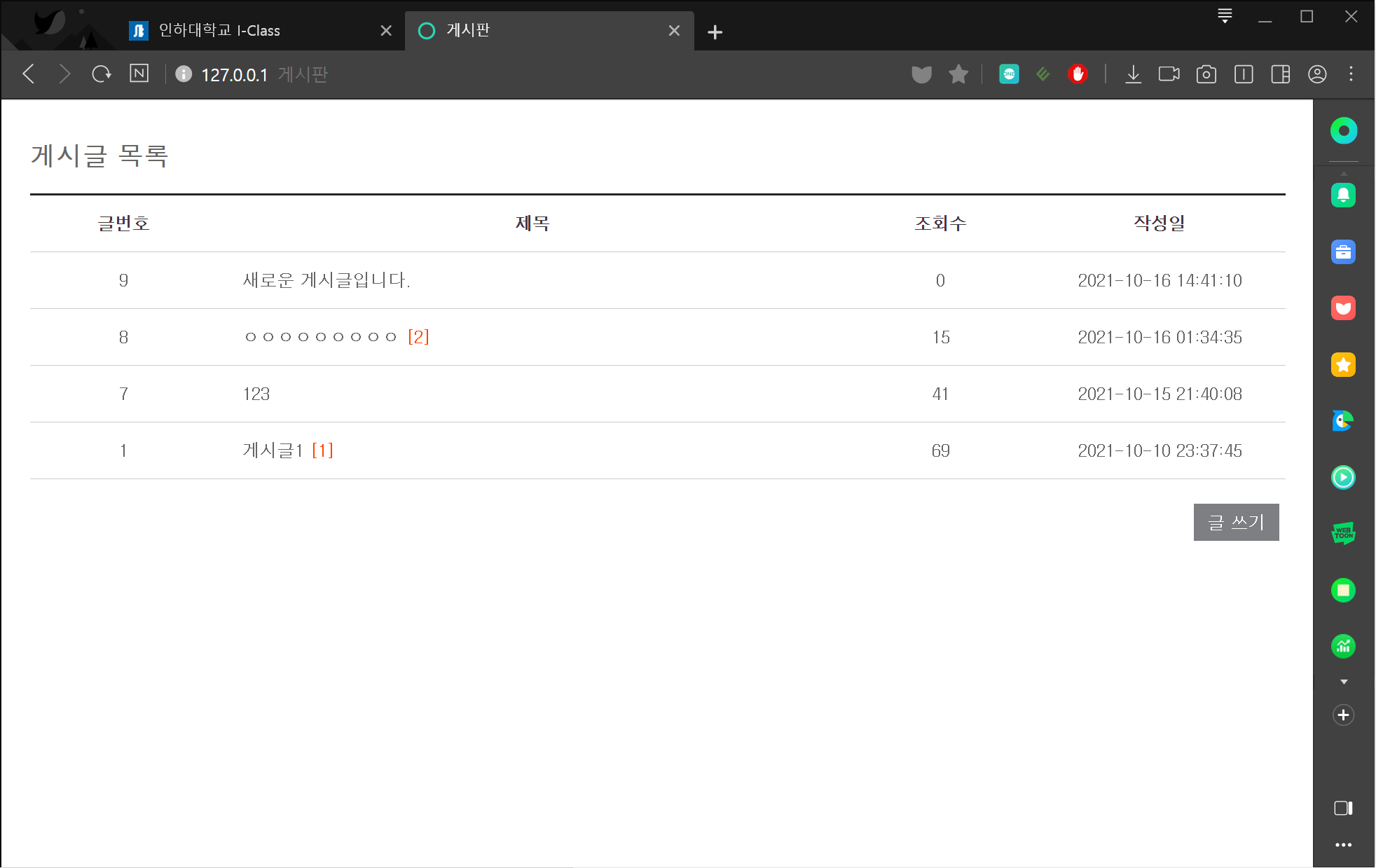
**웹서버프로그래밍 댓글작성과제**

12204172 최현식

1. **동작화면 캡처**
   1. 게시글 목록  
      
   2. 게시글 상세  
      텍스트이(가) 표시된 사진

      자동 생성된 설명
   3. 댓글 추가  
      텍스트이(가) 표시된 사진

      자동 생성된 설명
   4. 댓글 수정/삭제  
      텍스트이(가) 표시된 사진

      자동 생성된 설명
2. **테이블 캡처**
   1. T\_board  
      텍스트이(가) 표시된 사진

      자동 생성된 설명
   2. T\_Reply  
      텍스트이(가) 표시된 사진

      자동 생성된 설명
3. **소스코드**

|  |
| --- |
| **Controller (BoardController.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**> |