

# Python을 활용한 데이터 수집 II

## 1. 목표

- Python 기초 문법 실습
- 데이터 구조에 대한 분석과 이해
- 요청과 응답에 대한 이해
- API의 활용과 문서 분석

## 2. 준비사항

### A. 사용 데이터

- i. TMDb API(<https://developers.themoviedb.org/3>)
- ii. 영화 정보 및 API 서비스

### B. 개발언어/프로그램

- i. Python 3.8 이상

### C. 필수 라이브러리

- i. requests

## 3. 요구사항

커뮤니티 서비스 개발을 위한 데이터 수집 단계로, 전체 데이터 중 필요한 영화 데이터를 수집하는 과정입니다. 아래 기술된 사항은 필수적으로 구현해야 하는 내용입니다. 완성된 기능들은, 향후 커뮤니티 서비스에서 활용할 수 있습니다.

### A. **problem\_a.py**: 영화 개수 카운트 기능 구현

popular를 기준으로 가져온 영화 목록의 개수를 출력합니다.

#### i. 요청

1. 제공되는 tmdb.py를 이용하여 영화 리스트 조회 URL을 생성합니다.

2. requests 패키지를 이용하여 URL에 요청을 보냅니다.

## ii. 결과

1. popular를 기준으로 받아온 데이터에서 영화 리스트의 개수를 계산합니다.
2. 계산한 정보를 반환하는 함수 popular\_count를 완성합니다.

## B. problem\_b.py: 특정 조건에 맞는 영화 출력

popular를 기준으로 가져온 영화 목록 중 평점이 8 이상인 영화들의 목록을 출력합니다.

### i. 요청

1. 제공되는 tmdb.py를 이용하여 영화 리스트 조회 URL을 생성합니다.
2. requests 패키지를 이용하여 URL에 요청을 보냅니다.

### ii. 결과

1. 받아온 데이터에서 vote\_average를 기준으로 점수가 8 이상인 영화들의 목록을 리스트로 반환하는 함수 vote\_average\_movies를 완성합니다.

## C. problem\_c.py: 평점 순 정렬

영화목록을 평점순으로 출력하는 함수를 완성합니다. 해당 기능은 향후 커뮤니티 서비스에서 기본으로 제공되는 영화 정보로 사용됩니다.

### i. 요청

1. 제공되는 tmdb.py를 이용하여 영화 리스트 조회 URL을 생성합니다.
2. requests 패키지를 이용하여 URL에 요청을 보냅니다.

### ii. 결과

1. 받아온 데이터 중 평점이 높은 영화 다섯개의 정보를 리스트로 반환하는 함수 ranking을 완성합니다.

#### D. `problem_d.py`: 제목 검색, 영화 추천

제공된 영화 제목을 기준으로 추천영화 목록을 출력합니다.

##### i. 요청

1. 제공되는 `tmdb.py`를 이용하여 영화 제목을 기준으로 TMDB에서 사용하는 id를 검색합니다.
2. id를 기준으로 추천영화 목록 조회 URL을 생성합니다.
3. `requests` 패키지를 이용하여 URL에 요청을 보냅니다.

##### ii. 결과

1. TMDB에서 추천받은 영화 리스트에서 제목을 리스트에 저장합니다.
2. 저장된 리스트를 반환하는 함수 `recommendation`을 완성합니다.
3. 올바르지 않은 영화 제목으로 id가 없는 경우 `None`을 반환합니다.
4. id값은 있지만 추천영화가 없는 경우 빈 리스트를 반환합니다.

#### E. `problem_e.py`: 배우, 제작진 리스트 출력

영화에 출연한 배우들과 제작진의 정보가 저장된 딕셔너리를 출력합니다.

##### i. 요청

1. 제공되는 `tmdb.py`를 이용하여 영화 제목을 기준으로 TMDB에서 사용하는 id를 검색합니다.
2. id를 기준으로 배우와 제작진 목록 조회 URL을 생성합니다.
3. `requests` 패키지를 이용하여 URL에 요청을 보냅니다.

##### ii. 결과

1. `cast_id` 값이 10보다 작은 배우의 이름을 리스트에 저장합니다.
2. `department` 값이 `Directing`인 감독의 이름을 리스트에 저장합니다.
3. 반환되는 딕셔너리는 `cast`, `crew` 두개의 key를 가지고 각각 배우 리스트와 제작진 리스트를 value로 갖습니다.

4. 완성된 딕셔너리를 반환하는 함수 `credits`을 완성합니다.

#### F. 추가정보 수집

영화 데이터를 제공하는 다른 API를 사용하여 요청을 보내고 추가적인 정보를 수집하는 함수를 완성합니다.

##### i. 데이터

1. KMDB(<https://www.kmdb.or.kr/info/api/apiDetail/6>)
2. 영진위(<https://www.kobis.or.kr/kobisopenapi/homepg/main/main.do>)
3. 네이버 영화검색 API(<https://developers.naver.com/docs/search/movie/>)

#### 4. 제출

제출 마감시간은 18:00이며, 각 반 교수님을 `lab.ssafy.com` 프로젝트에 Maintainer로 등록해야 합니다.

반드시 활용하였던 데이터 정보 정리 및 저장한 파일에 대한 설명과 학습 내용을 README.md에 기록하여 제출합니다.

위에 명시된 사항은 최소 조건이며 추가적인 정보를 수정할 수 있습니다.

Gitlab에 제출하는 파일/폴더의 구조는 아래와 같습니다.

`pjt02/`

- `tmdb.py`
- `README.md`
- `problem_a.py`
- `problem_b.py`
- `problem_c.py`
- `problem_d.py`
- `problem_e.py`