

PJT 02

Python을 활용한 데이터 수집 2

INDEX

- 사전 학습
- Aladin API
- Spotify API
- 제출

개요

프로젝트 개요

- 커뮤니티 서비스 개발을 위한 데이터 구성 단계로,
필요한 데이터를 직접 추출하고 재구성하는 과정
- API를 활용한 요청과 이에 따른 응답 데이터를 활용

프로젝트 목표

- API에 대한 이해
- 서버와의 요청과 응답에 대한 이해
- 데이터 타입 List, Dictionary 조작하기
- JSON 데이터를 재구성하기

사전 학습

사전학습 목표

- API 이해하기
- API 가이드 문서 숙지 및 API 키 발급
- requests 라이브러리를 통한 데이터 요청
- 알라딘 API 요청 및 JSON 응답 데이터 다루기

API

API

(Application Programming Interface)

두 소프트웨어가 서로 통신할 수 있게 하는 메커니즘

Application

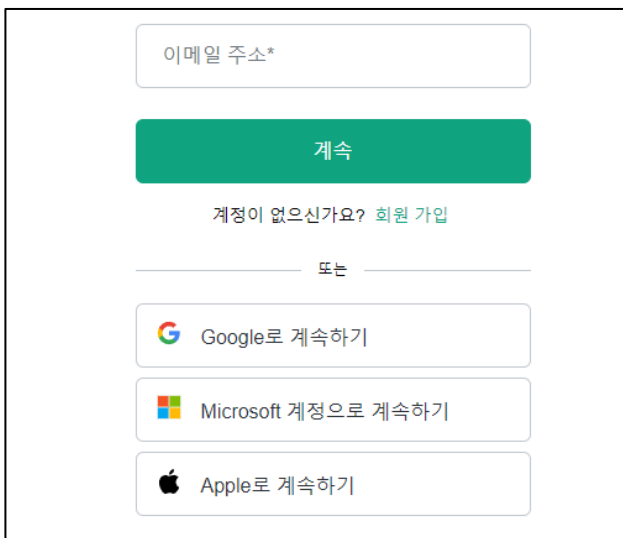
고유한 기능을 가진 모든 소프트웨어

Interface

사용자가 기기를 쉽게 동작 시키는 데 도움을 주는 시스템

- 서로 다른 두 개의 시스템 사이에서
정보나 신호를 주고받는 경우의 접점

API 예시 1 - 소셜 로그인 (1/4)





이메일 주소*


계속

계정이 없으신가요? [회원 가입](#)

또는

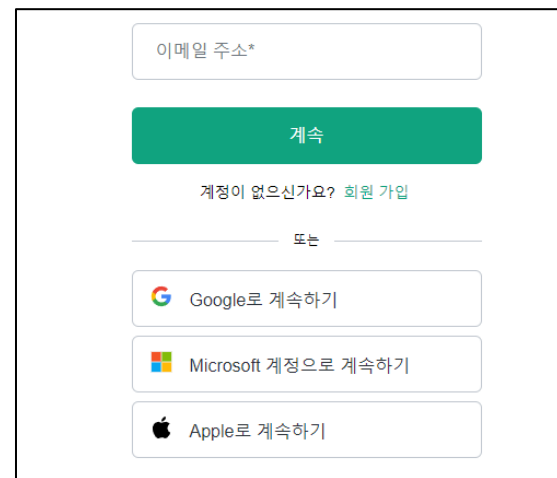
 Google로 계속하기

 Microsoft 계정으로 계속하기

 Apple로 계속하기

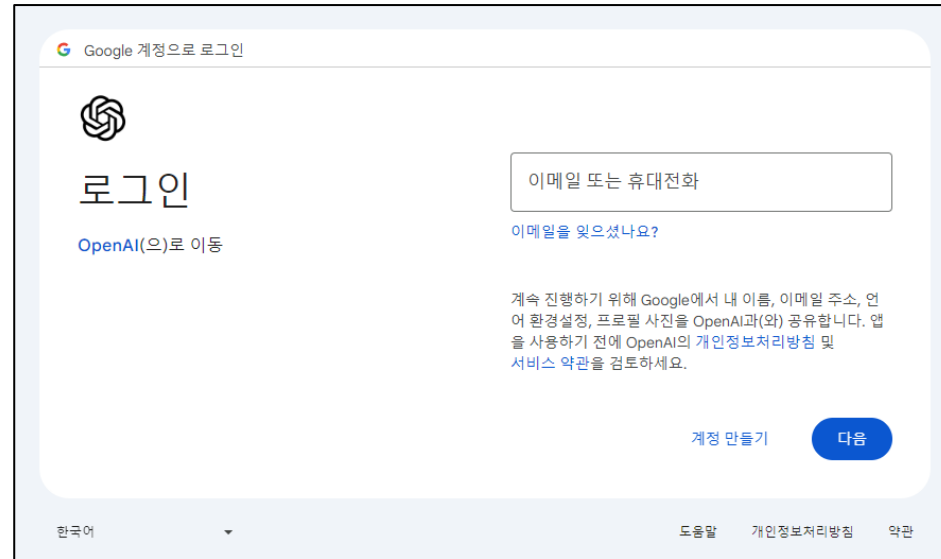
- 최근에는 특정 사이트에 직접 회원가입을 하지 않고 다른 소셜 미디어 계정으로 회원가입 및 로그인 하는 경우가 많음
- 어떻게 가능할까?

API 예시 1 - 소셜 로그인 (2/4)

A screenshot of the ChatGPT login interface. It features a text input field for '이메일 주소*' (Email address*), a green '계속' (Continue) button, and a link for '계정이 없으신가요? 회원 가입' (Don't have an account? Sign up). Below these are three social login options: 'Google로 계속하기' (Continue with Google), 'Microsoft 계정으로 계속하기' (Continue with Microsoft account), and 'Apple로 계속하기' (Continue with Apple).

- ChatGPT에서 Google로 회원가입 및 로그인을 진행하는 과정 알아보기
- 사용자는 ChatGPT 로그인 화면에서 Google로 계속하기를 클릭

API 예시 1 - 소셜 로그인 (3/4)



- ChatGPT가 제공한 Google 로그인 화면에서 Google 계정으로 로그인 진행

API 예시 1 - 소셜 로그인 (4/4)



ChatGPT

1. 어떤 사람이 Google로 내 사이트에 회원가입 하려고 하는데 어떤 사용자인지 정보를 줘



2. 알겠어. 이 사용자는 내 사이트에 올바르게 로그인 했고 확인해보니 ~~ 한 정보를 가진 사용자야



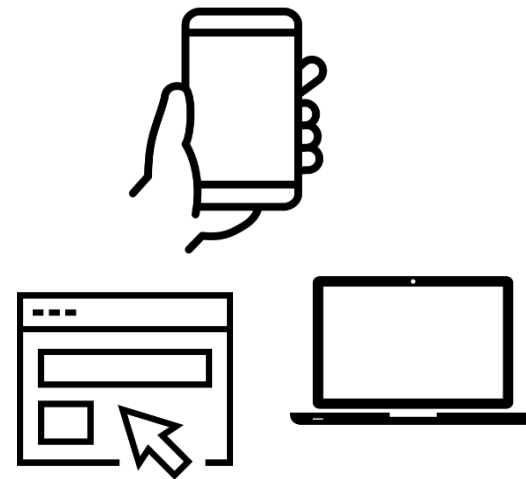
Google API

- Google 로그인 계정으로 로그인을 성공했을 경우 Google API는 ChatGPT에게 로그인에 성공한 인증된 사용자 정보를 넘겨줌
- 사용자 정보를 넘겨받은 ChatGPT는 해당 정보를 활용해 회원가입 및 로그인을 진행

API 예시 2 - 날씨 데이터 받기 (1/3)



기상청 시스템



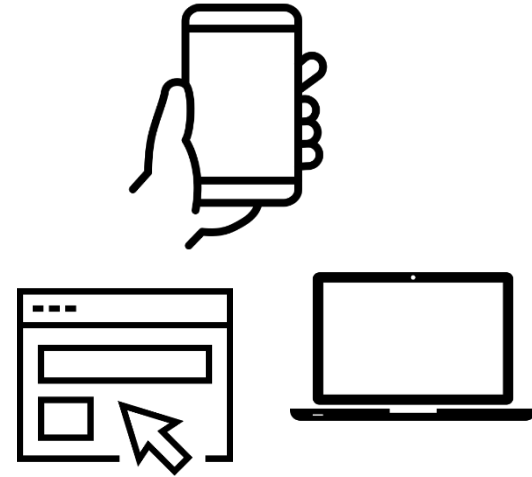
다양한 서비스

- 기상 데이터가 들어있는 기상청의 시스템
- 스마트 폰의 날씨 앱, 웹 사이트의 날씨 정보 등 다양한 서비스들이 이 기상청 시스템으로부터 데이터를 요청해서 받아감

API 예시 2 - 날씨 데이터 받기 (2/3)



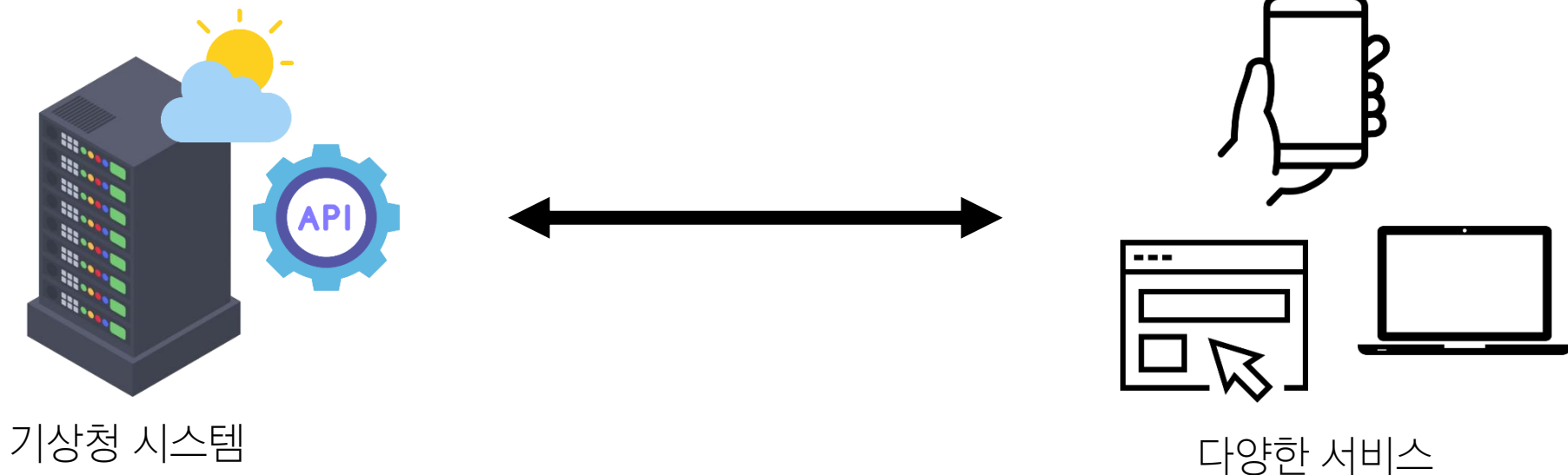
기상청 시스템



다양한 서비스

- 날씨 데이터를 얻으려면?
 - 기상청 시스템에는 정보들을 요청하는 지정된 형식이 있음
 - 지역, 날짜, 조회할 내용들(온도, 바람 등)을 제공하는 매뉴얼

API 예시 2 - 날씨 데이터 받기 (3/3)

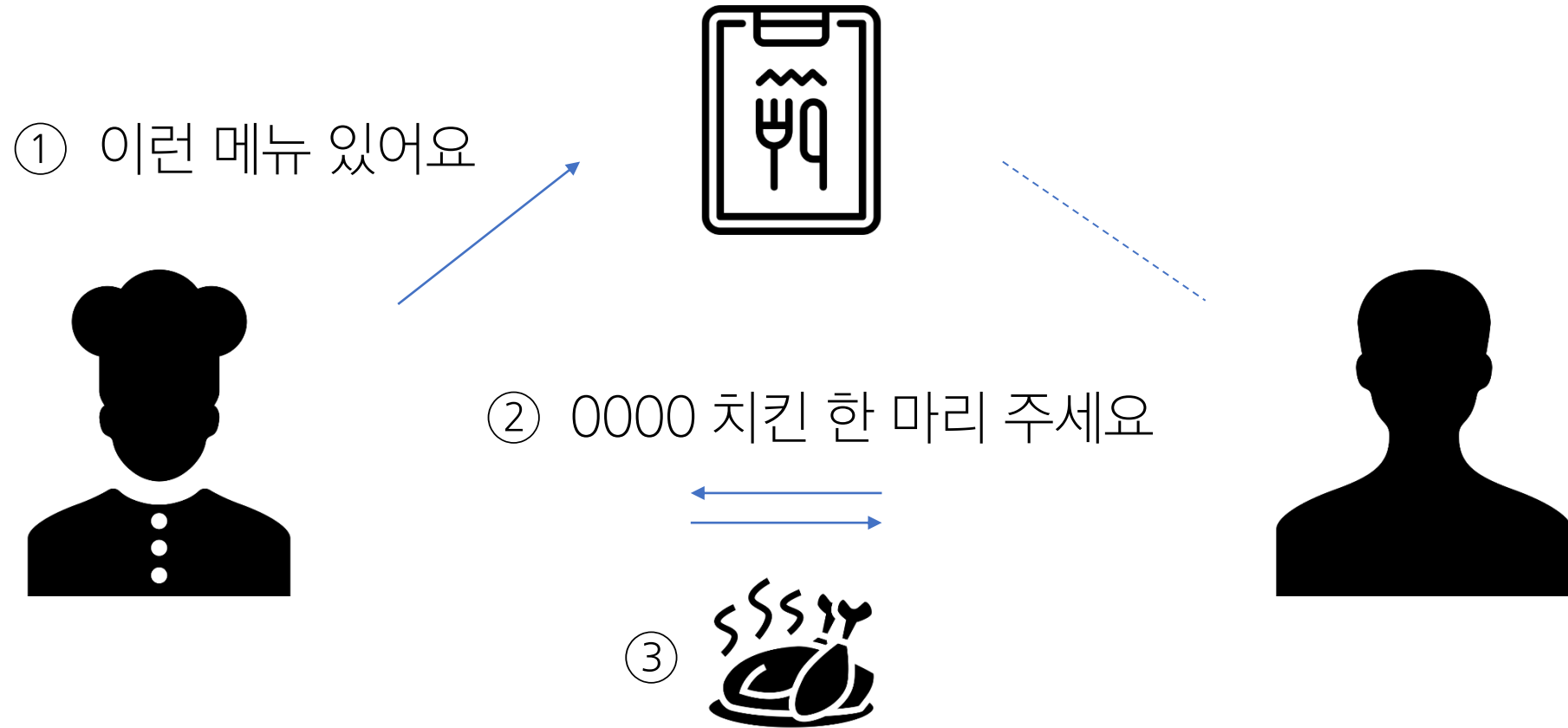


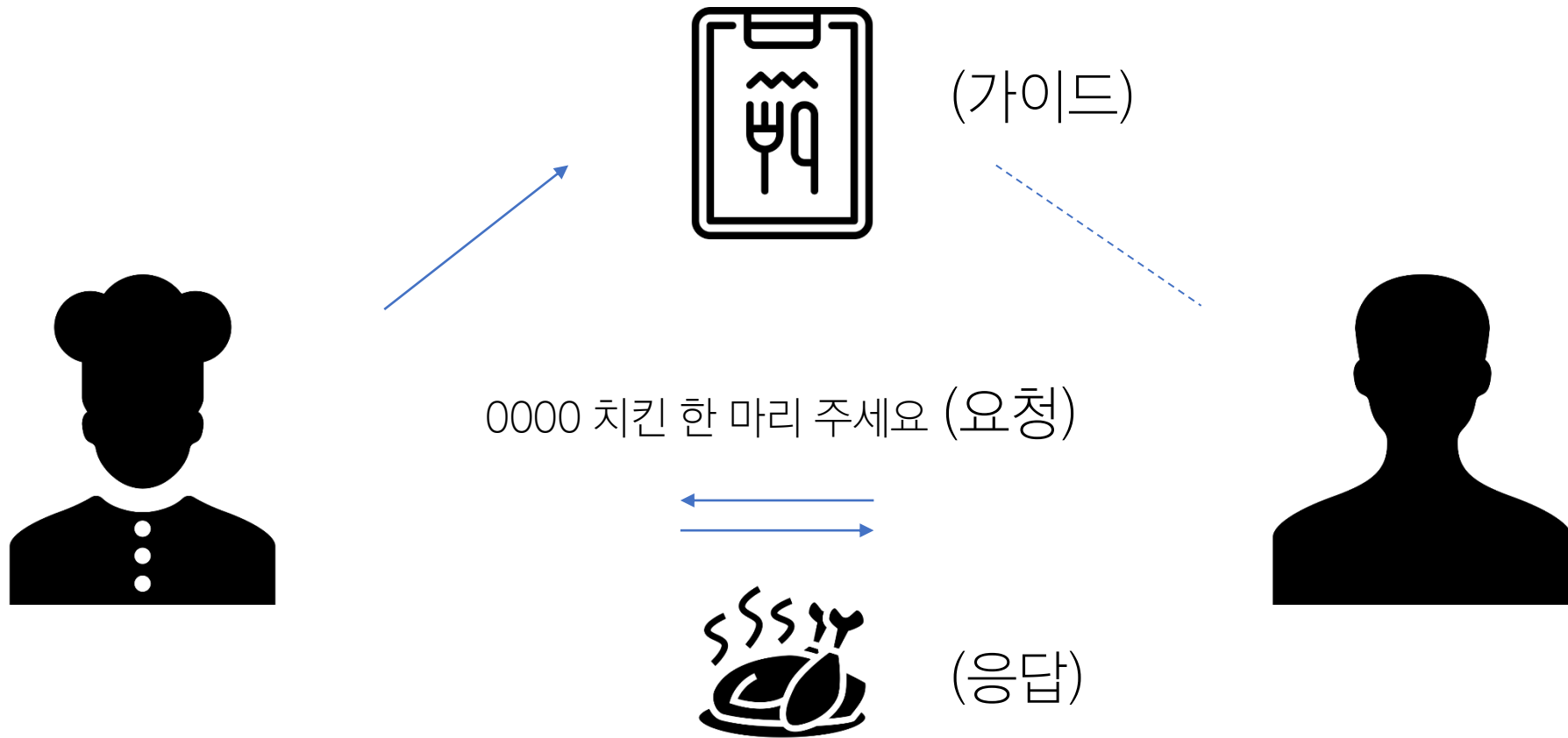
- “이렇게 요청을 보내면, 이렇게 정보를 제공 해줄 것이다”라는 매뉴얼
 - 소프트웨어와 소프트웨어 간 지정된 정의(형식)으로 소통하는 수단 → API
- 스마트폰의 날씨 앱은 기상청에서 제공하는 API를 통해 기상청 시스템과 대화하여 매일 최신 날씨 정보를 표시 할 수 있음

API

(Application Programming Interface)

두 소프트웨어가 서로 통신할 수 있게 하는 메커니즘





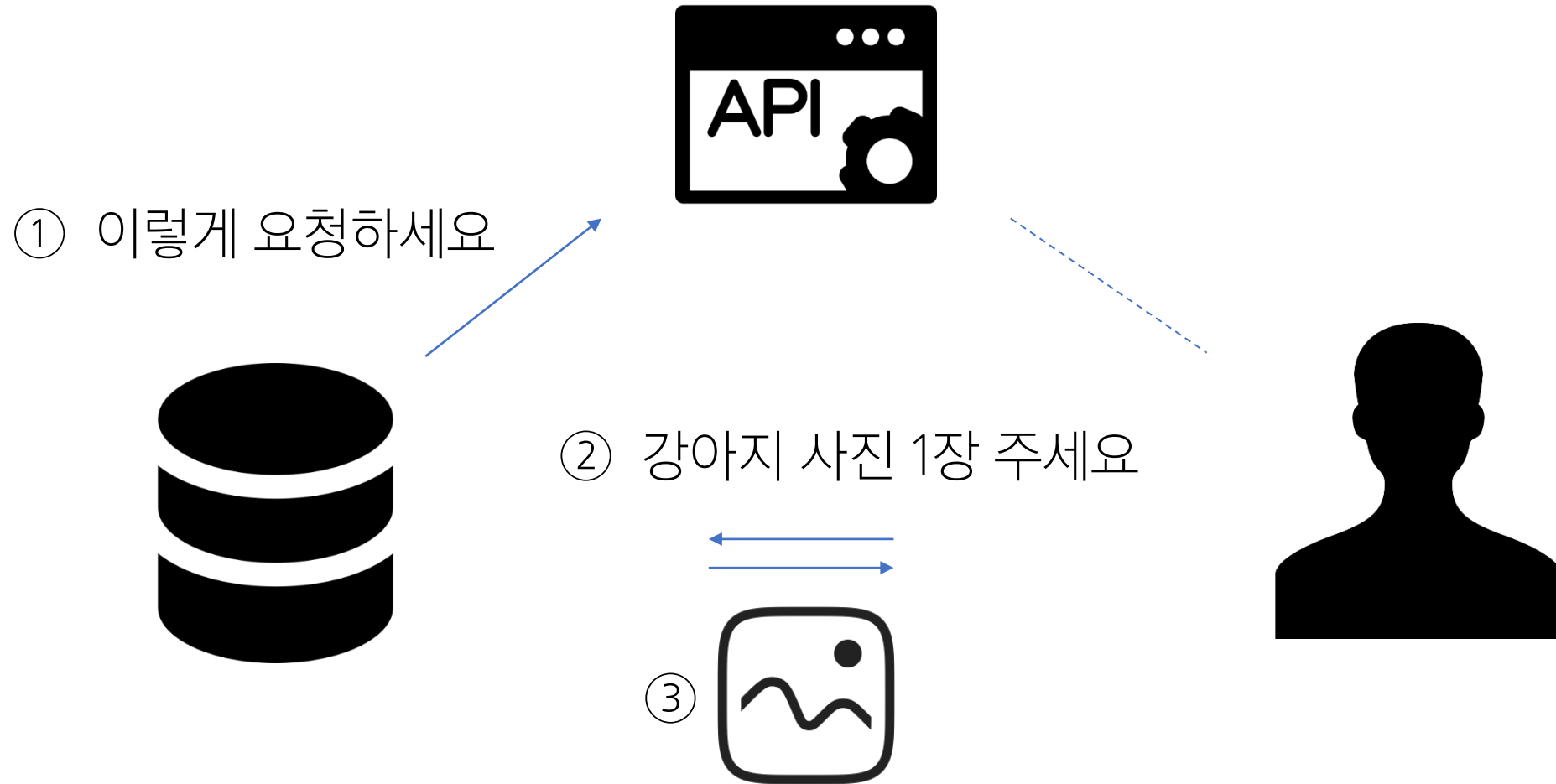
요리사가 만든 메뉴판(주문 **가이드**)에 맞춰
고객은 음식을 주문(**요청**)하고,
요리사는 주문 받은 음식을 전달(**응답**)한다.


API

(Application Programming Interface)

- 두 소프트웨어가 서로 통신할 수 있게 하는 메커니즘
- 가이드에 맞춰 요청하고 응답하는 통신 소프트웨어

01_example.py





Dog API

- Documentation
- Breeds list
- About
- Submit your dog

- Dog CEO Zine
- Buy me a dog treat
- View on GitHub
- Follow on Twitter

The internet's biggest collection of **open source dog pictures**.

Read our [documentation](#) to find out more or try it out for yourself below.

Want to add your dog to the collection? Submit your photos as a [pull request here](#).

Need more dog in your life? Get issue 1 of [Dog CEO Zine](#) - a quarterly business and lifestyle magazine for the modern dog. Featuring an exclusive interview with Scottie the Monopoly dog. [Order your copy](#) from Side Orders Publishing. Ships worldwide.


`https://dog.ceo/api/breeds/image/random`

Fetch!

JSON

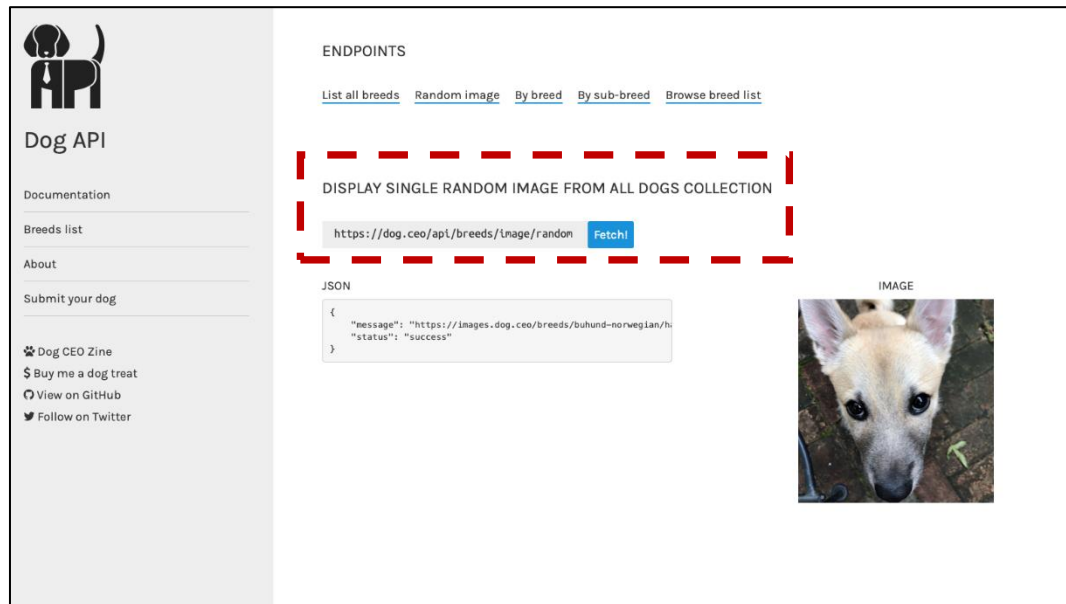
```
{  "message": "https://images.dog.ceo/breeds/boxer/n02108089_35",  "status": "success"}
```

IMAGE

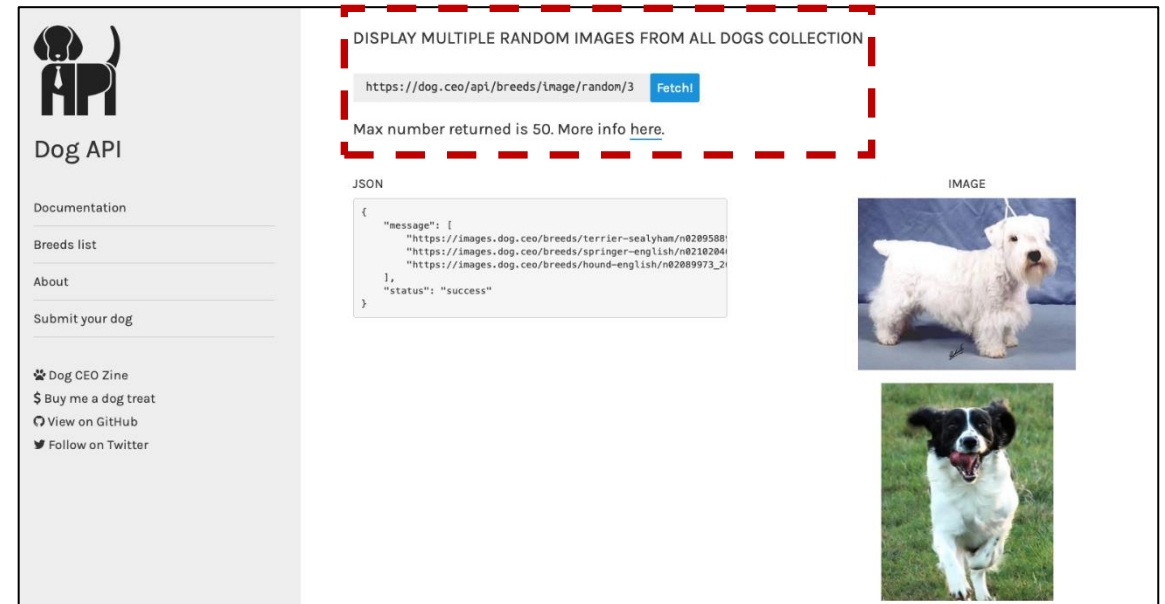


강아지 사진을 요청하면 응답해주는 Dog API
<https://dog.ceo/dog-api/>

무작위 강아지 사진 한 장 필요할 땐 이렇게 요청



무작위 강아지 사진 여러 장 필요할 땐 이렇게 요청



적절한 요청 방법과 응답 결과를 알려주는 가이드 문서

```
# 01_example.py

import requests

URL = 'https://dog.ceo/api/breeds/image/random'

response = requests.get(URL).json()
print(response)

results = response.get('message')
print(results)
```

무작위 강아지 사진 한 장 요청

```
{'message': 'https://images.dog.ceo/breeds/spaniel-brittany/n02101388_10373.jpg',  
 'status': 'success'}  
'https://images.dog.ceo/breeds/spaniel-brittany/n02101388_10373.jpg'
```

API는 message에 사진 경로 데이터를
담아 응답

02_example.py

알라딘 Open API 매뉴얼 ☆ ⓘ ⓘ ⓘ
파일 수정 보기 도구 도움말

1. 요청 (Request)
1) 상품 검색 API
2) 상품 리스트 API
3) 상품 조회 API
4) 중고상품 보유 매장 검색 API
2. 응답 (Response)
1) 상품 검색/상품 리스트/상품 조회 API
2) 상품 조회 API : 부가정보
3) 중고상품 보유 매장 검색 API
* 최종업데이트 : 2022-07-13

1. 요청 (Request)

1) 상품 검색 API

- 요청 방법

- 요청 URL : <http://www.aladin.co.kr/ttb/api/ItemSearch.aspx>
- 요청 URL 샘플 : [http://www.aladin.co.kr/ttb/api/ItemSearch.aspx?ttbkey=\[TTBKey\]&Query=aladdin&QueryType=Title&MaxResults=10&start=1&SearchTarget=Book&output=xml&Version=20131101](http://www.aladin.co.kr/ttb/api/ItemSearch.aspx?ttbkey=[TTBKey]&Query=aladdin&QueryType=Title&MaxResults=10&start=1&SearchTarget=Book&output=xml&Version=20131101)
- 결과샘플
 - XML형식 : http://www.aladin.co.kr/ttb/api/test/ItemSearch_20131101.xml
 - javascript형식 : http://www.aladin.co.kr/ttb/api/test/ItemSearch_20131101.js
- 한 페이지에 최대 50개, 총 결과는 200개까지만 검색 가능

- 상품 검색 API 요청(Request) 파라미터

구분	요청변수	변수종류	설명
필수	TTBKey	문자열	부여받은 TTBKey값
	Query	문자열	검색어
옵션 (옵션 조정 변수가 없을 경우에는 기본값으로 검색)	QueryType	Keyword (기본값) : 제목+저자 Title : 제목검색 Author : 저자검색 Publisher : 출판사검색	검색어 종류
	SearchTarget	Book(기본값) : 도서 Foreign : 외국도서 Music : 음반 DVD : DVD Used : 중고상(도서/음반/DVD 등) eBook: 전자책 All : 위의 모든 타겟(몰)	검색 대상 Mall
	Start	1이상, 양의 정수(기본값:1)	검색결과 시작페이지
	MaxResults	1이상 100이하, 양의 정수(기본값:10)	검색결과 한 페이지당 최대 출력 개수

도서 관련 정보를 제공하는 알라딘 API와 가이드 문서

https://docs.google.com/document/d/1mX-WxuoGs8Hy-QalhHcvuV17n50uGI2Sg_GHofgiePE/edit

알라딘을 비롯한 많은 서비스는 클라이언트에게
고유의 키를 발급해, 들어온 요청이 누구의
요청인지 확인(인증)한다.

- ❖ 따라서 나를 인증 할 수 있는 (일종의 주민등록번호)
키가 외부에 유출되지 않도록 잘 관리하여야 한다.

| 사전 준비

- 공용 문서 (<https://abit.ly/pb-document>)
 - “Aladin API Key 발급 가이드” 진행

| 요청 방법 1 (1/2)

- 무슨 자료를 요청할지 (endpoint)

[http://www.aladin.co.kr/ttb/api/ItemList.aspx](http://www.aladin.co.kr/ttb/api/ItemList.aspx?ttbkey={key}&QueryType={query_type})
`?ttbkey={key}&QueryType={query_type}`

| 요청 방법 1 (2/2)

- 자료 요청에 필요한 정보 (query)
 - endpoint와 query는 ‘?’ 로 연결 및 구분
 - 각각의 query는 ‘&’ 으로 연결

```
http://www.aladin.co.kr/ttb/api/ItemList.aspx  
?ttbkey={key}&QueryType={query_type}
```

요청 방법 2

```
# 02_example.py

URL = 'http://www.aladin.co.kr/ttb/api/ItemSearch.aspx'

params = {
    'ttbkey': '부여받은 TTBKey',
    'Query': '파울로 코엘료',
    'QueryType': 'Author',
    'MaxResults' : 20,
    'start' : 1,
    'SearchTarget' : 'Book',
    'output' : 'js',
    'Version' : '20131101'
}

response = requests.get(URL, params=params)
response = response.json()
```

무슨 자료를 요청할지 (endpoint)

요청에 필요한 정보를 `params`에 담는다.

`params`를 포함하여 URL에 `get` 방식으로 요청 후
`json` 형태로 변환

응답 데이터

- 응답 받은 결과 중, 'item'에 도서 정보를 담은 리스트 응답
해당 정보는 알라딘 사이트에서도 확인가능

```
{
  "version": "20131101",
  "logo": "http://image.aladin.co.kr/img/header/2011/aladin_logo_new.gif",
  "title": "알라딘 주독할 만한 신간 리스트 - 국내도서",
  "link": "https://www.aladin.co.kr/shop/common/wnew.aspx?NewType=SpecialNew&BranchType=1&partner=openAPI",
  "pubDate": "Thu, 06 Jul 2023 06:49:10 GMT",
  "totalResults": 1981,
  "startIndex": 1,
  "itemsPerPage": 20,
  "query": "QueryType=ITEMNEWSPECIAL;SearchTarget=Book",
  "searchCategoryId": 0,
  "searchCategoryName": "국내도서",
  "item": [
    {
      "title": "희망의 책 - 희망의 사도가 전하는 끝나지 않는 메시지",
      "link": "http://www.aladin.co.kr/shop/wproduct.aspx?ItemId=320162652&partner=openAPI&start=api",
      "author": "제인 구달, 더글러스 에이브럼스, 게일 허드슨 (지은이), 변용란 (옮긴이)",
      "pubDate": "2023-07-07",
      "description": "30년 넘게 동물과 인간, 환경의 권리를 위해 전 세계에서 활약해 온 제인 구달 박사가 5년 만에 한국을 다시 찾는다. '희망의 실천' 강연은 이번엔 웨시언스북스에서 나온 신작 '희망의 책: 희망의 사도가 전하는 끝나지 않는 메시지'가 처음으로 공개되는 현상이기도 하다.",
      "isbn": "K662834574",
      "isbn13": "9791192908076",
      "itemId": 320162652,
      "priceSales": 16200,
      "priceStandard": 18000,
      "mallType": "BOOK",
      "stockStatus": "예약판매",
      "mileage": 900,
      "cover": "https://image.aladin.co.kr/product/32016/26/coversum/k662834574_1.jpg",
      "categoryId": 51209,
      "categoryName": "국내도서>과학>생물과학>생태학",
      "publisher": "사이언스북스",
      "salesPoint": 0,
      "adult": false,
      "fixedPrice": true,
      "customerReviewRank": 0,
      "subInfo": {}
    },
    {
      "title": "과학이 특별 생이다! 사이다 7 - 시간×이상한 나라의 엘리스",
      "link": "http://www.aladin.co.kr/shop/wproduct.aspx?ItemId=320150463&partner=openAPI&start=api",
      "author": "국립과천과학관 박은지 (지은이), 김정진 (그림)",
      "pubDate": "2023-07-07",
      "description": "10살 천재 소녀 화가 엘리스는 다음 전시회를 준비 중이다. 다음 전시회를 위해 보름달 그림에 도전하고 있는데 고민이 하나 있다. 바로 달의 모양이 매일 조금씩 변한다는 것. 달의 모양이 매일 바뀌니 보름달 그림을 완성하기 어려워 작업을 미루고만 있는데...",
      "isbn": "K992834571",
      "isbn13": "9791185402819",
    }
  ]
}
```

The screenshot shows the Aladin website interface. At the top, there's a search bar and navigation links. The main content area displays search results for the book '희망의 책' (The Book of Hope) by Jane Goodall. The book cover is shown, along with its title, author, and price. Below the book, there's a description and a list of related books. The sidebar on the right contains navigation links and a list of recommended books.

URL 파라미터 (쿼리 파라미터)

1. URL에 포함된 파라미터로 ‘?’ 이후의 문자열
2. 여기서 파라미터는 [파라미터 Key]=[파라미터 Value]가 한 세트이다.
 - 파라미터 Key는 대소문자를 구분한다.
 - API 사용 시, 공식문서에 명시된 파라미터 Key를 사용해야 한다.
3. 파라미터가 여러 개면 ‘&’로 구분한다.

예시) `https://example.com?fRuiT=orange&gAmE=tic-tac-toe`

```
URL = 'https://example.com'
params = {
    'fRuiT': 'orange',
    'gAmE': 'tic-tac-toe'
}
response = requests.get(URL, params=params).json()
print(response)
```

요청에 대한 응답 예시

```
{'fruit_param': 'I love orange!', 'game': 'tic-tac-toe is my favorite!'}
```

Aladin API

준비사항

개발도구 및 라이브러리

- 개발도구
 - Visual Studio Code
 - Python 3.9+
- 필수 라이브러리
 - <https://requests.readthedocs.io/en/latest/>
- API
 - [알라딘 API](#) - 도서 정보 및 API 서비스

요구사항

| 공통 요구사항

- API 요청 시 버전은 가장 최신 버전인 **20131101**을 사용

| 필수 요구사항

- A. 저자의 도서 조회
- B. 저자의 도서 조회 & 조건 I
- C. 저자의 도서 조회 & 조건 II
- D. 특정 도서 저자의 다른 도서 조회
- E. 가장 저렴한 중고 도서 가격 조회

A. 저자의 도서 조회 - 요구사항

- problem_a.py 풀이
- 저자 '파울로 코엘료'의 도서 검색
- 최대 20개의 도서 제목을 추출
- 요구사항을 만족하는 함수 `get_author_books` 작성

A. 저자의 도서 조회 - 결과

- problem_a.py 실행 예시

```
['연금술사',  
'11분',  
'흐르는 강물처럼',  
'브리다',  
'포르토벨로의 마녀',  
'마법의 순간 (리커버)',  
'오 자히르',  
'불륜',  
'순례자 - 개정판',  
'다섯번째 산',  
'아처',  
'내가 빛나는 순간',  
'일러스트 연금술사',  
'스파이',  
'피에트라 강가에서 나는 울었네',  
'알레프',  
'아크라 문서',  
'승자는 혼자다 1',  
'악마와 미스 프랭',  
'연금술사 (문학동네 30주년 기념 특별판)']
```

B. 저자의 도서 조회 & 조건 I - 요구사항

- problem_b.py 풀이
- 저자 '파울로 코엘료'의 도서 20권 중 리뷰 평점(customerReviewRank)이 9점 이상인 도서 목록 추출
- 요구사항을 만족하는 함수 get_best_review_books 작성

B. 저자의 도서 조회 & 조건 I - 결과

- problem_b.py 실행 예시

```
[{'adult': False,
  'author': '파울로 코엘료 (지은이), 최정수 (옮긴이)',
  'categoryId': 50920,
  'categoryName': '국내도서>소설/시/희곡>스페인/중남미소설',
  'cover': 'https://image.aladin.co.kr/product/30/73/coversum/8982814477_3.jpg',
  'customerReviewRank': 9,
  'description': "세상을 두루두루 여행하기 위해 양치기가 된 청년 산티아고의 '자아의 신화' 찾기 여행담. 자칫 딱딱하게 보일 "
    "수 있는 제목과는 달리 간결하고 경쾌한 언어들로 쓰여 있어서, 물이 흘러가듯 수월하게 읽히는 작품이다.",
  'fixedPrice': True,
  'isbn': '8982814477',
  'isbn13': '9788982814471',
  'itemId': 307361,
  'link': 'http://www.aladin.co.kr/shop/wproduct.aspx?ItemId=307361&partner=openAPI&start=api',
  'mallType': 'BOOK',
  'mileage': 600,
  'priceSales': 10800,
  'priceStandard': 12000,
  'pubDate': '2001-12-01',
  'publisher': '문학동네',
  'salesPoint': 85613,
  'stockStatus': '',
  'subInfo': {},
  'title': '연금술사'},
{'adult': False,
  'author': '파울로 코엘료 (지은이), 오진영 (옮긴이)',
  'categoryId': 50920,
  'categoryName': '국내도서>소설/시/희곡>스페인/중남미소설',
  'cover': 'https://image.aladin.co.kr/product/29733/11/coversum/8954687482_1.jpg',
```

이하 생략

C. 저자의 도서 조회 & 조건 II - 요구사항

- problem_c.py 풀이
- 저자 '파울로 코엘료'의 도서 20권 중 판매 지수(salesPoint)가 높은 5개 목록을 추출
 - 판매 지수가 높은 순서대로 정렬
- 요구사항을 만족하는 함수 get_bestseller_books 작성
- 참고)
 - [sort 메서드](#)
 - [sorted 함수](#)

C. 저자의 도서 조회 & 조건 II - 결과

- problem_c.py 실행 예시

```
[{'제목': '연금술사', '판매지수': 84709},  
{ '제목': '11분', '판매지수': 10429},  
{ '제목': '흐르는 강물처럼', '판매지수': 10192},  
{ '제목': '브리다', '판매지수': 10161},  
{ '제목': '포르토벨로의 마녀', '판매지수': 8409}]
```

D. 특정 도서 저자의 다른 도서 조회 - 요구사항

- `problem_d.py` 풀이
- 제공된 도서 제목('베니스의 상인', '죄와 벌', '*')을 검색하여
응답 받은 첫 번째 도서의 저자가 집필한 다른 5개의 도서 제목을 추출
 - 응답 받은 첫 번째 도서를 제외한 다른 도서 5개를 출력해야 함
- 검색된 도서가 없는 경우 `None`을 반환
- 요구사항을 만족하는 함수 `get_author_other_books` 작성

D. 특정 도서 저자의 다른 도서 조회 - 결과

- problem_d.py 실행 예시

```
{'"베니스의 상인"의 저자 "윌리엄 셰익스피어"의 다른 도서 목록': ['햄릿',  
    '모든 슬픔이 사라진다 - 미션나무에서 아카시아까지 시가 된 꽃과 나무',  
    '셰익스피어 4대 비극 세트 : 햄릿.오셀로.맥베스.리어 왕 - 전4권',  
    '셰익스피어 4대 비극 에디션 세트 - 전4권 (리커버 특별판) - 햄릿 + 오셀로 + 리어 왕 + 맥베스',  
    '맥베스']}  
{'"죄와 벌"의 저자 "표도르 도스토옙스키"의 다른 도서 목록': ['카라마조프가의 형제들 세트 - 전3권', '죄와 벌 2', '죄와 벌 - 상', '죄와 벌 - 하', '지하로부터의 수기']}  
None
```

E. 가장 저렴한 중고 도서 가격 조회 - 요구사항

- `problem_e.py` 풀이
- 제공된 도서 제목('죄와 벌', '로미오와 줄리엣', '*')을 검색하여 응답 받은 첫 번째 도서의 중고 도서 정보 검색
- 중고 도서 배송처 중 가장 저렴하게 판매중인 중고 도서의 배송처와 가격을 출력
- 검색된 도서가 없는 경우 `None`을 반환
- 중고 도서 재고가 없는 경우 새 도서의 가격을 출력
- 요구사항을 만족하는 함수 `get_used_books` 작성

E. 가장 저렴한 중고 도서 가격 조회 - 결과

- problem_e.py 실행 예시

```
'도서 "죄와 벌"의 가장 저렴한 중고는 광활한 우주점이 보유 중이며, 6100원에 판매 중입니다.'  
'도서 "로미오와 줄리엣"의 중고 재고가 없으며, 새 제품은 13320원에 판매 중입니다.'  
None
```

도전 과제

도전 과제 안내

- 생성형 AI를 사용해 도전과제 요구사항을 해결하기
- 생성형 AI 제공 서비스는 자유롭게 선택 가능

도전 과제

- 도서 데이터를 제공하는 다른 API를 사용하여 내가 원하는 데이터를 추출해 재구성하기
- API 목록
 - 국가 자료 종합 목록
(<https://www.nl.go.kr/NL/contents/N31101030400.do>)
 - 네이버 도서검색 API
(<https://developers.naver.com/docs/serviceapi/search/book/book.md>)
 - 카카오 도서검색 API
(<https://developers.kakao.com/docs/latest/ko/daum-search/dev-guide>)

Spotify API

준비사항

개발도구 및 라이브러리

- 개발도구
 - Visual Studio Code
 - Python 3.9+
- 필수 라이브러리
 - <https://requests.readthedocs.io/en/latest/>
- API
 - [Spotify API](#) - 음악 정보 및 API 서비스

| 제공사항

- examples 폴더
 - 이번 프로젝트 해결을 위해 알아야 하는 혹은 직접적인 도움이 될 수 있는 코드

| 사전 준비 (1/2)

- 공용 문서 (<https://abit.ly/pb-document>)
 - “Spotify API Key 발급 가이드” 진행

사전 준비 (2/2)

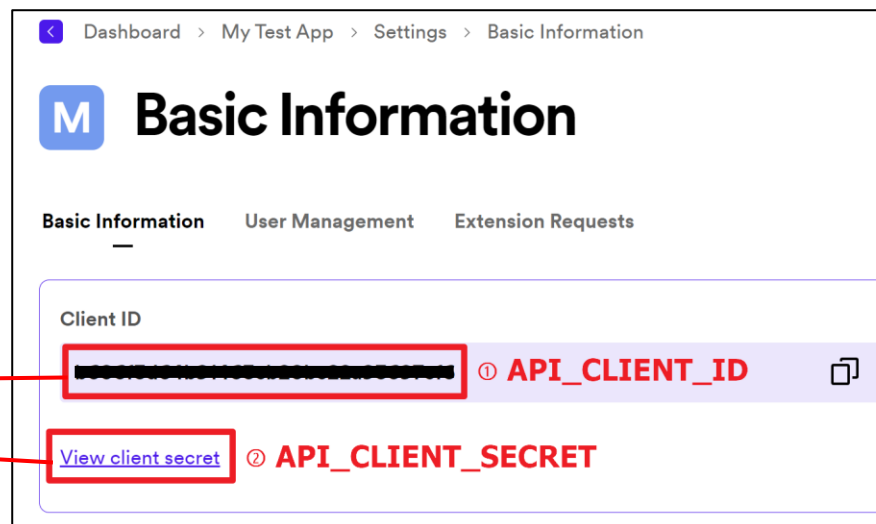
- 발급 받은 Client ID와 Client Secret ID를 각각 복사해서 examples 폴더 내 spotify_config.py에 입력

```
'''
Spotify에 요청을 보내기 위한 Header파일
'''

import requests

# 📌 여기에 발급받은 API_CLIENT_ID와 API_
API_URL = 'https://api.spotify.com/v1'
API_CLIENT_ID = ''
API_CLIENT_SECRET = ''
```

spotify_config.py



❖ 발급받은 API Key는 외부에 유출하지 않도록 주의

Spotify API 숙지사항

1. Spotify API를 통해 실습하기 위해서는 **examples** 폴더 하위에 있는 **spotify_config.py**를 먼저 완성해야만 실습이 가능
2. 또한, API를 요청할 때 모든 Spotify API의 **Headers**에 **spotify_config.py**의 **getHeaders()**가 요청 **headers**에 필수로 작성되어야 함

```
# search API 문서: https://developer.spotify.com/documentation/web-api/reference/search
URL = 'https://api.spotify.com/v1'

headers = getHeaders()
params = {
    'q': 'artist:BTS track:Take Two', # 필수 파라미터
    'type': 'track',                  # 필수 파라미터
    'market': 'KR',
    'limit': 1
}

# 요청 보내 받아온 결과는 requests 타입의 데이터이고, 파이썬에서 바로 쓸 수 없으며
response = requests.get(f'{URL}/search', headers=headers, params=params)
```


요구사항

공통 요구사항

- API 요청 시 시장(market)은 “대한민국”의 국가코드를 활용하여 진행
- 메인 참고 문서
 - <https://developer.spotify.com/documentation/web-api/reference/search>
- 추가 참고 문서
 - <https://developer.spotify.com/documentation/web-api/reference/get-recommendation-genres>
 - <https://developer.spotify.com/documentation/web-api/reference/get-an-artists-related-artists>
 - <https://developer.spotify.com/documentation/web-api/reference/get-track>
 - <https://developer.spotify.com/documentation/web-api/reference/get-an-artist>
 - <https://developer.spotify.com/documentation/web-api/reference/get-recommendations>

| 필수 요구사항

- A. 아티스트 정보 조회
- B. 아티스트 정보 조회 & 조건
- C. 트랙 정보 조회 & 조건
- D. 연관 아티스트 정보 조회
- E. 추천 트랙 정보 조회

A. 아티스트 정보 조회 - 요구사항

- `problem_a.py` 풀이
- k-pop 장르의 아티스트에 대한 카탈로그 정보 조회
- 최대 20명의 아티스트 이름 조회
- 요구사항을 만족하는 함수 `get_artists` 작성

A. 아티스트 정보 조회 - 결과

- problem_a.py 실행 예시

```
['Jack Harlow',  
'BTS',  
'Jung Kook',  
'Stray Kids',  
'JENNIE',  
'BLACKPINK',  
'LE SSERAFIM',  
'NewJeans',  
'TWICE',  
'Jimin',  
'ENHYPEN',  
'(G)I-DLE',  
'SEVENTEEN',  
'TOMORROW X TOGETHER',  
'ATEEZ',  
'ITZY',  
'Dave Stewart',  
'aespa',  
'IVE',  
'wave to earth']
```

❖ Spotify “Search for Item” 응답 데이터는 시기에 따라 예시 출력과 다를 수 있음

B. 아티스트 정보 조회 & 조건 - 요구사항

- `problem_b.py` 풀이
- k-pop 장르에서 인기도가 높은 아티스트에 대한 카탈로그 정보 조회
- 최대 20명의 아티스트 이름을 조회한 결과에서 인기도가 **80**이상인 아티스트 조회
- 요구사항을 만족하는 함수 `get_popular_artists` 작성

B. 아티스트 정보 조회 & 조건 - 결과

- problem_b.py 실행 예시

```
['Jack Harlow',  
'BTS',  
'Jung Kook',  
'Stray Kids',  
'BLACKPINK',  
'LE SSERAFIM',  
'NewJeans',  
'TWICE',  
'Jimin']
```

❖ Spotify “Search for Item” 응답 데이터는 시기에 따라 예시 출력과 다를 수 있음

C. 트랙 정보 조회 & 조건 - 요구사항

- `problem_c.py` 풀이
- 클래식 장르에서 가장 최근에 발매된 트랙에 대한 카탈로그 정보 조회
- 최대 10개의 트랙을 조회한 결과에서 발매일이 가장 빠른 순서대로 5개 트랙을 조회
 - 각 트랙의 앨범 이름, 아티스트 이름 그리고 발매일을 함께 출력
- 요구사항을 만족하는 함수 `get_recently_tracks` 작성
- 참고)
 - [sort 메서드](#)
 - [sorted 함수](#)

C. 트랙 정보 조회 & 조건 - 결과

- problem_c.py 실행 예시

```
[{'발매일': '2023-03-17',  
  '아티스트': 'Wolfgang Amadeus Mozart',  
  '앨범명': 'Abbado: Mozart'},  
{ '발매일': '2022-10-07',  
  '아티스트': 'Ludwig van Beethoven',  
  '앨범명': 'Beethoven - Inspiring Classics'},  
{ '발매일': '2020-08-07', '아티스트': 'Xuefei Yang', '앨범명': 'Sketches of China'},  
{ '발매일': '2018-03-29', '아티스트': 'Edvard Grieg', '앨범명': 'Grieg Slow Motions'},  
{ '발매일': '2012-01-01', '아티스트': 'Various Artists', '앨범명': 'The All-Baroque Box'}]
```

❖ Spotify “Search for Item” 응답 데이터는 시기에 따라 예시 출력과 다를 수 있음

D. 연관 아티스트 정보 조회 - 요구사항

- `problem_d.py` 풀이
- 특정 아티스트와 유사한 아티스트에 대한 카탈로그 정보 조회
- 제공된 아티스트 `NewJeans`와 `OldShirts` 활용
- 유사한 아티스트가 없을 경우 `None`을 반환
- 요구사항을 만족하는 함수 `get_related_artists` 작성

D. 연관 아티스트 정보 조회 - 결과

- problem_d.py 실행 예시

```
['LE SSERAFIM',  
'IVE',  
'NMIXX',  
'aespa',  
'STAYC',  
'NAYEON',  
'XG',  
'ITZY',  
'Kep1er',  
'HUH YUNJIN',  
'KISS OF LIFE',  
'BIBI',  
'JEON SOMI',  
'FIFTY FIFTY',  
'ENHYPEN',  
'MISAMO',  
'IZ*ONE',  
'Red Velvet',  
'TOMORROW X TOGETHER']  
None
```

❖ Spotify “Get Artist's Related Artists” 응답 데이터는 시기에 따라 예시 출력과 다를 수 있음

E. 추천 트랙 정보 조회 - 요구사항

- `problem_e.py` 풀이
- 특정 시드(seed) 데이터를 기반으로 추천하는 아티스트 및 트랙 정보 조회
 - 최대 5개의 추천 트랙을 조회
 - 추천 트랙의 이름과 아티스트 이름을 함께 출력
- 제공된 트랙 “HypeBoy”, 아티스트 “BTS”, 장르 “k-pop”과 “acoustic” 활용
- 제공된 트랙이나 아티스트에 대한 결과가 없을 경우 `None`을 반환
- 요구사항을 만족하는 함수 `get_recommended_tracks` 작성

E. 추천 트랙 정보 조회 - 결과

- problem_e.py 실행 예시

```
[{'아티스트': 'Justin Bieber', '트랙': 'Up'},  
 {'아티스트': 'BTS', '트랙': '친구'},  
 {'아티스트': 'Boyce Avenue', '트랙': 'Crash Into Me (2009)'},  
 {'아티스트': 'BLACKPINK', '트랙': 'Typa Girl'},  
 {'아티스트': 'JEON SOMI', '트랙': 'XOXO'}]
```

도전 과제

도전 과제 안내

- 생성형 AI를 사용해 도전과제 요구사항을 해결하기
- 생성형 AI 제공 서비스는 자유롭게 선택 가능

도전 과제

- 음악 데이터를 제공하는 다른 API를 사용하여 내가 원하는 데이터를 재구성하기
- API 목록
 - SoundCloud API
(<https://developers.soundcloud.com/docs/api/guide>)
 - Apple Music API
(<https://developer.apple.com/documentation/applemusicapi>)

제출

제출 시 주의사항

- 제출기한은 금일 18시까지 입니다. 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 강사님을 Maintainer로 설정해야 합니다.