**VS code 동작환경에서 실행시키는 것은 실패했다.**

**무슨 이유인지 oauth2 모듈을 못 불러온다.**

**알았다! spotipy.py라고 이름 붙여놓고 실행하면 못 가져온다. 이름이 똑같아서 그런가보다.**

**Jupyter 노트북에서는 잘 작동한다.**

**가장 먼저 깨우쳐야 할 것!!**

**- 1. Spotipy로 음악 검색, Json 파일 전처리, 데이터 내보내기**

**- 1-1. Spotipy로 검색하면 나오는 정보 파악**

**- 1-2. 다양한 검색조건과 방법들은 공식 Document 참조**

**- 2. Spotipy 연동해 사용자의 요청을 처리할 DB구조 생각하기**

**search() 함수**

**입력 가능한 parameter는 다음과 같다**

**q : 문자열이 들어간다. 다양한 검색어를 넣을 수 있다. 연도를 검색하려면 q=“year:연도”를 넣으면 된다.**

**그 외 album:앨범명, genre:장르명 그런 것도 넣을 수 있다**

**이것저것 하다보니 알게 됐는데, 그냥 일상적으로 검색하듯 아무거나 넣어도 작동한다.**

**예를 들어 q=”year:2021 BTS” 라던가, q=”Imagine Dragons” 넣어도 정보 잘 갖고 온다. 심지어 한글을 넣어도 된다.**

**type: 노래의 유형을 결정한다. comma 붙이면 여러 검색어를 넣을 수 있다. album, artist, playlist, track, show, episode, audiobook 이런 것들이 있다.**

**예: type='track, album'**

**검색 결과로 JSON 형식을 돌려주는데, 차례차례 보자**

**우선 검색 결과는 내가 기입한 'type=복수형'으로 돌아온다. type=”track”이라면 {'tracks': {'href':'주소', 'items': 어쩌고}}**

**type=”album”이라면 {'albums': {'href':'주소', 'items': 어쩌고}}**

**대략 이런 식이다. 따라서 검색 종류에 따라 JSON을 다르게 처리 해줘야 한다.**

**검색 결과에 여러가지가 보일 것이다.**

**album\_type, artists, id, name, available\_markets 등등...**

**JSON 명령어를 사용해 원하는 정보만 뽑고 웹페이지에 보여주려면 사실 JSON만 잘 접근해도 괜찮을 것 같다는 생각이 들었다.**

**그럼 우리 DB에는 뭘 저장해야 좋을까?**

**audio\_features() 함수**

**이 노래의 특징을 정량적으로 올려놓은 자료이다**

**'danceability' : 이거 틀어놓으면 얼마나 춤추기 좋은 지 나타내는 지표다. 1에 가까울 수록 춤추기 좋다.**

**'energy' : 대략 노래의 분위기다. 0에 가까우면 조용조용하고 1에 가까우면 템포가 빠르고 흥겹다**

**'key' : C장조, F 등등 화음? 비슷한 자료인 듯**

**'instrumentless' : 보컬이 얼마나 들어가느냐인 듯.**

**'loudness' : 노래의 시끄러운 정도라는데, 데시벨 값으로 나타낸다고 한다. -60부터 0DB 사이라는데, 상관관계는 아직 잘 모르겠다.**

**'speechiness' : 1에 가까우면 일상 대화하듯 읖조리는 정도고, 0.33 이하라면 노래에 가깝다고 한다. 0.66 이상이면 오디오북이나 텍스트 읽어주기 같은 느낌일 것이다.**

**valence : 1~0까지이며 지수가 높을 수록 밝고 활기차지만 낮을 수록 우울하고 처지는 느낌**

**track\_id를 입력하고 audio\_feature('track\_id')를 넣으면 위 자료를 반환해준다.**

**사이트 구성 단계:**

**1. Instagram 페이지와 비슷한 심플한 UI 설계.**

**회원가입 단계에서 좋아하는 장르나 키워드를 물어보고 이를 노래 검색에 활용한다.**

**2. audio\_feature()활용하여 '선호음악'에 앨범 띄우기**

**이 함수를 어떻게 사용할 것인가?**

**recommendations()**