

[문제 1] 다음과 같은 조건을 만족하는 프로그램을 작성 하시오

해당 구현 클래스 다이어그램과 클래스 구조를 참고하여 프로젝트를 완성하시오

프로젝트명 : chap12-collection-practice-question

1. 구현 클래스 다이어그램 (Class Diagram)

| BookDTO |
|--|
| - bNo : int // 도서 번호 - category : int // 도서분류코드 - title : String // 도서 제목 - author : String // 도서 저자 |
| + BookDTO() + BookDTO(category:int, title:String, author:String) + setter() / getter() + toString() : String |

| Application |
|-------------------------------------|
| |
| + <u>main(args:String[]) : void</u> |

| BookMenu |
|---|
| - sc : Scanner - bm : BookManager |
| + BookMenu() + menu() : void + inputBook() : BookDTO + inputBookTitle() : String |

| BookManager |
|--|
| |
| + addBook(book:BookDTO): void + deleteBook(index:int):void + searchBook(bTitle:String):int + printBook(index:int):void + displayAll():void + sortedBookList(select:int):ArrayList<BookDTO> + printBookList(br:ArrayList<BookDTO>):void |

* BookDTO클래스의 setter() / getter() 메소드는 직접 구현

* java.util.Comparator 인터페이스를 상속받은 정렬용 클래스들 작성함

→ AscCategory, DescCategory

3. 구현 클래스 설명

| Package명 | Class명 | Method | 설명 |
|--------------------------|-------------|-----------------------------|---|
| com.ohgiraffers.hw1.run | Application | +main(args:String[]) : void | BookMenu 클래스 객체 생성하고 menu() 메소드 실행 |
| com.ohgiraffers.hw1.view | BookMenu | +BookMenu() | 기본 생성자 |
| | | +menu() : void | 도서관리 프로그램에 해당하는 메인 메뉴 출력, 각 메뉴에 해당하는 BookManager 클래스의 메소드 실행 → 반복 출력되게 한다 |
| | | +inputBook() : BookDTO | BookDTO 객체의 필드 값을 키보드로 입력 받아 초기화 하고 객체 리턴 |
| | | +inputBookTitle() : String | 검색할 도서제목을 키보드로 입력 받아 리턴 |

* 위와 같이 추가, 삭제, 검색에 필요한 정보는 키보드로 입력 받도록 각각의 메소드 따로 구현

| Package명 | Class명 | Method | 설명 |
|--------------------------------|-------------|--|--|
| com.ohgiraffers.hw1.controller | BookManager | +BookManager() | 기본 생성자 -> ArrayList<BookDTO> 인스턴스 생성 |
| | | +addBook(book:BookDTO) : void | 전달받은 BookDTO객체를 생성자 시 선언한 ArrayList객체에 추가 |
| | | +deleteBook(index:int) : void | 전달받은 도서 번호로 ArrayList 안에있는 정보 삭제 |
| | | +searchBook(title:String) : int | 전달받은 도서 제목으로 도서를 검색하고 결과값 반환 |
| | | +printBook(index:int) : void | 전달받은 인덱스에 있는 도서정보 출력 |
| | | +displayAll() : void | 도서목록 전체 출력 |
| | | +sortedBookList(select:int) : ArrayList<BookDTO> | 전달받은 번호를 기준으로 해당 카테고리순으로 정렬해서 ArrayList<BookDTO>에 담아서 리턴 |
| | | +printBookList(br:ArrayList<BookDTO>) : void | 전달 받은 ArrayList<BookDTO>를 향상된for문을 이용하여 출력 |

| Package명 | Class명 | Method | 설명 |
|--------------------------------------|--------------|----------------------|--|
| com.ohgiraffers.hw1.model.dto | BookDTO | +BookDTO() | 기본 생성자 |
| | | +toString() : String | 필드 값 문자열 합친 후 리턴 category 분류 별로 출력 >> 1 : 인문 / 2 : 자연과학 3 : 의료 / 4 : 기타 |
| com.ohgiraffers.hw1.model.comparator | AscCategory | +compare() : int | 카테고리순 오름차순 정렬 처리 |
| | DescCategory | | 카테고리순 내림차순 정렬 처리 |

4. class 구조

```
public class BookMenu{
    public void menu() {
        *** 도서 관리 프로그램 ***

        1. 새 도서 추가          // addBook (inputBook()이 리턴한 객체) 실행
        2. 도서정보 정렬 후 출력 // sortedBookList(정렬종류번호) 실행
        => 입력받은 매개변수를 이용하여 정렬방식에 따라 출력
        정렬방식을 선택해주세요 (1. 오름차순, 2.내림차순)
        1,2 이외의 값을 입력했을 시에는 "번호를 잘못입력하셨습니다"를 출력
        => ArrayList<BookDTO> 리턴 받아 printBookList(ArrayList<BookDTO>) 실행
        3. 도서 삭제              // deleteBook (도서 번호) 실행
        => seachBook()메소드를 이용
        => 결과값 리턴 받아 0일 경우 "성공적으로 삭제"
        1일 경우 "삭제할 글이 존재하지 않음"
        4. 도서 검색출력          // searchBook (inputBookTitle()이 리턴한 도서 제목) 실행
        => index 리턴 받아 -1일 경우 "조회한 도서가 존재하지 않음"
        -1이 아닐 경우 printBook(index) 출력
        5. 전체 출력              // displayAll() 실행
        => 결과 리스트 리턴 받아 비어있을 경우 "출력할 도서가 없습니다."
        아닌 경우는 전체 출력
        6. 끝내기                // main()으로 리턴
        메뉴 번호 선택 : >> 입력 받음
        // 메뉴 화면 반복 실행 처리
        // 해당 메뉴 번호에 따라 BookManager 클래스 메소드 실행
    }

    public BookDTO inputBook(){
        // 도서 번호 : >> 임의 값 입
        // "도서 제목 : " >> 입력 받음
        // "도서 장르 (1:인문 / 2:자연과학 / 3:의료 / 4:기타) : " >> 입력 받음 (숫자로)
        // "도서 저자 : " >> 입력 받음
        // 매개변수 생성자를 이용하여 위의 초기값을 이용한 BookDTO객체 리턴
    }

    public String inputBookTitle() {
        // "도서 제목 : " >> 입력 받음 >> 리턴
    }
}
```

```
public class BookManager{
    private ArrayList<BookDTO> bookList;
    public BookManager(){
        // bookList 인스턴스 생성
    }
    public void addBook(BookDTO book) {
        //bookList에 도서 추가
    }
    public void deleteBook(int index){
        //전달받은 index로 삭제
    }
    public int searchBook(String title){
        //도서명이 일치하는 객체를 찾아 해당 인덱스를 리턴
        //도서명이 일치하는 객체가 리스트에 없으면 -1을 리턴
    }
    public void printBook(int index){
        // index에 해당하는 객체 출력
    }
    public void displayAll(){
        // 도서 전체 리스트를 출력
    }
    public ArrayList<BookDTO> sortedBookList(int select){
        // 전달받은 매개변수의 값에 따라 정렬(1. 오름차순,2. 내림차순)
        // for문을 이용하여 ArrayList<BookDTO> 에 추가
        // 해당 배열 주소 값 리턴
    }
    public void printBookList(ArrayList<BookDTO> br) {
        // 향상된 for문을 이용하여 전달받은 ArrayList<BookDTO> 전체 출력
    }
}
```