

2023. 05. 12 2차발표

---

조장 황정현 | 이수형 | 채경철 | 나주화 | 김휘현 | 라가현 | 유찬희 | 박소율

---

# COMMUNITY

## QNA

- User는 글을 작성할 수 있고 자신이 작성한 글만 확인할 수 있다.
- 댓글은 관리자만 작성할 수 있다.

tb\_user 1

select \* from tb\_user

Enter a SQL expression to filter results (use Ctrl+Space)

		123 user_no	ABC gr_no	ABC user_id	ABC user_pwd	ABC user_nm	ABC user_nicknm	123 user_gen	ABC user_ema	reg_date	ABC admin
1		1	0	earth0629	0629earth!	노지구	나안관리자	0	earth0629@gmail:48:41.372 +0900		N
2		2	0	earth0630	0630earth!	나지구	나관리자	0	earth0629@gmail:53:45.138 +0900		Y

회원 User					
🔑	회원 번호	user_no	Domain	BIGINT	NOT NULL
🔑	등급 번호	gr_no	Domain	char(1)	NOT NULL
	아이디	user_id	Domain	VARCHAR(15)	NOT NULL
	비밀번호	user_pwd	Domain	VARCHAR(15)	NOT NULL
	이름	user_nm	Domain	VARCHAR(30)	NOT NULL
	닉네임	user_nicknm	Domain	VARCHAR(10)	NOT NULL
	성별	user_gen	Domain	CAHR(1)	NOT NULL
	이메일	user_email	Domain	VARCHAR(150)	NOT NULL
	가입일	reg_date	Domain	TIMESTAMP TZ	NOT NULL
	관리자여부	admin	Domain	CHAR(1)	NOT NULL
	회원 이미지	image	Domain	TEXT	NULL
	팔로잉	following	Domain	BIGINT	NOT NULL
	팔로워	followers	Domain	BIGINT	NOT NULL
	신고개수	report_cnt	Domain	INT	NOT NULL

게시글 Article					
🔑	게시글번호	article_no	Domain	BIGINT	NOT NULL
🔑	회원 번호	user_no	Domain	BIGINT	NOT NULL
🔑	게시판종류	article_index_no	Domain	char(1)	NOT NULL
	제목	article_title	Domain	VARCHAR(255)	NULL
	내용	article_content	Domain	VARCHAR(65535)	NULL
	사진	article_image	Domain	TEXT	NULL
	등록날짜	article_create_dt	Domain	TIMESTAMP	NOT NULL
	작성자	article_create_user	Domain	VARCHAR(50)	NOT NULL
	수정일	article_mod_dt	Domain	TIMESTAMP	NULL
	좋아요	article_like_count	Domain	INT	NULL
	신고개수	report_cnt	Domain	INT	NULL

# COMMUNITY

## QNA

- User는 글을 작성할 수 있고 자신이 작성한 글만 확인할 수 있다.
- 댓글은 관리자만 작성할 수 있다.

tb_article 1					
SELECT * FROM tb_article where article					
	123 article_no	123 user_no	ABC article_index_no	ABC article_content	
1	2	2	2	나관리자임당	
2	1	1	2	노관리자임당	

<user\_no=1> 관리자가 아닐 경우

OTT

영화드라마예능애니게시판

Q

자유게시판

종료예정작

가격정보

Q&A

공지사항

글작성

제목	날짜
나는 관리자가 아님당	2023-05-11

1

<user\_no=2> 관리자일 경우

OTT

영화드라마예능애니게시판

Q

자유게시판

종료예정작

가격정보

Q&A

공지사항

글작성

나는 관리자임당	2023-05-11
나는 관리자가 아님당	2023-05-11

1

# COMMUNITY

## QNA

- User는 글을 작성할 수 있고 자신이 작성한 글만 확인할 수 있다.
- 댓글은 관리자만 작성할 수 있다.

## 로그인 CONTROLLER

```
@PostMapping("/login")
public String login(String user_id, String user_pwd, String toURL, boolean rememberId,
    HttpServletRequest request, HttpServletResponse response) throws UnsupportedEncodingException {

    if(!loginCheck(user_id, user_pwd)) {
        String msg = URLEncoder.encode("id 또는 pwd가 일치하지 않습니다", "utf-8");
        return "redirect:/login?msg="+msg;
    }

    if(rememberId) {
        Cookie cookie = new Cookie("id", user_id);
        response.addCookie(cookie);
    }
    else {
        Cookie cookie = new Cookie("id", user_id);
        cookie.setMaxAge(0);
        response.addCookie(cookie);
    }

    UserDTO userDTO = userDao.select(user_id);
    HttpSession session = request.getSession();
    session.setAttribute("id", user_id);
    session.setAttribute("admin", userDTO.getAdmin());
    session.setAttribute("no", userDTO.getUser_no());

    toURL = toURL == null || toURL.equals("") ? "/" : toURL;

    return "redirect:" +toURL;
}
```

# COMMUNITY

## QNA

- User는 글을 작성할 수 있고 자신이 작성한 글만 확인할 수 있다.
- 댓글은 관리자만 작성할 수 있다.

## QNA CONTROLLER

```
@Controller
@RequestMapping("/community")
public class QnaController {

    @Autowired
    QnaServiceImpl serviceImpl;
    @Autowired
    LoginUserDao loginUserDao;

    //QnA
    @GetMapping(value = "/QnA")
    public String QnA(SearchItem sc, Model m, HttpServletRequest request, HttpSession session) {

        try {
            UserDTO userDTO = loginUserDao.select((String) session.getAttribute("id"));
            char admin = (char) session.getAttribute("admin");
            if(admin=='Y') {
                int totalCnt = serviceImpl.getAllCount(sc);
                m.addAttribute("totalCnt", totalCnt);
                List<ArticleDTO> list = serviceImpl.getAllPage(sc);
                PageResolver pageResolver = new PageResolver(totalCnt, sc);
                m.addAttribute("list", list);
                m.addAttribute("pr", pageResolver);
            }else {
                int totalCnt = serviceImpl.getCount(sc, userDTO.getUser_no());
                m.addAttribute("totalCnt", totalCnt);
                List<ArticleDTO> list = serviceImpl.getPage(sc, userDTO.getUser_no());
                PageResolver pageResolver = new PageResolver(totalCnt, sc);
                m.addAttribute("list", list);
                m.addAttribute("pr", pageResolver);
            }
        } catch (Exception e) {e.printStackTrace();}
        return "/community/QnA/QnA";
    }
}
```

# COMMUNITY

## QNA

- User는 글을 작성할 수 있고 자신이 작성한 글만 확인할 수 있다.
- 댓글은 관리자만 작성할 수 있다.

### <관리자일 경우>

#### <QnA Controller>

```
if(admin=='Y') {  
    int totalCnt = serviceImpl.getAllCount(sc);  
    m.addAttribute("totalCnt", totalCnt);  
    List<ArticleDTO> list = serviceImpl.getAllPage(sc);  
    PageResolver pageResolver = new PageResolver(totalCnt, sc);  
    m.addAttribute("list", list);  
    m.addAttribute("pr", pageResolver);}
```

#### <ArticleQnADao>

```
@Repository  
public class ArticleQnADao implements ArticleNoticeDao {  
  
    @Autowired  
    private SqlSession session;  
    private static String namespace = "com.ottt.ottt.dao.community.qnaMapper.";  
  
    public List<ArticleDTO> selectAll(SearchItem sc) throws Exception{  
        return session.selectList(namespace+"selectAll", sc);  
    }  
  
    public int countAll(SearchItem sc) throws Exception {  
        return session.selectOne(namespace+"countAll", sc);  
    }  
}
```

#### <QnAServiceImpl>

```
@Service  
public class QnAServiceImpl implements ArticleService {  
  
    @Autowired  
    ArticleQnADao articleQnADao;  
  
    public List<ArticleDTO> getAllPage(SearchItem sc) throws Exception {  
        return articleQnADao.selectAll(sc);  
    }  
  
    public int getAllCount(SearchItem sc) throws Exception {  
        return articleQnADao.countAll(sc);  
    }  
}
```

#### <QnA Mapper>

```
<select id="selectAll" parameterType="SearchItem" resultType="ArticleDTO">  
    select *  
    from tb_article  
    where article_index_no = '2'  
</select>  
  
<select id="countAll" parameterType="SearchItem" resultType="int">  
    select count(*)  
    from tb_article  
    where article_index_no = '2'  
</select>
```

# COMMUNITY

## QNA

- User는 글을 작성할 수 있고 자신이 작성한 글만 확인할 수 있다.
- 댓글은 관리자만 작성할 수 있다.

### <관리자가 아닐 경우>

#### <QnA Controller>

```
else {  
    int totalCnt = serviceImpl.getCount(sc, userDTO.getUser_no());  
    m.addAttribute("totalCnt", totalCnt);  
    List<ArticleDTO> list = serviceImpl.getPage(sc, userDTO.getUser_no());  
    PageResolver pageResolver = new PageResolver(totalCnt, sc);  
    m.addAttribute("list", list);  
    m.addAttribute("pr", pageResolver);  
}
```

#### <ArticleQnADao>

```
@Repository  
public class ArticleQnADao implements ArticleNoticeDao {  
  
    @Autowired  
    private SqlSession session;  
    private static String namespace = "com.ottt.ottt.dao.community.qnaMapper.";  
  
    public List<ArticleDTO> selectPage(SearchItem sc, Integer user_no) throws Exception {  
        Map map = new HashMap();  
        map.put("searchItem", sc);  
        map.put("user_no", user_no);  
        map.put("pageSize", sc.getPageSize());  
        map.put("offset", sc.getOffset());  
        return session.selectList(namespace+"selectList", map);  
    }  
  
    public int count(SearchItem sc, Integer user_no) throws Exception {  
        Map map = new HashMap();  
        map.put("SearchItem", sc);  
        map.put("user_no", user_no);  
        return session.selectOne(namespace+"count", map);  
    }  
}
```

#### <QnAServiceImpl>

```
@Service  
public class QnAServiceImpl implements ArticleService {  
  
    @Autowired  
    ArticleQnADao articleQnADao;  
  
    public List<ArticleDTO> getPage(SearchItem sc, Integer user_no) throws Exception {  
        return articleQnADao.selectPage(sc, user_no);  
    }  
  
    public int getCount(SearchItem sc, Integer user_no) throws Exception {  
        return articleQnADao.count(sc, user_no);  
    }  
}
```

#### <QnA Mapper>

```
<select id="selectList" parameterType="map" resultType="ArticleDTO">  
    SELECT article_title, article_create_dt, article_no  
    FROM tb_article  
    where article_index_no = '2' and user_no = #{user_no}  
    order by article_no desc  
    limit #{pageSize} offset #{offset}  
</select>  
  
<select id="count" parameterType="map" resultType="int">  
    select count(*)  
    from tb_article  
    where article_index_no = '2' and user_no = #{user_no}  
</select>
```

# COMMUNITY

## 공지사항

- 관리자만 글을 작성할 수 있으며, User는 글을 작성할 수 없다.
- 댓글도 작성할 수 없다.

tb_user 1 ✕										
select * from tb_user <small>Enter a SQL expression to filter results (use Ctrl+Space)</small>										
	123 user_no ▼	ABC gr_no ▼	ABC user_id ▼	ABC user_pwd ▼	ABC user_nm ▼	ABC user_nicknm ▼	123 user_gen ▼	ABC user_ema ▼	🕒 reg_date ▼	ABC admin ▼
1	1	0	earth0629	0629earth!	노지구	나안관리자	0	earth0629@gmail:48:41.372 +0900		N
2	2	0	earth0630	0630earth!	나지구	나관리자	0	earth0629@gmail:53:45.138 +0900		Y

회원 User					
🔑	회원 번호	user_no	Domain	BIGINT	NOT NULL
🔑	등급 번호	gr_no	Domain	char(1)	NOT NULL
	아이디	user_id	Domain	VARCHAR(15)	NOT NULL
	비밀번호	user_pwd	Domain	VARCHAR(15)	NOT NULL
	이름	user_nm	Domain	VARCHAR(30)	NOT NULL
	닉네임	user_nicknm	Domain	VARCHAR(10)	NOT NULL
	성별	user_gen	Domain	CAHR(1)	NOT NULL
	이메일	user_email	Domain	VARCHAR(150)	NOT NULL
	가입일	reg_date	Domain	TIMESTAMP TZ	NOT NULL
	관리자여부	admin	Domain	CHAR(1)	NOT NULL
	회원 이미지	image	Domain	TEXT	NULL
	팔로잉	following	Domain	BIGINT	NOT NULL
	팔로워	followers	Domain	BIGINT	NOT NULL
	신고개수	report_cnt	Domain	INT	NOT NULL

게시글 Article					
🔑	게시글번호	article_no	Domain	BIGINT	NOT NULL
🔑	회원 번호	user_no	Domain	BIGINT	NOT NULL
🔑	게시판종류	article_index_no	Domain	char(1)	NOT NULL
	제목	article_title	Domain	VARCHAR(255)	NULL
	내용	article_content	Domain	VARCHAR(65535)	NULL
	사진	article_image	Domain	TEXT	NULL
	등록날짜	article_create_dt	Domain	TIMESTAMP	NOT NULL
	작성자	article_create_user	Domain	VARCHAR(50)	NOT NULL
	수정일	article_mod_dt	Domain	TIMESTAMP	NULL
	좋아요	article_like_count	Domain	INT	NULL
	신고개수	report_cnt	Domain	INT	NULL



# COMMUNITY

## 공지사항

- 관리자만 글을 작성할 수 있으며, User는 글을 작성할 수 없다.
- 댓글도 작성할 수 없다.

```
<!-- 검색 배너-->
<form action="<c:url value="/community/notice" />" method="get">
  <div class="search-var">
    <span>제목+내용</span>
    <input type="text" class="search-input" name="keyword" value="${param.keyword }"/>
    <button type="button" class="btn btn-outline-success" id="btn">검색</button>
  </div>
</form>
<c:if test="${userDTO.admin.toString() == 'Y' }">
  <div style="justify-content: flex-end; display: flex;">
    <button type="button" id="writeBtn" class="btn btn-outline-success">글작성</button>
  </div>
</c:if>
```

> QnA.jsp

<user\_no=1> 관리자가 아닐 경우

ONTT

영화드라마예능애니게시판

Q

자유게시판

종료예정작

가격정보

Q&A

공지사항

제목+내용

검색

제목

날짜

나는 관리자니까 글작성 할 수 있지 ㅎㅎㅎ

2023-05-11

저는 관리자랍니다

2023-05-11

1

<user\_no=2> 관리자일 경우

ONTT

영화드라마예능애니게시판

Q

자유게시판

종료예정작

가격정보

Q&A

공지사항

제목+내용

검색

제목

날짜

나는 관리자니까 글작성 할 수 있지 ㅎㅎㅎ

2023-05-11

저는 관리자랍니다

2023-05-11

1

# COMMUNITY

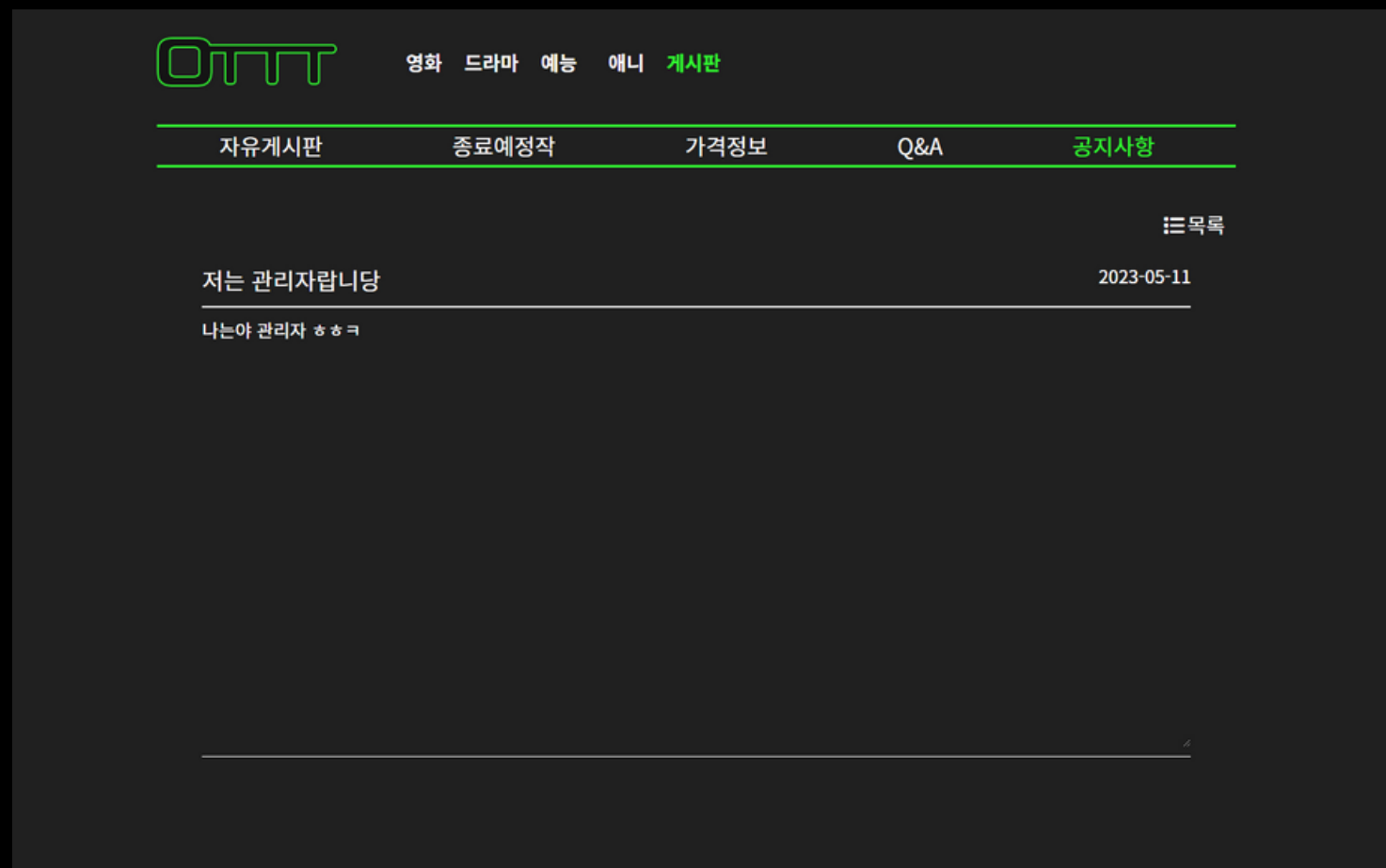
## 공지사항

- 관리자만 글을 작성할 수 있으며, User는 글을 작성할 수 없다.
- 댓글도 작성할 수 없다.

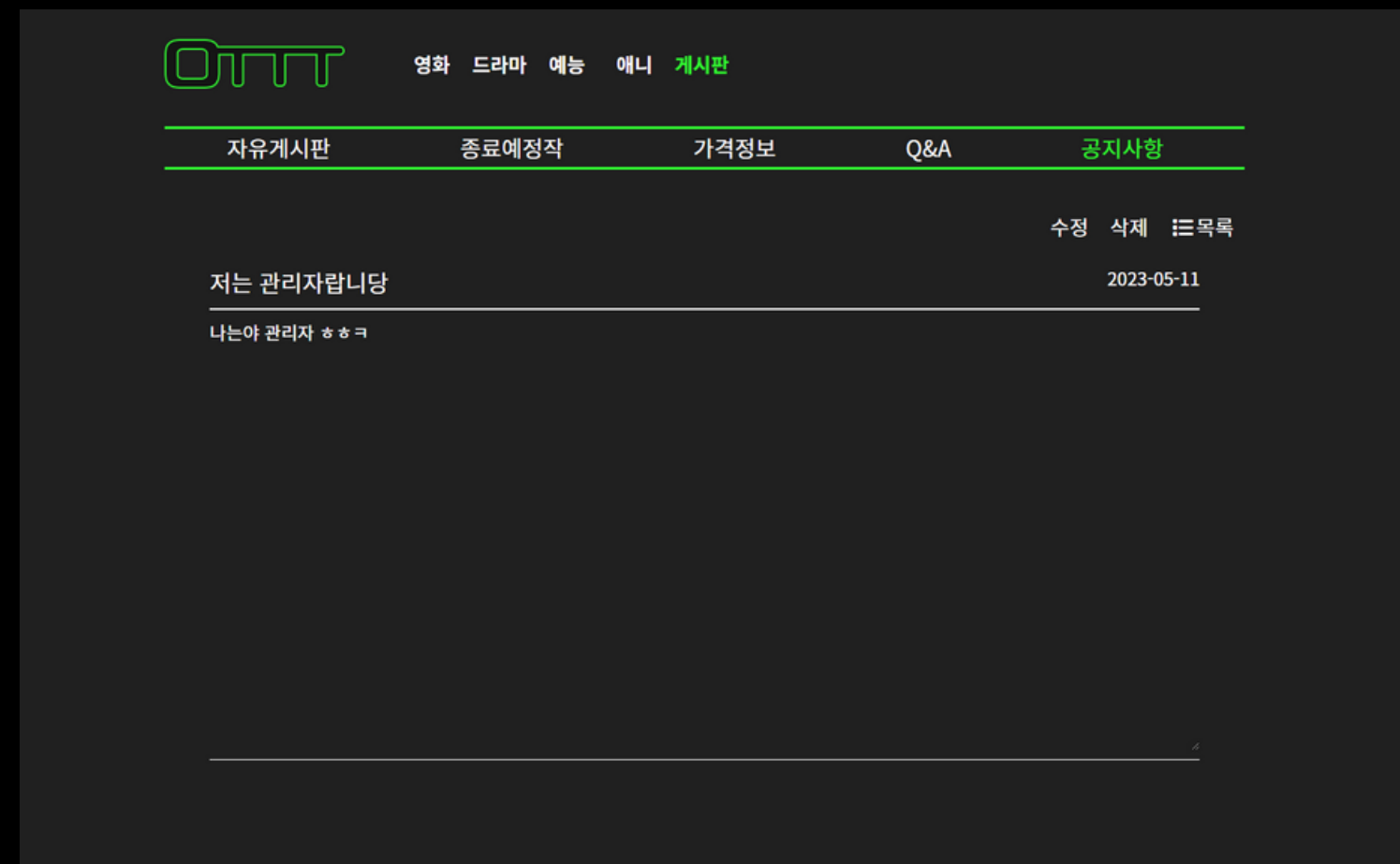
```
<c:if test="${userDTO.admin.toString() == 'Y'}">
  <!-- Button trigger modal -->
  <button type="button" class="btn btn-secondary" id="modi" data-bs-toggle="modal" data-bs-target="#exampleModal">
    수정
  </button>
  <!-- Button trigger modal -->
  <button type="button" class="btn btn-secondary" id="del" data-bs-toggle="modal" data-bs-target="#exampleModa2">
    삭제
  </button>
</c:if>
```

> QnA.jsp

<user\_no=1> 관리자가 아닐 경우



<user\_no=2> 관리자일 경우



# COMMUNITY

## 공지사항

- 관리자만 글을 작성할 수 있으며, User는 글을 작성할 수 없다.
- 댓글도 작성할 수 없다.

## notice.jsp

```
<!-- 검색 배너-->
<form action="<c:url value="/community/notice" />" method="get">
  <div class="search-var">
    <span>제목+내용</span>
    <input type="text" class="search-input" name="keyword" value="${param.keyword }"/>
    <button type="button" class="btn btn-outline-success" id="btn">검색</button>
  </div>
</form>
```

## notice Mapper

```
<mapper namespace="com.ottt.ottt.dao.community.noticeMapper">

  <select id="selectList" parameterType="SearchItem" resultType="ArticleDTO" >
    SELECT article_title, article_create_dt, article_no
    FROM tb_article
    where article_index_no = '3' and
    (article_title like concat('%', #{keyword}, '%')
    or article_content like concat('%', #{keyword}, '%'))
    order by article_no desc
    limit #{pageSize} offset #{offset}
  </select>
```

