
                                                <th scope="row">조회수</th>
                                                <td th:text="${board.hitCnt}"></td>
                                        </tr>
                                        <tr>
                                                <th scope="row">작성자</th>
                                                <td th:text="${board.creatorId}"></td>
                                                <th scope="row">작성일</th>
                                                <td th:text="${board.createdDatetime }"></td>
                                        </tr>
                                        <tr>
                                                <th scope="row">제목</th>
                                                <td colspan="3"><input type="text" id="title"
name="title" th:value="${board.title }"/></td>
                                        </tr>
                                        <tr>
                                                <td colspan="4" class="view_text">
                                                <textarea title="내용" id="contents"
name="contents" th:text="${board.contents }"></textarea>
                                                </td>
                                        </tr>
                                </tbody>
                        </table>
                        <input type="hidden" id="boardIdx" name="boardIdx"
th:value="${board.boardIdx }">
                </form>

                <!-- 댓글 목록 부분 -->
                <table class="board_list">
                        <colgroup>
                                <col width="*" />
                                <col width="20%" />
                                <col width="15%" />
                                <col width="15%" />
                        </colgroup>
                        <tbody>
                                <tr th:if="${#lists.size(reply)} > 0" th:each="reply :
${reply}">
                                        <td th:text="${reply.replyMemo}"></td>
                                        <td th:text="${reply.createdDatetime}"></td>
                                        <td th:text="${reply.creatorId}"></td>
                                        <td class="title">
                                                <a href="/board/openReplyDetail.do?replyNo="
th:attrappend="href=${reply.replyNo}" class="btn">
                                                수정/삭제
                                                </a>
                                        </td>
                                </tr>
                                <tr th:unless="${#lists.size(reply)} > 0">
                                        <td colspan="4">조회된 결과가 없습니다.</td>
                                </tr>
                        </tbody>
                </table>

                <a href="#this" id="list" class="btn">목록으로</a>
                <a href="#this" id="edit" class="btn">수정하기</a>
                <a href="#this" id="delete" class="btn">삭제하기</a>
                <a href="/board/openReplyWrite.do?boardIdx="
                        th:attrappend="href=${board.boardIdx}" class="btn">댓글추가</a>
                </div>
                <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
                <script type="text/javascript">
                $(document).ready(function(){
                        $("#list").on("click", function(){
                                location.href = "/board/openBoardList.do";
                        });
                        $("#edit").on("click", function(){
                                var frm = $("#frm")[0];
                                frm.action = "/board/updateBoard.do";
                                frm.submit();
                        });
                        $("#delete").on("click", function(){
                                var frm = $("#frm")[0];
                                frm.action = "/board/deleteBoard.do";
                                frm.submit();
                        });
                });
                </script>
        </body>
</html>
```

Template (boardList.html)

```
<!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org">

<head>
    <meta charset="UTF-8">
    <title>게시판</title>
    <link rel="stylesheet" th:href="@{/css/style.css}" />
</head>

<body>
    <div class="container">
        <h2>게시글 목록</h2>
        <table class="board_list">
            <colgroup>
                <col width="15%" />
                <col width="*" />
                <col width="15%" />
                <col width="20%" />
            </colgroup>
            <thead>
                <tr>
                    <th scope="col">글번호</th>
                    <th scope="col">제목</th>
                    <th scope="col">조회수</th>
                    <th scope="col">작성일</th>
                </tr>
            </thead>
            <tbody>
                <tr th:if="${#lists.size(list)} > 0" th:each="list : ${list}">
                    <td th:text="${list.boardIdx}"></td>
                    <td class="title">
                        <a

    href="/board/openBoardDetail.do?boardIdx="
                            th:attrappend="href=${list.boardIdx} "
                            th:text=" ${list.title}">
                        </a>
                        <span class="num_of_reply"
th:text="(${list.replyCnt}>0 ? '['+${list.replyCnt}+']': '')"></span>
                    </td>
                    <td th:text="${list.hitCnt}"></td>
                    <td th:text="${list.createdDatetime}"></td>
                </tr>
                <tr th:unless="${#lists.size(list)} > 0">
                    <td colspan="4">조회된 결과가 없습니다.</td>
                </tr>
            </tbody>
        </table>
        <p align="right">
            <a href="/board/openBoardWrite.do" class="btn">글 쓰기</a>
        </p>
    </div>
</body>
</html>
```

Template (boardWrite.html)

```
<!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org">
        <head>
                <meta charset="UTF-8">
                <title>board</title>
                <link rel="stylesheet" th:href="@{/css/style.css}"/>
        </head>
        <body>
                <div class="container">
                <h2>게시글 등록</h2>
                <form id="frm" name="frm" method="post" action="/board/insertBoard.do">
                        <table class="board_detail">
                                <tr>
                                        <td>제목</td>
                                        <td><input type="text" id="title" name="title"/></td>
                                </tr>
                                <tr>
                                        <td colspan="2">
                                        <textarea id="contents" name="contents"></textarea>
                                        </td>
                                </tr>
                        </table>
                        <input type="submit" id="submit" value="저장" class="btn">
                </form>
                </div>
        </body>
</html>
```

Template (commentWrite.html)

```
<!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org">

<head>
        <meta charset="UTF-8">
        <title>board</title>
        <link rel="stylesheet" th:href="@{/css/style.css}" />
</head>

<body>
        <div class="container">
                <h2>댓글 등록</h2>
                <form id="frm" name="frm" method="post" action="/board/insertReply.do">
                        <table class="board_detail">
                                <tr>
                                        <td colspan="2">
                                                <textarea id="replyMemo"
name="replyMemo"></textarea>
                                        </td>
                                </tr>
                        </table>
                        <input type="hidden" id="boardIdx" name="boardIdx" th:value="${idx}">
                        <input type="submit" id="submit" value="저장" class="btn">
                </form>
        </div>
</body>
```

```
</html>
```

Template (replyDetail.html)

```
<!DOCTYPE html><!DOCTYPE html>
<html lang="ko" xmlns:th="http://www.thymeleaf.org">
        <head>
                <meta charset="UTF-8">
                <title>댓글 상세 화면</title>
                <link rel="stylesheet" th:href="@{/css/style.css}"/>
        </head>
        <body>

                <div class="container">
                <h2>댓글 상세 화면</h2>
                <form id="frm" name="frm" method="post" action="/board/updateReply.do">
                        <table class="board_detail">
                                <colgroup>
                                        <col width="15%"/>
                                        <col width="35%"/>
                                        <col width="15%"/>
                                        <col width="35%"/>
                                </colgroup>
                                <caption>게시글 상세내용</caption>
                                <tbody>
                                        <tr>
                                                <th scope="row">댓글 번호</th>
                                                <td colspan="3"
th:text="${reply.replyNo}"></td>
                                        </tr>
                                        <tr>
                                                <th scope="row">작성자</th>
                                                <td th:text="${reply.creatorId}"></td>
                                                <th scope="row">작성일</th>
                                                <td th:text="${reply.createdDatetime }"></td>
                                        </tr>
                                        <tr>
                                                <th scope="row">댓글</th>
                                                <td colspan="3"><input type="text"
id="replyMemo" name="replyMemo" th:value="${reply.replyMemo}"/></td>
                                        </tr>
                                </tbody>
                        </table>
                        <input type="hidden" id="boardIdx" name="boardIdx"
th:value="${reply.boardIdx }">
                        <input type="hidden" id="replyNo" name="replyNo"
th:value="${reply.replyNo}">
                </form>

                <a href="/board/openBoardDetail.do?boardIdx="
th:attrappend="href=${reply.boardIdx}" class="btn">게시글상세</a>
                <a href="#this" id="edit" class="btn">수정하기</a>
                <a href="/board/deleteReply.do?replyNo=" th:attrappend="href=${reply.replyNo}"
class="btn">삭제하기</a>
                </div>

                <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
```

```
		<script type="text/javascript">
		$(document).ready(function(){
			$("#edit").on("click", function(){
				var frm = $("#frm")[0];
				frm.action = "/board/updateReply.do";
				frm.submit();
			});
		});
		</script>
	</body>
</html>
```

Query (sql-board.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-
3-mapper.dtd">

<mapper namespace="kr.ac.inha.board.board.mapper.BoardMapper">
	<select id="selectBoardList" resultType="kr.ac.inha.board.board.dto.BoardDto">
		<![CDATA[
			SELECT
				board_idx,
				title,
				hit_cnt,
				reply_cnt,
				DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS
created_datetime
			FROM
				t_board
			WHERE
				deleted_yn = 'N'
			ORDER BY board_idx DESC
		]]>
	</select>


	<insert id="insertBoard" parameterType="kr.ac.inha.board.board.dto.BoardDto"
useGeneratedKeys="true" keyProperty="boardIdx">
		<![CDATA[
		INSERT INTO t_board
		(
			title,
			contents,
			created_datetime,
			creator_id
		)
		VALUES
		(
			#{title},
			#{contents},
			NOW(),
			'admin'
		)
		]]>
	</insert>

	<select id="selectBoardDetail" parameterType="int"
```

```
resultType="kr.ac.inha.board.board.dto.BoardDto">
                <![CDATA[
                SELECT
                board_idx, title, contents, hit_cnt, DATE_FORMAT(created_datetime,
'%Y.%m.%d %H:%i:%s') AS created_datetime, creator_id
                FROM
                t_board
                WHERE
                board_idx = #{boardIdx}
                AND deleted_yn = 'N'
                ]]>
        </select>

        <update id="updateHitCount" parameterType="int">
                <![CDATA[
                UPDATE
                t_board
                SET
                hit_cnt = hit_cnt + 1
                WHERE
                board_idx = #{boardIdx}
                ]]>
        </update>
        <update id="updateBoard" parameterType="kr.ac.inha.board.board.dto.BoardDto">
                <![CDATA[
                        UPDATE t_board
                                SET title = #{title},
                                contents = #{contents},
                                updated_datetime = Now(),
                                updater_id = 'admin'
                        WHERE
                                board_idx = #{boardIdx}
                ]]>
        </update>
        <update id="deleteBoard" parameterType="int">
                <![CDATA[
                UPDATE t_board
                SET deleted_yn = 'Y',
                updated_datetime = Now(),
                updater_id = 'admin'
                WHERE
                board_idx = #{boardIdx}
                ]]>
        </update>


        <select id="selectReplyList" resultType="kr.ac.inha.board.board.dto.ReplyDto">
                <![CDATA[
                        SELECT
                                reply_no,
                                board_idx,
                                reply_memo,
                                DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS
created_datetime,
                                creator_id
                        FROM
                                t_reply
                        WHERE
                                board_idx = #{boardIdx}
                        ORDER BY created_datetime DESC
                ]]>
        </select>
        <select id="selectReplyDetail" resultType="kr.ac.inha.board.board.dto.ReplyDto">
                <![CDATA[
                        SELECT
                                reply_no,
                                board_idx,
                                reply_memo,
                                DATE_FORMAT(created_datetime, '%Y-%m-%d %H:%i:%s') AS
created_datetime,
                                creator_id
                        FROM
                                t_reply
                        WHERE
                                reply_no = #{replyNo}
                ]]>
        </select>
        <insert id="insertReply" parameterType="kr.ac.inha.board.board.dto.ReplyDto"
useGeneratedKeys="true" keyProperty="replyNo">
                <![CDATA[
                        INSERT INTO t_reply
                        (
                                board_idx,
                                reply_memo,
                                created_datetime,
                                creator_id
                        )
                        VALUES
                        (
                                #{boardIdx},
                                #{replyMemo},
                                NOW(),
                                'admin'
                        )
                ]]>
        </insert>
        <update id="updateReplyCount">
                <![CDATA[
                        UPDATE t_board
                                SET reply_cnt = ifnull(reply_cnt, 0) + #{interval}
                        WHERE board_idx = #{boardIdx}
                ]]>
        </update>
        <update id="updateReply" parameterType="kr.ac.inha.board.board.dto.ReplyDto">
                <![CDATA[
                        UPDATE t_reply
                                SET reply_memo = #{replyMemo}
                        WHERE reply_no = #{replyNo}
                ]]>
        </update>
        <delete id="deleteReply" parameterType="int">
                <![CDATA[
                        DELETE FROM t_reply
                        WHERE reply_no = #{replyNo}
                ]]>
        </delete>
</mapper>
```