

Mybatis 게시판 제작(with paging, search)2

프로젝트명 : B07dMybatisBoard

작성하기

컨트롤러 : com.edu.springboot.MainController.java

```
66 @GetMapping("/write.do")
67 public String boardWriteGet(Model model) {
68     return "write";
69 }
70
71 @PostMapping("/write.do")
72 public String boardWritePost(Model model, HttpServletRequest req) {
73     String name = req.getParameter("name");
74     String title = req.getParameter("title");
75     String content = req.getParameter("content");
76     int result = dao.write(name, title, content);
77     System.out.println("글쓰기결과:" + result);
78     return "redirect:list.do";
79 }
```

인터페이스 : com.edu.springboot.jdbc.IBoardService.java

```
8 @Mapper
9 public interface IBoardService {
10     //Paging처리를 위해 게시물의 갯수를 카운트
11     public int getTotalCount(ParameterDTO parameterDTO);
12     //게시판 목록(페이징 기능 추가)
13     public ArrayList<BoardDTO> listPage(ParameterDTO parameterDTO);
14     //작성하기
15     public int write(@Param("_name") String name,
16                     @Param("_title") String title,
17                     @Param("_content") String content);
18     //내용보기
19     public BoardDTO view(BoardDTO boardDTO);
20     //수정하기
21     public int edit(BoardDTO boardDTO);
22     //삭제하기
23     public int delete(String idx);
24 }
```

매퍼 : resources/mapper/BoardDAO.xml

```
42<insert id="write">
43    INSERT INTO myboard (idx, name, title, content)
44    VALUES
45    (seq_board_num.nextval, #{_name}, #{_title}, #{_content})
46</insert>
47
```

뷰 : webapp/WEB-INF/views/write.jsp

```
<body>
    <h2>게시판 작성(Mybatis)</h2>
    <form name="writeFrm" method="post"
        action="./write.do" onsubmit="return validateForm(this);">
    <table border="1" width="90%">
        <tr>
            <td>작성자</td>
            <td>
                <input type="text" name="name" style="width:150px;" />
            </td>
        </tr>
        <tr>
            <td>제목</td>
            <td>
                <input type="text" name="title" style="width:90%;" />
            </td>
        </tr>
        <tr>
            <td>내용</td>
            <td>
                <textarea name="content" style="width:90%;height:100px;"></textarea>
            </td>
        </tr>
        <tr>
            <td colspan="2" align="center">
                <button type="submit">작성 완료</button>
                <button type="reset">RESET</button>
                <button type="button" onclick="location.href='./list.do';">
                    목록 바로가기
                </button>
            </td>
        </tr>
    </table>
</form>
</body>
```

```
</td>
</tr>
</table>
</form>
</body>
```

여기까지 작성하세요.

퀴즈]

‘작성완료’ 버튼을 누를때 작성자, 제목, 내용에 입력된 값이 있는지 검증하는 Javascript 코드를 작성하시오.
submit 이벤트가 발생할 때 validateForm() 함수를 호출하고 있으므로 이 함수를 정의하면된다.

게시판

localhost:8586 내용:
작성자를 입력하세요.

확인

작성자

제목

내용

작성 완료 RESET 목록 바로가기

내용보기

컨트롤러 : com.edu.springboot.MainController.java

```
94 @RequestMapping("/view.do")
95 public String boardView(Model model, BoardDTO boardDTO) {
96     boardDTO = dao.view(boardDTO);
97     boardDTO.setContent(boardDTO.getContent()
98         .replace("\r\n", "<br/>"));
99     model.addAttribute("boardDTO", boardDTO);
100     return "view";
101 }
```

인터페이스 : com.edu.springboot.jdbc.IBoardService.java

```
8 @Mapper
9 public interface IBoardService {
10     //Paging처리를 위해 게시물의 갯수를 카운트
11     public int getTotalCount(ParameterDTO parameterDTO);
12     //게시판 목록(페이징 기능 추가)
13     public ArrayList<BoardDTO> listPage(ParameterDTO parameterDTO);
14     //작성하기
15     public int write(@Param("_name") String name,
16         @Param("_title") String title,
17         @Param("_content") String content);
18     //내용보기
19     public BoardDTO view(BoardDTO boardDTO);
20     //수정하기
21     public int edit(BoardDTO boardDTO);
22     //삭제하기
23     public int delete(String idx);
24 }
```

매퍼 : resources/mapper/BoardDAO.xml

```
59 <select id="view" resultType="com.edu.springboot.jdbc.BoardDTO"
60     parameterType="com.edu.springboot.jdbc.BoardDTO">
61     SELECT * FROM myboard WHERE idx=#{idx}
62 </select>
63
```

view : webapp/WEB-INF/views/view.jsp

```
<body>
    <h2>게시판 읽기(Mybatis)</h2>
    <form name="writeFrm">
```

```

        <input type="hid-den" name="idx" value="${boardDTO.idx}" />
</form>
<table border="1" width="90%">
    <colgroup>
        <col width="15%" /> <col width="35%" />
        <col width="15%" /> <col width="*" />
    </colgroup>
    <!-- 게시글 정보 -->
    <tr>
        <td>번호</td> <td>${ boardDTO.idx }</td>
        <td>작성자</td> <td>${ boardDTO.name }</td>
    </tr>
    <tr>
        <td>작성일</td> <td>${ boardDTO.postdate }</td>
        <td>조회수</td> <td>${ boardDTO.visitcount }</td>
    </tr>
    <tr>
        <td>제목</td>
        <td colspan="3">${ boardDTO.title }</td>
    </tr>
    <tr>
        <td>내용</td>
        <td colspan="3" height="100">
            ${ boardDTO.content }
        </td>
    </tr>
    <!-- 하단 메뉴(버튼) -->
    <tr>
        <td colspan="4" align="center">
            <button type="button" onclick="location.href='./edit.do?idx=${ param.idx }';">
                수정하기
            </button>
            <button type="button" onclick="deletePost(${ param.idx });">
                삭제하기
            </button>
            <button type="button" onclick="location.href='./list.do';">
                목록 바로가기
            </button>
        </td>
    </tr>

```

</tr>

</table>

</body>