

UI아키텍처설계 포트폴리오 평가과제

김보성

developBo@naver.com
한국ICT인재개발원 - 2021.02.19

스프링(Spring) 프레임워크와 마이바티스(Mybatis)를 연동하는 서비스

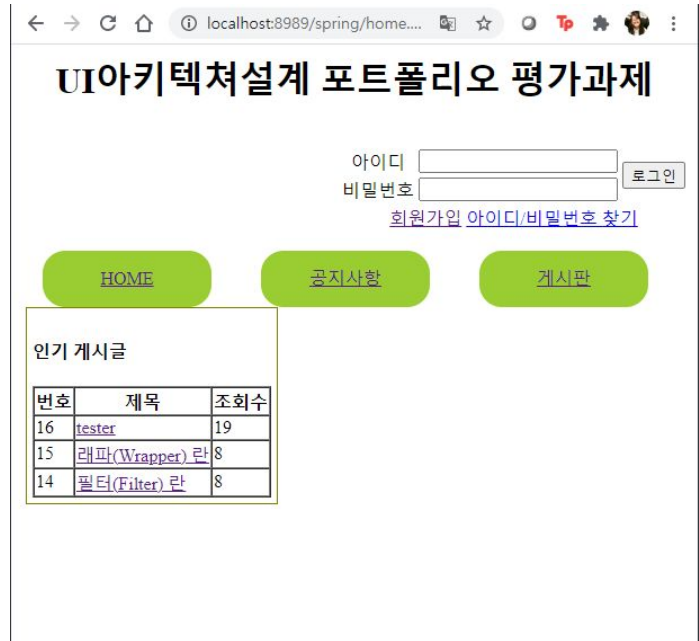
실행 흐름 :

실행 흐름 : 게시글 목록조회, 게시글 목록 페이지별로 출력 처리,
게시글 상세보기/첨부파일 다운로드/게시글 수정/게시글 삭제, 새
게시글 등록/파일첨부, 인기 게시글 3개 조회, 페이지 처리가 적용된
게시글 검색 서비스가 구동되게 코드를 구현하시오.

1. home.jsp

- jquery ajax 기능을 사용하여, 페이지 로딩시 조회수 많은 인기 게시글 3개를 조회해서 테이블로 출력 및 제목 클릭시 상세보기로 연결함

```
home.jsp | menuBar.jsp
41 //조회수 많은 인기 게시물 3개 자동 출력되게 함
42 $.ajax({
43   url: "${ pageContext.request.contextPath }/btop3.do",
44   type: "post",
45   dataType: "json",
46   success: function(data){
47     console.log("success : " + data);
48
49     //object ==> string
50     var jsonStr = JSON.stringify(data);
51     //string ==> json
52     var json = JSON.parse(jsonStr);
53
54     var values = "";
55     for(var i in json.list){
56       values += "<tr><td>" + json.list[i].bid
57         + "</td><td><a href='${ pageContext.request.contextPath }/bdetail.do?bid="
58         + json.list[i].bid + "'>"
59         + decodeURIComponent(json.list[i].btitle).replace(/&/gi, " ")
60         + "</a></td><td>" + json.list[i].bcount
61         + "</td></tr>";
62     } //for in
63
64     $("#top_board").html($("#top_board").html() + values);
65   },
66   error: function(jqXHR, textStatus, errorThrown){
67     console.log("error : " + jqXHR + ", " + textStatus + ", "
68       + errorThrown);
69   }
70 });
71 }); //jquery document ready
72 </script>
```



2. filedownloadview.java

- 게시글의 첨부파일을 다운로드 처리하는 뷰 클래스 작성함.

```

240
249 //게시글 첨부파일 다운로드 요청 처리용
250 @RequestMapping("bfdown.do")
251 public ModelAndView fileDownloadMethod(
252     @RequestParam("ofile") String originalFilename,
253     @RequestParam("rfile") String renameFilename,
254     HttpServletRequest request) {
255     String savePath = request.getSession().getServletContext()
256         .getRealPath("resources/board_files");
257     File renameFile = new File(savePath + "\\\" + renameFilename);
258
259     Map<String, Object> model = new HashMap<String, Object>();
260     model.put("renameFile", renameFile);
261     model.put("originalFilename", originalFilename);
262     return new ModelAndView("filedown2", "downFile", model);
263 }
264
265

```

```

17
18 <!-- Resolves views selected for rendering by @Controllers to .jsp resources in the /WEB-INF/views directory -->
19 <beans:bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
20     <beans:property name="prefix" value="/WEB-INF/views/" />
21     <beans:property name="suffix" value=".jsp" />
22     <beans:property name="order" value="1" />
23 </beans:bean>
24
25 <!-- 파일 다운을 위한 뷰리플버 지정 (JsonView 도 해당됨)
26 주의 : id 속성을 지정하지 않는다.
27 여러 개의 뷰리플버를 지정할 때는 실행 순서를 정해야 함
28 파일 다운 뷰리플버가 반드시 맨 처음 실행되게 순서를 지정해야 함
29 즉, 컨트롤러 메소드가 리턴하는 뷰이름과 동일한 뷰클래스(bean)가 있으면
30 자동으로 해당 뷰클래스(파일다운로드 처리용 뷰클래스)를 실행함
31 -->
32 <beans:bean
33     class="org.springframework.web.servlet.view.BeanNameViewResolver">
34     <beans:property name="order" value="0" />
35 </beans:bean>
36

```

```

1 package com.ict.spring.common;
2
3 import java.io.File;
4
14
15 //스프링에서 BeanNameViewResolver 에 의해 실행되는 뷰클래스가 되려면
16 //반드시 스프링이 제공하는 AbstractView 를 상속받은 후손클래스여야 함
17 @Component("filedown")
18 public class FileDownloadView extends AbstractView {
19
20     @Override
21     protected void renderMergedOutputModel(Map<String, Object> model,
22         HttpServletRequest request,
23         HttpServletResponse response) throws Exception {
24         // 컨트롤러에서 뷰리플버로 리턴한 model 정보가 자동 전달됨
25         File downFile = (File)model.get("downFile");
26
27         //한글 파일명 인코딩 처리를 위해 파일명만 추출함
28         String fileName = downFile.getName();
29
30         response.setContentType("text/plain; charset=utf-8");
31         response.addHeader("Content-Disposition", "attachment; filename=\""
32             + new String(fileName.getBytes("utf-8"), "ISO-8859-1") + "\"");
33         response.setContentLength((int)downFile.length());
34
35         //저장된 볼더에서 대상 파일을 읽어서, 요청한 클라이언트 브라우저로 보내기 위한
36         //입출력 스트림 생성함
37         OutputStream out = response.getOutputStream();
38
39         try(FileInputStream fin = new FileInputStream(downFile)){
40             FileCopyUtils.copy(fin, out);
41         }catch(Exception e) {
42             e.printStackTrace();
43         }
44
45         out.flush();
46         out.close();
47     }
48 }
49
50

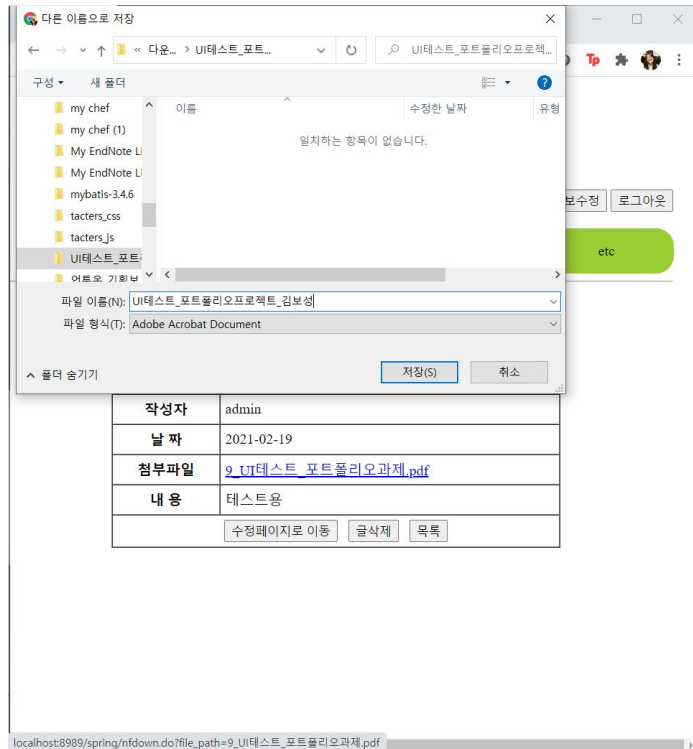
```

전 슬라이드의 코드 작성 결과

BoardController.java

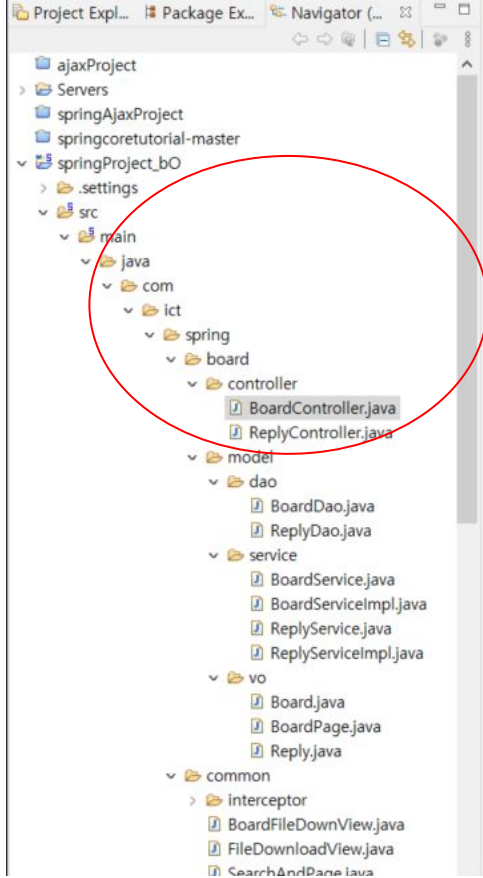
servlet-context.xml

filedownloadview.java



3. BoardController.java

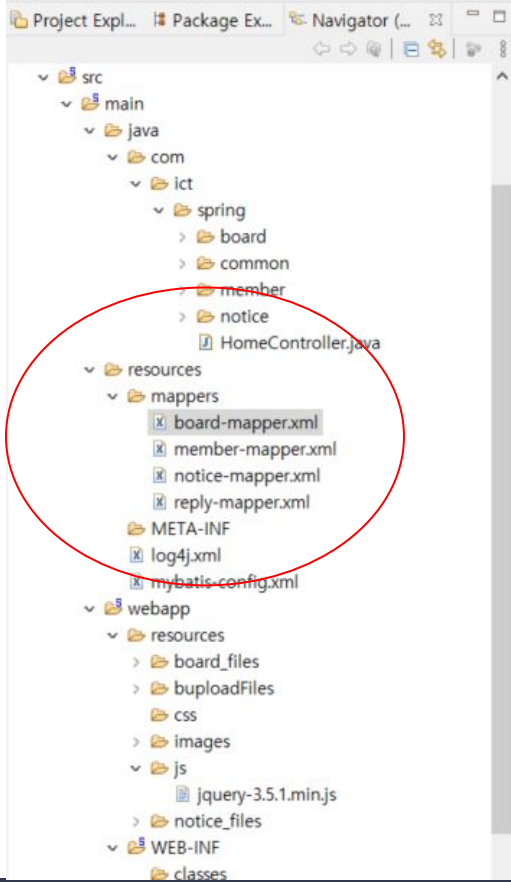
- src/main/java 폴더 아래 해당 패키지 아래에 컨트롤러 클래스 만듦



```
1 package com.ict.spring.board.controller;
2
3 import java.io.File;
4
5
6
7
8
9
10
11 @Controller
12 public class BoardController {
13     @Autowired
14     private BoardService boardService;
15
16     // 글쓰기 페이지 이동 요청 처리용
17     @RequestMapping("bwmove.do")
18     public String moveBoardWriteForm() {
19         return "board/boardWriteForm";
20     }
21
22     // 글 수정페이지로 이동 요청 처리용
23     @RequestMapping("bupview.do")
24     public String moveBoardUpdateView(@RequestParam("bid") int bid,
25         @RequestParam("page") int currentPage, Model model) {
26         Board board = boardService.selectBoard(bid);
27
28         if(board != null) {
29             model.addAttribute("board", board);
30             model.addAttribute("page", currentPage);
31             return "board/boardUpdateView";
32         } else {
33             model.addAttribute("msg", bid + "번 글 수정페이지로 이동 실패.");
34             return "common/errorPage";
35         }
36     }
37 }
```

4. board-mapper.xml

- src/main/resources/mappers 폴더 아래에 Board 테이블 sql 구문에 대한 엘리먼트들을 작성함.



```
board-mapper.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd" >
3 <mapper namespace="boardMapper">
4
5 <resultMap type="Board" id="resultBoard">
6   <id column="bid" property="bid"/>
7   <result column="btitle" property="btitle"/>
8   <result column="bwriter" property="bwriter"/>
9   <result column="bcontent" property="bcontent"/>
10  <result column="original_filename" property="original_filename"/>
11  <result column="rename_filename" property="rename_filename"/>
12  <result column="bcount" property="bcount"/>
13  <result column="b_create_date" property="b_create_date"/>
14  <result column="b_modify_date" property="b_modify_date"/>
15  <result column="b_status" property="b_status"/>
16 </resultMap>
17
18 <resultMap type="Board" id="resultTop3">
19   <id column="bid" property="bid"/>
20   <result column="btitle" property="btitle"/>
21   <result column="bcount" property="bcount"/>
22 </resultMap>
23
24 <!-- 비교연산자를 쿼리문에 그냥 사용하면, 마이바티스에서
25   태그 엘리먼트의 각식괄호 등(<, >, &) 와 혼동하는 문제가 발생할 수 있음
26   이 문제를 미리 방지하기 위해 xml 코드 중 xml parser 에 의해 해석하지 말고
27   문자 그대로 해석하라는 뜻으로 <![CDATA[ 쿼리문장 ]]> 사용함
28   -->
29
30 <select id="selectTop3" resultMap="resultTop3">
```

5. Board.java

- `src/main/java` 폴더 아래에 해당 패키지 안에 클래스 만듦
- `Serializable`, 기본/매개변수 있는 생성자, getters and setters, `toString()` overriding 작성함

```

package com.ict.spring.notice.model.vo;

import java.sql.Date;

public class Notice implements java.io.Serializable {
    private static final long serialVersionUID = 2222L;

    private int nid;
    private String ntitle;
    private String nwriter;
    private String ncontent;
    private Date n_create_date;
    private Date n_modify_date;
    private String file_path;

    public Notice() {}

    public Notice(int nid, String ntitle, String nwriter, String ncontent, Date n_create_date, Date
n_modify_date,
                String file_path) {
        super();
        this.nid = nid;
        this.ntitle = ntitle;
        this.nwriter = nwriter;
        this.ncontent = ncontent;
        this.n_create_date = n_create_date;
        this.n_modify_date = n_modify_date;
        this.file_path = file_path;
    }
}

```

```

    public int getNid() { return nid; }
    public void setNid(int nid) { this.nid = nid; }
    public String getNtitle() { return ntitle; }
    public void setNtitle(String ntitle) { this.ntitle = ntitle; }
    public String getNwriter() { return nwriter; }
    public void setNwriter(String nwriter) { this.nwriter = nwriter; }
    public String getNcontent() { return ncontent; }
    public void setNcontent(String ncontent) { this.ncontent = ncontent; }
    public Date getN_create_date() { return n_create_date; }
    public void setN_create_date(Date n_create_date) { this.n_create_date =
n_create_date; }
    public Date getN_modify_date() { return n_modify_date; }
    public void setN_modify_date(Date n_modify_date) { this.n_modify_date =
n_modify_date; }
    public String getFile_path() { return file_path; }
    public void setFile_path(String file_path) { this.file_path = file_path; }
    public static long getSerialversionuid() { return serialVersionUID; }
    @Override
    public String toString() {
        return "Notice [nid=" + nid + ", ntitle=" + ntitle + ", nwriter=" + nwriter + ",
ncontent=" + ncontent
                + ", n_create_date=" + n_create_date + ", n_modify_date=" +
n_modify_date + ", file_path=" + file_path
                + "]\n";
    }
}

```

6. NoticeService.java

- 인터페이스로 작성함

```
1 package com.ict.spring.board.model.service;
2
3 import java.util.ArrayList;
4
5
6
7
8
9
10 public interface BoardService {
11     ArrayList<Board> selectTop3();
12     int getListCount();
13     ArrayList<Board> selectList(int currentPage, int limit);
14     Board selectBoard(int bid);
15     int addReadCount(int bid);
16     int insertBoard(Board board);
17     int updateBoard(Board board);
18     int deleteBoard(int bid);
19
20     ArrayList<Board> selectSearchTitle(SearchAndPage searches);
21     ArrayList<Board> selectSearchWriter(SearchAndPage searches);
22     ArrayList<Board> selectSearchDate(SearchAndPage searches);
23
24     int getSearchTitleListCount(String keyword);
25     int getSearchWriterListCount(String keyword);
26     int getSearchDateListCount(SearchDate dates);
27
28 }
29
```

7. BoardServiceImpl.java

- **BoardService** 인터페이스를 상속받아 선언된 메소드들을 오버라이딩 처리 함.
- **BoardDao** 의 메소드를 호출함


```

board-mapper.xml BoardServiceImpl.java BoardService.java
1 package com.ict.spring.board.model.service;
2
3 import java.util.ArrayList;
4
5 @Service("boardService")
6 public class BoardServiceImpl implements BoardService {
7     @Autowired
8     private BoardDao boardDao;
9
10    @Override
11    public ArrayList<Board> selectTop3() {
12        return boardDao.selectTop3();
13    }
14
15    @Override
16    public int getListCount() {
17        return boardDao.getListCount();
18    }
19
20    @Override
21    public ArrayList<Board> selectList(int currentPage, int limit) {
22        return boardDao.selectList(currentPage, limit);
23    }
24
25    @Override
26    public Board selectBoard(int bid) {
27        return boardDao.selectBoard(bid);
28    }
29
30    @Override
31    public int addReadCount(int bid) {
32        return boardDao.addReadCount(bid);
33    }
34
35    @Override
36    public int insertBoard(Board board) {
37        return boardDao.insertBoard(board);
38    }
39
40    <

```

```

board-mapper.xml BoardServiceImpl.java BoardService.java
51
52
53 @Override
54 public int deleteBoard(int bid) {
55     return boardDao.deleteBoard(bid);
56 }
57
58 @Override
59 public ArrayList<Board> selectSearchTitle(SearchAndPage searches) {
60     return boardDao.selectSearchTitle(searches);
61 }
62
63 @Override
64 public ArrayList<Board> selectSearchWriter(SearchAndPage searches) {
65     return boardDao.selectSearchWriter(searches);
66 }
67
68 @Override
69 public ArrayList<Board> selectSearchDate(SearchAndPage searches) {
70     return boardDao.selectSearchDate(searches);
71 }
72
73 @Override
74 public int getSearchTitleListCount(String keyword) {
75     return boardDao.getSearchTitleListCount(keyword);
76 }
77
78 @Override
79 public int getSearchWriterListCount(String keyword) {
80     return boardDao.getSearchWriterListCount(keyword);
81 }
82
83 @Override
84 public int getSearchDateListCount(SearchDate dates) {
85     return boardDao.getSearchDateListCount(dates);
86 }
87
88 }
89

```

8. BoardDao.java

- SqlSessionTemplate 를 사용하여
boardMapper 와 연동 실행하는 메소드를
작성함

```

board-mapper.xml  BoardServiceImpl.java  BoardService.java  BoardDao.java
1 package com.ict.spring.board.model.dao;
2
3 import java.util.ArrayList;
4
5 @Repository("boardDao")
6 public class BoardDao {
7     @Autowired
8     private SqlSessionTemplate sqlSession;
9
10    public ArrayList<Board> selectTop3() {
11        List<Board> list = sqlSession.selectList("boardMapper.selectTop3");
12        return (ArrayList<Board>)list;
13    }
14
15    public int getListCount() {
16        return sqlSession.selectOne("boardMapper.getListCount");
17    }
18
19    public ArrayList<Board> selectList(int currentPage, int limit) {
20        //전달된 값을 이용해서 출력할 시작행과 끝행들 계산함
21        int startRow = (currentPage - 1) * limit - 1;
22        int endRow = startRow + limit - 1;
23
24        List<Board> list = sqlSession.selectList("boardMapper.selectList",
25            new BoardPage(startRow, endRow));
26        return (ArrayList<Board>)list;
27    }
28
29    public Board selectBoard(int bid) {
30        return sqlSession.selectOne("boardMapper.selectBoard", bid);
31    }
32
33    public int addReadCount(int bid) {
34        return sqlSession.update("boardMapper.updateBCount", bid);
35    }
36
37
38
39
40
41
42
43
44
45
46
47
48
49

```

```

board-mapper.xml  BoardServiceImpl.java  BoardService.java  BoardDao.java
63 public int deleteBoard(int bid) {
64     return sqlSession.update("boardMapper.changeBstatusN", bid);
65 }
66
67 public ArrayList<Board> selectSearchTitle(SearchAndPage searches) {
68     List<Board> list = sqlSession.selectList("boardMapper.searchTitle");
69     return (ArrayList<Board>)list;
70 }
71
72
73 public ArrayList<Board> selectSearchWriter(SearchAndPage searches) {
74     List<Board> list = sqlSession.selectList("boardMapper.searchWriter");
75     return (ArrayList<Board>)list;
76 }
77
78
79 public ArrayList<Board> selectSearchDate(SearchAndPage searches) {
80     List<Board> list = sqlSession.selectList("boardMapper.searchDate");
81     return (ArrayList<Board>)list;
82 }
83
84
85 public int getSearchTitleListCount(String keyword) {
86     return sqlSession.selectOne("boardMapper.getSearchTitleListCount", keyword);
87 }
88
89
90
91 public int getSearchWriterListCount(String keyword) {
92     return sqlSession.selectOne("boardMapper.getSearchWriterListCount", keyword);
93 }
94
95
96 public int getSearchDateListCount(SearchDate dates) {
97     return sqlSession.selectOne("boardMapper.getSearchDateListCount", dates);
98 }
99 }
100

```

9. Pilot Test

- 파일럿 사용성 테스트를 통해 구현된 서비스들을

View --> Controller --> Service --> Repository

순으로 시험 작동해 보고, 테스트 화면들을 캡처하여 이미지 파일로 보고서에 추가함.

```

1 package com.ict.spring.notice.model.dao;
2
3 import java.util.ArrayList;
4
12 @Repository("noticeDao")
14 public class NoticeDao {
15     //스프링-마이바티스 연동 객체 연결함 : root-context.xml 에 선언되어 있음
16     @Autowired
17     private SqlSessionTemplate sqlSession;
18
19     public NoticeDao() {}
20
21     public ArrayList<Notice> selectList(){
22         List<Notice> list = sqlSession.selectList("noticeMapper.selectAll");
23         return (ArrayList<Notice>)list;
24     }
25
26     public Notice selectOne(int nid) {
27         return sqlSession.selectOne("noticeMapper.selectNotice", nid);
28     }
29
30     public int insertNotice(Notice notice) {
31         return sqlSession.insert("noticeMapper.insertNotice", notice);
32     }
33
34     public int updateNotice(Notice notice) {
35         return sqlSession.update("noticeMapper.updateNotice", notice);
36     }
37
38     public int deleteNotice(int nid) {
39         return sqlSession.delete("noticeMapper.deleteNotice", nid);
40     }
41
42     public ArrayList<Notice> selectNewTop3(){
43         List<Notice> list = sqlSession.selectList("noticeMapper.selectNewTop3");
44         return (ArrayList<Notice>)list;
45     }
46

```

```

47     public ArrayList<Notice> selectSearchTitle(String keyword) {
48         List<Notice> list = sqlSession.selectList("noticeMapper.searchTitle", keyword); //searchTitle = notice-mapper.xml 에 사용했던 id 명
49         // keyword는 전달할 인자 (매개변수)
50         return (ArrayList<Notice>)list;
51     }
52
53     public ArrayList<Notice> selectSearchWriter(String keyword) {
54         List<Notice> list = sqlSession.selectList("noticeMapper.searchWriter", keyword); //searchWriter = notice-mapper.xml 에 사용했던 id 명
55         // keyword는 전달할 인자 (매개변수)
56         return (ArrayList<Notice>)list;
57     }
58
59     public ArrayList<Notice> selectSearchDate(SearchDate dates) {
60         List<Notice> list = sqlSession.selectList("noticeMapper.searchDate", dates); //searchTitle = notice-mapper.xml 에 사용했던 id 명
61         // dates는 전달할 인자 (매개변수)
62         return (ArrayList<Notice>)list;
63     }
64 }
65

```

게시판 목록

first

+

localhost:8989/spring/blist.do?page=1

☆

UI아키텍처설계 포트폴리오 평가과제

관리자님 환영합니다!!

정보수정

로그아웃

HOME

공지사항

게시판

etc

글쓰기

검색할 항목을 선택하십시오.

☒ 제목 ☐ 작성자 ☐ 날짜

검색할 제목을 입력하십시오 :

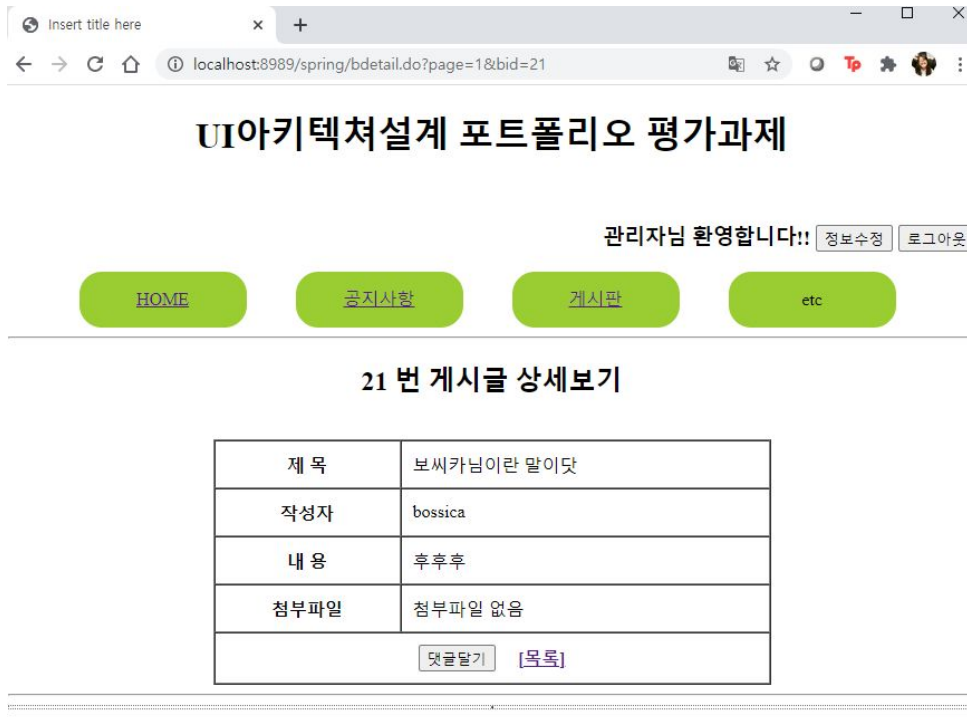
검색

전체 목록 보기

번호	제목	작성자	날짜	조회수	첨부파일
21	보쌈카님이란 말이닷	bossica	2021-02-08	0	
16	tester	admin	2021-02-04	19	📎
15	래퍼(Wrapper) 란	user01	2018-10-21	8	
14	필터(Filter) 란	user02	2018-10-21	8	
13	ajax 란	admin	2018-10-21	0	
12	jQuery 란	admin	2018-10-21	2	
11	JavaScript 란	user01	2018-03-01	0	
10	CSS3	user02	2018-03-01	2	

[맨처음] [이전] [1] 2 [다음그룹] [맨끝]

게시글 상세보기



게시판

새 게시물

등록 / 파일첨부

Insert title here

localhost:8989/spring/bupview.do?bid=41&page=1

UI아키텍처설계 포트폴리오 평가과제

관리자님 환영합니다!!

정보수정

로그아웃

HOME

공지사항

게시판

etc

41번 게시물 수정 페이지

제 목	UI아키텍처설계 포트폴리오
작성자	admin
첨부파일	새로 첨부할 파일 선택 : Choose File mds_profile.jpg
내 용	테스트 1
<div>수정하기</div> <div>수정취소</div> <div>목록</div>	

게시글 첨부파일 다운로드

The screenshot shows a web browser window with a file download dialog open. The dialog title is "다른 이름으로 저장" (Save with another name). The address bar shows the URL "localhost:8989/spring/nfdown.do?file_path=9_UI테스트_포트폴리오과제.pdf". The file name in the dialog is "9_UI테스트_포트폴리오과제" and the file type is "Adobe Acrobat Document".

Below the dialog, a table displays article details:

작성자	admin
날 짜	2021-02-19
첨부파일	9_UI테스트_포트폴리오과제.pdf
내 용	테스트용

A "목록" (List) button is located at the bottom right of the table.

At the bottom of the browser window, the address bar shows the URL: `localhost:8989/spring/nfdown.do?file_path=9_UI테스트_포트폴리오과제.pdf`

게시글 첨부파일 수정

Insert title here x +

localhost:8989/spring/bupview.do?bid=41&page=1

UI아키텍처설계 포트폴리오 평가과제

관리자님 환영합니다!! 정보수정 로그인아웃

HOME 공지사항 게시판 etc

41번 게시글 수정 페이지

제 목	UI아키텍처설계 포트폴리오
작성자	admin
첨부파일	mds_profile.jpg <input checked="" type="checkbox"/> 파일삭제 바꿀 파일 선택 : <input type="button" value="Choose File"/> No file chosen
내 용	테스트

게시글 첨부파일 삭제

first

localhost:8989/spring/blist.do?page=1

☆

🔍

🔖

👤

⋮

UI아키텍처설계 포트폴리오 평가과제

관리자님 환영합니다!!

HOME

공지사항

게시판

etc

글쓰기

검색할 항목을 선택하십시오.

☒ 제목 ☐ 작성자 ☐ 날짜

검색할 제목을 입력하십시오 :

전체 목록 보기

번호	제목	작성자	날짜	조회수	첨부파일
41	UI아키텍처설계 포트폴리오 평가과제	admin	2021-02-26	3	
21	보씨가남이란 말이닷	bossica	2021-02-08	1	
16	tester	admin	2021-02-04	19	📎
15	래퍼(Wrapper) 란	user01	2018-10-21	8	
14	필터(Filter) 란	user02	2018-10-21	8	
13	ajax 란	admin	2018-10-21	0	
12	jQuery 란	admin	2018-10-21	2	
11	JavaScript 란	user01	2018-03-01	0	

[맨처음] [이전] [1] 2 [다음그룹] [맨끝]

공지사항 검색 서비스 (제목)

UI테스트 포트폴리오 x 9_UI테스트_포트폴리 x Insert title here x +

localhost:8989/spring/nsearchTitle.do

UI테스트 포트폴리오 평가과제

관리자님 환영합니다!! [정보수정](#) [로그아웃](#)

[HOME](#) [공지사항](#) [게시판](#) [etc](#)

공지사항

[글쓰기](#)

검색할 항목을 선택하십시오.

☒ 제목 ☐ 작성자 ☐ 날짜

검색할 제목을 입력하십시오 : [검색](#)

[전체 목록 보기](#)

번호	제목	작성자	첨부파일	날짜
21	테스트용 새 공지글	admin		2021-02-08