

## 베스트앨범

Python3

## 문제 설명

스트리밍 사이트에서 장르 별로 가장 많이 재생된 노래를 두 개씩 모아 베스트 앨범을 출시하려 합니다. 노래는 고유 번호로 구분하며, 노래를 수록하는 기준은 다음과 같습니다.

1. 속한 노래가 많이 재생된 장르를 먼저 수록합니다.
2. 장르 내에서 많이 재생된 노래를 먼저 수록합니다.
3. 장르 내에서 재생 횟수가 같은 노래 중에서는 고유 번호가 낮은 노래를 먼저 수록합니다.

노래의 장르를 나타내는 문자열 배열 `genres`와 노래별 재생 횟수를 나타내는 정수 배열 `plays`가 주어질 때, 베스트 앨범에 들어갈 노래의 고유 번호를 순서대로 return 하도록 solution 함수를 완성하세요.

## 제한사항

- `genres[i]`는 고유번호가 `i`인 노래의 장르입니다.
- `plays[i]`는 고유번호가 `i`인 노래가 재생된 횟수입니다.
- `genres`와 `plays`의 길이는 같으며, 이는 1 이상 10,000 이하입니다.
- 장르 종류는 100개 미만입니다.
- 장르에 속한 곡이 하나라면, 하나의 곡만 선택합니다.
- 모든 장르는 재생된 횟수가 다릅니다.

## 입출력 예

genres	plays	return
["classic", "pop", "classic", "classic", "pop"]	[500, 600, 150, 800, 2500]	[4, 1, 3, 0]

## 입출력 예 설명

classic 장르는 1,450회 재생되었으며, classic 노래는 다음과 같습니다.

- 고유 번호 3: 800회 재생
- 고유 번호 0: 500회 재생
- 고유 번호 2: 150회 재생

pop 장르는 3,100회 재생되었으며, pop 노래는 다음과 같습니다.

- 고유 번호 4: 2,500회 재생
- 고유 번호 1: 600회 재생

따라서 pop 장르의 [4, 1]번 노래를 먼저, classic 장르의 [3, 0]번 노래를 그다음에 수록합니다.

※ 공지 - 2019년 2월 28일 테스트케이스가 추가되었습니다.

## solution.py

```
1 def solution(genres, plays):
2     answer = []
3     return answer
```

## 실행 결과

실행 결과가 여기에 표시됩니다.

초기화

코드 실행

제출 후 채점하기