

DB 작성 문서

현재까지 작성한 country model data

```
"country_name": "",
"country_code": "GH", # https://ko.wikipedia.org/wiki/ISO_3166 참조 api param 으로 사용
# 메
"exchange_rate_code": "GHS", # 통화 코드 환율을 받아오기 위한 param 으로 사용하려 했음
"ysd_er": "",
"cur_nm": "",
"status": "▶20.9.1.부터 코토카 국제공항의 국제선 항공 운항 재개(20.8.30.)(단, 육로 및 해로 국경은봉쇄 유지)\r\n* 탑승 전 72시간 이내 검사받은 코로나19 PCR 음",
"status_wrt_dt": "2022-02-16",
"events": [],
"visa": [],
```



- api parameter 는 postman에서 url을 찍어보는 게 가장 수월하다고 생각
- 일단은 api로 db를 만들어두고 crontab 이나 apscheduler 를 통해 주기적으로 갱신 하는 방향으로 생각했었음
- 하단의 모델 작성란 수정되지 않은 부분이 있어, 흐름 참고만 하는 걸 추천

DB에 반영 안된 API

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/f0bc89b6-35e3-42ee-aac8-f0bb0dfc1c55/18.%EA%B8%B0%EC%88%A0%EB%AC%B8%EC%84%9C_%EC%99%B8%EA%B5%90%EB%B6%80_%EA%B3%B5%EC%A7%80%EC%82%AC%ED%95%AD_v1.2.docx

```
# 18. 외교부 공지사항
# https://www.data.go.kr/iim/api/selectAPIAccountView.do
import requests
api_key = "<API KEY>"
url = f"http://apis.data.go.kr/1262000/NoticeService2/getNoticeList2?serviceKey={api_key}%2F0BWaoU5xsw5hNE4zys87bTkD3C3y0Gm5RXWURYvakg"

response = requests.request("GET", url)
```

- news api
- country params로 검색할 수 있을 거라 생각

```
import requests

API_KEY = '<API KEY>'

url = f"https://newsapi.org/v2/top-headlines?q=travel&category=business&language=en&country=us&apiKey={API_KEY}"

payload={}
headers = {}

response = requests.request("GET", url, headers=headers, data=payload)

print(response.json()['articles'])
# response
[{'author': 'Andrea Cavallier',
 'content': 'Thousands of flights were delayed and hundreds were cancelled '
            'across the United States as severe weather hampered travel on '
            'one of the busiest holiday travel weekends of the year.\r\n'
            'As of Sunday night... [+4545 chars]',
 'description': 'Thousands of flights were delayed and hundreds were canceled '
                'across the United States as severe weather hampered travel '
                'on one of the busiest holiday travel weekends of the year.',
 'publishedAt': '2022-11-28T05:31:04Z',
 'source': {'id': None, 'name': 'Daily Mail'},
 'title': 'Thousands of flights delayed or cancelled after severe weather '
           'sets off travel chaos - Daily Mail',
 'url': 'https://www.dailymail.co.uk/news/article-11476065/Thousands-flights-delayed-cancelled-severe-weather-sets-travel-chaos.html',
 'urlToImage': 'https://i.dailymail.co.uk/1s/2022/11/28/05/64999065-0-image-a-60_1669613352327.jpg'}}]
```

DB 반영 API

- 공공데이터 사용

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/08cb559a-2a1b-4d86-b65e-65c818e66d72/22.%EA%B8%B0%EC%88%A0%EB%AC%B8%EC%84%9C_%EC%99%B8%EA%B5%90%EB%B6%80_%EA%B5%AD%EA%B0%80-%EC%A7%80%EC%97%AD%EB%B3%84_%ED%91%9C%EC%A4%80%EC%BD%94%EB%93%9C_v1.2_\(1\).docx](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/08cb559a-2a1b-4d86-b65e-65c818e66d72/22.%EA%B8%B0%EC%88%A0%EB%AC%B8%EC%84%9C_%EC%99%B8%EA%B5%90%EB%B6%80_%EA%B5%AD%EA%B0%80-%EC%A7%80%EC%97%AD%EB%B3%84_%ED%91%9C%EC%A4%80%EC%BD%94%EB%93%9C_v1.2_(1).docx)

```
# 22. 외교부_국가_지역별_표준코드
# https://www.data.go.kr/iim/api/selectAPIAccountView.do
# country code, name 를 받아오는데 사용
import requests
from urllib.parse import urlencode, unquote
import json

API_KEY = '<API KEY>'

url = "http://apis.data.go.kr/1262000/CountryCodeService2/getCountryCodeList2"
qs = "?" + urlencode(
    {
        'ServiceKey': unquote('<API KEY>'),
        'pageNo': 1,
        'numOfRows': 250,
    }
)

response = requests.get(url + qs)
## response
```

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e522cc5c-42d8-4977-8060-56d2b15133b5/13.%EA%B8%B0%EC%88%A0%EB%AC%B8%EC%84%9C_%EC%99%B8%EA%B5%90%EB%B6%80_%EA%B5%AD%EA%B0%80-%EC%A7%80%EC%97%AD%EB%B3%84_%EC%9E%85%EA%B5%AD%ED%97%88%EA%B0%80%EC%9A%94%EA%B1%B4_v1.2_\(1\).docx](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e522cc5c-42d8-4977-8060-56d2b15133b5/13.%EA%B8%B0%EC%88%A0%EB%AC%B8%EC%84%9C_%EC%99%B8%EA%B5%90%EB%B6%80_%EA%B5%AD%EA%B0%80-%EC%A7%80%EC%97%AD%EB%B3%84_%EC%9E%85%EA%B5%AD%ED%97%88%EA%B0%80%EC%9A%94%EA%B1%B4_v1.2_(1).docx)

```
# 13. 외교부_국가_지역별_입국허가요건
# https://www.data.go.kr/iim/api/selectAPIAccountView.do
# visa 를 받아오는데 사용

import requests
API_KEY = '<API KEY>'
url = f"http://apis.data.go.kr/1262000/EntranceVisaService2/getEntranceVisaList2?serviceKey={API_KEY}%2F0BWaoU5xsw5hNE4zys87bTkD3C3yOG"

response = requests.request("GET", url)

print(response.json()['data'][0])
## response.jsno()
{
  "currentCount": 1,
  "data": [
    {
      "country_eng_nm": "Ghana",
      "country_iso alp2": "GH",
      "country_nm": "가나",
      "dplmt_pspt_visa_cn": "X",
      "dplmt_pspt_visa_yn": "Y",
      "gnrl_pspt_visa_cn": "X",
      "gnrl_pspt_visa_yn": "Y",
      "have_yn": "Y",
      "id": 83,
      "nvisa_entry_evdc_cn": "",
      "ofclpspt_visa_cn": "X",
      "ofclpspt_visa_yn": "Y",
      "remark": ""
    }
  ],
  "numOfRows": 1,
  "pageNo": 1,
  "resultCode": 0,
  "resultMsg": "정상",
}
```

```

    "totalCount": 170
}

```

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/d98cf07a-affc-49db-8a26-bafba8086a52/44.%EA%B8%B0%EC%88%A0%EB%AC%B8%EC%84%9C_%EC%99%B8%EA%B5%90%EB%B6%80_%EA%B5%AD%EA%B0%80%EB%B3%84_%ED%95%B4%EC%99%B8%EC%9E%85%EA%B5%AD%EC%9E%90_%EC%A1%B0%EC%B9%98%ED%98%84%ED%99%A9_v1.1.docx

```

# 44. 외교부 국가별 해외입국자 조치현황
# https://www.data.go.kr/iim/api/selectAPIAccountView.do
# 22 에서 가져온 country_code 를 cond[country_iso_alp2::EQ] 검색 parameter로 사용
# status, status_wrt_dt 를 받아오는데 사용

import requests

API_KEY = '<API KEY>'
url = f"http://apis.data.go.kr/1262000/CountryOverseasArrivalsService/getCountryOverseasArrivalsList?serviceKey={API_KEY}/0BWaoU5xsw5h"

response = requests.request("GET", url)

print(response.json())

## response.json()['data']
{
  "currentCount": 1,
  "data": [
    {
      "country_eng_nm": "Netherlands",
      "country_iso_alp2": "NL",
      "country_nm": "네덜란드",
      "html_origin_cn": "<p style='line-height: 1.5;'><span style='font-size: 12pt;'>▶출발지별 입국 제한 (모든 교통수단 망라)</span>
      "notice_id": "ATC0000000008799",
      "title": "각국의 해외입국자에 대한 조치 현황(국가 순서별)",
      "txt_origin_cn": "▶출발지별 입국 제한 (21.8.8.부터 적용, 모든 교통수단 망라)\r\n가. EU 역내\r\n(새로운 변이바이러스 우려가 있는 경우를 제외하고는)
      "wrt_dt": "2022-03-28"
    }
  ],
  "numOfRows": 10,
  "pageNo": 1,
  "resultCode": 0,
  "resultMsg": "정상",
  "totalCount": 180
}

```

- 다른 API 사용

<https://www.predicthq.com/apis/event-api>

- 이벤트

```

# events 를 받아오는데 사용

import requests

API_KEY = '<API KEY>'

response = requests.get(
    url="https://api.predicthq.com/v1/events/",
    headers={
      "Authorization": f"Bearer {API_KEY}",
      "Accept": "application/json"
    },
    params={
      "category": "public-holidays,festivals,concerts",
      "start.gte": "2022-12-02", # 시작 날짜
      "country": "JP" # 22 country_code 를 검색 parameter 로 사용
    }
)

print(response.json()['results'][0])

## response.json()['results'][0]
{'aviation_rank': 100,
 'brand_safe': True,
 'category': 'public-holidays',
 'country': 'JP',

```

```
{
  'description': '',
  'duration': 86399,
  'end': '2023-02-23T23:59:59Z',
  'entities': [{
    'category': 'public-holidays',
    'entity_id': 'hX6QcnxhsaStrAiHGn9sEd',
    'labels': [
      'event-group',
      'holiday',
      'holiday-national',
      'recurring'
    ],
    'name': 'Emperor's Birthday',
    'type': 'event-group'
  }],
  'first_seen': '2020-01-01T00:05:39Z',
  'geo': {
    'geometry': {
      'coordinates': [138.252924, 36.204824],
      'type': 'Point'
    }
  },
  'id': 'pkjY2B23jEYYJossBi',
  'labels': [
    'holiday',
    'holiday-national'
  ],
  'local_rank': None,
  'location': [138.252924, 36.204824],
  'phq_attendance': None,
  'place_hierarchies': [
    ['6295630', '6255147', '1861060']
  ],
  'private': False,
  'rank': 90,
  'relevance': 0.0,
  'scope': 'country',
  'start': '2023-02-23T00:00:00Z',
  'state': 'active',
  'timezone': None,
  'title': 'Emperor's Birthday',
  'updated': '2022-11-25T02:39:32Z'
}
```

어제자 환율

<https://www.koreaexim.go.kr/ir/HPHKIRO20M01?apino=2&viewtype=C&searchselect=&searchword=>

```
# ysd_er, cur_nm 을 받아오는데 사용
# 현재 짜둔 코드로는 일일 요청 초과 및 지원하는 국가가 적어 다른 api 고려 필요

import requests

API_KEY = '<API KEY>'

url = f"https://www.koreaexim.go.kr/site/program/financial/exchangeJSON?authkey={API_KEY}&searchdate=20180102&data=AP01"

payload={}
headers = {
    'Cookie': 'JSESSIONID=t43WUsr3iRZGDaGRoE1cp4wCA4mZ9EulPEyoGS676VdFLYFLb3mGZW0MBE0Su9aI.aG1wX2RvbwFpbi9ITVBNUzEy; WMONID=-ECmhyU_vnR'
}

response = requests.request("GET", url, headers=headers, data=payload)

print(response.json()[0])
# response
{
  "result": 1,
  "cur_unit": "AED",
  "ttb": "288.78",
  "tts": "294.61",
  "deal_bas_r": "291.7",
  "bkpr": "291",
  "yy_efee_r": "0",
  "ten_dd_efee_r": "0",
  "kftc_bkpr": "291",
  "kftc_deal_bas_r": "291.7",
  "cur_nm": "아랍에미리트 디르함"
}
```

DB 반영 json

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/7f2e765e-6ce3-4ac1-b401-8166a741258b/iso4217.json>

```
# exchange_rate_code 를 가져오기 위해 사용
import json
```

```

file_path = "./iso4217.json"

with open(file_path, 'r') as file:
    er_list = json.load(file)

```

모델 작성

```

import requests
from urllib.parse import urlencode, unquote
import json
import datetime
from dateutil.relativedelta import relativedelta

td = (datetime.datetime.now() - relativedelta(days=1)).strftime('%Y-%m-%d')

url = "http://apis.data.go.kr/1262000/CountryCodeService2/getCountryCodeList2"
qs = "?" + urlencode(
    {
        'ServiceKey': unquote('<DATA API KEY>'),
        'pageNo': 1,
        'numOfRows': 250,
    }
)

response = requests.get(url + qs)
#
url_status = f"http://apis.data.go.kr/1262000/CountryOverseasArrivalsService/getCountryOverseasArrivalsList?serviceKey=<DATA API KEY>/"
response_status = requests.request("GET", url_status)
#
region_code = []
fest_json = []

file_path = "./iso4217.json"
cnt = 1
with open(file_path, 'r') as file:
    # iso4217 파일 열기
    er_code_list = json.load(file)
    for res in response.json()['data']:
        new_dict = {}
        rs = list(res.values())
        # 모델 데이터 json 파일 작성
        new_dict['model'] = 'country.country'
        new_dict['fields'] = {}
        if not rs[0]:
            continue
        new_dict['fields']['country_name'] = rs[0]
        new_dict['fields']['country_code'] = rs[1]
        query = new_dict['fields']['country_code']
        for er in er_code_list:
            if er['ISO3166-1-Alpha-2'] == query:
                new_dict['fields']['exchange_rate_code'] = er['ISO4217-currency_alphabetic_code']
                break
            else:
                new_dict['fields']['exchange_rate_code'] = None

# er_url = f"https://www.koreaexim.go.kr/site/program/financial/exchangeJSON?authkey=<API KEY>&searchdate={td}&data=AP01"

# er_headers = {
#     'Cookie': 'JSESSIONID=h95M11w7C1PjUxXq13Ka43ztoEUzD5UCG23pN5axk1G3kbyfbEwwZFam3CUnKtFV.aG1wX2RvbWfPbi9ITVBNUzEy; WMO
# }
#
# er_response = requests.request("GET", er_url, headers=er_headers)
# for e in er_response.json():
#     print(e['cur_unit'], new_dict['fields']['exchange_rate_code'])
#     if e['cur_unit'] == new_dict['fields']['exchange_rate_code']:
#         new_dict['fields']['ysd_er'] = e['kftc_deal_bas_r']
#         new_dict['fields']['cur_nm'] = e['cur_nm']
#         break

status_url = f"http://apis.data.go.kr/1262000/CountryOverseasArrivalsService/getCountryOverseasArrivalsList?serviceKey=Eenkiy"
status_response = requests.request("GET", status_url)
# print(status_response.json()['data'][0]['txt_origin_cn'])
try:
    new_dict['fields']['status'] = status_response.json()['data'][0]['txt_origin_cn']
    new_dict['fields']['status_wrt_dt'] = status_response.json()['data'][0]['wrt_dt']
except:
    pass

fest_response = requests.get(

```

```

url="https://api.predicthq.com/v1/events/",
headers={
    "Authorization": "Bearer <predicthq api key>",
    "Accept": "application/json"
},
params={
    "category": "public-holidays,festivals,concerts",
    "start.gte": "2022-12-02",
    "country": f"{query}"
}
)
<-----
# 이 부분 오류 있을듯?
try:
    for festival in fest_response.json()['results']:
        fest_dict = {}
        fest_dict['model'] = 'country.event'
        fest_dict['fields'] = {}
        fest_dict['fields']['country_code'] = query
        fest_dict['fields']['title'] = festival['title']
        fest_dict['fields']['description'] = festival['description']
        fest_dict['fields']['category'] = festival['category']
        fest_dict['fields']['st_dt'] = festival['start'][:10]
        fest_dict['fields']['ed_dt'] = festival['end'][:10]
        fest_dict['location'] = festival['location']
        fest_json.append(fest_dict)
        # new_dict['fields']['events'].append(fest_dict)
except:
    pass
< -----
permission_url = "http://apis.data.go.kr/1262000/EntranceVisaService2/getEntranceVisaList2?serviceKey=<data api key>%2F0BWaoU5
permission_response = requests.request("GET", permission_url)

new_dict['fields']['visa'] = []
try:
    for permission in permission_response.json()['data']:
        permission_dict = {}
        permission_dict['have_yn'] = permission['have_yn']
        permission_dict['gnrl_pspt_visa_yn'] = permission['gnrl_pspt_visa_yn']
        permission_dict['gnrl_pspt_visa_cn'] = permission['gnrl_pspt_visa_cn']
        permission_dict['ofclpspt_visa_yn'] = permission['ofclpspt_visa_yn']
        permission_dict['ofclpspt_visa_cn'] = permission['ofclpspt_visa_cn']
        permission_dict['dplmt_pspt_visa_yn'] = permission['dplmt_pspt_visa_yn']
        permission_dict['nvisa_entry_evdc_cn'] = permission['nvisa_entry_evdc_cn']
        permission_dict['remark'] = permission['remark']
        new_dict['fields']['visa'].append(permission_dict)

except:
    pass

region_code.append(new_dict)
print(cnt, len(response.json()['data']))
cnt += 1
if cnt == 20:
    break

with open('country_test2.json', 'w', encoding='utf-8') as make_file:
    json.dump(region_code, make_file, ensure_ascii=False, indent=2)

```

```

import json
import requests
# events 를 제외한 db 작성 후 실행
file_path = "./country_test.json"

currency_set = set()

fest_json = []

with open(file_path, 'rt', encoding='UTF-8') as file:
    countries = json.load(file)
    cnt = 1
    for country in countries:
        q = country['fields']['country_code']
        fest_dict = {}
        fest_dict['model'] = 'countries.event'
        fest_dict['fields'] = {}
        fest_response = requests.get(
            url="https://api.predicthq.com/v1/events/",
            headers={
                "Authorization": "Bearer <API KEY>",
                "Accept": "application/json"
            },
            params={

```

```

"category": "public-holidays, festivals, concerts",
"start.gte": "2022-12-02",
# "category": "public-holidays, festivals, concerts",
"country": f"{q}"
})
try:
    for festival in fest_response.json()['results']:
        fest_dict['fields']['country_code'] = q
        fest_dict['fields']['title'] = festival['title']
        fest_dict['fields']['description'] = festival['description']
        fest_dict['fields']['category'] = festival['category']
        fest_dict['fields']['st_dt'] = festival['start'][:10]
        fest_dict['fields']['ed_dt'] = festival['end'][:10]
        fest_dict['location'] = festival['location']
        fest_json.append(fest_dict)
except:
    pass
print(cnt)
cnt += 1

with open('event_test.json', 'w', encoding='utf-8') as make_file:
    json.dump(fest_json, make_file, ensure_ascii=False, indent=2)

```