

베스트 앨범

genres

0	1	2	3	4
classic	pop	classic	classic	pop

1. 장르별로 2곡씩

1-1. 속한 노래가 많이 재생된 장르 : 장르별 내림차순

1-2. 장르 내에서 많이 재생된 노래 : 재생수별 내림차순

1-3. 장르 내에서 재생수 같은 경우 고유번호로 정렬

plays

0	1	2	3	4
500	600	150	800	2500

1. 각 곡의 장르 알아야 함

장르, 곡, 재생횟수

classic, {{0, 500}, {2, 150}, {3, 800}}
pop, {{1, 600}, {4, 2500}}
⇒ Map<String, ArrayList<int[]>>

2. 장르별 재생횟수

Map<String, Integer>

3. 장르별 재생횟수 내림차순 정렬
entry 이용해서 list 만든 뒤 정렬
⇒ {pop: 3100}, {class: 1450}

4. 장르의 곡들은 가려내기 재생횟수 내림차순 정렬

pop이면 {1, 600}, {4, 2500} → {4, 2500}, {1, 600}

5. 오버인덱스 ArrayList에 넣기

1. 장르별 곡과 장르별 재생횟수 값을 HashMap 생성

```
Map<String, ArrayList<int[]>> genreMap = new HashMap<>();
```

```
Map<String, Integer> playMap = new HashMap<>();
```

2. 정답은 값을 ArrayList 생성

```
List<Integer> answer = new ArrayList<>();
```

3. 장르별곡, 장르별 재생횟수 값을 리턴

```
for (int i=0; i < genres.length; i++) {
```

```
    String genre = genres[i];
```

```
    int play = plays[i];
```

3-1. ArrayList는 초기화를 해주어야 컴파일 에러 안남.

```
if (!genreMap.containsKey(genre)) {  
    genreMap.put(genre, new ArrayList<>());  
}
```

3-2. 값 넣기

```
genreMap.get(genre).add(new int[] {i, play});
```

ArrayList<int[]> ← int 배열로 초기화해서 값 넣어줘야 함.

```
playMap.put(genre, playMap.getOrDefault(genre, 0) + play);
```

```
}
```

4. playMap의 entry를 ArrayList로 만든 value를 기준으로 내림차순 정렬

```
List<Map.Entry<String, Integer>> playEntry = new ArrayList<>(playMap.entrySet());
```

```
playEntry.sort((e1, e2) -> (e2.getValue().compareTo(e1.getValue())));
```

⇒ {pop: 4100}, {classic: 1450} 이런 식으로 얻기

5. entry를 돌면서 곡들을 정렬해서 붙여야함

```
for (Map<String, Integer> categoryMap : playEntry) {  
    String category = categoryMap.getKey();
```

5-1. genreMap에서 장르별 곡들 리스트 뽑기

```
List<int[]> songs = ArrayList<>(genreMap.get(category));
```

5-2. 곡 별로 재생횟수 내림차순 정렬

```
songs.sort((s1, s2) -> (s2[1] - s1[1]));
```

5-3. songs 돌면서 2개씩 가져와서 answer에 기록번트 넣기

```
for (int i = 0; i < Math.min(2, songs.size()); i++) {  
    answer.add(songs.get(i)[0]);  
}  
}
```

6. ArrayList를 Array로 변환 후 반환

```
return answer.stream().mapToInt(i -> i).toArray();
```