

# 01 PJT

# Python을 활용한 데이터 수집 1

# INDEX

---

- Python을 활용한 데이터 수집 1
  - 목표
  - 준비사항
  - 요구사항
  - 제출

# 목표

## 프로젝트 목표

- Python 기본 문법 습득
- 파일 입출력에 대한 이해
- 데이터 구조에 대한 분석과 이해
- 데이터를 가공하고 JSON 형태로 구성하기

# 준비사항

## 개발도구 및 라이브러리

- 개발도구
  - Visual Studio Code
  - Python 3.9+
- 필수 라이브러리
  - [json](#)

# 요구사항

## 공통 요구사항

- 커뮤니티 서비스 개발을 위한 데이터 구성 단계로, 필요한 영화 데이터를 직접 추출하고 구성하는 과정입니다.
- 반드시 제공된 examples/ 폴더의 예시 파일을 먼저 참고합니다.
  - 예시 파일에는 이번 프로젝트 해결을 위해 알아야 하는 혹은 직접적인 도움이 될 수 있는 코드가 작성되어 있습니다.

## A. 제공되는 영화 데이터의 주요내용 수집 (problem\_a)

- 샘플 영화 데이터(movie.json)가 주어집니다.

이중 서비스 구성에 필요한 정보만 추출해 반환하는 함수를 단계적으로 완성합니다.

완성된 함수는 다음 문제의 기본 기능으로 사용됩니다.

### 1. 데이터

- 제공되는 movie.json을 활용합니다.
- movie.json은 영화 '쇼생크 탈출' 정보를 담고 있습니다.

## A. 제공되는 영화 데이터의 주요내용 수집 (problem\_a)

### 2. 풀이

- 필요한 정보
  - id, title, poster\_path, vote\_average, overview, genre\_ids
  - movie.json에서 필요한 정보에 해당하는 값을 추출합니다.
  - 필요한 정보를 새로운 dictionary로 반환하는 함수 movie\_info를 완성합니다.

## A. 제공되는 영화 데이터의 주요내용 수집 (problem\_a)

### 3. 결과

- problem\_a.py 실행 예시

```
{'genre_ids': [18, 80],  
 'id': 278,  
 'overview': '촉망받는 은행 간부 앤디 뉴프레인은 아내와 그녀의 정부를 살해했다는 누명을 쓴다. 주변의 증언과 살해 현장의  
 '그럴듯한 증거들로 그는 종신형을 선고받고 악질범들만 수용한다는 지옥같은 교도소 쇼생크로 향한다. 인간 말종  
 '쓰레기들만 모인 그곳에서 그는 이루 말할 수 없는 억압과 짐승보다 못한 취급을 당한다. 그러던 어느 날, 간수의  
 '세금을 면제받게 해 준 덕분에 그는 일약 교도소의 비공식 회계사로 일하게 된다. 그 와중에 교도소 소장은 죄수들을  
 '이리저리 부리면서 검은 돈을 긁어 모으고 앤디는 이 돈을 세탁하여 불려주면서 그의 돈을 관리하는데...',  
 'poster_path': '/3h06DIGRBaJQj2NLEYBMwpcz88D.jpg',  
 'title': '쇼생크 탈출',  
 'vote_average': 8.7}
```

주의) pprint 함수로 인해 dictionary의 key 순서가 정렬되어서 출력됩니다.

## B. 제공되는 영화 데이터의 주요내용 수정 (problem\_b)

- 이전 단계에서 만들었던 데이터 중 genre\_ids를 장르 번호가 아닌 장르 이름 리스트 genre\_names로 바꿔 반환하는 함수를 완성합니다.  
완성된 함수는 다음 문제의 기본 기능으로 사용됩니다.

### 1. 데이터

- 제공되는 movie.json, genres.json을 활용합니다.
- genres.json은 모든 장르의 id, name 정보를 담고 있습니다.

## B. 제공되는 영화 데이터의 주요내용 수정 (problem\_b)

### 2. 풀이

- 필요한 정보
  - id, title, poster\_path, vote\_average, overview, genre\_names
  - movie.json에서 필요한 정보에 해당하는 값을 추출합니다.
    - 이때, genre\_names대신 genre\_ids를 추출합니다.
  - genres.json을 이용하여 genre\_ids를 각 장르 번호에 맞는 name 값으로 대체한 genre\_names 키를 생성합니다.
  - 위 요구사항을 반영한 새로운 dictionary를 반환하는 함수 movie\_info를 완성합니다.

## B. 제공되는 영화 데이터의 주요내용 수정 (problem\_b)

### 3. 결과

- problem\_b.py 실행 예시

```
{'genre_names': ['Drama', 'Crime'],
'id': 278,
'overview': '촉망받는 은행 간부 앤디 뉴프레인은 아내와 그녀의 정부를 살해했다는 누명을 쓴다. 주변의 증언과 살해 현장의
'그럴듯한 증거들로 그는 종신형을 선고받고 악질범들만 수용한다는 지옥같은 교도소 쇼생크로 향한다. 인간 말종
'쓰레기들만 모인 그곳에서 그는 이루 말할 수 없는 억압과 짐승보다 못한 취급을 당한다. 그러던 어느 날, 간수의
'세금을 면제받게 해 준 덕분에 그는 일약 교도소의 비공식 회계사로 일하게 된다. 그 와중에 교도소 소장은 죄수들을
'이리저리 부리면서 검은 돈을 긁어 모으고 앤디는 이 돈을 세탁하여 불려주면서 그의 돈을 관리하는데...', 
'poster_path': '/3h06DIGRBaJQj2NLEYBMwpcz88D.jpg',
'title': '쇼생크 탈출',
'veote_average': 8.7}
```

주의) pprint 함수로 인해 dictionary의 key 순서가 정렬되어서 출력됩니다.

## C. 다중 데이터 분석 및 수정 (problem\_c)

- movies.json에는 평점이 높은 20개의 영화 데이터가 주어집니다.  
이 중 서비스 구성에 필요한 정보만 추출해 반환하는 함수를 완성합니다.  
완성된 함수는 향후 커뮤니티 서비스에서 제공되는 영화 목록을 제공하기 위한 기능  
으로 사용됩니다.

### 1. 데이터

- 제공되는 movies.json, genres.json을 활용합니다.
- movies.json은 전체 영화 정보를 담고 있습니다.

## C. 다중 데이터 분석 및 수정 (problem\_c)

### 2. 풀이

- 필요한 정보
  - id, title, poster\_path, vote\_average, overview, genre\_names
- 이전 단계의 함수 구조를 재사용합니다.
- 위 요구사항을 반영한 새로운 list를 반환하는 함수 movie\_info를 완성합니다.

## C. 다중 데이터 분석 및 수정 (problem\_c)

### 3. 결과

- problem\_c.py 실행 예시

```
[{'genre_names': ['Drama', 'Crime'],
  'id': 278,
  'overview': '촉망받는 은행 간부 앤디 듀프레인은 아내와 그녀의 정부를 살해했다는 누명을 쓴다. 주변의 증언과 살해 현장의 그럴듯한 증거들로 그는 종신형을 선고받고 악질범들만 수용한다는 지옥같은 교도소 쇼생크로 향한다. 인간 말종 쓰레기들만 모인 그곳에서 그는 이루 말할 수 없는 억압과 짐승보다 못한 취급을 당한다. 그러던 어느 날, 간수의 세금을 면제받게 해 준 덕분에 그는 일약 교도소의 비공식 회계사로 일하게 된다. 그 와중에 교도소 소장은 죄수들을 이리저리 부리면서 검은 돈을 긁어 모으고 앤디는 이 돈을 세탁하여 불려주면서 그의 돈을 관리하는데...',  
  'poster_path': '/3h06DIGRBaJQj2NLEYBMwpcz88D.jpg',
  'title': '쇼생크 탈출',
  'vote_average': 8.7},
 {'genre_names': ['Drama', 'Crime'],
  'id': 238,
  'overview': '시실리에서 이민온 뒤, 정치권까지 영향력을 미치는 거물로 자리잡은 돈 꿀레오네는 갖가지 고민을 호소하는 사람들의 문제를 해결해주며 대부라 불리운다. 한편 솔로소라는 인물은 꿀레오네가와 라이벌인 탓타리아 패밀리와 손잡고 새로운 마약 사업을 제안한다. 돈 꿀레오네가 마약 사업에 참여하지 않기로 하자, 돈 꿀레오네를 저격해 그는 중상을 입고 사경을 헤매게 된다. 그 뒤, 돈 꿀레오네의 아들 소니는 조직력을 총 동원해 다른 패밀리들과 피를 부르는 전쟁을 시작하는데... 가족의 사업과 상관없이 대학에 진학한 뒤 인텔리로 지내왔던 막내 아들 마이클은 아버지가 총격을 당한 뒤, 아버지를 구하기 위해 위험천만한 협상 자리에 나선다.',  
  'poster_path': '/c0wVs8eYA4G9ZQs7hIRSoiZr46Q.jpg',
  'title': '대부',
  # 이하 생략
}]
```

## D. 알고리즘을 사용한 데이터 출력 (problem\_d)

- 영화 세부 정보 중 수입 정보(revenue)를 이용하여 모든 영화 중 가장 높은 수익을 낸 영화를 출력하는 알고리즘을 작성합니다. 해당 과정은 향후 커뮤니티 서비스에서 수익순으로 영화를 정렬하여 출력하는 정보로 사용됩니다.

### 1. 데이터

- 제공되는 movies.json과 movies 폴더 내부의 파일들을 활용합니다.
- movies 폴더 내부의 파일들은 각 영화의 세부 정보를 가지고 있습니다.

## D. 알고리즘을 사용한 데이터 출력 (problem\_d)

### 2. 풀이

- 반복문을 통해 movies 폴더 내부의 파일들을 오픈해야 합니다.
  - 03\_example 또는 파일 하단의 코드를 참고합니다.
- 제공된 영화 데이터에서 수익이 같은 영화는 없습니다.
- 수익이 가장 높은 영화의 제목을 출력하는 함수 max\_revenue를 완성합니다.
  - 단, original\_title이 아닌 title을 사용합니다.

## D. 알고리즘을 사용한 데이터 출력 (problem\_d)

### 3. 결과

- problem\_d.py 실행 예시

반지의 제왕: 왕의 귀환

## E. 알고리즘을 사용한 데이터 출력 (problem\_e)

- 영화 세부 정보 중 개봉일 정보(release\_date)를 이용하여 모든 영화 중 12월에 개봉한 영화들의 제목 리스트를 출력하는 알고리즘을 작성합니다. 해당 과정은 향후 커뮤니티 서비스에서 추천 기능의 일부로 사용됩니다.

### 1. 데이터

- 제공되는 movies.json과 movies 폴더 내부의 파일들을 활용합니다.
- movies 폴더 내부의 파일들은 각 영화의 세부 정보를 가지고 있습니다.

## E. 알고리즘을 사용한 데이터 출력 (problem\_e)

### 2. 풀이

- 반복문을 통해 movies 폴더 내부의 파일들을 오픈해야 합니다.
  - 03\_example 또는 파일 하단의 코드를 참고합니다.
- 개봉일이 12월인 영화들의 제목을 리스트로 출력하는 함수  
dec\_movies를 완성합니다.
  - 단, original\_title이 아닌 title을 사용합니다.

## E. 알고리즘을 사용한 데이터 출력 (problem\_e)

### 3. 결과

- problem\_e.py 실행 예시

```
['그린 마일', '인생은 아름다워', '반자의 제왕: 왕의 귀환', '스파이더맨: 뉴 유니버스']
```

주의) 실행 예시와 영화 정렬 순서가 달라도 무관합니다.

## F. 선택 과제

- 제공된 영화 데이터를 사용하여 내가 원하는 데이터를 추출하고 나만의 데이터 구조를 만들어봅니다.
- 예시
  - 90년대 개봉작 중 많은 수입을 올린 영화 순위
  - 배급한 영화가 많은 순으로 배급사 정렬하기

# 제출

## 제출 시 주의사항

- 제출기한은 금일 18시까지입니다. 제출기한을 지켜 주시기 바랍니다.
- 반드시 README.md 파일에 단계별로 구현 과정 중 학습한 내용, 어려웠던 부분, 새로 배운 것들 및 느낀 점을 등을 상세히 기록하여 제출합니다.
  - 단순히 완성된 코드만을 나열하지 않습니다.
- 위에 명시된 요구사항은 최소 조건이며, 추가 개발을 자유롭게 진행할 수 있습니다.
- <https://lab.ssafy.com/>에 프로젝트를 생성하고 제출합니다.
  - 프로젝트 이름은 '프로젝트 번호 + pjt'로 지정합니다. (ex. **01\_pjt**)
- 반드시 각 반 담당 교수님을 Maintainer로 설정해야 합니다.