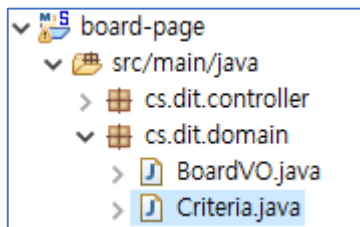


Chap13

MyBatis와 스프링에서 페이징 처리

동의과학대학교
컴퓨터정보과
김진숙

페이지 기본 설정 객체



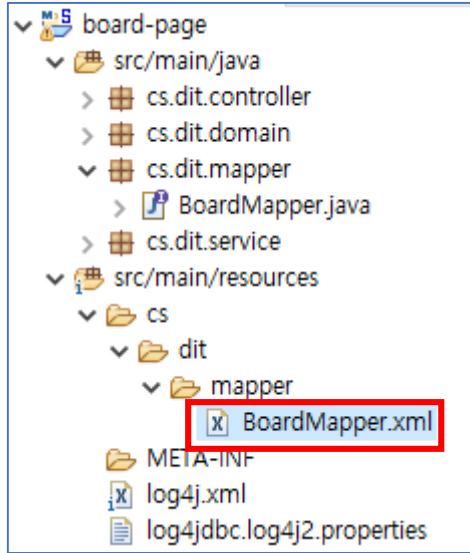
```
7 @Setter
8 @Getter
9 @ToString
10 public class Criteria {
11
12     private int pageNum;
13     private int amount;
14
15     public Criteria() {
16         this(1, 10);
17     }
18
19     public Criteria(int pageNum, int amount) {
20         super();
21         this.pageNum = pageNum;
22         this.amount = amount;
23     }
24 }
```

BoardMapper.java 인터페이스 수정

```
10 public interface BoardMapper {  
11  
12     public List<BoardVO> getList();  
13  
14     public List<BoardVO> getListPage(Criteria cri);  
15  
16     public int insert(BoardVO board);  
17  
18     public BoardVO selectOne(Long bno);  
19  
20     public int delete(Long bno);  
21  
22     public int update(BoardVO board);  
23  
24 }
```

BoardMapper.xml 파일 수정

- 한 화면에 10개씩 레코드 출력 SQL문



```
6 <mapper namespace="cs.dit.mapper.BoardMapper">
7
8 <select id="getList" resultType="cs.dit.domain.BoardVO">
9     select * from tbl_board order by bno desc
10 </select>
11
12 <select id="getListPage" resultType="cs.dit.domain.BoardVO">
13     select * from tbl_board order by bno desc limit #{pageStart}, #{amount}
14 </select>
15
```

page	numOfRecords	limit	한 화면 출력 레코드
1	10	0, 10	1 ~ 10
2	10	10, 10	11 ~ 20
3	10	20, 10	21 ~ 30
4	10	30, 10	31 ~ 40
5	10	40, 10	41 ~ 50

(Page -1)

SELECT * FROM table
LIMIT 0, 10

SELECT * FROM table
LIMIT 10 OFFSET 0

Criteria.java 에
메서드 추가

```
public int getPageStart() {
    return (this.pageNum - 1) * this.amount;
}
```

BoardMapperTests.java 에 추가

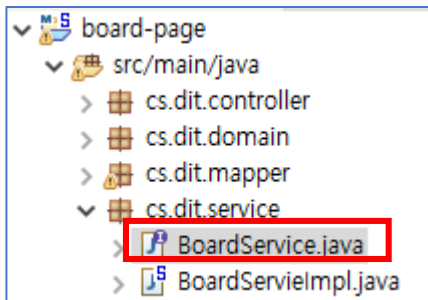
```
@Test
public void testGetListPage() {
    Criteria cri = new Criteria();
    cri.setPageNum(3); //3번째 페이지
    cri.setAmount(10); //10개씩

    boardMapper.getListPage(cri).forEach(board -> Log.info(board));
}
```

INFO : jdbc.resultsettable -

bno	title	content	writer	regdate	updatedate
115	테스트9	테스트내용9	홍길동	2022-11-02	2022-11-02
114	테스트8	테스트내용8	홍길동	2022-11-02	2022-11-02
112	테스트6	테스트내용6	홍길동	2022-11-02	2022-11-02
111	테스트5	테스트내용5	홍길동	2022-11-02	2022-11-02
110	테스트4	테스트내용4	홍길동	2022-11-02	2022-11-02
109	테스트3	테스트내용3	홍길동	2022-11-02	2022-11-02
108	테스트2	테스트내용2	홍길동	2022-11-02	2022-11-02
107	테스트1	테스트내용1	홍길동	2022-11-02	2022-11-02
106	테스트10	테스트내용10	홍길동	2022-11-02	2022-11-02
105	테스트9	테스트내용9	홍길동	2022-11-02	2022-11-02

BoardService.java 인터페이스 수정



```
public interface BoardService {  
  
    public List<BoardVO> getList();  
  
    public List<BoardVO> getList(Criteria cri);  
  
    public BoardVO getOne(Long bno);  
  
    public int register(BoardVO board);  
  
    public int modify(BoardVO board);  
  
    public int remove(Long bno);  
  
}
```

BoardServiceImpl.java 수정

```
@Service
@Log4j
@RequiredArgsConstructor //final로 객체주입(DI)
@ToString
public class BoardServieImpl implements BoardService{

    private final BoardMapper mapper;

    @Override
    public List<BoardVO> getList() {

        return mapper.getList();
    }

    @Override
    public List<BoardVO> getList(Criteria cri) {

        log.info("get List with Criteria : " + cri);

        return mapper.getListPage(cri);
    }
}
```

getList() 오버로딩

Paging이 적용된 화면

게시판 목록

num	subject	writer	regDate
59	안녕46	gildong11	2022. 9. 18.
58	안녕45	gildong	2022. 9. 18.
57	안녕44	gildong	2022. 9. 18.
56	안녕43	gildong	2022. 9. 18.
54	안녕41	gildong	2022. 9. 18.
53	안녕40	gildong	2022. 9. 18.
52	안녕39	gildong	2022. 9. 18.
51	안녕38	gildong	2022. 9. 18.
47	안녕34	gildong	2022. 9. 18.
44	안녕31	gildong	2022. 9. 18.

[홈으로](#) [게시글 등록](#)

Prev 1 2 3 4 5 Next

- 전체 레코드의 개수 : DB연동 필요
 - count
- 한 화면에 출력되는 레코드 수 : 10개
 - numOfRecords
- 한 화면에 보여지는 페이지 번호 개수 : 5개
 - numOfPages
- 한 화면의 시작 페이지 번호 : 1, 6, 11, 16, ...
 - startNum
- 마지막 출력 페이지 번호
 - lastNum

Prev 6 Next

startNum 찾기

- 한 화면의 시작 페이지 번호 : **startNum**

1	1	2	3	4	5
2	6	7	8	9	10
3	11	12	13	14	15
4	16	17	18	19	20
5	21	22	23	24	25
				

현재 페이지 번호(p)가 4일 때 startNum? → 1

현재 페이지 번호(p)가 8일 때 startNum? → 6

현재 페이지 번호(p)가 22일 때 startNum? → 21

5 - 4 = 1	10 - 4 = 6	15 - 4 = 11
4 - 3 = 1	9 - 3 = 6	14 - 3 = 11
3 - 2 = 1	8 - 2 = 6	13 - 2 = 11
2 - 1 = 1	7 - 1 = 6	12 - 1 = 11
1 - 0 = 1	6 - 0 = 6	11 - 0 = 11

- 나머지 계산 필요 : $(p - 1) \% 5$
- $\text{startNum} = p - (p - 1) \% 5$

페이지 처리에 필요한 정보들

페이징의 끝 번호(endPage) 계산

```
this.endPage = (int)(Math.ceil(페이지번호 / 10.0)) * 10;
```

페이징의 시작 번호(startPage) 계산

```
this.startPage = this.endPage - 9;
```

total을 통한 endPage의 재계산

```
realEnd = (int) (Math.ceil((total * 1.0) / amount) );
```

```
if(realEnd < this.endPage) {  
    this.endPage = realEnd;  
}
```

이전(prev) 계산

```
this.prev = this.startPage > 1;
```

다음(next) 계산

```
this.next = this.endPage < realEnd;
```

PageDTO.java 객체 생성

```
@Getter
@ToString
public class PageDTO {
    private int startPage; //한 화면의 첫 페이지 번호
    private int endPage;   //전체 게시글의 마지막 페이지 번호
    private boolean prev, next; //페이지네이션 링크에 표시되는 이전, 다음 링크 표시

    private int total; //전체 게시글의 갯수
    private Criteria cri; //pageNum, amount을 설정한 객체

    public PageDTO(Criteria cri, int total) { //필드가 있는 생성자
        this.cri = cri;
        this.total = total;

        int numOfPages = 5; //한 화면에 보여지는 페이지의 수

        //마지막 페이지 계산 - 정밀한 계산은 아니나 endPage 계산을 위해 미리 계산
        this.endPage = (int)Math.ceil(((float)cri.getPageNum()/((float)numOfPages))*numOfPages);
        //첫 페이지 계산
        this.startPage = this.endPage - (numOfPages-1);

        //전체 레코드의 수로 계산한 실제 마지막 페이지 계산
        int realEnd = (int)Math.ceil(((float)total/((float)cri.getAmount()));
        //이전에 계산한 마지막 페이지가 실제 마지막 페이지보다 크면 실제 페이지가 적용되도록 마지막페이지 설정
        this.endPage = this.endPage > realEnd ? realEnd : this.endPage;

        //첫 페이지가 1보다 크면, 즉 (한 화면의 페이지 수가 5일 경우) 6, 11, 16... 이 되면 prev가 표시
        this.prev = this.startPage > 1;
        //next 링크는 realEnd가 끝번호 보다 큰경우에만 존재
        this.next = this.endPage < realEnd;
    }
}
```

BoardController.java 수정

```
@Controller
@RequiredArgsConstructor
@RequestMapping("/board/*")
@Log4j
public class BoardController {

    private final BoardService service;

    @GetMapping("/list")
    public void list(Criteria cri, Model model) {
        log.info("list : " + cri);

        int total = service.getTotal(cri);

        model.addAttribute("list", service.getList(cri));

        model.addAttribute("pageMaker", new PageDTO(cri, total));
    }
}
```

JSP에서 페이지 번호 출력

- Bootstrap이 적용된 다음의 코드로 페이지 출력
 - https://www.w3schools.com/bootstrap4/bootstrap_pagination.asp 에서 복사
 - list.jsp 파일의 테이블 아래에 다음의 코드를 위치시킴

```
<div class="pull-right">
  <ul class="pagination">
    <li class="page-item"><a class="page-link" href="#">Previous</a></li>
    <li class="page-item"><a class="page-link" href="#">1</a></li>
    <li class="page-item active"><a class="page-link" href="#">2</a></li>
    <li class="page-item"><a class="page-link" href="#">3</a></li>
    <li class="page-item"><a class="page-link" href="#">Next</a></li>
  </ul>
</div> <!-- end of pagination -->
```

list.jsp 일부

JSP에서 페이지 번호 출력

- Controller에서 model에 저장한 pageMaker객체로 화면에 링크 표시

```
<div class="pull-right">
  <ul class="pagination">
    <c:if test="${pageMaker.prev}">
      <li class="page-item">
        <a class="page-link" href="${pageMaker.startPage-1 }">Previous</a>
      </li>
    </c:if>

    <c:forEach var="num" begin="${pageMaker.startPage}" end="${pageMaker.endPage}">
      <li class="page-item ${pageMaker.cri.pageNum == num ? 'active': ''}">
        <a class="page-link" href="${num}">${num}</a>
      </li>
    </c:forEach>

    <c:if test="${pageMaker.next}">
      <li class="page-item">
        <a class="page-link" href="${pageMaker.endPage+1 }">Next</a>
      </li>
    </c:if>
  </ul>
</div><!-- end of pagination -->
```

list.jsp 일부

페이지 번호 이벤트 처리

- 페이지 이벤트 발생 시에 같이 전송되어야 하는 값들을 form에 설정
 - 페이지번호의 링크는 페이지 번호만 가지도록 함
 - 별도의 <form>태그와 이벤트를 통해서 pageNum, amount 값 전달

```
<form id="pageForm" action="/board/list" method="get">  
  <input type="hidden" name="pageNum" value="${pageMaker.cri.pageNum}">  
  <input type="hidden" name="amount" value="${pageMaker.cri.amount}">  
</form>
```

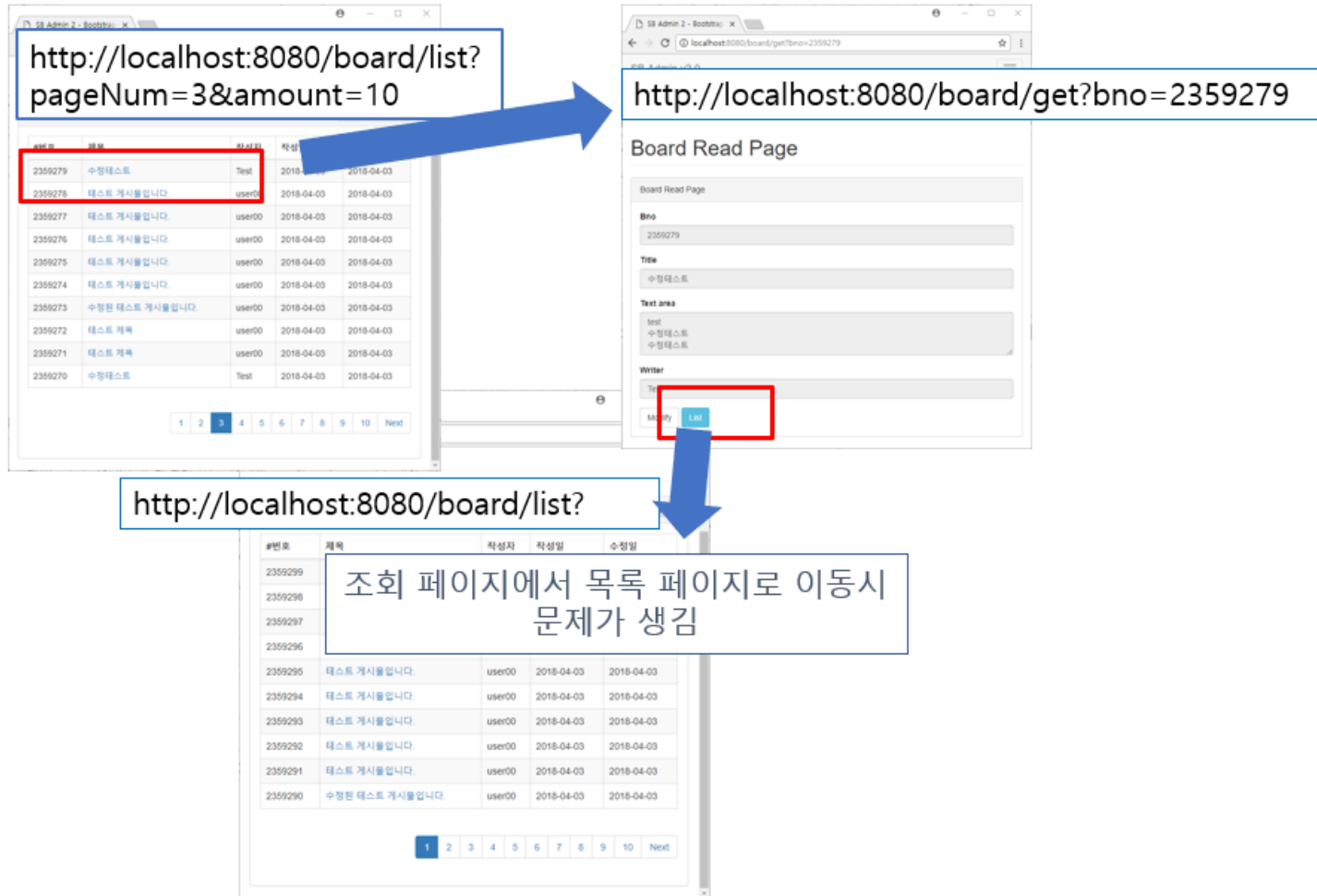
list.jsp 일부

```
//page 클릭 이벤트로 이동  
var pageForm = $("#pageForm");  
  
$(".page-link").on("click", function(e){  
  
    e.preventDefault(); //기본 동작 막기 : a태그의 기본동작은 이동 -> 이동을 막음  
  
    var targetPage = $(this).attr("href");  
  
    console.log(targetPage);  
    pageForm.find("input[name='pageNum']").val(targetPage);  
    pageForm.submit();  
});
```

list.jsp 일부

조회 페이지로 이동

- 조회 페이지 이동시 현재 페이지 번호를 같이 전달하는 방식으로 처리되어야 함



조회 페이지로 이동

```
<table width="100%" class="table table-striped table-bordered table-hover">
  <thead>
    <tr>
      <th>번호</th>
      <th>제목</th>
      <th>작성자</th>
      <th>작성일</th>
      <th>수정일</th>
    </tr>
  </thead>
  <tbody>
    <c:forEach var="board" items="${list}">
      <tr class="odd gradeX">
        <td><c:out value='${board.bno}' /></td>
        <td><a class="move" href="<c:out value='${board.bno}' />"><c:out value='${board.title}' /></a></td>
        <td><c:out value='${board.writer}' /></td>
        <td><fmt:formatDate pattern="yyyy-MM-dd" value='${board.regDate}' /></td>
        <td><fmt:formatDate pattern="yyyy-MM-dd" value='${board.updateDate}' /></td>
      </tr>
    </c:forEach>
  </tbody>
</table>
<!-- /.table-responsive -->
```

list.jsp 일부

이벤트 프로시저로 처리

조회 페이지로 이동을 이벤트로 처리

list.jsp 일부

```
$(".move").on("click", function(e){  
    e.preventDefault();  
  
    var targetBno = $(this).attr("href");  
    console.log(targetBno);  
    pageForm.append("<input type='hidden' name='bno' value='" + targetBno + "'>");  
    pageForm.attr("action", "/board/getOne");  
    pageForm.submit();  
});
```

The screenshot shows two web pages. The left page is the 'Board List Page' with a table of board entries. The right page is the 'Board Read Page' showing details for a specific board.

Board List Page:

#번호	제목	작성자	작성일	수정일
2359289	테스트 제목	user00	2018-04-03	2018-04-03
2359288	Test	Test	2018-04-03	2018-04-03
2359287	테스트 새글 제목	user00	2018-04-03	2018-04-03
2359286	새로 작성하는 글	newbie	2018-04-03	2018-04-03
2359285	새로 작성하는 글 select key	newbie	2018-04-03	2018-04-03
2359284	수정된 제목	user00	2018-04-03	2018-04-03
2359283	테스트 제목	user00	2018-04-03	2018-04-03
2359282	테스트 제목	user00	2018-04-03	2018-04-03
2359281	테스트 제목	user00	2018-04-03	2018-04-03
2359280	테스트 제목	user00	2018-04-03	2018-04-03

The URL for the list page is: `http://localhost:8080/board/list?pageNum=2&amount=10`

Board Read Page:

The URL for the read page is: `http://localhost:8080/board/get?pageNum=2&amount=10&bno=2359285`

The read page shows the following details:

- Bno:** 2359285
- Title:** 새로 작성하는 글 select key
- Text area:** 새로 작성하는 내용 select key
- Writer:** newbie
- Buttons:** Modify, List

조회 페이지(getOne.jsp)-> 목록 페이지 이동

- 페이지 번호 유지 방법

BoardController.java 일부

```
@GetMapping({"/getOne", "/modify"})  
public void getOne(@RequestParam("bno") Long bno,  
                  @ModelAttribute("cri") Criteria cri, Model model) {  
    log.info("get.....");  
    model.addAttribute("board", service.getOne(bno));  
}
```

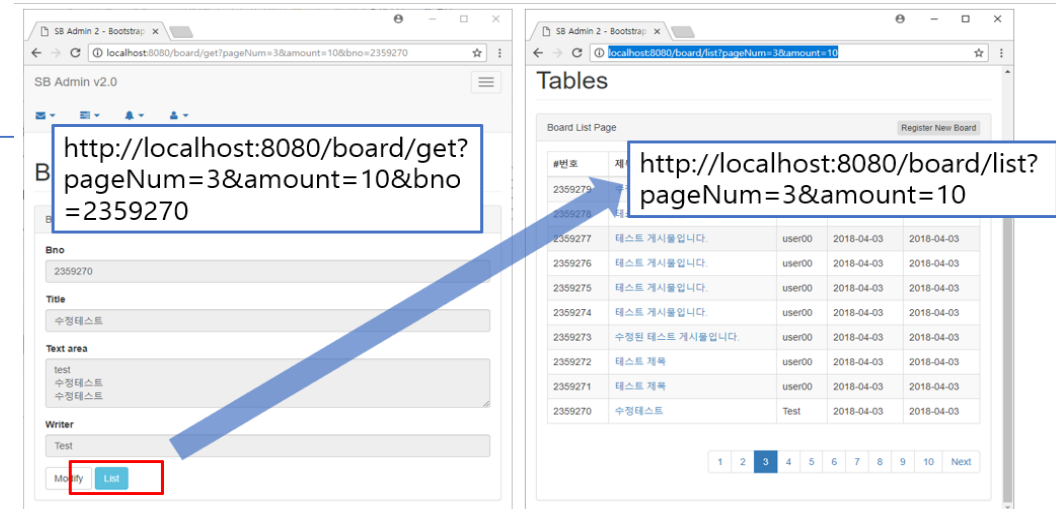
pageNum, amount를 처리하기 위해 전달

조회 페이지(getOne.jsp)-> 목록 페이지 이동

- 페이지 번호 유지 방법
 - 전달받은 페이지 번호를 이용해서 원래 페이지로 이동

getOne.jsp 일부

```
<form id="pageForm" action="/board/list" method="get">
  <input type="hidden" name="pageNum" value="${cri.pageNum}">
  <input type="hidden" name="amount" value="${cri.amount}">
  <input type="hidden" name="bno" value="${board.bno}">
</form>
<button class="btn btn-default modBtn" data-oper="modify">
  <a href="/board/modify?bno=<c:out value='${board.bno}']/>">Modify</a>
</button>
<button class="btn btn-default listBtn" data-oper="list">
  <a href="/board/list">List</a>
</button>
```



조회 페이지(getOne.jsp)-> 목록 페이지 이동

- list 버튼과 modify 버튼의 이벤트 프로시저

```
$(".listBtn").on("click", function(e){  
    e.preventDefault();  
  
    pageForm.find("input[name='bno']").remove();  
    pageForm.submit();  
  
});  
  
$(".modBtn").on("click", function(e){  
    e.preventDefault();  
  
    pageForm.attr("action", "/board/modify");  
    pageForm.submit();  
  
});
```

getOne.jsp 일부

수정/삭제

- 수정/삭제 페이지에서도 목록으로 이동할 수 있도록 전달

<http://localhost:8080/boards/list?pageNum=5&amount=10>

SB Admin 2 - Bootstrap

Tables

Board List Page

번호	제목	작성자	작성일	수정일
	새로 작성하는 글 select			
	수정된 제목	user00	2018-04-03	2018-04-03
2359257	테스트 제목	user00	2018-04-03	2018-04-03
2359256	테스트 제목	user00	2018-04-03	2018-04-03
2359255	테스트 제목	user00	2018-04-03	2018-04-03
2359254	테스트 제목	user00	2018-04-03	2018-04-03
2359253	새로 작성하는 글	newbie	2018-04-03	2018-04-03
2359252	수정테스트	Test	2018-04-03	2018-04-03
2359251	Test	Test	2018-04-03	2018-04-03
2359250	테스트 새글 제목	user00	2018-04-03	2018-04-03

1 2 3 4 5 6 7 8 9 10 Next

<http://localhost:8080/board/get?pageNum=5&amount=10&bno=2359251>

SB Admin v2.0

Board Read Page

Bno

2359251

Title

Test

Text area

Test

Writer

Test

Modify List

<http://localhost:8080/board/modify?bno=2359251&pageNum=5&amount=10>

SB Admin v2.0

Board Modify Page

Bno

2359251

Title

Test

Text area

Test

Writer

Test

Modify Remove List

수정/삭제 – BoardController.java 수정

```
@PostMapping("/modify")
public String modify(BoardVO board,
    @ModelAttribute("cri") Criteria cri, RedirectAttributes rttr) {
    log.info("modify : " + board);
    int count = service.modify(board);

    if(count==1)
        rttr.addFlashAttribute("result", "modified");

    rttr.addAttribute("pageNum", cri.getPageNum());
    rttr.addAttribute("amount", cri.getAmount());
    return "redirect:/board/list";
}

@PostMapping("/remove")
public String remove(@RequestParam("bno") Long bno,
    @ModelAttribute("cri") Criteria cri, RedirectAttributes rttr) {
    log.info("remove : " + bno);
    int count = service.remove(bno);

    if(count==1)
        rttr.addFlashAttribute("result", "removed");

    rttr.addAttribute("pageNum", cri.getPageNum());
    rttr.addAttribute("amount", cri.getAmount());
    return "redirect:/board/list";
}
```