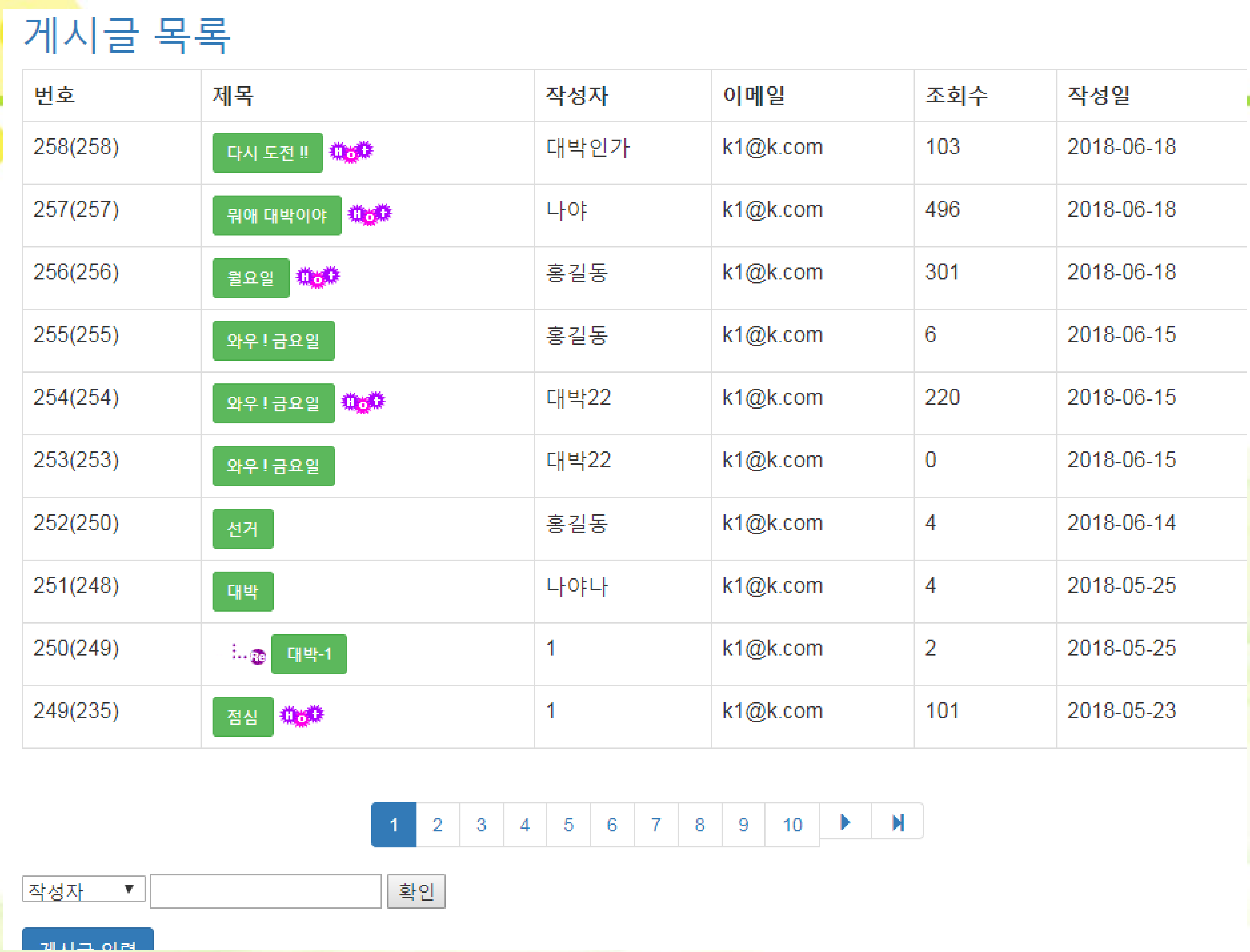
게시판(Mybatis)

강사 : 강병준

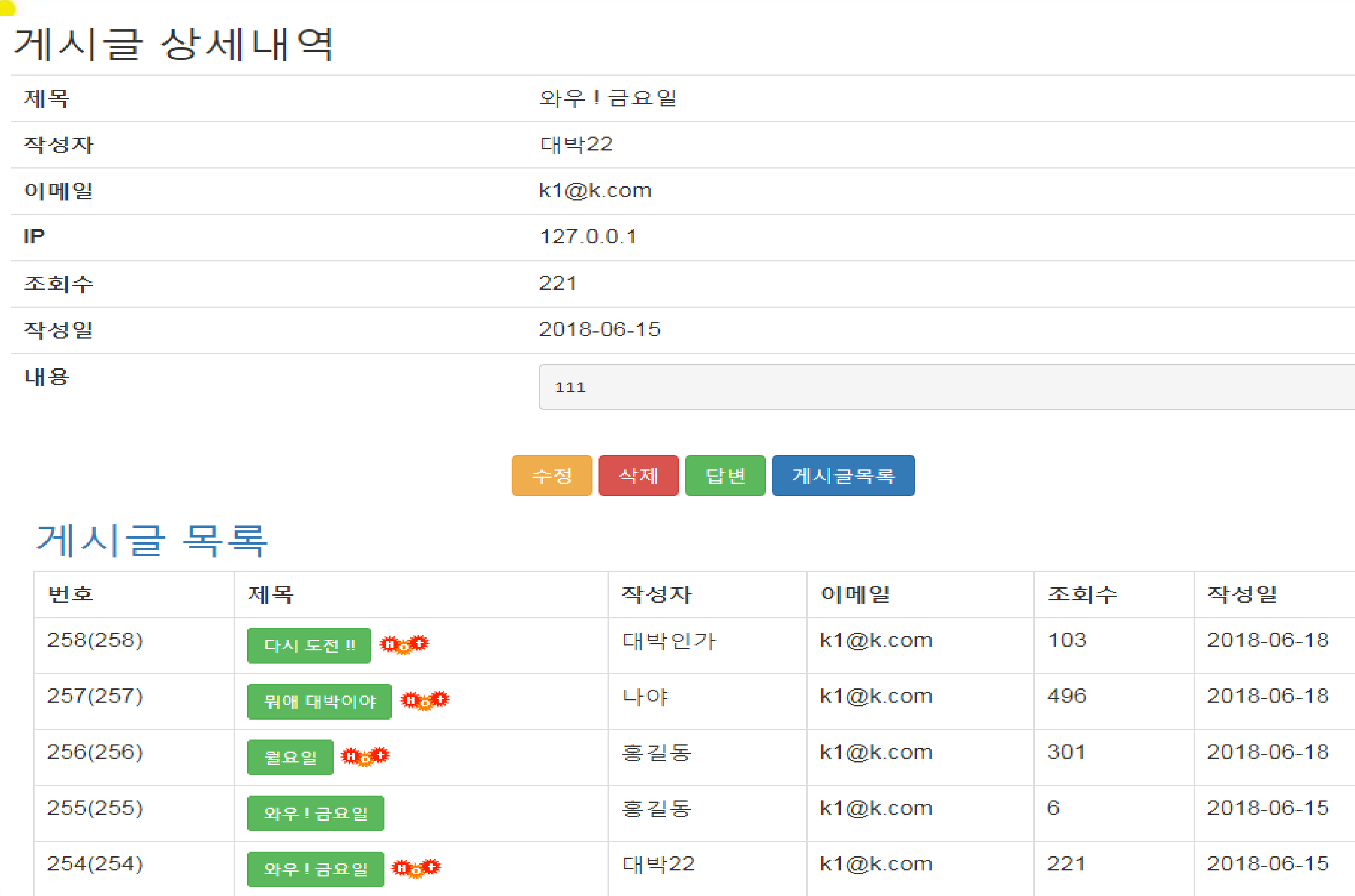
답변 게시판

```
create table board (
    num number primary key, -- key
    writer varchar2(20) not null, -- 작성자
    subject varchar2(50) not null, -- 제목
    content varchar2(500) not null, -- 본문
    readcount number default 0, -- 읽은 횟수
    password varchar2(100) not null, -- 암호
    ref number not null, -- 답변글끼리 그룹
    re_step number not null, -- ref내의 순서
    re_level number not null, -- 들여쓰기
    reg_date date not null, -- 작성일
    del char(1)
);
```



게시글 목록

번호	제목	작성자	이메일	조회수	작성일
258(258)	다시 도전 !! Hot	대박인가	k1@k.com	103	2018-06-18
257(257)	뭐얘 대박이야 Hot	나야	k1@k.com	496	2018-06-18
256(256)	월요일 Hot	홍길동	k1@k.com	301	2018-06-18
255(255)	와우 ! 금요일	홍길동	k1@k.com	6	2018-06-15
254(254)	와우 ! 금요일 Hot	대박22	k1@k.com	220	2018-06-15
253(253)	와우 ! 금요일	대박22	k1@k.com	0	2018-06-15
252(250)	선거	홍길동	k1@k.com	4	2018-06-14
251(248)	대박	나야나	k1@k.com	4	2018-05-25
250(249)	Re 대박-1	1	k1@k.com	2	2018-05-25
249(235)	점심 Hot	1	k1@k.com	101	2018-05-23



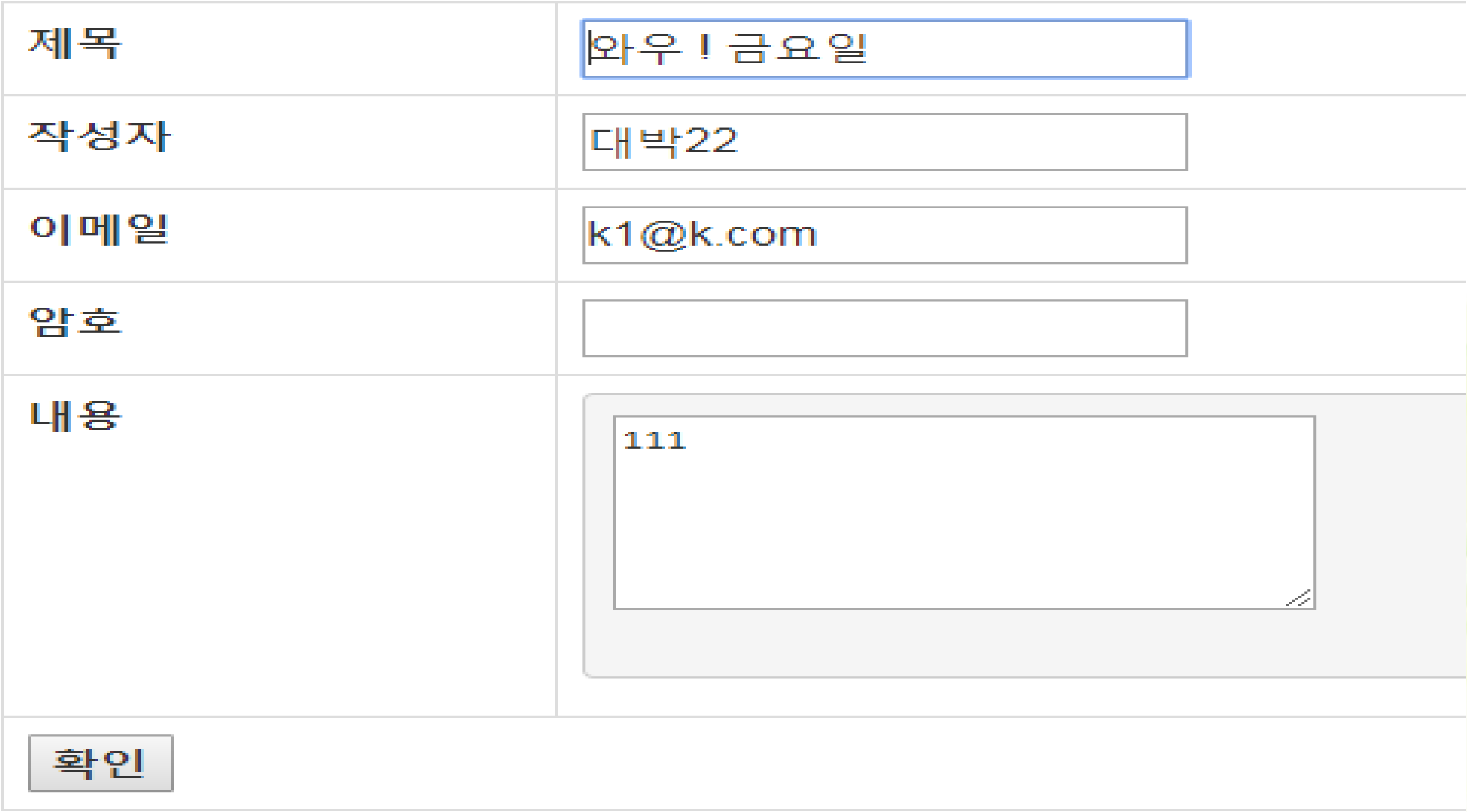
게시글 상세내역

제목	와우 ! 금요일
작성자	대박22
이메일	k1@k.com
IP	127.0.0.1
조회수	221
작성일	2018-06-15
내용	111

게시글 목록

번호	제목	작성자	이메일	조회수	작성일
258(258)	다시 도전 !! Hot	대박인가	k1@k.com	103	2018-06-18
257(257)	뭐애 대박이야 Hot	나야	k1@k.com	496	2018-06-18
256(256)	월요일 Hot	홍길동	k1@k.com	301	2018-06-18
255(255)	와우 ! 금요일	홍길동	k1@k.com	6	2018-06-15
254(254)	와우 ! 금요일 Hot	대박22	k1@k.com	221	2018-06-15

게시글 수정



제목	와우 ! 금요일
작성자	대박22
이메일	k1@k.com
암호	
내용	111

configuration.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
    <typeAliases>
        <typeAlias alias="board" type="com.ch.ch10.model.Board" />
    </typeAliases>
    <mappers>
        <mapper resource="board.xml" />
    </mappers>
</configuration>
```

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="boardns">
	<!-- Use type aliases to avoid typing the full classname every time. -->
	<resultMap id="boardResult" type="board">
		<result property="num" column="num" />
		<result property="writer" column="writer" />
		<result property="subject" column="subject" />
		<result property="content" column="content" />
		<result property="email" column="email" />
		<result property="readcount" column="readcount" />
		<result property="password" column="password" />
		<result property="ref" column="ref" />
		<result property="re_step" column="re_step" />
		<result property="re_level" column="re_level" />
		<result property="ip" column="ip" />
		<result property="reg_date" column="reg_date" />
		<result property="del" column="del" />
	</resultMap>
```

board.xml

```
<select id="list" parameterType="board" resultMap="boardResult">
	select * from (select rowNum rn, a.* from (
	select * from board1
	<where>
		<if test="keyword!=null and keyword!=''">
		<if test="search=='subcon'">
			subject like '%'||#{keyword}||'%' or
			content like '%'||#{keyword}||'%'
		</if>
		<if test="search!='subcon'">
			${search} like '%'||#{keyword}||'%'
		</if>
		</if>
	</where>
	order by ref desc, re_step) a)
	where rn between #{startRow} and #{endRow}
</select>
```

board.xml

```
<select id="getTotal" parameterType="board" resultType="int">
    select count(*) from board1
    <where>
        <if test="keyword!=null and keyword!=''">
        <if test="search=='subcon'">
            subject like '%'||#{keyword}||'%' or
            content like '%'||#{keyword}||'%'
        </if>
        <if test="search!='subcon'">
            ${search} like '%'||#{keyword}||'%'
        </if>
        </if>
    </where>
</select>
<select id="maxNum" resultType="int">
    select nvl(max(num),0) + 1 from board1
</select>
<select id="select" parameterType="int" resultType="board">
    select * from board1 where num=#{num}
</select>
```

board.xml

```
    <insert id="insert" parameterType="board">
        insert into board1 values (#{num},#{writer},#{subject},
        #{content},#{email},0,#{passwd},#{ref},#{re_step},
        #{re_level},#{ip},sysdate,'n')
    </insert>
    <update id="updateReadcount" parameterType="int">
        update board1 set readcount = readcount+1 where num=#{num}
    </update>
    <update id="update" parameterType="board">
        update board1 set subject=#{subject}, writer=#{writer},
        content=#{content}, email=#{email}
        where num=#{num}
    </update>
    <update id="delete" parameterType="int">
        update board1 set del='y' where num=#{num}
    </update>
    <update id="updateRe_step" parameterType="board">
        update board1 set re_step=re_step+1
        where ref=#{ref} and re_step > #{re_step}
    </update>
</mapper>
```

Board.java

```
public class Board {
      private int num;                      private String writer;
      private String subject;               private String content;
      private String email;                 private int readcount;
      private String password;              private int ref;
      private int re_step;                  private int re_level;
      private String ip;                    private Date reg_date;
      private String del;
      // paging 시작
      private int startRow;                 private int endRow;
      public int getStartRow() { return startRow; }
      public void setStartRow(int startRow) { this.startRow = startRow; }
      public int getEndRow() {return endRow;}
      public void setEndRow(int endRow) {    this.endRow = endRow;   }
      // 검색 시작
      private String search;                private String keyword;
      public String getSearch() {           return search;    }
      public void setSearch(String search) {   this.search = search;        }
      public String getKeyword() {  return keyword;  }
      public void setKeyword(String keyword) { this.keyword = keyword;  }
//   setter and getter 추가
```

PagingBean.java

```
package com.ch.ch10.service;
public class PagingBean {
        private int currentPage;private int rowPerPage;
        private int total;private int pagePerBlock = 10;
        private int totalPage;       private int startPage;
        private int endPage;
        public PagingBean(int currentPage, int rowPerPage, int total) {
                this.currentPage = currentPage;
                this.rowPerPage   = rowPerPage;
                this.total = total;
                totalPage = (int)Math.ceil((double)total/rowPerPage);
                startPage = currentPage - (currentPage - 1)%pagePerBlock;
                endPage = startPage + pagePerBlock -1;
                if (endPage > totalPage) endPage = totalPage;
        }

        # getter and setter method
}
```

BoardService.java

```
import java.util.List;
import com.ch.ch10.model.Board;
public interface BoardService {
	// List<Board> list(int startRow, int endRow);
	List<Board> list(Board board);
	int getTotal(Board board);
	int insert(Board board);
	int maxNum();
	void updateReadCount(int num);
	Board select(int num);
	int update(Board board);
	int delete(int num);
	void updateRe_step(Board board);

}
```

BoardServiceImpl.java

```
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.ch.ch10.dao.BoardDao;
import com.ch.ch10.model.Board;
@Service
public class BoardServiceImpl implements BoardService {
        @Autowired
        private BoardDao bd;
//      public List<Board> list(int startRow, int endRow) {
//              return bd.list(startRow, endRow);
//      }
        public List<Board> list(Board board) {
                return bd.list(board);
        }
        public int getTotal(Board board) {
                return bd.getTotal(board);
        }
        public int insert(Board board) {
                return bd.insert(board);
        }
```

BoardServiceImpl.java

```
	public int maxNum() {
		return bd.maxNum();
	}
	public void updateReadCount(int num) {
		bd.updateReadCount(num);
	}
	public Board select(int num) {
		return bd.select(num);
	}
	public int update(Board board) {
		return bd.update(board);
	}
	public int delete(int num) {
		return bd.delete(num);
	}
	@Override
	public void updateRe_step(Board board) {
		bd.updateRe_step(board);
	}
}
```

BoardDao.java

```
package com.ch.ch10.dao;
import java.util.List;
import com.ch.ch10.model.Board;
public interface BoardDao {
            // List<Board> list(int startRow, int endRow);
            List<Board> list(Board board);
            int getTotal(Board board);
            int insert(Board board);
            int maxNum();
            void updateReadCount(int num);
            Board select(int num);
            int update(Board board);
            int delete(int num);
            void updateRe_step(Board board);

}
```

BoardDaoImpl.java

```
import org.mybatis.spring.SqlSessionTemplate;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Repository;
import com.ch.ch10.model.Board;
@Repository
public class BoardDaoImpl implements BoardDao {
        @Autowired
        private SqlSessionTemplate sst;
//      public List<Board> list(int startRow, int endRow) {
//              HashMap<String, Integer> hm = new HashMap<String, Integer>();
//              hm.put("startRow", startRow);
//              hm.put("endRow", endRow);
//              return sst.selectList("boardns.list", hm);
//      }
        public List<Board> list(Board board) {
                return sst.selectList("boardns.list", board);
        }
        public int getTotal(Board board) {
                return sst.selectOne("boardns.getTotal", board);
        }
        public int maxNum() {
                return sst.selectOne("boardns.maxNum");
        }
```

BoardDaoImpl.java

```
    public int insert(Board board) {
        return sst.insert("boardns.insert", board);
    }
    public void updateReadCount(int num) {
        sst.update("boardns.updateReadcount", num);
    }
    public Board select(int num) {
        return sst.selectOne("boardns.select", num);
    }
    public int update(Board board) {
        return sst.update("boardns.update", board);
    }
    public int delete(int num) {
        return sst.update("boardns.delete", num);
    }
    public void updateRe_step(Board board) {
        sst.update("boardns.updateRe_step", board);
    }
}
```

BoardController.java

```
@Controller
public class BoardController {
	@Autowired
	private BoardService bs;
	@RequestMapping("list")
	public String list(String pageNum, Board board, Model model) {
		if (pageNum==null || pageNum.equals("")) pageNum = "1";
		int currentPage = Integer.parseInt(pageNum);
		int total = bs.getTotal(board);  int rowPerPage = 10;
		int startRow = (currentPage - 1) * rowPerPage + 1;
		int endRow    = startRow + rowPerPage - 1;
		board.setStartRow(startRow);    board.setEndRow(endRow);
		// List<Board> list = bs.list(startRow, endRow);
		List<Board> list = bs.list(board);    int no = total - startRow + 1;
		PagingBean pb = new PagingBean(currentPage,rowPerPage,total);
		String[] t = {"작성자","제목","내용","제목+내용"};
		model.addAttribute("t", t);         model.addAttribute("board", board);
		model.addAttribute("list", list);   model.addAttribute("no", no);
		model.addAttribute("pb", pb);
		return "list";
	}
```

BoardController.java

```java
@RequestMapping("insertForm")
public String insertForm(String pageNum, String nm, Model model){
    int num=0, ref=0, re_level=0, re_step=0;
    // 답변글시작
    if (nm!=null && !nm.equals("")) {
        num = Integer.parseInt(nm);
        Board board = bs.select(num);
        ref = board.getRef();
        re_level = board.getRe_level();
        re_step  = board.getRe_step();
    }
    // 답변글 끝
    model.addAttribute("pageNum", pageNum);
    model.addAttribute("num", num);
    model.addAttribute("ref", ref);
    model.addAttribute("re_level", re_level);
    model.addAttribute("re_step", re_step);
    return "insertForm";
}
```

BoardController.java

```
@RequestMapping("insert")
public String insert(Board board, String pageNum, Model model,
        HttpServletRequest request) {
        board.setIp(request.getRemoteAddr());
        int number = bs.maxNum();
        // 답변글은 num이 0이 아님
        if (board.getNum() != 0) {
                bs.updateRe_step(board);
                board.setRe_level(board.getRe_level()+1);
                board.setRe_step(board.getRe_step()+1);
        } else
                board.setRef(number); // 답변글이 아닐때만 num고 일치
        board.setNum(number);
        int result = bs.insert(board);
        model.addAttribute("result", result);
        model.addAttribute("pageNum", pageNum);
        return "insert";
}
```

BoardController.java

```
@RequestMapping("view")
public String view(int num, String pageNum, Model model) {
        bs.updateReadCount(num);
        Board board = bs.select(num);
        model.addAttribute("board", board);
        model.addAttribute("pageNum", pageNum);
        return "view";
}
@RequestMapping("updateForm")
public String updateForm(int num, String pageNum, Model model) {
        Board board = bs.select(num);
        model.addAttribute("pageNum", pageNum);
        model.addAttribute("board", board);
        return "updateForm";
}
```

BoardController.java

```java
    @RequestMapping("update")
    public String update(Board board, String pageNum, Model model) {
        int result = bs.update(board);
        model.addAttribute("result", result);
        model.addAttribute("pageNum", pageNum);
        return "update";
    }
    @RequestMapping("deleteForm")
    public String deleteForm(int num, String pageNum, Model model) {
        Board board = bs.select(num);
        model.addAttribute("board", board);
        model.addAttribute("pageNum", pageNum);
        return "deleteForm";
    }
    @RequestMapping("delete")
    public String delete(int num, String pageNum, Model model) {
        int result = bs.delete(num);
        model.addAttribute("result", result);
        model.addAttribute("pageNum", pageNum);
        return "delete";
    }
}
```

DataNotfound.jsp

```
<%@ page contentType="text/html;charset=UTF-8"%>
<%@ include file="../header.jsp"%>
<html><head><title>해당 데이터를 찾을 수 없습니다.</title>
</head>
<body>
<div align="center" class="body">
<h2><font color="red">해당 데이터를 찾을 수 없습니다.</font></h2>
<a href="main.html">■목록으로 돌아가기</a></div>
</body></html>
```

exception.jsp

```
<%@ page contentType="text/html;charset=UTF-8"%>
<%@ include file="../header.jsp"%>
<html><head><title>현재 쇼핑 애플리케이션은 이용할 수 없습니다.</title>
</head><body>
<div align="center" class="body">
<h2><font color="red">현재 쇼핑 애플리케이션은 이용할 수 없습니
다.</font></h2>
</div>
</body>  </html>
```

list.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ include file="header.jsp" %>
<!DOCTYPE html>
<html><head><meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
<title>Insert title here</title></head><body>
<div class="container">
      <h2 class="text-primary">게시글 목록</h2>
      <table class="table table-bordered table-hover">
            <tr><th>번호</th><th>제목</th><th>작성자</th><th>이메일</th>
            <th>조회수</th><th>작성일</th></tr>
            <c:if test="${empty list}">
                        <tr><th colspan="6">데이터가 없습니다</th></tr>
            </c:if>
```

list.jsp

```
<c:if test= "${not empty list}">
    <c:forEach var= "board" items= "${list}">
    <tr><td>${no}(${board.num})</td>
        <c:if test= "${board.del=='y'}">
                <td colspan= "5">삭제된 글입니다</td></c:if>
        <c:if test= "${board.del!='y' }"><td title= "${board.content}">
            <c:if test= "${board.re_level > 0}">
                <img alt= "" src= "images/level.gif" height= "5"
                        width= "${board.re_level*10}">
                <img alt= "" src= "images/re.gif"></c:if>
            <a class= "btn btn-success btn-sm"
               href= "view.do?num=${board.num}&pageNum=${pb.currentPage}">
               ${board.subject}</a>
            <c:if test= "${board.readcount > 50}">
                <img alt= "" src= "images/hot.gif">
            </c:if></td>
            <td>${board.writer }</td><td>${board.email }</td>
            <td>${board.readcount }</td><td>${board.reg_date }</td>
        </c:if></tr>
    <c:set var= "no" value= "${no - 1}"></c:set>
</c:forEach></c:if></table>
```

list.jsp

```
<div align="center"><ul class="pagination">
<c:if test="${pb.startPage > pb.pagePerBlock }">
<li><a href=
        "list.do?pageNum=1&search=${board.search}&keyword=${board.keyword}">
        <span class="glyphicon glyphicon-step-backward"></span></a></li>
<li><a href="list.do?pageNum=${pb.startPage-
1}&search=${board.search}&keyword=${board.keyword}">
<span class="glyphicon glyphicon-triangle-left"></span>
</a></li>
</c:if>
<c:forEach var="i" begin="${pb.startPage}" end="${pb.endPage}">
<c:if test="${i!=pb.currentPage }">
<li><a href=
"list.do?pageNum=${i}&search=${board.search}&keyword=${board.keyword}">${i}</a>
</li>
</c:if>
<c:if test="${i==pb.currentPage }">
<li class="active"><a
href="list.do?pageNum=${i}&search=${board.search}&keyword=${board.keyword}">${i
}</a></li>
</c:if>
</c:forEach>
```

list.jsp

```
<c:if test="${pb.endPage < pb.totalPage }">
<li><a
href="list.do?pageNum=${pb.endPage+1}&search=${board.search}&keywo
rd=${board.keyword}">
<span class="glyphicon glyphicon-triangle-right"></span>
</a></li>
</c:if>
<li><a
href="list.do?pageNum=${pb.totalPage}&search=${board.search}&keywo
rd=${board.keyword}">
<span class="glyphicon glyphicon-step-forward"></span>
</a></li>
</ul></div>
<div>
<form action="list.do">
<input type="hidden" name="pageNum" value="1">
<!--<select name="search">
<option value="writer">작성자</option>
<option value="subject">제목</option>
<option value="content">내용</option>
<option value="subcon">제목+내용</option>
</select> -->
```

list.jsp

```
<select name="search">
<c:forTokens var="sh" items="writer,subject,content,subcon"
 delims="," varStatus="i">
 <c:if test="${sh!=board.search}">
 <option value="${sh}">${t[i.index]}</option>
 </c:if>
<c:if test="${sh==board.search}">
 <option value="${sh}"
 selected="selected">${t[i.index]}</option>
 </c:if>
</c:forTokens>
</select>
<input type="text" name="keyword">
<input type="submit" value="확인">
</form>
</div>
<a href="insertForm.do?pageNum=1" class="btn btn-primary">게시글
입력</a>
</div>
</body></html>
```

view.jsp

```
<%@ include file="header.jsp" %>
<title>Insert title here</title>
<script type="text/javascript">
	$(function() {
		$('#boardDisp').load('list.do?pageNum=${pageNum}');
	});
</script></head><body>
<div class="container">
	<h2>게시글 상세내역</h2>
	<table class="table table-board table-hover">
		<tr><th>제목</th><td>${board.subject}</td></tr>
		<tr><th>작성자</th><td>${board.writer}</td></tr>
		<tr><th>이메일</th><td>${board.email}</td></tr>
		<tr><th>IP</th><td>${board.ip}</td></tr>
		<tr><th>조회수</th><td>${board.readcount}</td></tr>
		<tr><th>작성일</th><td>${board.reg_date}</td></tr>
		<tr><th>내용</th><td><pre>${board.content}</pre></td></tr>
</table>
```

view.jsp

```
    <div align="center">
          <a class="btn btn-warning"
            href="updateForm.do?num=${board.num}&pageNum=${pageNum}">
          수정</a>
          <a class="btn btn-danger"
            href="deleteForm.do?num=${board.num}&pageNum=${pageNum}">삭
          제</a>
          <a class="btn btn-success"
           href="insertForm.do?pageNum=${pageNum}&nm=${board.num}">답변
          </a>
          <a class="btn btn-primary" href="list.do?pageNum=${pageNum}">게시
          글목록</a>
     </div>
     <div id="boardDisp"></div>
</div>
</body>
</html>
```

updateForm.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ include file="header.jsp" %>
<!DOCTYPE html>
<html><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Insert title here</title>
<script type="text/javascript">
      function chk() {
            if (frm.passwd.value != frm.passwd2.value) {
                  alert("암호가 다릅니다");frm.passwd.focus();
                  frm.passwd.value = "";  return false;
            }
      }
</script></head><body>
<div class="container">
<h2>게시글 수정</h2>
```

updateForm.jsp

```
<form action="update.do" method="post" name="frm"  onsubmit="return chk()">
    <input type="hidden" name="pageNum" value="${pageNum}">
    <input type="hidden" name="num" value="${board.num}">
    <input type="hidden" name="passwd2" value="${board.passwd}">
    <table class="table table-bordered table-hover">
        <tr><th>제목</th><td><input type="text" name="subject"
        required="required" autofocus="autofocus"
        value="${board.subject}"></td></tr>
        <tr><th>작성자</th><td><input type="text" name="writer"
        required="required" value="${board.writer}"></td></tr>
        <tr><th>이메일</th><td><input type="email" name="email"
        required="required" value="${board.email}"></td></tr>
        <tr><th>암호</th><td><input type="password" name="passwd"
        required="required"></td></tr>
        <tr><th>내용</th><td><pre><textarea rows="5" cols="30"
        required="required" name="content">${board.content}</textarea>
        </pre></td></tr>
        <tr><th colspan="2"><input type="submit" value="확인"></th></tr>
    </table></form>
</div></body></html>
```

update.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ include file="header.jsp" %>
<title>Insert title here</title></head>
<body>
<c:if test="${result > 0 }">
    <script type="text/javascript">
        alert("수정 대박 ㅋㅋ");
        location.href="list.do?pageNum=${pageNum}";
    </script>
</c:if>
<c:if test="${result == 0 }">
    <script type="text/javascript">
        alert("수정 실패 ㅠㅠ");
        history.go(-1);
    </script>
</c:if>
</body></html>
```

insertForm.jsp

```
<div class="container">
<h2>게시글 입력</h2>
<form action="insert.do" method="post">
    <input type="hidden" name="pageNum" value="${pageNum}">
    <input type="hidden" name="num" value="${num}">
    <input type="hidden" name="ref" value="${ref}">
    <input type="hidden" name="re_level" value="${re_level}">
    <input type="hidden" name="re_step" value="${re_step}">
    <table class="table table-bordered table-hover">
        <tr><th>제목</th><td><input type="text" name="subject"
        required="required" autofocus="autofocus"></td></tr>
        <tr><th>작성자</th><td><input type="text" name="writer"
        required="required"></td></tr>
        <tr><th>이메일</th><td><input type="email" name="email"
        required="required"></td></tr>
        <tr><th>암호</th><td><input type="password" name="passwd"
        required="required"></td></tr>
        <tr><th>내용</th><td><textarea rows="5" cols="30"
        required="required" name="content"></textarea></td></tr>
        <tr><th colspan="2"><input type="submit" value="확인"></th></tr>
    </table></form></div>
```

설정파일에서 SqlSessionTemplate생성

```
<!-- 스프링 jdbc 즉 스프링으로 oracle 디비 연결 -->
<bean id="sqlSessionFactory"
    class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource" />
    <property name="configLocation"
    value="classpath:configuration.xml" />
</bean>
<bean id="session" class="org.mybatis.spring.SqlSessionTemplate">
    <constructor-arg index="0" ref="sqlSessionFactory" />
</bean>
<context:property-placeholder location="classpath:db.properties" />
<bean id="dataSource"
    class="com.mchange.v2.c3p0.ComboPooledDataSource"
    destroy-method="close">
    <property name="driverClass" value="${driver}" />
    <property name="jdbcUrl" value="${url}" />
    <property name="user" value="${userId}" />
    <property name="password" value="${password}" />
    <property name="maxPoolSize" value="${maxPoolSize}" />
</bean>
```

AOP활용

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:context="http://www.springframework.org/schema/context"
xmlns:mvc="http://www.springframework.org/schema/mvc"
xmlns:aop="http://www.springframework.org/schema/aop"
xmlns:tx="http://www.springframework.org/schema/tx"
xsi:schemaLocation="http://www.springframework.org/schema/tx
http://www.springframework.org/schema/tx/spring-tx-3.1.xsd
        http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans.xsd
http://www.springframework.org/schema/aop
http://www.springframework.org/schema/aop/spring-aop-3.1.xsd
http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context-3.1.xsd
http://www.springframework.org/schema/mvc
        http://www.springframework.org/schema/mvc/spring-mvc-3.1.xsd">
<context:component-scan base-package="controllers,dao,logic,service" />
<mvc:annotation-driven />
```

```
<!-- Data Source -->
<context:property-placeholder location="classpath:mybatis/db.properties"/>
<bean id="dataSource"
class="org.springframework.jdbc.datasource.SimpleDriverDataSource">
<property name="driverClass" value="${driver}" />
<property name="url" value="${url}" />
<property name="username" value="${user.username}" />
<property name="password" value="${user.password}" />
</bean>
<!-- 스프링 jdbc 즉 스프링으로 oracle 디비 연결 -->
<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
<property name="dataSource" ref="dataSource" />
<property name="configLocation" value="classpath:mybatis/configuration.xml" />
</bean>

<!-- transactionmanager 선언 -->
<bean id="transactionManager"
class="org.springframework.jdbc.datasource.DataSourceTransactionManager">
<property name="dataSource" ref="dataSource" />
</bean>
```

```
<!-- Transaction 기본 설정 -->
<tx:advice id= "txAdvice" transaction-manager="transactionManager">
<tx:attributes>
<tx:method name= "*" propagation="REQUIRED"/>
 <!-- name속성값을 *로 지정하면 모든 메서드에 적용 즉 어떠한 메서드를 사용해도
된다
. 트랜잭션 속성 REQUIRED 를 적용. 즉 값을 가져오는 역할을 함. -->
<tx:method name= "get*" read-only="true"/>
<!--  get 으로 시작하는 모든 메서드를 받아들인다. 그리고 단지 읽기 속성을 하기
위해서  true로 설정 -->
</tx:attributes>
</tx:advice>

<!-- AOP 설정을 사용해서 Transaction 적용 -->
<aop:config>
<aop:advisor advice-ref= "txAdvice" pointcut="execution(* service.*Service.*(..))"/>
<!-- service 패키지의  Service 포함하는 모든 클래스,인터페이스의 모든 메서드를
포인트 컷 한다.   -->
</aop:config>
<!-- 이 설정은 검색해서 마이바티스 매퍼 XML파일을 모두 로드할 것이다. -->
<bean id= "session" class="org.mybatis.spring.SqlSessionTemplate">
<constructor-arg index= "0" ref="sqlSessionFactory" />
</bean>
</beans>
```

메일 보내기

- 메일을 보내기 위해서는 메일서버를 구축해야 함
- 그러나 개발자들이 자신의 PC에서 개발한 프로그램은 운영서버에서 실행
- 개발자가 자신의 PC에 메일서버 구축은 실제 운영에서 사용하지 않음
- 운영서버의 메일서버를 사용해야 함
- 여기서는 네이버 메일 서버를 사용하여 보내는 방법을 설명
- 네이버나 다음메일을 이용할 때는 그 회사의 개발자 센터를 방문하여 개발자로 등록
- 그곳에서 제공하는 방법으로 사용할 것

Java 보안 해결

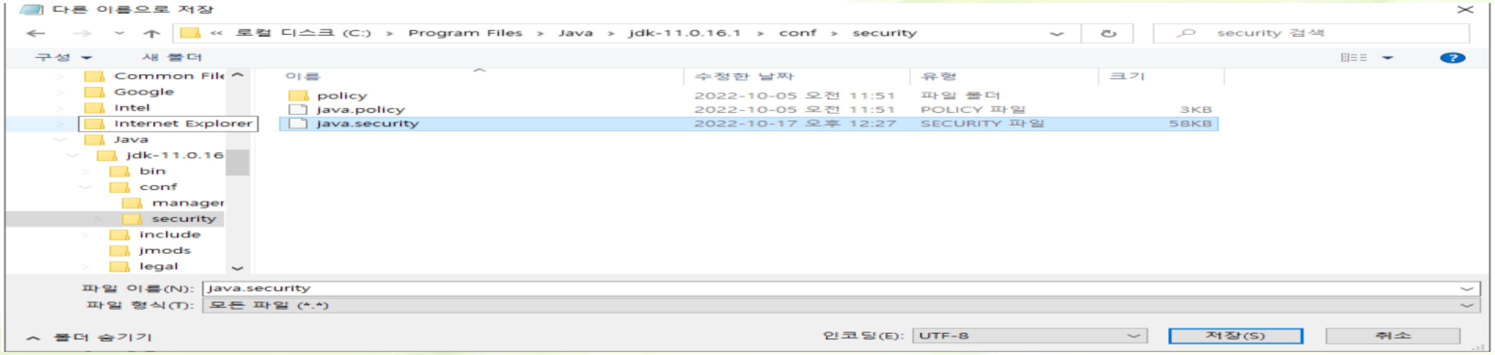
이메일에 Connection fail인 경우
Java보안에 관련 된 문제이다

해결 방법은

1. JDK가 설치된 폴더의 경로로 이동한다.
2. C:₩Program Files₩Java₩jdk-11.0.16.1₩conf₩security에서 java security 파일을 수정한다.
3. **jdk.tls.disabledAlgorithms** 을 검색해서 나오는 부분 중 TLS 1, TLS1.1을 지워주면 된다.

jdk.tls.disabledAlgorithms=SSLv3, ~~TLSv1, TLSv1.1~~, RC4, DES, MD5withRSA
TLSv1, TLSv1.1 지우기

4. 이후, 저장을 하려고 하면 "액세스가 거부되었다" 라는 에러와 함께 저장이 안될 수도 있는데, 바탕화면 저장하는 데 파일형식을 모든 파일을 선택해야 한다



이름	수정한 날짜	유형	크기
policy	2022-10-05 오전 11:51	파일 폴더	
java.policy	2022-10-05 오전 11:51	POLICY 파일	3KB
java.security	2022-10-17 오후 12:27	SECURITY 파일	58KB

파일 이름(N): java.security
파일 형식(T): 모든 파일 (*.*)
인코딩(E): UTF-8

Pom.xml에 추가할 내용

```
 <!-- javax.mail이 포함된 디펜던시 -->
 <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context-support</artifactId>
            <version>4.1.3.RELEASE</version>
</dependency>
 <dependency>
             <groupId>javax.mail</groupId>
             <artifactId>mail</artifactId>
             <version>1.4.7</version>
</dependency>
</dependencies>
```

설정파일(service-context.xml)에 추가

```
<!-- SET default mail properties -->
 <bean id="jMailSender"
class="org.springframework.mail.javamail.JavaMailSenderImpl">
        <property name="host" value="smtp.naver.com"/>
        <property name="port" value="465"/>
        <property name="username" value="kbj010"/>
        <property name="password" value=“xxxxxxxx"/>
        <property value="smtps" name="protocol"/>
        <property value="utf-8" name="defaultEncoding"/>
        <property name="javaMailProperties">
            <props>
                <prop key="mail.smtp.auth">true</prop>
                <prop key="mail.smtp.starttls.enable">true</prop>
                <prop key="mail.smtps.ssl.checkserveridentity">true</prop>
               <prop key="mail.smtps.ssl.trust">*</prop>
             <!--     <prop key="mail.debug">true</prop>  -->
            </props>
        </property>
    </bean>
```

Email

```
package com.ch.ch10.model;
public class Email {
    private String email;
    private String title;
    private String content;

    public String getEmail() {return email;}
    public void setEmail(String email) {
        this.email = email;
    }
    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;
    }
}
```

emailForm.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ include file="../header.jsp" %>
<title>Insert title here</title></head><body>
<div class="container">
<h2 class="text-primary">메일작성</h2>
<form action="email.do" method="post">
<table class="table table-bordered table-hover">
      <tr><th>받는사람</th><td><input type="email" name="email"
            required="required" autofocus="autofocus"></td></tr>
      <tr><th>제목</th><td><input type="text" name="title"
            required="required"></td></tr>
      <tr><th>내용</th><td><textarea rows="5" cols="30" name="content"
            required="required"></textarea></td></tr>
      <tr><th colspan="2"><input type="text" value="보내기"></th></tr>
</table></form>
</div>
</body>
</html>
```

EtcConytroller

EmailController.java

```
@Autowired
	JavaMailSender jMailSender;
	@RequestMapping("email")
	public String email(Email email, Model model) {
		MimeMessage mm = jms.createMimeMessage();
		try{MimeMessageHelper mmh =
				new MimeMessageHelper(mm, true, "utf-8");
			mmh.setSubject(email.getTitle());
			mmh.setText(email.getContent());
			mmh.setTo(email.getEmail());
			mmh.setFrom("kbj010@naver.com");
			jms.send(mm);
			model.addAttribute("message", "메일보내기 성공");
		}catch(Exception e) {
				System.out.println(e.getMessage());
				model.addAttribute("message", "뷁 실패");
		}
		return "etc/email";
	}
}
```

JSON test

```java
@Controller
public class EtcController {
	@Autowired
	private JavaMailSenderImpl jms;

	@RequestMapping("emailForm")
	public String emailForm() {
		return "etc/emailForm";
	}
	@RequestMapping("json")
	public String test() {
		return "etc/json";
	}
```

pom.xml에추가

```xml
<!-- restcontroller -->
<dependency>
<groupId>com.fasterxml</groupId>
<artifactId>classmate</artifactId>
<version>1.4.0</version>
</dependency>
 <dependency>
<groupId>com.fasterxml.jackson.core</groupId>
<artifactId>jackson-databind</artifactId>
<version>2.9.2</version>
</dependency
```

json.jsp

```
<script type="text/javascript">
 $(function() {
     $.getJSON("http://localhost:8080/ch10/jsonBoard.do",
             function(data) {
             $.each(data, function() {
                          $('#treeData').append(
      '<tr><td>'+this.subject+'</td>'+
      '<td>'+this.writer+'</td>'+
      '<td>'+this.reg_date+'</td>'+
      '<td>'+this.readcount+'</td></tr>');
                    });
      });
  });
</script></head><body><div class="container">
<h2 class="text-primary">게시글 목록</h2>
<table class="table table-bordered" id="treeData">
      <tr><td>제목</td><td>작성자</td><td>작성일</td>
             <td>조회수</td></tr>
</table></div>
```

RestBoardController

```
package com.ch.ch10.controller;
import java.util.Collection;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;
import com.ch.ch10.model.Board;
import com.ch.ch10.service.BoardService;
@RestController
public class RestBoardController {
        @Autowired
        private BoardService bs;
        @RequestMapping("jsonBoard")
        public Collection<Board> jsonBoard(Board board) {
                board.setStartRow(1);
                board.setEndRow(10);
                Collection<Board> list = bs.list(board);
                return list;
        }
}
```