

Spring Boot 4일차

db에 나이를 연산해서 넣지 않는다고?????

음?????

아 entity 같은 연산작업은 서비스(service)에서 한다!! db의 성능을 위해서.



시작

등록 작업 - register

service 패키지 생성

BoardService 인터페이스 생성

```
@Transactional
public interface BoardService {

    Long register(BoardDTO boardDTO);
}
```

BoardServiceImpl 클래스 생성

```
@Service
@Log4j2
@RequiredArgsConstructor
public class BoardServiceImpl implements BoardService {

    private final ModelMapper modelMapper;

    private final BoardRepository boardRepository;

    @Override
    public Long register(BoardDTO boardDTO) {
        // DTO -> entity
    }
}
```

```

        // repository save( save는 등록 수정때 사용 ) -> Long
        return null;
    }
}

```

```

@Service
@Log4j2
@RequiredArgsConstructor
public class BoardServiceImpl implements BoardService {

    private final ModelMapper modelMapper;

    private final BoardRepository boardRepository;

    @Override
    public Long register(BoardDTO boardDTO) {
        // DTO -> entity
        Board board = modelMapper.map(boardDTO, Board.class);
        // repository save( save는 등록 수정때 사용 ) -> Long
        boardRepository.save(board);
        return board.getBno();
    }
}

```

테스트 ㄱㄱ

test 에다가 service 패키지 test 계층을 만들어 줘야 한다.

test - service 패키지 생성 - BoardServiceTests 생성

그다음 database 에 Board 테이블 지우기

```

@SpringBootTest
@Log4j2
public class BoardServiceTests {

    @Autowired
    private BoardService boardService;

    @Test
    public void testRegister() {
        IntStream.rangeClosed(1,200).forEach(i -> { // service 계층이니까 DTO 가
필요
            BoardDTO boardDTO = BoardDTO.builder()
                .title("Title" + i)
                .content("Content" + i)
                .writer("user" + (i%10))
                .build();
            Long bno = boardService.register(boardDTO);
            log.info("BNO" + bno);
        });
    }
}

```

```
}
```

board 테이블 자동 생성되면서 콘솔에 짜라락 찍히면서 결과물이 딱

```
Hibernate:
```

```
create table board (  
    bno bigint not null auto_increment,  
    content varchar(255),  
    mod_date datetime(6),  
    reg_date datetime(6),  
    title varchar(255),  
    writer varchar(255),  
    primary key (bno)  
    ) engine=InnoDB
```

```
2021-10-07 09:50:05.606 INFO 1264 --- [    Test worker] org.zerock.sb.service.BoardServiceTests : BN01
```

```
2021-10-07 09:50:07.972 INFO 1264 --- [    Test worker] org.zerock.sb.service.BoardServiceTests : BN01
```

```
Hibernate:  
insert  
into  
    board  
    (content, mod_date, reg_date, title, writer)  
values  
    (?, ?, ?, ?, ?)
```

```
2021-10-07 09:50:07.979 INFO 1264 --- [    Test worker] org.zerock.sb.service.BoardServiceTests : BN02
```

```
Hibernate:  
insert  
into  
    board  
    (content, mod_date, reg_date, title, writer)  
values  
    (?, ?, ?, ?, ?)
```

```
2021-10-07 09:50:07.986 INFO 1264 --- [    Test worker] org.zerock.sb.service.BoardServiceTests : BN03
```

```
Hibernate:  
insert  
into
```

Profiler Terminal Build Services Endpoints Dependencies Spring

java x BoardServiceImpl.java x BoardServiceTests.java x console_2 x board x RootConfig.java x console_1 x Board.java

200 rows

WHERE ORDER BY

	bno	content	mod_date	reg_date	title	writer
1	1	Content1	2021-10-07 09:50:07.934740	2021-10-07 09:50:07.934740	Title1	user1
2	2	Content2	2021-10-07 09:50:07.974556	2021-10-07 09:50:07.974556	Title2	user2
3	3	Content3	2021-10-07 09:50:07.981539	2021-10-07 09:50:07.981539	Title3	user3
4	4	Content4	2021-10-07 09:50:07.987762	2021-10-07 09:50:07.987762	Title4	user4
5	5	Content5	2021-10-07 09:50:07.993745	2021-10-07 09:50:07.993745	Title5	user5
6	6	Content6	2021-10-07 09:50:08.001725	2021-10-07 09:50:08.001725	Title6	user6
7	7	Content7	2021-10-07 09:50:08.008760	2021-10-07 09:50:08.008760	Title7	user7
8	8	Content8	2021-10-07 09:50:08.017734	2021-10-07 09:50:08.017734	Title8	user8
9	9	Content9	2021-10-07 09:50:08.024717	2021-10-07 09:50:08.024717	Title9	user9
10	10	Content10	2021-10-07 09:50:08.033778	2021-10-07 09:50:08.033778	Title10	user0
11	11	Content11	2021-10-07 09:50:08.042753	2021-10-07 09:50:08.042753	Title11	user1
12	12	Content12	2021-10-07 09:50:08.049733	2021-10-07 09:50:08.049733	Title12	user2
13	13	Content13	2021-10-07 09:50:08.057712	2021-10-07 09:50:08.057712	Title13	user3
14	14	Content14	2021-10-07 09:50:08.064638	2021-10-07 09:50:08.064638	Title14	user4
15	15	Content15	2021-10-07 09:50:08.072616	2021-10-07 09:50:08.072616	Title15	user5
16	16	Content16	2021-10-07 09:50:08.080596	2021-10-07 09:50:08.080596	Title16	user6
17	17	Content17	2021-10-07 09:50:08.088575	2021-10-07 09:50:08.088575	Title17	user7
18	18	Content18	2021-10-07 09:50:08.095053	2021-10-07 09:50:08.095053	Title18	user8
19	19	Content19	2021-10-07 09:50:08.102034	2021-10-07 09:50:08.102034	Title19	user9
20	20	Content20	2021-10-07 09:50:08.109015	2021-10-07 09:50:08.109015	Title20	user0
21	21	Content21	2021-10-07 09:50:08.115998	2021-10-07 09:50:08.115998	Title21	user1

PageRequestDTO

PageRequestDTO 클래스 생성

```
import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;
import lombok.NoArgsConstructor;

@Builder
@AllArgsConstructor
@NoArgsConstructor
@Data
public class PageRequestDTO {

    @Builder.Default
    private int page = 1;

    @Builder.Default
    private int size = 10;

    private String type;

    private String keyword;

    public int getSkip() {
```

```

        return (page - 1) * size;
    }

    public char[] getTypes() {
        if(type == null || type.isEmpty()){
            return null;
        }
        return type.toCharArray();
    }
}

```

PageResponseDTO ㄱㄱ

```

import lombok.Data;

import java.util.List;

@Data
public class PageResponseDTO<E> { // 제네릭을 사용함

    //현재 페이지 번호
    private int page;
    //페이지당 게시물의 수
    private int size;

    //전체 데이터의 수
    private int count;

    //시작 페이지 번호
    private int start;
    // 끝 페이지 번호
    private int end;

    //이전 페이지 여부
    private boolean prev;
    //다음 페이지 여부
    private boolean next;

    //결과 데이터
    private List<E> dtoList;

    public PageResponseDTO(PageRequestDTO pageRequestDTO, int count){
        this.page = pageRequestDTO.getPage();
        this.size = pageRequestDTO.getSize();
        this.count = count;
        //임시 end 3/10.0 = 0.3 => 1 * 10 => 10
        int tempEnd = (int)(Math.ceil(page/(double)size)) * 10;

        this.start = tempEnd - 9;
        this.prev = start > 1;

        if(this.start < 1){ this.start = 1; }
    }
}

```

```

        //진짜 끝 페이지 count = 70 이고 tempEnd 10이라면
        if(tempEnd * size >= count ) {
            this.end = (int)(Math.ceil(count/(double)size));
            this.next = false;
        }else {
            this.end = tempEnd;
            this.next = true;
        }
    }

    public PageResponseDTO(PageRequestDTO pageRequestDTO, int count, List<E>
dtoList){
        this(pageRequestDTO, count);
        this.dtoList = dtoList;
    }
}

```

리스트 설계

BoardService ☞☞

```

@Transactional
public interface BoardService {

    Long register(BoardDTO boardDTO);

    PageResponseDTO<BoardDTO> getList(PageRequestDTO pageRequestDTO); // 추가
}

```

BoardServiceImpl ☞☞

implement method 해주기

```

@Service
@Log4j2
@RequiredArgsConstructor
public class BoardServiceImpl implements BoardService {

    private final ModelMapper modelMapper;

    private final BoardRepository boardRepository;

    @Override
    public Long register(BoardDTO boardDTO) {
        // DTO -> entity
        Board board = modelMapper.map(boardDTO, Board.class);
        // repository save( save는 등록 수정때 사용 ) -> Long
        boardRepository.save(board);
        return board.getBno();
    }
}

```

```

@Override // 이부분 작업
public PageResponseDTO<BoardDTO> getList(PageRequestDTO pageRequestDTO) {

    // 이부분 테스트 부분에서 복붙하움
    char[] typeArr = pageRequestDTO.getType();
    String keyword = pageRequestDTO.getKeyword();
    Pageable pageable = PageRequest.of(
        pageRequestDTO.getPage() - 1,
        pageRequestDTO.getSize(),
        Sort.by("bno").descending());

    Page<Board> result = boardRepository.search1(typeArr, keyword,
pageable);

    // get( ) 은 stream() 이다.
    List<BoardDTO> dtoList = result.get().map(
        board -> modelMapper.map(board, BoardDTO.class))
        .collect(Collectors.toList());
    long totalCount = result.getTotalElements();

    return new PageResponseDTO<>(pageRequestDTO, (int)totalCount, dtoList);
}
}

```

Controller 생성

```

@Controller
@Log4j2
@RequiredArgsConstructor
@RequestMapping("/board")
public class BoardController {

    private final BoardService boardService;

    @GetMapping("/list")
    public void list(PageRequestDTO pageRequestDTO, Model model) {

        model.addAttribute("responseDTO", boardService.getList(pageRequestDTO));
    }

}

```

resources 밑에다가 demo3 에 있는 static, templates 복붙

templates - board 디렉토리 생성

board 디렉토리 - list.html 생성

```

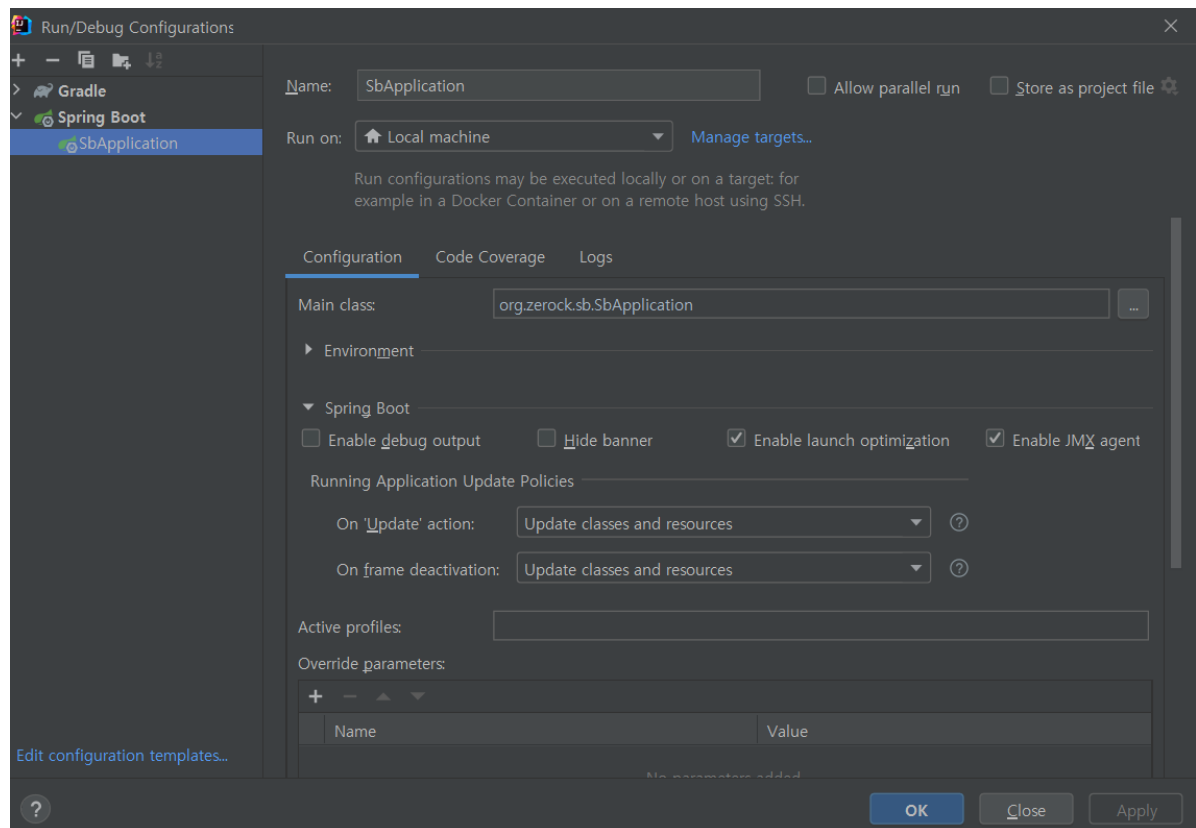
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
    layout:decorator="layout/layout2">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>

<div layout:fragment="content">
    [[${responseDTO}]]
</div>

</body>
</html>

```

그다음 밑에꺼 자동 새로그침 꼭 확인.



application.properties

```
spring.thymeleaf.cache=false
```

추가

그다음 실행

리스트 출력 확인

Start Bootstrap	Toggle Menu	Home Link Dropdown
Dashboard	PageResponseDTO(page=1, size=10, count=200, start=1, end=10, prev=false, next=true, dtoList=[BoardDTO(bno=200, title=Title200, content=Content200, writer=user0, regDate=2021-10-07T09:50:09.100365, modDate=2021-10-07T09:50:09.100365), BoardDTO(bno=199, title=Title199, content=Content199, writer=user9, regDate=2021-10-07T09:50:09.095379, modDate=2021-10-07T09:50:09.095379), BoardDTO(bno=198, title=Title198, content=Content198, writer=user8, regDate=2021-10-07T09:50:09.088397, modDate=2021-10-07T09:50:09.088397), BoardDTO(bno=197, title=Title197, content=Content197, writer=user7, regDate=2021-10-07T09:50:09.083416, modDate=2021-10-07T09:50:09.083416), BoardDTO(bno=196, title=Title196, content=Content196, writer=user6, regDate=2021-10-07T09:50:09.077428, modDate=2021-10-07T09:50:09.077428), BoardDTO(bno=195, title=Title195, content=Content195, writer=user5, regDate=2021-10-07T09:50:09.072440, modDate=2021-10-07T09:50:09.072440), BoardDTO(bno=194, title=Title194, content=Content194, writer=user4, regDate=2021-10-07T09:50:09.067455, modDate=2021-10-07T09:50:09.067455), BoardDTO(bno=193, title=Title193, content=Content193, writer=user3, regDate=2021-10-07T09:50:09.062467, modDate=2021-10-07T09:50:09.062467), BoardDTO(bno=192, title=Title192, content=Content192, writer=user2, regDate=2021-10-07T09:50:09.056486, modDate=2021-10-07T09:50:09.056486), BoardDTO(bno=191, title=Title191, content=Content191, writer=user1, regDate=2021-10-07T09:50:09.050498, modDate=2021-10-07T09:50:09.050498)])	
Shortcuts		
Overview		
Events		
Profile		
Status		

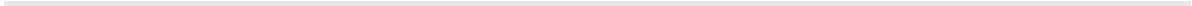
페이징 처리 확인

Start Bootstrap	Toggle Menu	Home Link Dropdown
Dashboard	PageResponseDTO(page=2, size=10, count=200, start=1, end=10, prev=false, next=true, dtoList=[BoardDTO(bno=190, title=Title190, content=Content190, writer=user0, regDate=2021-10-07T09:50:09.045513, modDate=2021-10-07T09:50:09.045513), BoardDTO(bno=189, title=Title189, content=Content189, writer=user9, regDate=2021-10-07T09:50:09.040526, modDate=2021-10-07T09:50:09.040526), BoardDTO(bno=188, title=Title188, content=Content188, writer=user8, regDate=2021-10-07T09:50:09.036536, modDate=2021-10-07T09:50:09.036536), BoardDTO(bno=187, title=Title187, content=Content187, writer=user7, regDate=2021-10-07T09:50:09.031549, modDate=2021-10-07T09:50:09.031549), BoardDTO(bno=186, title=Title186, content=Content186, writer=user6, regDate=2021-10-07T09:50:09.027561, modDate=2021-10-07T09:50:09.027561), BoardDTO(bno=185, title=Title185, content=Content185, writer=user5, regDate=2021-10-07T09:50:09.023571, modDate=2021-10-07T09:50:09.023571), BoardDTO(bno=184, title=Title184, content=Content184, writer=user4, regDate=2021-10-07T09:50:09.018584, modDate=2021-10-07T09:50:09.018584), BoardDTO(bno=183, title=Title183, content=Content183, writer=user3, regDate=2021-10-07T09:50:09.013598, modDate=2021-10-07T09:50:09.013598), BoardDTO(bno=182, title=Title182, content=Content182, writer=user2, regDate=2021-10-07T09:50:09.008610, modDate=2021-10-07T09:50:09.008610), BoardDTO(bno=181, title=Title181, content=Content181, writer=user1, regDate=2021-10-07T09:50:09.004621, modDate=2021-10-07T09:50:09.004621)])	
Shortcuts		
Overview		
Events		
Profile		
Status		

페이지, 검색 타입, 검색 키워드 확인

Start Bootstrap	Toggle Menu	Home Link Dropd
Dashboard	PageResponseDTO(page=2, size=10, count=12, start=1, end=2, prev=false, next=false, dtoList=[BoardDTO(bno=100, title=Title100, content=Content100, writer=user0, regDate=2021-10-07T09:50:08.646579, modDate=2021-10-07T09:50:08.646579), BoardDTO(bno=10, title=Title10, content=Content10, writer=user0, regDate=2021-10-07T09:50:08.033778, modDate=2021-10-07T09:50:08.033778)])	
Shortcuts		
Overview		
Events		

모두 확인되면 굿!!!



웹 화면에 테이블 이쁘게 만들기

list.html ㄱㄱ

```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
    layout:decorator="layout/layout2">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>

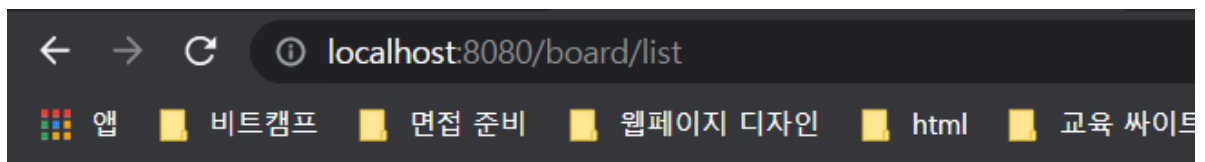
<div layout:fragment="content">

```

```

<div>
  <table>
    <tr>
      <td>BNO</td>
      <td>TITLE</td>
      <td>WRITER</td>
      <td>REGDATE</td>
    </tr>
    <tr th:each="dto: ${responseDTO.dtoList}">
      <td>BNO</td>
      <td>TITLE</td>
      <td>WRITER</td>
      <td>REGDATE</td>
    </tr>
  </table>
</div>
</div>
</body>
</html>

```



Start Bootstrap	Toggle Menu
Dashboard	BNO TITLE WRITER REGDATE
Shortcuts	BNO TITLE WRITER REGDATE
Overview	BNO TITLE WRITER REGDATE
Events	BNO TITLE WRITER REGDATE
Profile	BNO TITLE WRITER REGDATE
Status	BNO TITLE WRITER REGDATE

그다음

```
<div layout:fragment="content">
```

```

<div>
  <table>
    <tr>
      <td>BNO</td>
      <td>TITLE</td>
      <td>WRITER</td>
      <td>REGDATE</td>
    </tr>
    <tr th:each="dto: ${responseDTO.dtoList}">
      <td th:text="${dto.bno}">BNO</td>
      <td th:text="${dto.title}">TITLE</td>
      <td th:text="${dto.writer}">WRITER</td>
      <td th:text="${dto.regDate}">REGDATE</td>
    </tr>
  </table>
</div>
</div>

```

일단 빨간 줄 무시

```

<div layout:fragment="content">

  <div>
    <table>
      <tr>
        <td>BNO</td>
        <td>TITLE</td>
        <td>WRITER</td>
        <td>REGDATE</td>
      </tr>
      <tr th:each="dto: ${responseDTO.dtoList}">
        <td th:text="${dto.bno}">BNO</td>
        <td th:text="${dto.title}">TITLE</td>
        <td th:text="${dto.writer}">WRITER</td>
        <td th:text="${dto.regDate}">REGDATE</td>
      </tr>
    </table>
  </div>

```

결과

<div> <div>← → ↺</div> <div>localhost8080/board/list</div> </div> <div> <div>앱</div> <div>비트캠프</div> <div>면접 준비</div> <div>웹페이지 디자인</div> <div>html</div> <div>교육 사이트</div> <div>Team Project</div> </div>	
Start Bootstrap	Toggle Menu
Dashboard	<div> <div>BNO</div> <div>TITLE</div> <div>WRITER</div> <div>REGDATE</div> </div> <div> <div>200</div> <div>Title200</div> <div>user0</div> <div>2021-10-07T09:50:09.100365</div> </div>
Shortcuts	<div> <div>199</div> <div>Title199</div> <div>user9</div> <div>2021-10-07T09:50:09.095379</div> </div> <div> <div>198</div> <div>Title198</div> <div>user8</div> <div>2021-10-07T09:50:09.088397</div> </div>
Overview	<div> <div>197</div> <div>Title197</div> <div>user7</div> <div>2021-10-07T09:50:09.083416</div> </div> <div> <div>196</div> <div>Title196</div> <div>user6</div> <div>2021-10-07T09:50:09.077428</div> </div>
Events	<div> <div>195</div> <div>Title195</div> <div>user5</div> <div>2021-10-07T09:50:09.072440</div> </div> <div> <div>194</div> <div>Title194</div> <div>user4</div> <div>2021-10-07T09:50:09.067455</div> </div>
Profile	<div> <div>193</div> <div>Title193</div> <div>user3</div> <div>2021-10-07T09:50:09.062467</div> </div> <div> <div>192</div> <div>Title192</div> <div>user2</div> <div>2021-10-07T09:50:09.056486</div> </div>
Status	<div> <div>191</div> <div>Title191</div> <div>user1</div> <div>2021-10-07T09:50:09.050498</div> </div>

웹 페이징 및 검색 기능 구축

```

<div layout:fragment="content">

    <div>
        <table>
            <tr>
                <td>BNO</td>
                <td>TITLE</td>
                <td>WRITER</td>
                <td>REGDATE</td>
            </tr>
            <tr th:each="dto: ${responseDTO.dtoList}">
                <td th:text="${dto.bno}">BNO</td>
                <td th:text="${dto.title}">TITLE</td>
                <td th:text="${dto.writer}">WRITER</td>
                <td th:text="${dto.regDate}">REGDATE</td>
            </tr>
        </table>
    </div>
    <!--페이지 용 div-->
    <div>
        <ul class="pagination">
            <li class="page-item" th:if="${responseDTO.prev}">
                <a class="page-link" href="#" tabindex="-1">Previous</a>
            </li>
            <li class="page-item"><a class="page-link" href="#">1</a></li>
        </ul>
    </div>

```


←

→

↺

localhost:8080/board/list#

앱

비트캠프

면접 준비

웹페이지 디자인

html

교육 사이트

Team Project

Start Bootstrap	Toggle Menu
Dashboard	<div> <div>BNO</div> <div>TITLE</div> <div>WRITER</div> <div>REGDATE</div> </div> <div> <div>200</div> <div>Title200</div> <div>user0</div> <div>2021-10-07T09:50:09.100365</div> </div>
Shortcuts	<div> <div>199</div> <div>Title199</div> <div>user9</div> <div>2021-10-07T09:50:09.095379</div> </div> <div> <div>198</div> <div>Title198</div> <div>user8</div> <div>2021-10-07T09:50:09.088397</div> </div>
Overview	<div> <div>197</div> <div>Title197</div> <div>user7</div> <div>2021-10-07T09:50:09.083416</div> </div> <div> <div>196</div> <div>Title196</div> <div>user6</div> <div>2021-10-07T09:50:09.077428</div> </div>
Events	<div> <div>195</div> <div>Title195</div> <div>user5</div> <div>2021-10-07T09:50:09.072440</div> </div> <div> <div>194</div> <div>Title194</div> <div>user4</div> <div>2021-10-07T09:50:09.067455</div> </div>
Profile	<div> <div>193</div> <div>Title193</div> <div>user3</div> <div>2021-10-07T09:50:09.062467</div> </div> <div> <div>192</div> <div>Title192</div> <div>user2</div> <div>2021-10-07T09:50:09.056486</div> </div>
Status	<div> <div>191</div> <div>Title191</div> <div>user1</div> <div>2021-10-07T09:50:09.050498</div> </div> <div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> <div>6</div> <div>7</div> <div>8</div> <div>9</div> <div>10</div> <div>Next</div> </div>

일단 강사님께서로

```
<div class="card-body">

    <div class="float-end" >
        <select id="searchType" >
            <option value="" th:selected="${pageRequestDTO.type} == null">-----
--</option>
            <option value="T" th:selected="${pageRequestDTO.type} == 'T'">제목
</option>
            <option value="C" th:selected="${pageRequestDTO.type} == 'C'">내용
</option>
            <option value="W" th:selected="${pageRequestDTO.type} == 'W'">작성자
</option>
            <option value="TC" th:selected="${pageRequestDTO.type} == 'TC'">제목/내용
</option>
            <option value="TCW" th:selected="${pageRequestDTO.type} == 'TCW'" >제목
혹은 내용 혹은 작성자</option>
        </select>
        <input type="text" id="keyword" th:value="${pageRequestDTO.keyword}">
        <button id="searchBtn" class="btn btn-info"
onlick="search()">Search</button>
    </div>

    <table class="table">
        <thead>
            <tr>
                <th scope="col">Bno</th>
                <th scope="col">Title</th>
```

```

        <th scope="col">writer</th>
        <th scope="col">RegDate</th>
    </tr>
</thead>
<tbody>
    <tr th:each="board:${responseDTO.dtoList}">
        <th scope="row">[[${board.bno}]]</th>
        <td><a th:href="|javascript:movePage(${responseDTO.page}, ${board.bno}
)|">[[${board.title}]]</a></td>
        <td>[[${ board.writer}]]</td>
        <td>[[${#temporals.format(board.regDate, 'yyyy/MM/dd')}]]</td>
    </tr>
</tbody>
</table>
</div>

<div class="d-flex justify-content-center">
    <ul class="pagination">
        <li th:if="${responseDTO.prev}" class="page-item">
            <a class="page-link" th:href="|javascript:movePage(${responseDTO.start-
1})|" aria-label="Previous">
                <span aria-hidden="true">&laquo;</span>
                <span class="sr-only">Previous</span>
            </a>
        </li>

        <li th:each="num: ${#numbers.sequence( responseDTO.start ,
responseDTO.end)}" th:class="|page-item ${responseDTO.page ==
num?'active':''}|">
            <a class="page-link" th:href="|javascript:movePage(${num})|">[[${num}]]
        </a>
        </li>

        <li th:if="${responseDTO.next}" class="page-item">
            <a class="page-link"
th:href="|javascript:movePage(${responseDTO.end+1})|" aria-label="Next">
                <span aria-hidden="true">&raquo;</span>
                <span class="sr-only">Next</span>
            </a>
        </li>
    </ul>
</div>

```

Bno	Title	Writer	RegDate
200	Title200	user0	2021/10/07
199	Title199	user9	2021/10/07
198	Title198	user8	2021/10/07
197	Title197	user7	2021/10/07
196	Title196	user6	2021/10/07
195	Title195	user5	2021/10/07
194	Title194	user4	2021/10/07
193	Title193	user3	2021/10/07
192	Title192	user2	2021/10/07
191	Title191	user1	2021/10/07

위에꺼 그냥 복붙 하면 위에처럼 게시판이 저렇게 나온다.

여기서 핵심

여기서 눈여겨 봐야할것은 javascript 처리이다.

layout2.html ㄱㄱ

맨밑에 <script> 맨 밑으로 ㄱㄱ

```
<!-- Core theme JS-->
<script th:src="@{/js/scripts.js}"></script>

<th:block layout:fragment="script">

</th:block>
```

list.html ㄱㄱ

맨밑에 가서

```
</div>

<th:block layout:fragment="script">
  <script th:inline="javascript">

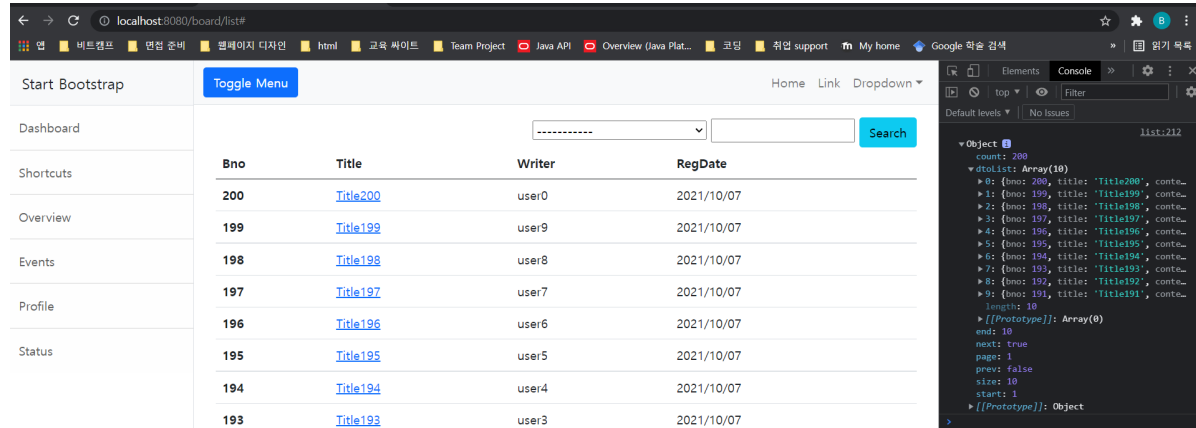
    console.dir([[${responseDTO}]]);
    alert("AAA")

  </script>
</th:block>

</body>
</html>
```

th:block 은 안보이는 창이다.

alter "AAA" 창 뜨고



그리고 마지막으로

저 위에

```
<th:block layout:fragment="script">
  <script th:inline="javascript">

    console.dir([[${responseDTO}]]);
    alert("AAA")

  </script>
</th:block>
```

이 부분을 밑에처럼 싹 바꿔주면!!!

```
<th:block layout:fragment="script">

  <div class="modal" tabindex="-1">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <h5 class="modal-title">Modal title</h5>
          <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
        </div>
        <div class="modal-body">
          <p>Modal body text goes here.</p>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
          <button type="button" class="btn btn-primary">Save changes</button>
        </div>
      </div>
    </div>
  </div>
</th:block>
```

```

<script th:inline="javascript">

    const myModal = new bootstrap.Modal(document.querySelector(".modal"))

    const result = [{${result}]}

    if(result){
        myModal.show()
    }

    const pageRequest = [{${pageRequestDTO}]}

    function movePage(page, bno) {

        console.log(pageRequest)

        let uri = "";

        const type = pageRequest.type
        const keyword = pageRequest.keyword

        let searchQuery = ''
        if(type && keyword){
            searchQuery += `&type=${type}&keyword=${keyword}`
        }

        if(bno) {
            uri = `/board/read?bno=${bno}&page=${page}` + searchQuery
        }else {
            uri = `/board/list?page=${page}` + searchQuery
        }
        self.location = uri
    }

    function search() {
        const searchSelect = document.querySelector("#searchType")

        const type = searchSelect.options[searchSelect.selectedIndex].value

        const keyword = encodeURIComponent(document.querySelector("#keyword").value)

        console.log("type", type, "keyword", keyword)

        self.location = `/board/list?page=1&type=${type}&keyword=${keyword}`
    }

</script>

```

```

</th:block>

```

localhost:8080/board/list?page=2&type=W&keyword=9

Start Bootstrap Toggle Menu Home Link Dropdown

Dashboard Shortcuts Overview Events Profile Status

작성자 9 Search

Bno	Title	Writer	RegDate
99	Title99	user9	2021/10/07
89	Title89	user9	2021/10/07
79	Title79	user9	2021/10/07
69	Title69	user9	2021/10/07
59	Title59	user9	2021/10/07
49	Title49	user9	2021/10/07
39	Title39	user9	2021/10/07
29	Title29	user9	2021/10/07
19	Title19	user9	2021/10/07
9	Title9	user9	2021/10/07

1 2

검색과 페이징이 된다.

웹 등록 구현 작업

post redirect get 방식 이용

get post 이거 반복 노가다 안하려면 그때 상속을 사용한다고 한다.

BoardController ㄱㄱ

```
@GetMapping("/register")
public void register() {

}

@PostMapping("/register")
public String registerPost(BoardDTO boardDTO, RedirectAttributes
redirectAttributes) {

    Long bno = boardService.register(boardDTO);

    redirectAttributes.addFlashAttribute("result", bno);

    return "redirect:/board/list";

}
```

resources - templates - board - register.html 생성

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org"
xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
    layout:decorator="layout/layout2">

<div layout:fragment="content">

</div>
```

그다음 안에

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org"
      xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
      layout:decorator="layout/layout2">

<div class="card" layout:fragment="content">
  <div class="card-header">
    Board Register
  </div>

  <div class="card-body">
    <form th:action="@{/board/register}" method="post">
      <div class="input-group mb-3">
        <span class="input-group-text">Title</span>
        <input type="text" name="title" class="form-control"
placeholder="Title">
      </div>

      <div class="input-group mb-3">
        <span class="input-group-text">Writer</span>
        <input type="text" name="writer" class="form-control"
placeholder="writer">
      </div>

      <div class="input-group">
        <span class="input-group-text">Content</span>
        <textarea class="form-control" aria-label="with textarea"
name="content" rows="4"></textarea>
      </div>
      <div class="my-4">
        <div class="float-end">
          <button type="submit" class="btn btn-
primary">Submit</button>
          <button type="result" class="btn btn-
secondary">Reset</button>
        </div>
      </div>
    </form>
  </div>
</div>
```

Start Bootstrap

Toggle Menu

Home Link Dropdown

Dashboard

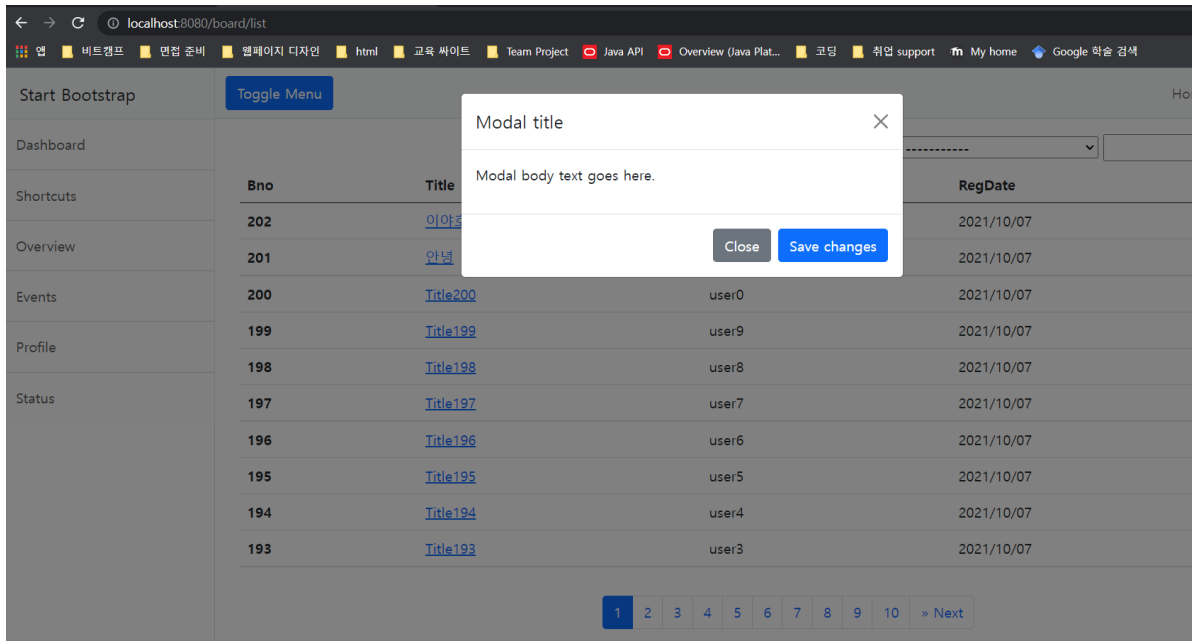
Board Register

Title 이야호

Writer 후하하

Content 다음주 수업 화요일 금요일 출

Submit Reset



Bno	Title	Writer	RegDate
202	이아호	후하하	2021/10/07

웹에 조회 페이지 구축 - Read

BoardService ㄱㄱ

```
@Transactional
public interface BoardService {

    Long register(BoardDTO boardDTO);

    PageResponseDTO<BoardDTO> getList(PageRequestDTO pageRequestDTO);

    BoardDTO read(Long bno);
}
```

BoardServiceImpl ㄱㄱ

implement method 해주기

```
@Override
public BoardDTO read(Long bno) {
    Optional<Board> result = boardRepository.findById(bno);

    if (result.isEmpty()) {
        throw new RuntimeException("Not Found");
    }

    // get() -> 내용물
```

```

        BoardDTO dto = modelMapper.map(result.get(), BoardDTO.class);
        // get() 으로 받아온 내용물을 BoardDTO 타입으로 바꿔준다.

        return dto;
    }

```

BoardController ㄱㄱ

```

@GetMapping("/read")
public void read(Long bno, PageRequestDTO pageRequestDTO, Model model) {

    model.addAttribute("dto", boardService.read(bno) );
}

```

그다음

read.html 생성

그후

```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org"
      xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
      layout:decorator="layout/layout2">

<div class="card" layout:fragment="content">

    <div class="card">
        <div class="card-header">
            Board Read
        </div>
        <div class="card-body">
            <div class="input-group mb-3">
                <span class="input-group-text">Bno</span>
                <input type="text" name="bno" class="form-control" readonly
                    th:value="${dto.bno}" >
            </div>
            <div class="input-group mb-3">
                <span class="input-group-text">Title</span>
                <input type="text" name="title" class="form-control" readonly
                    th:value="${dto.title}" >
            </div>

            <div class="input-group mb-3">
                <span class="input-group-text">Writer</span>
                <input type="text" name="writer" class="form-control" readonly
                    th:value="${dto.writer}">
            </div>

            <div class="input-group mb-3">

```

```

        <span class="input-group-text">RegDate</span>
        <input type="text" name="regDate" class="form-control" readonly
            th:value="${#temporals.format(dto.regDate, 'yyyy/MM/dd
HH:mm:ss')}" >
    </div>

    <div class="input-group mb-3">
        <span class="input-group-text">ModDate</span>
        <input type="text" name="modDate" class="form-control" readonly
            th:value="${#temporals.format(dto.modDate, 'yyyy/MM/dd
HH:mm:ss')}">
    </div>

    <div class="input-group">
        <span class="input-group-text">Content</span>
        <textarea class="form-control" aria-label="with textarea"
name="content" rows="4">[[${dto.content}]]</textarea>
    </div>

    <div class="my-4">
        <div class="my-4">
            <div class="float-end">
                <button type="button" class="btn btn-primary"
th:onclick="|modify(${dto.bno})|">Modify</button>
                <button type="button" class="btn btn-secondary"
th:onclick="|moveList()|">List</button>
            </div>
        </div>
    </div>
</div>
</div>
</div>
</div>

<th:block layout:fragment="script">

    <script th:inline="javascript">
        function modify(bno){

            const page = [[${pageRequestDTO.page}]]
            const size = [[${pageRequestDTO.size}]]
            const type = [[${pageRequestDTO.type}]]
            const keyword = [[${pageRequestDTO.keyword}]]

            let searchQuery = ''

            if(type && keyword){
                searchQuery += `&type=${type}&keyword=${keyword}`
            }

            self.location=`/board/modify?
bno=${bno}&page=${page}&size=${size}`+searchQuery

        }

        function moveList(){

            const page = [[${pageRequestDTO.page}]]

```

```

const size = [[${pageRequestDTO.size}]]
const type = [[${pageRequestDTO.type}]]
const keyword = [[${pageRequestDTO.keyword}]]

let searchQuery = ''

if(type && keyword){
    searchQuery += `&type=${type}&keyword=${keyword}`
}

self.location=`/board/list?page=${page}&size=${size}` + searchQuery

}

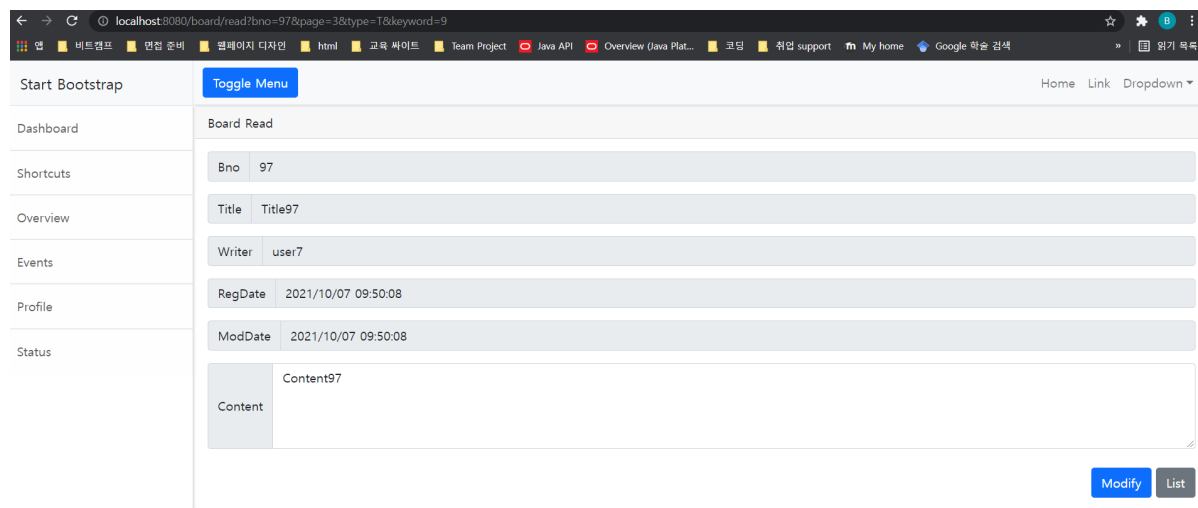
</script>
</th:block>

```

복붙!!

그다음

실행



잘 뜬다

그리고 read 페이지에서 list 로 가면 page 가 잘 유지되고 검색 후에 read 페이지 들어갔다가 다시 list 로 가면 검색조건과 page 가 유지되면 성공!!

웹에서 수정 삭제 구현

BoardService ㄱ ㄱ


```

@Transactional
public interface BoardService {

    Long register(BoardDTO boardDTO);

    PageResponseDTO<BoardDTO> getList(PageRequestDTO pageRequestDTO);

    BoardDTO read(Long bno);

    void modify(BoardDTO boardDTO); // 추가

    void remove(Long bno); // 추가
}

```

BoardServiceImpl ㄱㄱ

implement method 해주자.

```

@Service
@Log4j2
@RequiredArgsConstructor
public class BoardServiceImpl implements BoardService {
    .
    .
    .
    @Override
    public void modify(BoardDTO boardDTO) {
        Optional<Board> result = boardRepository.findById(boardDTO.getBno());

        // 예외는 메시지다. 고객에게 알려주는 메시지다. 뭔가 없다고 또는 뭔가 안된다고.
        // 즉 상부에게 보고하는 거라고 보면 된다.
        if (result.isEmpty()) {
            throw new RuntimeException("Not Found");
        }
        Board board = result.get();
        board.change(board.getTitle(), board.getContent());
        boardRepository.save(board);
    }

    @Override
    public void remove(Long bno) {

        boardRepository.deleteById(bno);
    }
}

```

그다음

test ㄱㄱ

BoardServiceTests ㄱㄱ

```

@Test
public void testModify() {
    BoardDTO boardDTO = BoardDTO.builder()
        .bno(200L)
        .title("그럴치 그럴치")
        .content("작동 되어라").build();

    boardService.modify(boardDTO);
}

```

Bno	Title	Writer	RegDate
202	한글한글	후하하	2021/10/07
201	안녕	하하	2021/10/07
200	Title200	user0	2021/10/07

test 실행 ㄱㄱ

Hibernate:

select

board0_.bno as bno1_0_0_,
 board0_.content as content2_0_0_,
 board0_.mod_date as mod_date3_0_0_,
 board0_.reg_date as reg_date4_0_0_,
 board0_.title as title5_0_0_,
 board0_.writer as writer6_0_0_

from

board board0_

where

board0_.bno=?

Hibernate:

update

board

set

content=?,
 mod_date=?,
 reg_date=?,
 title=?,
 writer=?

where

bno=?

2021-10-07 12:34:38.805 INFO 15152 --- [ionShu

콘솔에서 select 해서 해당 bno의 게시글을 확인하고

save 를 통해 update 시켜버렸다.

그다음 웹 확인

				제목	그렇치	Search
Bno	Title	Writer	RegDate			
200	그렇치, 그렇치	user0	2021/10/07			
				1		

Bno 200번 게시글이 수정되었고 검색에서 한글로 검색이 되는것을 확인 할 수 있다.

오늘 끝!!!!!!