

[문제 1] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오

MVC패턴을 이용한 ArrayList를 사용하는 문제이다. 해당 구현 클래스 다이어그램과 클래스 구조를 참고하여 프로젝트를 완성하시오

1. 구현 클래스 다이어그램 (Class Diagram)

Run
+ <u>main(args:String[]) : void</u>

Book
- title:String - author:String - category : String - price : int
+ Book() + Book(title:String, author:String, category:String, price:int) + setter() / getter() + toString() : String + hashCode() : int + equals() : boolean + compareTo() : int

BookController
- list:List = new ArrayList() + BookController() + insertBook(bk:Book):void + selectList():ArrayList + searchBook(keyword:String):ArrayList + deleteBook(title:String, author:String):Book + ascBook() : int

BookMenu
- sc:Scanner = new Scanner(System.in) - bc : BookController = new BookController() + mainMenu() : void + insertBook() : void + selectList() : void + searchBook() : void + deleteBook() : void + ascBook() : void

2. 구현 클래스 설명

Package명	Class명	Method	설명
com.bs.practice. list.library.run	Run	<u>+main(args:String[]) : void</u>	BookMenu 객체를 생성 후 mainMenu() 실행
com.bs.practice. list.library.view	BookMenu	+ mainMenu() : void	사용자가 직접 메인 메뉴를 선택 할 수 있음. 종료 전까지 반복 실행. 각 메뉴를 누를 시 해당 메소드로 이동.
		+insertBook() : void	도서 추가를 위해 정보를 받는 메소드
		+selectList() : void	전체 도서 목록 출력 성공을 알리는 메소드
		+searchBook() : void	특정 도서 검색 결과를 보여주는 메소드
		+deleteBook() : void	특정 도서 삭제 성공을 알리는 메소드
		+ascBook() : void	책 명 오름차순 정렬 성공을 알리는 메소드
com.bs.practice. list.library.controller	BookController	+BookController()	초기 값을 넣어줄 기본 생성자
		+insertBook(bk:Book):void	리스트에 값 저장 메소드
		+selectList():ArrayList	모든 리스트를 출력하는 메소드
		+searchBook(keyword:String):ArrayList	키워드로 책을 검색하는 메소드
		+deleteBook(title:String, author:String):Book	책 제목과 저자 명으로 책을 삭제하는 메소드
		+ascBook():int	책 명 오름차순 정렬하는 메소드
com.bs.practice. list.library.model.vo	Book	+ Book()	기본 생성자
		+ Book(title:String, author:String, category:String, price:int)	매개변수가 있는 생성자
		+ toString() : String	객체의 정보를 리턴하는 메소드
		+ hashCode() : int	해시코트를 리턴하는 메소드
		+ equals():boolean	객체가 같은지 여부를 리턴하는 메소드
		+ compareTo(o:Object):int	정렬 기준을 가지고 있는 메소드

3. class 구조

```
public class BookMenu {  
    // BookController 객체 생성  
    // Scanner 객체  
    public void mainMenu() {  
        System.out.println("== Welcome BS Library ==");  
  
        ***** 메인 메뉴 *****  
  
        1. 새 도서 추가                // insertBook () 실행  
        2. 도서 전체 조회              // selectList ()  
        3. 도서 검색 조회              // searchBook ()  
        4. 도서 삭제                  // deleteBook ()  
        5. 도서 명 오름차순 정렬        // ascBook()  
        9. 종료                        // "프로그램을 종료합니다." 출력 후 main()으로 리턴  
  
        메뉴 번호 선택 : >> 입력 받음  
        // 메뉴 화면 반복 실행 처리  
        // 잘 못 입력 하였을 경우 "잘못 입력하였습니다. 다시 입력해주세요" 출력 후 반복  
    }  
  
    // 1. 새 도서 추가용 view 메소드  
    public void insertBook() {  
        1. 도서명 입력받기 (String title)  
        2. 저자명 입력받기 (String author)  
        3. 장르 입력받기 (int category) --> 숫자로 입력받기 (1.인문 / 2.자연과학 / 3.의료 / 4.기타)  
        4. 가격 입력받기 (int price)  
        5. 매개변수 생성자를 이용하여 Book 객체 생성  
           (객체 생성 시 장르 번호 별로 조건식 이용해 장르번호가 아닌 장르 명으로 값 넘겨야 함)  
        6. bc(BookController)의 insertBook으로 위의 Book 객체 전달  
    }  
}
```

// 2. 도서 전체용 view 메소드

```
public void selectList() {
```

1. bc(BookController)의 selectList() 메소드를 호출 후

--> 결과 값을 임의의 리스트(ArrayList<Book> bookList)에 대입

2. 조건식 이용해서

2_1. bookList가 "비어있는 경우" --> "존재하는 도서가 없습니다." 라는 알람 문구 출력

2_2. bookList가 "비어있지 않은 경우" --> 반복문을 통해 bookList 안의 Book 객체들 출력

```
}
```

// 3. 도서 검색용 view 메소드

```
public void searchBook() {
```

1. 검색할 도서명 키워드로 입력 받기 (String keyword)

2. bc(BookController)의 searchBook() 메소드로 위의 keyword 값 전달 후

--> 결과 값을 임의의 리스트(ArrayList<Book> searchList)에 대입

** 결과 값이 한 도서의 Book 객체가 아니라 리스트인 이유

보통 프로그램에서 검색 할 때 풀 네임을 쳐서 검색하지 않고 키워드로 검색

즉, 키워드를 포함한 결과가 하나가 아닌 여러 개일 수 있기 때문에 리스트로 받기

3. 조건식 이용해서

3_1. searchList가 "비어있는 경우" --> "검색 결과가 없습니다."라는 알람 문구 출력

3_2. searchList가 "비어있지 않은 경우" --> 반복문으로 searchList 안의 Book 객체들 출력

```
}
```

// 4. 도서 삭제용 view 메소드

```
public void deleteBook() {
```

1. 삭제할 도서명 입력 받기 (String title)

2. 삭제할 저자명 입력 받기 (String author)

** 도서명과 저자명을 같이 입력 받는 이유

--> 같은 도서명이라고 해도 저자명이 다른 도서가 있을 수 있기 때문에

3. bc(BookController)의 deleteBook() 메소드로 위의 title, author 값 전달 후

--> 결과 값을 임의의 Book(Book remove)에 대입

4. 조건식 이용해서

4_1. remove가 존재하는 경우 --> "성공적으로 삭제되었습니다."라는 문구 출력

4_2. remove가 존재하지 않은 경우 --> "삭제할 도서를 찾지 못했습니다."라는 문구 출력

```
}
```

```
}
```

// 5. 도서 명 오름차순 정렬용 view 메소드

```
public void ascBook() {
```

```
    bc(BookController)의 ascBook() 메소드 값에 따라
```

```
    성공 시 "정렬에 성공하였습니다.", 실패 시 "정렬에 실패하였습니다." 출력
```

```
}
```

```
}
```

```
public class BookController{
```

```
    // ArrayList 객체 생성
```

```
    public BookController(){
```

```
        // 초기 값 4개 추가
```

```
        bookList.add(new Book("자바의 정석", "남궁 성", "기타", 20000));
```

```
        bookList.add(new Book("쉽게 배우는 알고리즘", "문병로", "기타", 15000));
```

```
        bookList.add(new Book("대화의 기술", "강보람", "인문", 17500));
```

```
        bookList.add(new Book("암 정복기", "박신우", "의료", 21000));
```

```
    }
```

```
    public void insertBook(Book bk){
```

```
        //전달 받은 bk를 bookList에 추가
```

```
    }
```

```
    public ArrayList selectList(){
```

```
        // 해당 bookList의 주소 값 반환
```

```
    }
```

```
    public ArrayList searchBook(String keyword){
```

```
        // 검색 결과 리스트를 담아줄 리스트(ArrayList searchList) 선언 및 생성
```

```
        // 반복문을 통해 list의 책 중 책 명에 전달 받은 keyword가 포함되어있는 경우
```

```
        // searchList에 해당 책 추가하고 searchList 반환
```

```
    }
```

```
    public Book deleteBook(String title, String author){
```

```
        // 삭제된 도서를 담을 Book객체 (Book removeBook) 선언 및 null로 초기화
```

```
        // 반복문을 통해 bookList의 책 중 책 명이 전달 받은 title과 동일하고
```

```
        // 저자 명이 전달 받은 author와 동일한 경우 해당 인덱스 도서 삭제 후 빠져나감
```

```
        // 이 때 해당 인덱스 도서를 removeBook에 대입 후 removeBook 반환
```

```
    }
```

```

public int ascBook(){

    // 책 이름으로 오름차순 후 1 반환

}

}

```

```

===== Welcome BS Library =====

=====***** 메인 메뉴 *****=====
1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

메뉴 번호 입력 : 1
===== 새 도서 추가 =====
책 정보를 입력해주세요.
도서 명 : 갈증에 한 수위
저자 명 : 헛개수
장르(1. 인문 / 2. 과학 / 3. 외국어 / 4. 기타) : 1
가격 : 15000

=====***** 메인 메뉴 *****=====
1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

메뉴 번호 입력 : 2
===== 도서 전체 조회 =====
      (자바의 정석/남궁 성/기타/27000)
      (쉽게 배우는 알고리즘/문병로/기타/15000)
      (대화의 기술/강보람/인문/17500)
      (암 정복기/박신우/의료/21000)
      (갈증에 한 수위/헛개수/인문/15000)

=====***** 메인 메뉴 *****=====
1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

```

메뉴 번호 입력 : 3

===== 도서 검색 =====

검색 키워드 : 의

(자바의 정석/남궁 성/기타/27000)

(대화의 기술/강보람/인문/17500)

=====***** 메인 메뉴 *****=====

1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

메뉴 번호 입력 : 4

===== 도서 삭제 =====

삭제할 도서 명 : 갈증에 한 수위

삭제할 저자 명 : 헛개수

성공적으로 삭제되었습니다.

=====***** 메인 메뉴 *****=====

1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

메뉴 번호 입력 : 2

===== 도서 전체 조회 =====

(자바의 정석/남궁 성/기타/27000)

(쉽게 배우는 알고리즘/문병로/기타/15000)

(대화의 기술/강보람/인문/17500)

(암 정복기/박신우/의료/21000)

=====***** 메인 메뉴 *****=====

1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

메뉴 번호 입력 : 5

정렬에 성공하였습니다.

===== 메인 메뉴 =====

1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

메뉴 번호 입력 : 2

===== 도서 전체 조회 =====

(대화의 기술/강보람/인문/17500)
(쉽게 배우는 알고리즘/문병로/기타/15000)
(암 정복기/박신우/의료/21000)
(자바의 정석/남궁 성/기타/27000)

===== 메인 메뉴 =====

1. 새 도서 추가
2. 도서 전체 조회
3. 도서 검색 조회
4. 도서 삭제
5. 도서 명 오름차순 정렬
9. 종료

메뉴 번호 입력 : 9

프로그램 종료