

## 실습2 음악 재생기

다수의 음악 정보를 다루는 앨범 기능을 추가로 작성한다. 앨범은 다수의 음악으로 구성되며 다양한 형태의 음악을 이용해서 앨범을 구성할 수 있다.

앨범을 다루는 클래스로 Album 을 작성하고 앨범의 이름을 다루기 위한 title 프로퍼티를 선언한다. 앨범에 저장하는 음악 목록을 다루기 위해서 배열이 필요한데, 음악을 추가하거나 삭제하려면 가변형 타입의 배열을 사용해야 한다.

앨범을 구성하기 위해서는 앨범에 음악을 추가한다. 앨범에 음악을 추가하는 기능은 addSong: 메소드로 작성한다. addSong: 메소드의 파라미터는 음악 객체이고 배열에 음악 객체를 추가한다.

앨범에는 앨범에 속한 음악을 찾는 기능을 제공한다. 음악을 찾는 기능은 searchSong: 메소드로 작성하고 찾으려는 음악 이름을 파라미터로 전달받도록 작성한다. 이 메소드는 앨범이 음악을 저장하는 용도로 사용하는 배열을 순회하면서 음악 제목과 음악 객체를 비교한다. 찾으려는 음악이 있으면 음악 객체를 반환하고, 없으면 nil을 반환한다.

음악 제목과 음악 객체를 비교하려면 음악 클래스인 Song 에는 비교 기능을 작성해야 한다. 이 메소드를 isEqualToSongTitle: 이름으로 작성하고 문자열 타입으로 음악 제목을 입력받도록 정의한다. searchSong: 메소드는 파라미터로 입력받은 음악 제목과 음악 객체를 비교하기 위해서 Song 클래스의 isEqualToSongTitle: 메소드를 사용한다.

앨범에 음악을 추가하는 기능이 있으면 삭제하는 기능도 제공해야 한다. 음악을 삭제하는 기능은 deleteSong: 메소드로 작성하고 파라미터로 삭제하려는 음악의 제목을 입력받도록 정의한다. 음악을 삭제하려면 먼저 삭제하려는 음악을 찾아야 하므로 searchSong: 메소드를 이용한다. 삭제하려는 음악 객체를 찾은 다음에 배열에서 음악 객체를 삭제한다.

앨범은 재생 기능을 제공한다. 앨범에서의 재생 기능은 play 메소드로 작성하되, 앨범에 속한 모든 음악을 재생해야 한다. 앨범의 제목을 먼저 콘솔에 출력하고, 앨범에 속한 모든 음악 객체를 순회하면서 음악 클래스의 재생 기능을 사용하도록 작성한다.

이렇게 앨범 클래스와 음악 클래스를 작성했으면, main.m 에서 음악 객체를 5개 이상 만들고 음악 정보를 입력한다. 그리고 앨범 객체를 2개 이상 만들고, 음악을 이용해서 앨범을 구성해보자. 앨범에서 음악을 찾는 기능을 사용해보고, 앨범 단위로 재생하도록 작성한다.

< 작성해야 할 것들 >

## **Album 클래스 메소드**

- addSong:
- searchSong:
- deleteSong:
- play

## **Album클래스 프로퍼티**

- title
- songs

## **Song 클래스 메소드**

- isEqualToSongTitle: