

My Book | 네책 Your Book | 내책

이승재의 포트폴리오



INDEX

1. 프로그램 개요

-네책내책

2. 이용자별 주요 기능

-Process

3. 다이어그램

-UseCase/ Class

4. 구현했던 기능

-이용자별 페이지

5. 오류해결 또는

까다로웠던 부분



프로그램 개요
My B Your B

My B Your B ▼

My B Your B

MYBOOK (네책 내책)

내가 현재 보고 있는 책이 네 책이 될 수 있다고 하는 의미를 담고 있는 중의적 의미.

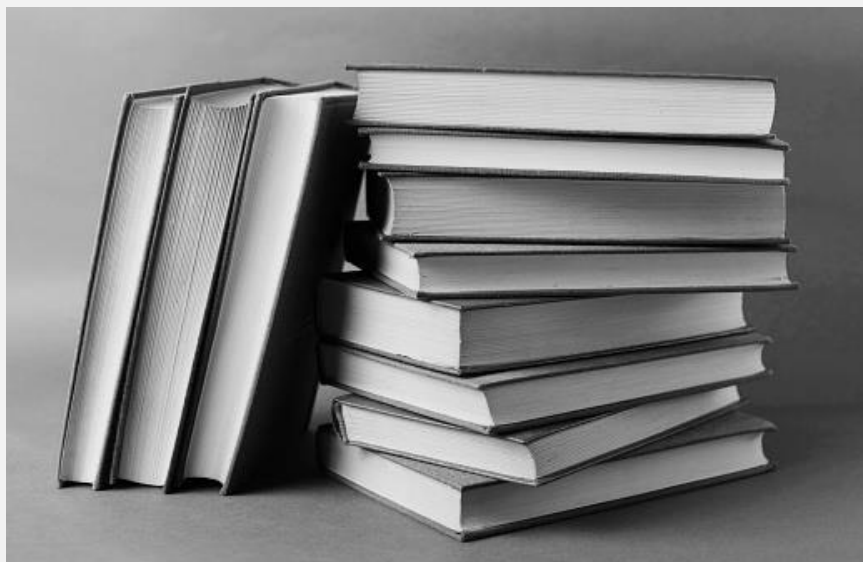


당신의 책은

책장 속에서 잠자고 있지 않나요?



내게 필요 없는 책을 필요한 누군가에게
그리고 누군가에게 필요없는 책을 나에게



죽은 책 살리기

현재 트렌드인 중고 거래에 추가적으로 도서 쉐어 서비스를 제공.

새 책의 경우 비회원 구매 회원구매가 가능.

중고 도서 상태에 따라 구분하여, 대여/판매를 구분.
비회원의 경우 구매만 가능.

이용자 별 주요 기능



구매자

저렴한 가격에 구매 가능, 대
여 기능을 통해 구매 전 미리
읽고, 구매 가능



기증자

자리만 차지하던 책을 기증하여
포인트 적립 및 포인트를 통해 대
여 가능

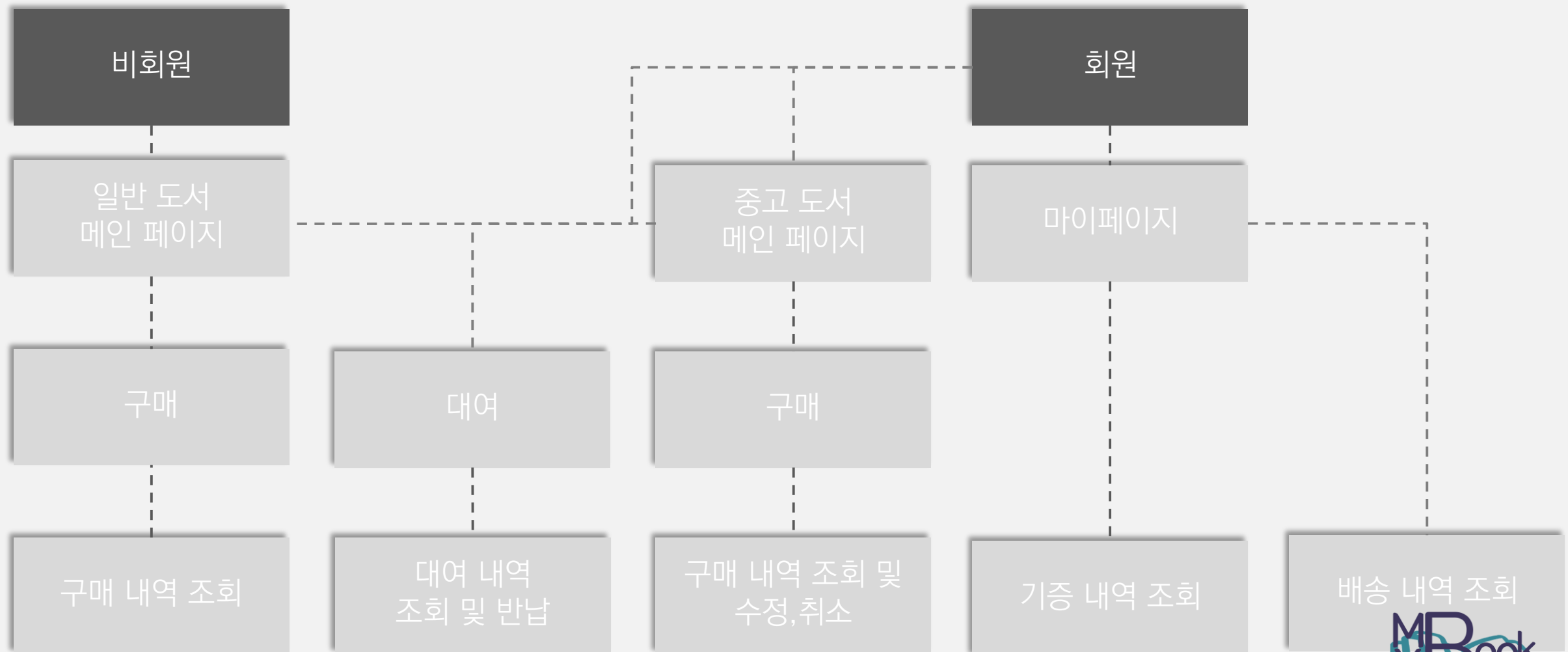


관리자

매달 매출 현황, 베스트 셀러,
독서왕, 구매왕 등
전반적인 고객관리

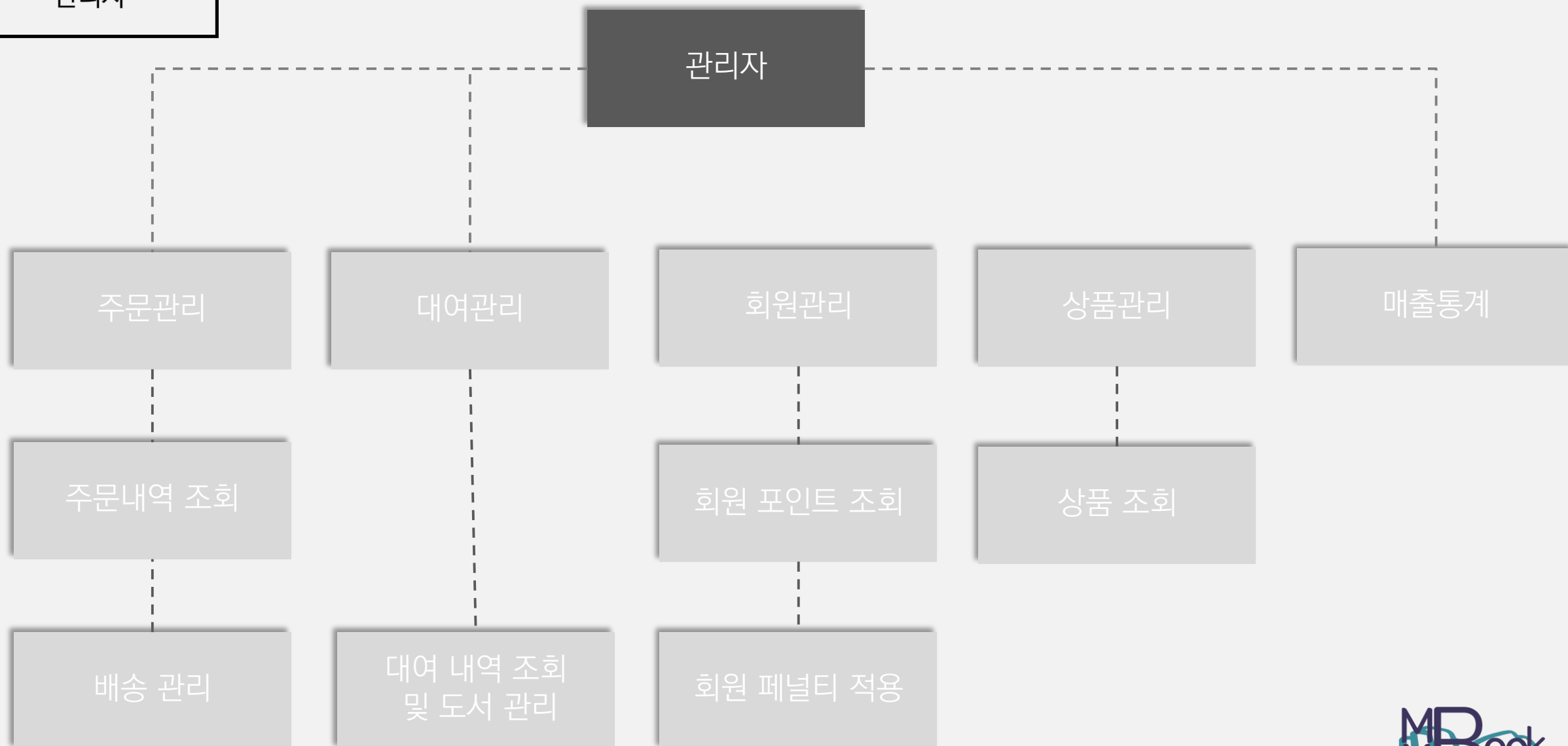
Process
이용자

Process | 이용자

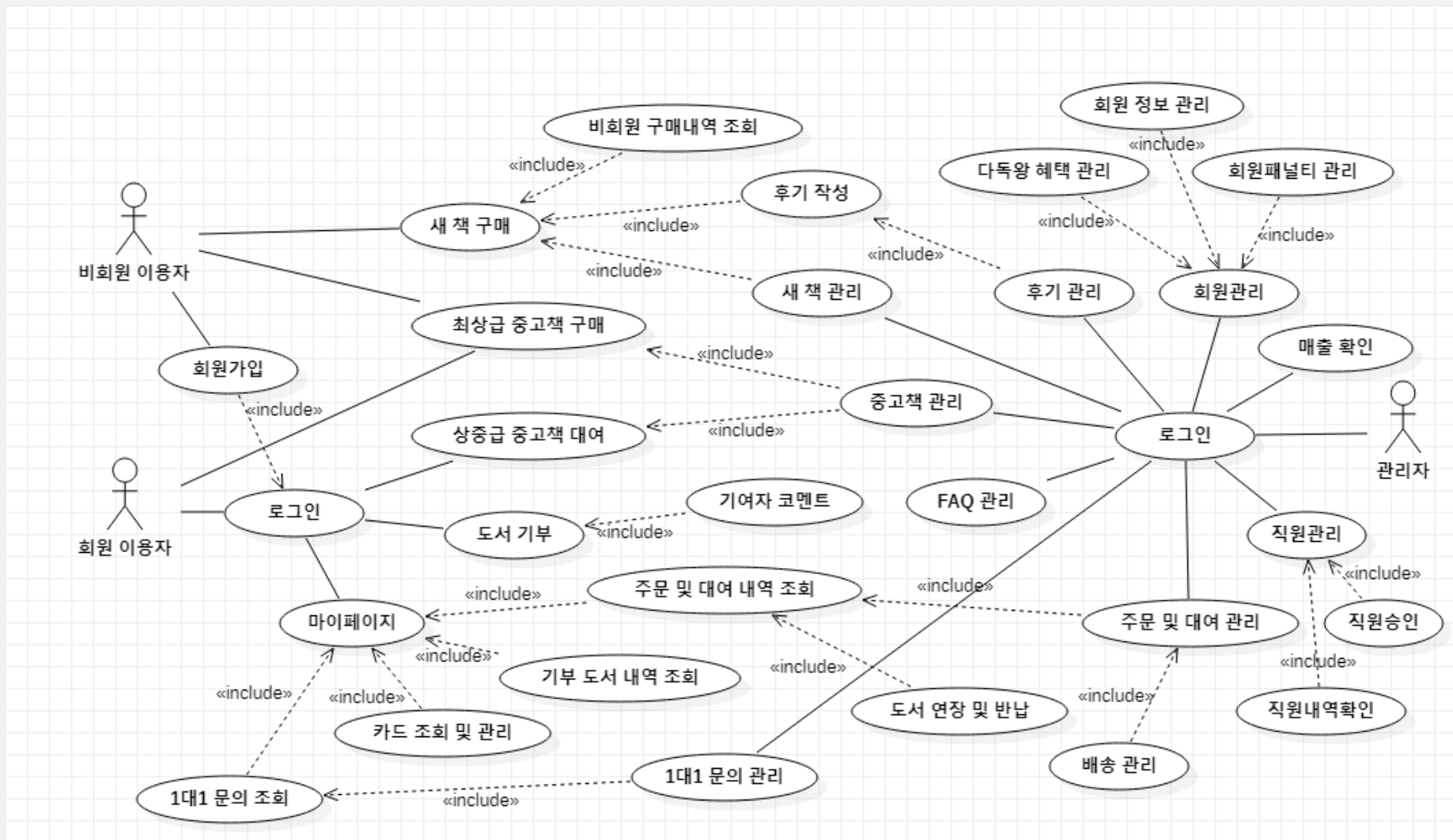


Process
관리자

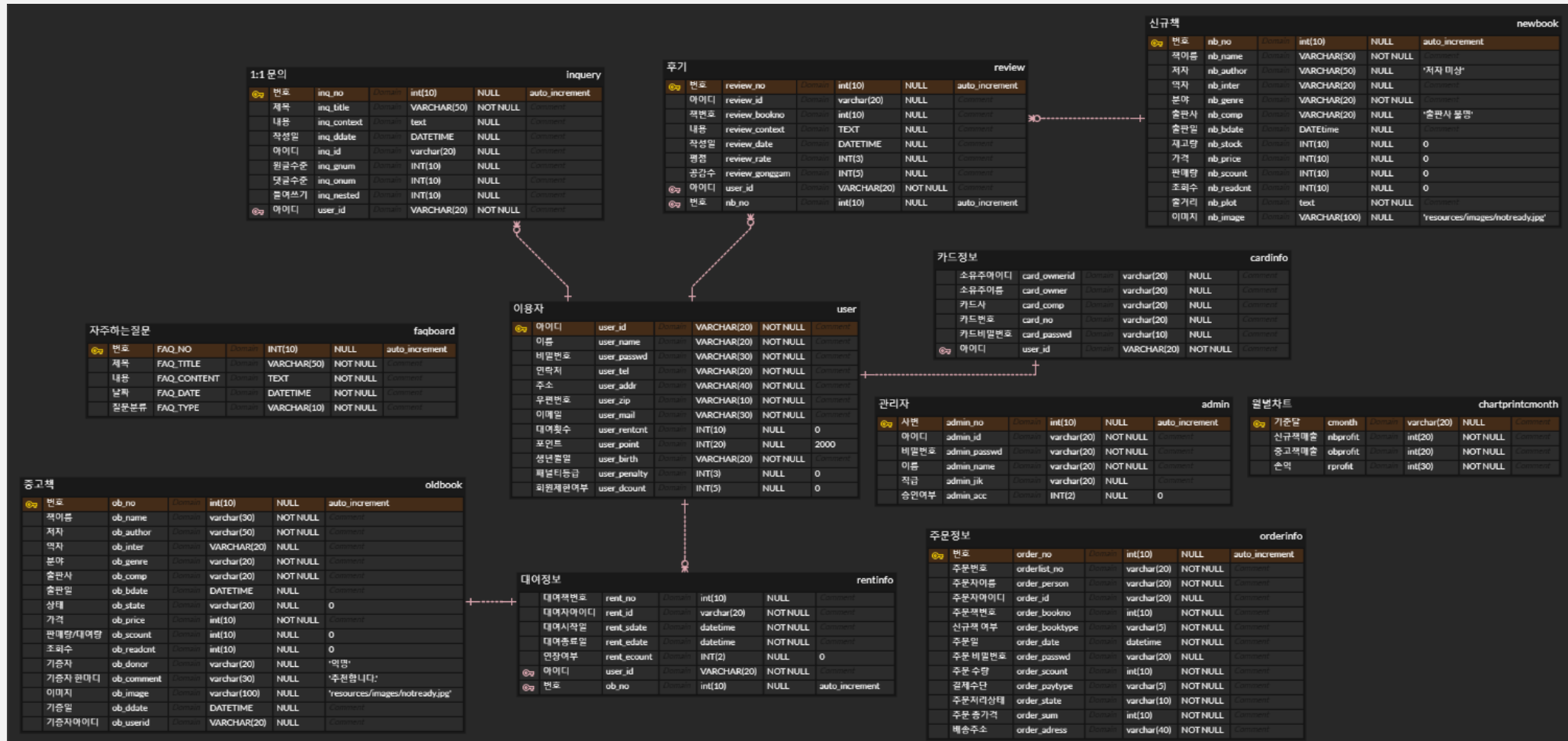
Process | 관리자



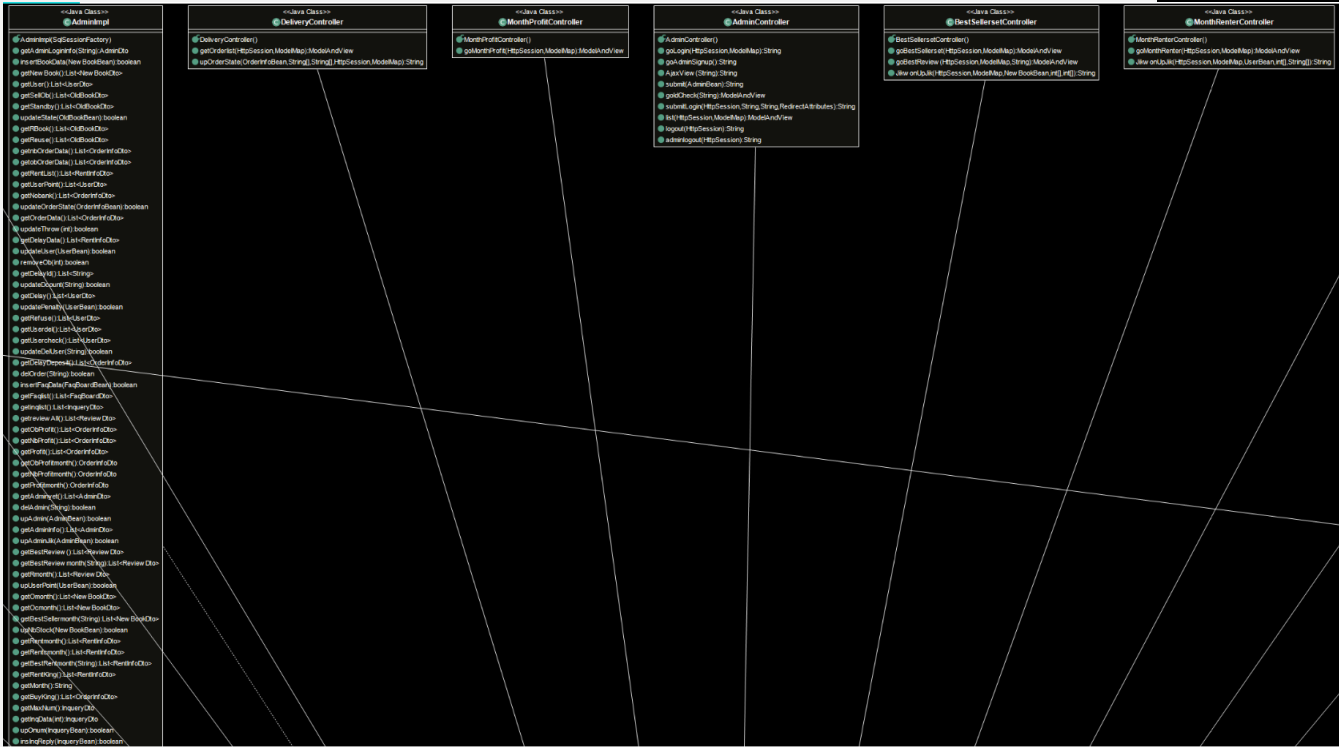
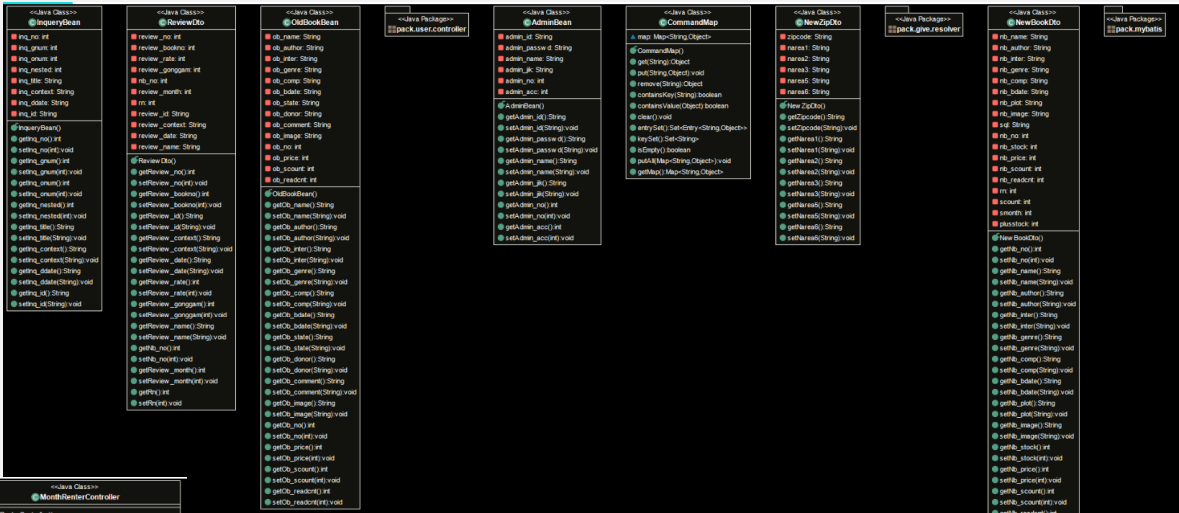
UseCase Model



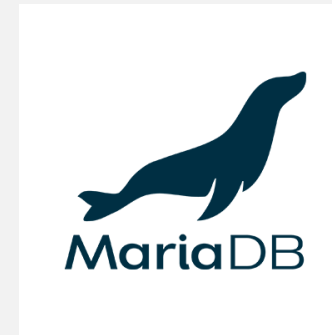
ER-D



Class Diagram



- 여기서 제가 구현한 기능들은, 새책 메인 페이지와
- 중고 메인페이지, 새책(중고) 구매 정보, 주문번호 생성기,
- 비회원 주문조회, 구매 완료페이지를 구현하였습니다.
- 아래에는 제가 구현한 기능들을 설명하였습니다.
- 제가 사용한 사용 기술입니다.



새책 메인 페이지

My B Your B Home 로그인 회원가입 비회원주문조회 1:1문의 제목 ▼ 도서 검색하기 검색

전체보기 장르 베스트30 신간 중고책관련

베스트 셀러

조희수 TOP3

방문자님 환영합니다.

책 기증하기

포인트를 얻고 싶다면?

오늘의 추천

영어 전치사 연구
의미와 활용
ENGLISH PREPOSITIONS

털털한 아롱이

한계전의 멍치퀴

TOEIC 로익 정기시험 기술문제집 2
1000 READING

미국 영어말을 무자랑 따라하기
영어 공부 용어

King Sejong the Great
김태종

EBS 수석교수 최정호는 AI 기초

시간을 파는 상점

미국 영어말을 무자랑 따라하기
영어 공부 용어

- 새 책은 조희수 순 TOP3노출
- 오늘의 추천은 랜덤으로 10가지의 책을 노출

- 장르 버튼 클릭 드랍다운 메뉴 노출



중고책 메인 페이지

My B Your B Home 로그인 회원가입 비회원주문조회 1:1문의 방문해주셔서 감사합니다

제목 ▼ 도서 검색하기 검색

장르별보기

미분류(default)
소설/시/희곡
에세이
어린이
유아
경제경영
인문학
경제경영
사회과학
수험서/자격증
대학교재
컴퓨터/모바일
전체보기

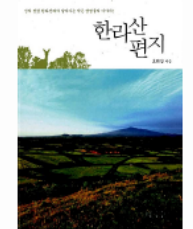
최상급 전체 보기

| 책 | 제목 | 가격 | 출판년도 | 출판사 |
|---|------------|---------|-------------|-------------|
|  | 하루부터 질문 수업 | 13500 원 | 2016년04월15일 | 경향BP |
|  | 뉴욕스타일 포토샵 | 38800 원 | 2012년12월26일 | 피어슨에듀케이션코리아 |

내 책 기증하기

방문자님 환영합니다.

↓ 이달의 베스트 대여도서 ↓



이달의 다독왕은
장하윤 님 입니다

고객 센터

- 장르별로 책 조회 가능

2~3등급 전체 보기

| 책 | 제목 | 가격 | 출판년도 | 출판사 | 대여여부 |
|---|-------------|---------|-------------|------|------|
|  | 바이러스에서 살아남기 | 19600 원 | 2015년11월25일 | 아이세움 | 대여가능 |

- 중고 서적은 상태별 분류 후
- 1등급은 판매
- 2~3등급은 대여

중고책 메인 페이지

My B Your B Home 로그아웃 마이페이지 장바구니 1:1문의 새책 구매하기 강호동님 로그인중입니다.

제목 ▾

도서 검색하기

검색

상위메뉴

최상급 전체 보기

| 책 | 제목 | 가격 | 출판년도 | 출판사 |
|---|--------------------|---------|-------------|-------|
|  | 하루부터 질문 수업 | 13500 원 | 2016년04월15일 | 경향BP |
|  | 2시간에 끝내는 한글영어 발음천사 | 11400 원 | 2016년07월04일 | 마이클리시 |

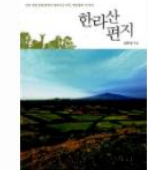
2~3등급 전체 보기

| 책 | 제목 | 가격 | 출판년도 | 출판사 | 대여여부 |
|---|-------------------|---------|-------------|-------|------|
|  | 멜로디 봉봉 : 크리스마스 캐럴 | 15500 원 | 2019년11월15일 | 스마트베어 | 대여가능 |
|  | 배당주 투자 바이블 | 25000 원 | 2014년10월31일 | 참돌 | 대여가능 |

내 책 기증하기
대여목록 보기

강호동님 환영합니다.
보유포인트는 2000point

↓ 이달의 베스트 대여도서 ↓



이달의 다독왕은
장하윤 님 입니다

고객 센터

- 매 달마다 가장 많이 팔린 베스트 셀러를 출력
- 회원은 중고도서 기증가능



중고책 구매 페이지

- 포인트는 10%를 넘게 사용할수 없음

My B Your B Home 로그아웃 마이페이지 장바구니 1:1문의 새책 구매하기 강호동님 로그인중입니다.

제목 도서 검색하기

검색

장르별보기

| 책 정보 | | 조회수 : 120 |
|--|-------------------|-----------|
| | 책 제목 : 기본간호학 | |
| | 저자 : 간호학 문제집 편집위원 | |
| | 등록일 : 2017년05월31일 | |
| | 출판사 : 대한간호협회 | |
| | 회원가 : 12500 원 | |
| 장바구니에 담기 바로 구매하기 | | |
| <div>기증자 한마디</div> <div>간호학 시험 수험서</div> | | |

3팀 프로젝트
02-123-4567

1:1문의
고객센터

- 책에 대한 정보 출력
- 조회수 상시 업데이트

- 주문번호 생성
yyyyMMDD형식으
로 구현함

| 책 정보 | |
|----------------------|---|
| 책제목 | 기본간호학 |
| 강호동 님의 사용가능한 포인 트 | 2000 점 |
| 포인트 사용하기 | <input type="text" value="200"/> 적용하기 |
| 가격 | 12500 원 |
| 주소 | 서울시 마포구 |
| 상세주소 | <input type="text" value="상세주소 입력"/> |
| 포인트 적용 후 | <input type="text" value="12300"/> 원 |
| 결제방법 | <input checked="" type="radio"/> 무통장입금 <input type="radio"/> 카드결제 |
| 결제하기 | |

| 주문이 완료되었습니다 | |
|----------------------|---------------|
| 주문번호 | 20210107-72 |
| 성함 | 강호동 |
| 회원 여부 | 정회원 |
| 주문일 | 2021년 01월 07일 |
| 결제 유형 | 무통장입금 |
| 금액 | 12300 원 |
| 주소지 | 서울시 마포구 |
| 메인으로 | |

비회원 주문조회페이지

주문이 완료되었습니다

| | |
|-------|---------------------------|
| 주문번호 | 20210107-11 |
| 성함 | 홍길동 |
| 회원 여부 | 비회원 |
| 주문일 | 2021년 01월 07일 |
| 결제 유형 | 무통장입금 |
| 금액 | 6750 원 |
| 주소지 | 경기 부천시 작동로50번길 10 (작동) |

메인으로

주문조회하기

주문번호

20210107-11

비밀번호

....

조회하기

주문조회 결과

주문번호

20210107-11

고객명

홍길동 님

책유형

중고책

주문일

2021년 01월 07일

결제유형

무통장 결제

금액

6750

돌아가기

메인페이지에 장르별로 똑같은 페이지인데 데이터만 다르게 받아오려고 하다보니 무한루프가 발생하여 새로운 Controller 로직을 만들고 그 부분에서 View로 돌려주었던 에러 해결법입니다.

J NewbooklistController.java ☒

```
6
7 @Controller
8 public class NewbooklistController {
9
10     @RequestMapping("newbooklist")
11     public String main(@RequestParam("book") String book) {
12
13         if(book.equals("a")) { //미분류
14             return "redirect:/list?book=a";
15
16         }else if(book.equals("b")){ //소설류
17             return "redirect:/list?book=b";
18
19         }else if(book.equals("c")){ // 에세이
20             return "redirect:/list?book=c";
21
22         }else if(book.equals("d")){ // 어린이
23             return "redirect:/list?book=d";
24
25         }else if(book.equals("e")){ // 유아
26             return "redirect:/list?book=e";
27
28         }else if(book.equals("f")){ //경제경영
29             return "redirect:/list?book=f";
30
31         }else if(book.equals("g")){ // 인문학
32             return "redirect:/list?book=g";
33
34         }else if(book.equals("h")){ // 외국어
35             return "redirect:/list?book=h";
36
37         }else if(book.equals("i")){ // 사회과학
38             return "redirect:/list?book=i";
39
40         }else if(book.equals("j")){ // 수험서
41             return "redirect:/list?book=j";
42
43         }else if(book.equals("k")){ // 대학교재
44             return "redirect:/list?book=k";
```

J ListController.java ☒

```
1 package pack.user.controller;
2
3 import java.util.List;
4
14
15 @Service
16 @Controller
17 public class ListController {
18     @Autowired
19     private NewBookInter newBookInter;
20
21     @RequestMapping("list")
22     public ModelAndView list(@RequestParam("book") String book) {
23         ModelAndView modelAndView = new ModelAndView();
24         if (book.equals("a")) { // 미분류
25             String nb_genre = "성공학";
26             List<NewBookDto> list = newBookInter.getGenre(nb_genre);
27             modelAndView.setViewName("list");
28             modelAndView.addObject("newbooklist", list);
29             return modelAndView;
30
31         } else if (book.equals("b")) { // 소설
32             String nb_genre = "소설";
33             List<NewBookDto> list = newBookInter.getGenre(nb_genre);
34             modelAndView.setViewName("list");
35             modelAndView.addObject("newbooklist", list);
36             return modelAndView;
37
38         } else if (book.equals("c")) { // 에세이
39             String nb_genre = "에세이";
40             List<NewBookDto> list = newBookInter.getGenre(nb_genre);
41             modelAndView.setViewName("list");
42             modelAndView.addObject("newbooklist", list);
43             //System.out.println(list);
44             return modelAndView;
45
46         } else if (book.equals("d")) { // 어린이
47             String nb_genre = "어린이";
48             List<NewBookDto> list = newBookInter.getGenre(nb_genre);
49             modelAndView.setViewName("list");
```

아이디가 같지 않을때
View에서 되돌려주는 것이 아니라
Controller로직에서 바로 계정이 일
치하지 않게 된다면 뒤로 보내는 형
식으로 코드를 구현하였습니다.

```
// 해당책의 리뷰 쓰기
@RequestMapping(value = "plusGonggam", method = RequestMethod.GET)
public String plusGonggam(@RequestParam("review_no") String review_no) {
    ReviewDto dto = reviewInter.selectNewbookReview(Integer.parseInt(review_no));
    boolean b = reviewInter.plusGonggam(Integer.parseInt(review_no));

    if(b) {
        return "redirect:/newbook?book_no="+dto.getReview_bookno();
    }else {
        return "error";
    }
}

// 해당책의 리뷰 지우기
@RequestMapping(value = "deleteReview", method = RequestMethod.GET)
public String deleteReview(HttpSession session ,@RequestParam("review_no") String review_no,
    HttpServletRequest request, HttpServletResponse response) throws Exception {
    ReviewDto dto = reviewInter.selectNewbookReview(Integer.parseInt(review_no));
    String id = (String)session.getAttribute("id");

    //아이디가 같을 때만 지울 수 있다.
    if(id.equals(dto.getReview_id())) {
        boolean b = reviewInter.deleteReview(Integer.parseInt(review_no));

        if(b) {
            return "redirect:/newbook?book_no="+dto.getReview_bookno();
        }else {
            return "error";
        }
    }else {
        response.setContentType("text/html; charset=UTF-8");

        PrintWriter out = response.getWriter();
        out.println("<script>alert('일치하지 않는 계정입니다'); history.back(); </script>");
        out.flush();
        return "";
    }
}
```

주문번호 생성기 구현 로직입니다.
Date와 SimpleDateFormat 그리고 math랜덤함수를 이용하여 구현하였습니다.

```
//public int count = 5;
@Override
public OrderInfoDto order(HttpSession session,
    String user_id, String order_person, int order_sum, String radioPaytype, String order_passwd, String address, String ob_no) {
    //sql 주문번호 +
    String sql = "insert into orderinfo(order_no,orderlist_no,order_person,order_id,order_bookno,order_booktype,order_date,"
        + "order_passwd,order_scount, order_paytype, order_state, order_sum, order_address) values(default,?,?,?,?,?,curdate(),?,?,?,?,?,?)";
    OrderInfoDto dto = new OrderInfoDto();
    try {
        String url = "jdbc:mysql://localhost:3306/test";
        String id = "root";
        String pw = "123";
        Class.forName("org.mariadb.jdbc.Driver");
        conn = DriverManager.getConnection(url, id, pw);

        Date now = new Date();
        SimpleDateFormat vans = new SimpleDateFormat("yyyyMMdd");
        String wdate = vans.format(now);
        OldBookDto oldBookDto = oldBookInter.view(ob_no);
        DecimalFormat df = new DecimalFormat("00");

        if(session.getAttribute("id") != null) { //회원
            pstmt = conn.prepareStatement(sql);

            //pstmt.setInt(1, count); //번호

            pstmt.setString(1, wdate + "-" + df.format(count)); // 주문번호 2
            pstmt.setString(2, order_person ); //주문자 3

            pstmt.setString(3, user_id); //주문자 4
            pstmt.setInt(4, oldBookDto.getOb_no()); //책번호 5
            pstmt.setString(5, "2" ); //1은 새책 2는 중고책 6
            pstmt.setString(6, " "); //비밀번호 7
            pstmt.setInt(7, 1); //수량 8
            pstmt.setString(8, radioPaytype); //결제수단 9
            pstmt.setString(9, "0"); //배송현황 10
            pstmt.setInt(10, order_sum); //금액 11
            pstmt.setString(11, address); //주소
            pstmt.executeUpdate();
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
    return dto;
}
```


Thank you!
감사합니다!

더 자세한 코드 내용은
<https://github.com/lsj8367/Project>
를 참고해주시면 감사하겠습니다.

