

[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>

MusicView
- sc:Scanner = new Scanner(System.in) - mc:MusicController = new MusicController();
+ mainMenu() : void + addList() : void + addAtZero() : void + printAll() : void + searchMusic() : void + removeMusic() : void + setMusic() : void + ascTitle() : void + descSinger() : void

AscTitle
+ compare(o1:Object, o2:Object):int

MusicController
- list : List = new ArrayList<Music>() + addList(music:Music):boolean + addAtZero(music:Music):int + printAll():List<Music> + searchMusic(title:String):Music + removeMusic(title:String):Music + setMusic(title:String, music:Music):Music + ascTitle():int + descSinger():int

Music
- title:String - singer:String
+ Music() + Music(title:String, singer:String) + setter() / getter() + toString() : String + hashCode() : int + equals(obj:Object) : boolean + compareTo(o:Object) : int

## 2. 구현 클래스 설명

Package명	Class명	Method	설명
com.kh.chap1_list.practice.run	Run	<u>+main(args:String[]) : void</u>	MusicView 객체를 생성 후 mainMenu() 실행
com.kh.chap1_list.practice.view	MusicView	+ mainMenu() : void	사용자가 직접 메인 메뉴를 선택할 수 있음. 종료 전까지 반복 실행. 각 메뉴를 누를 시 해당 메소드로 이동.
		+addList() : void	마지막 위치에 곡 추가 성공을 알리는 메소드
		+addAtZero() : void	첫 위치에 곡 추가 성공을 알리는 하는 메소드
		+printAll() : void	전체 곡 목록을 출력 성공을 알리는 메소드
		+searchMusic() : void	특정 곡 검색의 결과를 보여주는 메소드
		+removeMusic() : void	특정 곡 삭제 결과를 보여주는 메소드
		+setMusic() : void	특정 곡 정보 수정 결과를 보여주는 메소드
		+ascTitle() : void	곡 명 오름차순 정렬 성공을 알리는 메소드
		+descSinger() : void	가수 명 내림차순 정렬 성공을 알리는 메소드
com.kh.chap1_list.practice.model.compare	AscTitle	+compare(o1:Object, o2:Object):int	곡 명을 오름 차순 정렬의 기준을 제공하는 메소드
com.kh.chap1_list.practice.controller	MusicController	+addList(music:Music):boolean	리스트 마지막에 값 저장 메소드
		+addAtZero(music:Music):int	리스트 첫 번째에 값 저장 메소드
		+printAll():List<Music>	모든 리스트를 출력하는 메소드
		+searchMusic(title:String):Music	곡 명을 검색하는 메소드
		+removeMusic(title:String):Music	곡 명으로 찾아 삭제하는 메소드
		+setMusic(title:String, music:Music):Music	곡 명으로 찾아 내용을 수정하는 메소드
		+ascTitle():int	곡 명 오름차순 정렬하는 메소드

		+descSinger():int	가수 명을 내림차순 정렬하는 메소드
com.kh.chap1_list.practice. model.vo	Music	+ Music()	기본 생성자
		+ Music(title:String, singer:String)	매개변수가 있는 생성자
		+ toString() : String	객체의 정보를 리턴하는 메소드
		+ hashCode() : int	해시코드 오버라이딩한 메소드
		+ equals(o:Object) : boolean	equals 오버라이딩한 메소드
		+ compareTo(o:Object) : int	정렬 기준을 가지고 있는 메소드

### 3. class 구조

```
public class MusicView{
    // 스캐너 객체 생성
    // MusicController 객체 생성
    public void mainMenu {
        ***** 메인 메뉴 *****

        1. 마지막 위치에 곡 추가          // addList() 실행
        2. 첫 위치에 곡 추가              // addAtZero()
        3. 전체 곡 목록 출력              // printAll()
        4. 특정 곡 검색                  // searchMusic()
        5. 특정 곡 삭제                  // removeMusic()
        6. 특정 곡 정보 수정              // setMusic()
        7. 곡명 오름차순 정렬            // ascTitle()
        8. 가수명 내림차순 정렬          // descSinger()
        9. 종료                          // "종료" 출력 후 main()으로 리턴

        메뉴 번호 선택 : >> 입력 받음
        // 메뉴 화면 반복 실행 처리
    }

    public void addList(){
        ***** 마지막 위치에 곡 추가 *****

        // 곡 명과 가수 명을 사용자에게 입력 받는다.
        // MusicController에 addList()를 이용해서 입력한 정보를 넘기고
        // 추가 성공 시 "추가 성공", 추가 실패 시 "추가 실패" 콘솔창에 출력
    }

    public void addAtZero(){
        ***** 첫 위치에 곡 추가 *****

        // 곡 명과 가수 명을 사용자에게 입력 받는다.
        // MusicController에 addAtZero()를 이용해서 입력한 정보를 넘기고
        // 추가 성공 시 "추가 성공", 추가 실패 시 "추가 실패" 콘솔 창에 출력
    }
}
```

```

public void printAll(){
    ***** 전체 곡 목록 출력 *****

    // MusicController에 printAll()을 통해 List를 반환받아 출력
}

public void searchMusic(){
    ***** 특정 곡 검색 *****

    // 사용자에게 곡 이름을 받고 MusicController에 있는 searchMusic으로 값을 넘긴다.
    // searchMusic()의 반환 값에 따라 반환 값이 없으면 "검색한 곡이 없습니다."
    // 반환 값이 있으면 "검색한 곡은 000(곡 명, 가수 명) 입니다." 콘솔 창에 출력
}

public void removeMusic(){
    ***** 특정 곡 삭제 *****

    // 사용자에게 삭제할 곡의 이름을 직접 받고 MusicController에 removeMusic으로
    // 값을 넘긴다. removeMusic()의 반환 값에 따라 반환 값이 없으면 "삭제할 곡이 없습니다."
    // 반환 값이 있으면 "000(곡 명, 가수 명)을 삭제 했습니다" 콘솔 창에 출력
}

public void setMusic(){
    ***** 특정 곡 정보 수정 *****

    // 사용자에게 수정할 곡을 입력 하여 MusicController에 setMusic으로 검색 할 수 있게
    // 값을 넘기며 수정할 곡 명과 가수 명을 받아 setMusic으로 값을 넘긴다.
    // 검색 결과 값이 없으면 "수정할 곡이 없습니다.", 검색 결과 값이 있고
    // 수정 했으면 "000(곡 명, 가수 명)의 값이 변경 되었습니다." 콘솔 창에 출력
}

public void ascTitle(){
    ***** 곡 명 오름차순 정렬 *****

    // MusicController에 ascTitle()의 값에 따라 성공 시 "정렬 성공", 실패 시 "정렬 실패"
}

public void descSinger(){
    ***** 가수 명 내림차순 정렬 *****

    // musicController에 descSinger()의 값에 따라 성공 시 "정렬 성공", 실패 시 "정렬 실패"
}
}

```

```

public class AscTitle implements Comparator{
    public int compare(Object o1, Object o2) {
        // Title의 내림차순으로 정렬하는 기준을 마련한다.
    }
}

public class MusicController {
    // ArrayList 객체 생성
    public boolean addList(Music music) {
        // 마지막 리스트에 객체 저장 성공 시 true, 실패 시 false
    }
    public int addAtZero(Music music) {
        // 첫 번째 리스트에 객체 저장 성공 시 1 리턴, 실패 시 0 리턴
    }
    public List<Music> printAll() {
        // 곡을 저장하고 있는 List 객체 리턴
    }
    public Music searchMusic(String title) {
        // 곡 명으로 객체 검색, 객체가 있으면 객체 정보 리턴, 없으면 null 리턴
    }
    public Music removeMusic(String title) {
        // 곡 명으로 객체 검색, 객체가 있으면 객체 정보 삭제, 없으면 null 리턴
    }
    public Music setMusic(String title, Music music) {
        // 곡 명으로 객체 검색, 객체가 있으면 객체 정보 수정, 없으면 null 리턴
    }
    public int ascTitle() {
        // 리스트 곡 명 오름차순 정렬, 제목이 같으면 가수 명으로 오름차순 정렬, 1 리턴
    }
    public int descSinger() {
        // 리스트 가수 명 내림차순 정렬, 1 리턴
    }
}

```

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 1

\*\*\*\*\* 마지막 위치에 곡 추가 \*\*\*\*\*

곡 명 : 소우주

가수 명 : 방탄소년단

추가 성공

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 2

\*\*\*\*\* 첫 위치에 곡 추가 \*\*\*\*\*

곡 명 : 작은 것들을 위한 시

가수 명 : 방탄소년단

추가 성공

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 3

\*\*\*\*\* 전체 곡 목록 출력 \*\*\*\*\*

[방탄소년단 - 작은 것들을 위한 시, 방탄소년단 - 소우주]

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 1

\*\*\*\*\* 마지막 위치에 곡 추가 \*\*\*\*\*

곡 명 : 주저하는 연인들을 위해

가수 명 : 잔나비

추가 성공

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 3

\*\*\*\*\* 전체 곡 목록 출력 \*\*\*\*\*

[방탄소년단 - 작은 것들을 위한 시, 방탄소년단 - 소우주, 잔나비 - 주저하는 연인들을 위해]

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 4

\*\*\*\*\* 특정 곡 검색 \*\*\*\*\*

검색할 곡 명 : 소우주

방탄소년단 - 소우주



===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 5

\*\*\*\*\* 특정 곡 삭제 \*\*\*\*\*

삭제할 곡 명 : 소우주

방탄소년단 - 소우주를(를) 삭제했습니다.

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 6

\*\*\*\*\* 특정 곡 수정 \*\*\*\*\*

검색할 곡 명 : 작은 것들을 위한 시

수정할 곡 명 : 작은 것

수정할 가수 명 : 방탄소년단

방탄소년단 - 작은 것들을 위한 시가 값이 변경되었습니다.

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 3

\*\*\*\*\* 전체 곡 목록 출력 \*\*\*\*\*

[방탄소년단 - 작은 것, 잔나비 - 주저하는 연인들을 위해]

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 7

정렬 성공

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 3

\*\*\*\*\* 전체 곡 목록 출력 \*\*\*\*\*

[방탄소년단 - 작은 것, 잔나비 - 주저하는 연인들을 위해]

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 8

정렬 성공

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 3

\*\*\*\*\* 전체 곡 목록 출력 \*\*\*\*\*

[잔나비 - 주저하는 연인들을 위해, 방탄소년단 - 작은 것]

===== 메인 메뉴 =====

1. 마지막 위치에 곡 추가
2. 첫 위치에 곡 추가
3. 전체 곡 목록 출력
4. 특정 곡 검색
5. 특정 곡 삭제
6. 특정 곡 수정
7. 곡 명 오름차순 정렬
8. 가수 명 내림차순 정렬
9. 종료

메뉴 번호 입력 : 9

프로그램 종료