Python

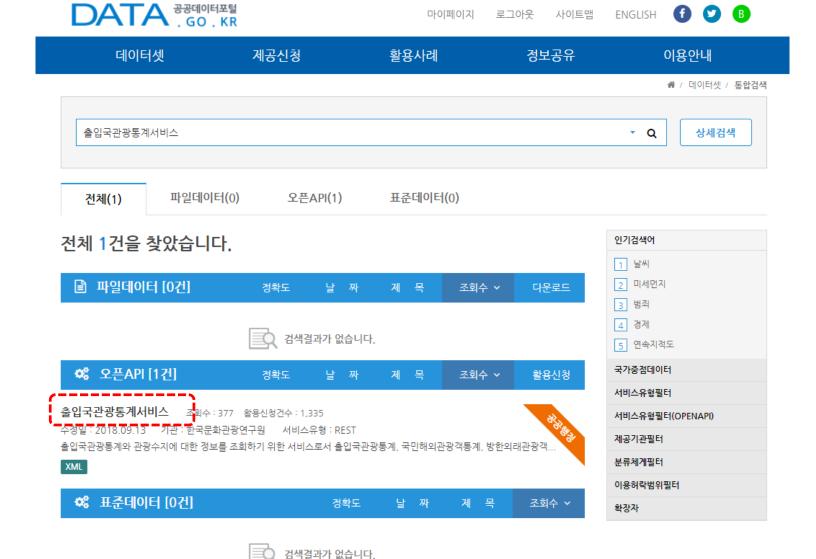
공공데이터수집하기

참고문헌: 파이썬을 이용한 빅데이터 수집, 분석과 시각화 - 비팬북스, 이원화

공공데이터 포털 가입

- 공공데이터포털 : http://www.data.go.kr
- 일반 회원으로 가입하여 개발계정 인증키를 발급받는다.
- 데이터 이용: 외국인의 입국 숫자에 따른 관광지 입장 수

출입국관광통계서비스





출입국관광통계서비스

M ENGLISH

출입국관광통계와 관광수지에 대한 정보를 조회하기 위한 서비스로서 출입국관광통계, 국민해외관광객통계, 방한외래관광객평균체재일,관광수지 등을 조회할 수 있다.

활용신청 (바로가기) 건수: 1,333

※ 서비스 오류가 있을지 오류신고 버튼을 이용해주세요.

XML

출입국관광통계서비스

활용신청 ○ 달기

오류신고

출입국관광통계서비스						
End Point	http://openapi.tour.go.kr/openapi/se	스크랩				
데이터포맷	XML	API 유형	REST			
비용부과유무	무료	활용신청 건수	1333			

.....

.....

참고문서	한국문화관광연구원 출입국관광통계서비	미리보기	
관리부서명	통계평가센터,통계평가센터	관리부서 전화번호	02-2669-8473

Q 연관데이터셋

오픈 API	청양통계	충청남도 청양군	활용신청 건수(바로가기 횟수) : 17
오픈 API	부산시 통계정보	부산광역시	활용신청 건수(바로가기 횟수) : 33

마이페이지

♠ > OPEN API > 개발계정 신청

개발계정 신청



오픈API

- 개발계정

- 활용현황

- 운영계정

- 인증키 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

> 기본정보

서비스명	출입국관광통계서비스	서비스 유형	REST
심의여부	자동승인	신청유형	개발계정 활용신청
처리상태	신청	활용기간	승인일로부터 24개월 간 활용가능

▶ 시스템유형 ★ 시스템 유형 ★ 시스템 유형 ★ 시스템 유형 ★ 서비 구축 ★ 서비 구축: OpenAPI 서비스를 호출하여 응답받은 결과값을 서비에 저장하지 않고 사용할 경우 (서비 미구축) ★ 서비 구축: OpenAPI 서비스를 호출하여 응답받은 결과값을 서비에 저장하거나 DB화 하여 사용할 경우

> 상세기능정보 필수 입력 정보입니다.

*자동승인 상세기능은 신청과 동시에 활용 가능합니다.

		상세기능	설명	일일 트래픽
		관광수지조회	월별 관광 수입과 지출, 1인당 평균 소비액을 제공하는 기능	1000
		방한외래관광객평균체재 일조회	월별, 국적의 검색 조건에 따라 방한외래관광객 평균체재일수를 제공하는 기능	1000
		방한외래관광객통계조회	월별, 국적, 성별, 연령대, 여행목적, 입국항의 검색 조건에 따라 방한외 래관광객수를 제공하는 기능	1000
2		국민해외관광객통계조회	월별, 성별, 연령대, 출국항의 검색 조건에 따라 국민해외관광객수를 제 공하는 기능	1000
Ť	•	출입국관광통계조회	기간, 국가의 검색조건에 따라 관광출입국자수를 제공하는 기능	1000

> 라이센스표시

저작자표시 이용허락범위 (사유:) **⊘**동의합니다



System Message

개발계정 신청이 완료되었습니다. '마이페이지>OPEN API>개발계정'에서 신청내 역을 확인하실 수 있습니다.

확인

마이페이지

오픈API - 개발계정 - 활용현황 - 운영계정 - 인증키 발급현황 DATA 나의 문의 나의 관심 회원정보 공공데이터 제공신청 ☆ > 마이페이지 > OPEN API > 개발계정

개발계정





상세검색 🔽

^읍총1건



신청일:2018-12-20 [활용신청] 만료예정:2020-12-20

서비스유형:REST 분류:일반공공행정 > 정부자원관리

제공기관:한국문화관광연구원

« < 1 > »

오픈API - 개발계정 - 활용현황 - 운영계정 - 인증키 발급현황 DATA 나의 문의 나의 관심 회원정보 공공데이터 제공신청

					_	
•	운영계정 신청	▶ 면장 신청	▶ 변경 신청	신청 ▶ 일반 인증5	기 받기 → 목록	
> 기본정보						
서비스명	출입국관광통기	계서비스 상세설명				
서비스 유형	REST	일일트래픽	0	평균응답속도(초)	0	
심의여부	자동승인	신청유형	개발계정 활용신청	처리상태	승인	
활용기간	2018-12-20	3-12-20 ~ 2020-12-20				
> 서비스정보						
일반 인증키 (UTF-8)						
End Point	http://	http://openapi.tour.go.kr/openapi/service				
데이터포맷	XML					
참고문서	한국문	화관광연구원 출입국	관광통계서비스 활용가이드	_v1.1.docx		

> 상세기능정보 개발가이드 활용제 미리보기 일일 트래픽 상세기능 설명 심의결과 NO 한 여부 다운로드 기간, 국가의 검색조건에 따라 관광출 출입국관광통계조회 1000 승인 실행 1 입국자수를 제공하는 기능

> 서비스정보

참고문서

오픈API > - 개발계정 - 활용현황 - 운영계정 - 인증키 발급현황 DATA 나의 문의 나의 문의

공공데이터 제공신청

회원정보

→ 5	운영계정 신청 📗 > 연	장 신청 →	변경 신청 🕒 중지 신	청 ♪ 일반 인증키 기	재발급	
> 기본정보						
서비스명	출입국관광통계서비	스 상세설명				
서비스 유형	REST	일일트래픽	0	평균응답속도(초)	0	
심의여부	자동승인	신청유형	개발계정 활용신청	처리상태	승인	
활용기간	2018-12-20 ~ 202	0-12-20				

일반 인증키 (UTF-8)	VM1liwkP6fkbqeg%2BBNrwMhb8nTeo5R%2FlfQOO73WkDxDu%2Fj9x1RGmqqJA%2BH eNKVcc%2BeWUdbihGCvrcmNx%2BzcNxA%3D%3D
End Point	http://openapi.tour.go.kr/openapi/service
데이터포맷	XMI

한국문화관광연구원 출입국관광통계서비스 활용가이드_v1.1.docx

▶ 상세기능정보						
NO	상세기능	시대	활용제 한 여부	일일 트래픽	심의결과	미리보기 다운로드
1	출입국관광통계조회	기간, 국가의 검색조건에 따라 관광출 입국자수를 제공하는 기능	-	1000	승인	실행

마이페이지

▲ > 마이페이지 > OPEN API > 개발계정

개발계정



오픈API

- 개발계정
- 활용현황
- 운영계정
- 인증키 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

활용자가 개발계정을 신청중인 단계

• 반려 0건 신청

• 보류 0건 0건

활용자의 개발계정신청이 승인되어 활용증인 단계

활용 • 반려 0건 2건

· 변경신청

·보류 0건

0건

활용자가 중지신청하여 운영이 중지된 단계

중지

0건

상세검색 🔽

ॿ 총 2 건

[승인] 관광자원통계서비스

서비스유형 :REST

분류 :일반공공행정 > 정부자원관리

신청일:2018-12-20 [활용신청] 만료예정:2020-12-20

제공기관 :한국문화관광연구원

[승인] 출입국관광통계서비스

신청일:2018-12-20 [활용신청] 만료예정:2020-12-20

서비스유형:REST 분류:일반공공행정 > 정부자원관리 제공기관:한국문화관광연구원

« < 1 > »

유료관광지방문객수조회

기간별 외국인방문객수와 내국인방문객수, 관광지의 주소 정보, 지역코드 정보를 제공하는 기능

- · 활용승인 절차 개발계정: 자동승인 / 운영계정: 자동승인
- · 신청가능 트래픽 100000 / 운영계정은 활용사례 등록시 신청하면 트래픽 증가 가능
- ·요청주소 http://openapi.tour.go.kr/openapi/service/TourismResourceStatsService/getPchrgTrrsrtVisitorList
- · 서비스URL



요청변수(Request Parameter)

항목명(국문)	항목명(영문)	항목크기	항목구분	샘플데이터	항목설명
년월	YM	6	필수	201201	년월
시도	SIDO	4000	옵션	부산광역시	시도
시군구	GUNGU	4000	옵션	해운대구	시군구
관광지	RES_NM	100	옵션	부산시립미술관	관광지

▶ 출력결과(Response Element)

항목명(국문)	항목명(영문)	항목크기	항목구분	샘플데이터	항목설명
결과코드	resultCode	4	필수	0000	결과코드
결과메시지	resultMsg	50	필수	ОК	결과메시지
한 페이지 결과 수	numOfRows	2	0	10	한 페이지 결과 수
페이지 번호	pageNo	5	0	1	페이지 번호
전체 결과 수	totalCount	7	0	12334	전체 결과 수
지역코드	addrCd	4		2635	지역코드
외국인방문객 수	csForCnt	6	0	1824	외국인방문객수
내국인방문객 수	csNatCnt	6	0	33367	내국인방문객수
시군구	gungu	4000	필수	해운대구	시군구
관광지	resNm	100	필수	부산시립	관광지
시도	sido	4000	필수	부산광역시	시도
년월	ym	6	필수	200409	년월
결과값 연번	rnum		필수	1	결과값 나열 순서

```
import os
   import sys
   import urllib.request
 4 | import datetime
   import time
6 import ison
    import math
    def get_request_url(url):
13
        req = urllib.request.Request(url)
14
15
       try:
16
            response = urllib.request.urlopen(req)
            if response.getcode() = 200:
                print ("[%s] Url Request Success" % datetime.datetime.now())
18
                return response.read().decode('utf-8')
19
20
        except Exception as e:
21
            print(e)
22
            print("[%s] Error for URL : %s" % (datetime.datetime.now(), url))
23
            return None
```

11

20

21

24 25

26

28 29

31 32

35

36

```
#[CODE 1]
    def getTourPointVisitor(yyyymm, sido, gungu, nPagenum, nItems):
        end point = "http://openapi.tour.go.kr/openapi/service/TourismResourceStatsService/getPchrgTrrsrtVisitorList"
        parameters = "?_type=json&serviceKey=" + access_key
        parameters += "&YM=" + vvvvmm
        parameters += "&SIDO=" + urllib.parse.guote(sido)
        parameters += "&GUNGU=" + urllib.parse.guote(gungu)
        parameters += "&RES NM=&pageNo=" + str(nPagenum)
        parameters += "&numOfRows=" + str(nItems)
12
13
        url = end point + parameters
14
15
        retData = get request url(url)
16
        if (retData = None):
18
            return None
19
        else:
            return ison.loads(retData)
   #[CODE 2]
    def getTourPointData(item, yyyymm, jsonResult):
        addrCd = 0 if 'addrCd' not in item.keys() else item['addrCd']
        gungu = '' if 'gungu' not in item.keys() else item['gungu']
        sido = '' if 'sido' not in item.keys() else item['sido']
        resNm = '' if 'resNm' not in item.keys() else item['resNm']
        rnum = 0 if 'rnum' not in item.keys() else item['rnum']
        ForNum = 0 if 'csForCnt' not in item.keys() else item['csForCnt']
        NatNum = 0 if 'csNatCnt' not in item.kevs() else item['csNatCnt']
        jsonResult.append({'yyyymm': yyyymm, 'addrCd': addrCd,
34
                        'gungu': gungu, 'sido': sido, 'resNm': resNm,
                        'rnum': rnum, 'ForNum': ForNum, 'NatNum': NatNum})
        return
37
```

```
def main():
 2
       isonResult = []
 4
 5
       sido = '서울특별시'
       gungu = ''
 6
       nPagenum = 1
       nTotal = 0
9
       nltems = 100
10
11
       nStartYear = 2011
       nEndYear = 2019
12
13
14
       for year in range(nStartYear, nEndYear):
15
            for month in range(1, 13):
16
               yyyymm = "{0}{1:0>2}".format(str(year), str(month))
17
               nPagenum = 1
19
               while True:
21
                    jsonData = getTourPointVisitor(yyyymm, sido, gungu, nPagenum, nItems)
23
                    if (jsonData['response']['header']['resultMsg'] = 'OK'):
24
                       nTotal = [sonData['response']['body']['totalCount']
25
26
                        if nTotal = 0:
27
                           break
                        for item in jsonData['response']['body']['items']['item']:
                           getTourPointData(item, yyyymm, jsonResult)
31
                       nPage = math.ceil(nTotal / 100)
                        if (nPagenum == nPage):
34
                           break
35
36
                       nPagenum += 1
                   else:
                        break
40
       with open('%s_관광지입장정보_%d_%d.json' % (sido, nStartYear, nEndYear-1), 'w', encoding='utf8') as outfile:
41
42
           retJson = json.dumps(jsonResult,indent=4, sort keys=True, ensure ascii=False)
           outfile.write(retJson)
43
44
       print ('%s_관광지입장정보_%d_%d.json SAYED' % (sido, nStartYear, nEndYear-1))
45
```

```
if __name__ = '__main__':
        main()
[2018-12-20 22:21:03.525624] Url Reauest Success
[2018-12-20 22:21:03.771358] Url Request Success
[2018-12-20 22:21:04.029859] Url Request Success
[2018-12-20 22:21:04.295045] Url Request Success
[2018-12-20 22:21:04.551682] Url Request Success
[2018-12-20 22:21:04.810408] Url Request Success
[2018-12-20 22:21:05.093090] Url Reguest Success
[2018-12-20 22:21:05.351705] Url Reguest Success
[2018-12-20 22:21:05.624242] Url Reguest Success
[2018-12-20 22:21:05.874234] Url Request Success
[2018-12-20 22:21:06.128480] Url Reguest Success
[2018-12-20 22:21:06.395198] Url Reauest Success
[2018-12-20 22:21:06.661475] Url Reguest Success
[2018-12-20 22:21:06.912319] Url Request Success
[2018-12-20 22:21:07.172973] Url Request Success
[2018-12-20 22:21:21.204802] Url Reguest Success
[2018-12-20 22:21:21.236039] Url Request Success
[2018-12-20 22:21:21.282904] Url Request Success
[2018-12-20 22:21:21.314069] Url Request Success
[2018-12-20 22:21:21.345292] Url Reguest Success
서울특별시 관광지입장정보 2011 2018.json SAVED
```

출입국관광통계서비스

기술문서: 한국문화관광연구원 출입국관광통계서비스 활용가이드_v1.1.docx

웹서비스 클라이언트 개발가이드(J2EE) 웹서비스 클라이언트 개발가이드(.NET)

• 관광수지조회

· 방한외래관광객평균체재일조회

· 방한외래관광객통계조회

· 국민해외관광객통계조회

· 출입국관광통계조회

출입국관광통계조회

기간, 국가의 검색조건에 따라 관광출입국자수를 제공하는 기능

- · 활용승인 절차 개발계정: 자동승인 / 운영계정: 자동승인
- · 신청가능 트래픽 100000 / 운영계정은 활용사례 등록시 신청하면 트래픽 증가 가능
- ·요청주소 http://openapi.tour.go.kr/openapi/service/EdrcntTourismStatsService/getEdrcntTourismStatsList
- ·서비스URL





요청변수(Request Parameter)

항목명(국문)	항목명(영문)	항목크기	항목구분	샘플데이터	항목설명
연월	YM	12	필수	201201	연월
국가코드	NAT_CD	6	옵션	112	국가코드
출입국구분코 드	ED_CD	2	옵션	D	출입국구분코드

▶ 출력결과(Response Element)

항목명(국문)	항목명(영문)	항목크기	항목구분	샘플데이터	항목설명
결과코드	resultCode	4	필수	0000	결과코드
결과메시지	resultMsg	50	필수	ОК	결과메시지
한 페이지 결과 수	numOfRows	2	옵션	10	한 페이지 결과 수
페이지 번호	pageNo	5	옵션	1	페이지 번호
전체 결과 수	totalCount	7	옵션	12334	전체 결과 수
출입국 구분	ed	14	필수	방한외래관광객	출입국구분
출입국 구분코 드	edCd	2	필수	Е	출입국 구분코드
국가코드	natCd	6	필수	112	국가코드
국가	natKorNm	80	필수	중국	국가
출입국자수	num	10	옵션	179508	출입국자수
연월	ym	12	필수	201206	연월
결과값 연번	rnum	2	필수	1	결과값 나열순서

```
import os
   import sys
   import urllib.request
   import datetime
   import time
   import json
   import matplotlib.pyplot as plt
   import matplotlib
   from matplotlib import font_manager, rc
   access_key="YM+1+wkP6fkbqeg%2BBNnwMinb3nTeb5R%2F1fQ0073WkBxDa%2Fj9xtRamqqJA%2BHeNK4cc%2BeWUdbthG6vrcmNx%2BzcNxA%3B%3D"
   def get request url(url):
15
16
       req = urllib.request.Request(url)
18
       try:
           response = urllib.request.urlopen(req)
19
            if response.getcode() = 200:
               print ("[%s] Url Request Success" % datetime.datetime.now())
21
               return response.read().decode('utf-8')
22
23
       except Exception as e:
24
           print(e)
25
           print("[%s] Error for URL : %s" % (datetime.datetime.now(), url))
26
           return None
```

```
#[CODE 1]
   def getNatVisitor(yyyymm, nat_cd, ed_cd):
       end point = "http://openapi.tour.go.kr/openapi/service/EdrcntTourismStatsService/getEdrcntTourismStatsList"
       parameters = "?_type=json&serviceKey=" + access_key
       parameters += "&YM=" + yyyymm
       parameters += "&NAT_CD=" + nat_cd
       parameters += "&ED_CD=" + ed_cd
10
11
       url = end_point + parameters
12
       retData = get_request_url(url)
13
14
15
       if (retData = None):
16
            return None
17
       else:
18
            return json.loads(retData)
```

```
def main():
    isonResult = []
    #중국: 112 / 일본: 130 / 미국: 275
    national code = "275"
    ed cd = "E"
    nStartYear = 2011
    nEndYear = 2017
    for year in range(nStartYear, nEndYear):
        for month in range(1, 13):
            yyyymm = "{0}{1:0>2}".format(str(year), str(month))
            jsonData = getNatVisitor(yyyymm, national code, ed cd)
            print (ison.dumps(isonData,indent=4, sort keys=True,ensure ascii=False))
            if (isonData['response']['header']['resultMsg'] = 'OK'):
                krName = jsonData['response']['body']['items']['item']["natKorNm"]
                krName = krName.replace(' ', '')
                iTotalVisit = jsonData['response']['body']['items']['item']["num"]
                print('%s_%s : %s' %(krName, yyyymm, iTotalVisit))
                isonResult.append({'nat name': krName, 'nat_cd': national_code,'yyyymm': yyyymm, 'visit_cnt': iTotalVisit})
    cnVisit = []
    VisitYM = []
    index = []
    i = 0
    for item in isonResult:
        index.append(i)
        cnVisit.append(item['visit_cnt'])
        VisitYM.append(item['vvvvmm'])
        i = i + 1
    with open('%s(%s)_해외방문객정보_%d_%d.json' % (krName, national_code, nStartYear, nEndYear-1), 'w', encoding='utf8') as outfile:
        retJson = json.dumps(jsonResult,indent=4, sort keys=True,ensure ascii=False)
        outfile.write(retJson)
```

8

10

11 12

13

14

15 16 17

18 19

21

24 25

26 27

29

34 35

36

37

40

```
41
42
       #[CODE 2]
43
       font_location = "c:/Windows/fonts/malgun.ttf"
44
       font_name = font_manager.FontProperties(fname=font_location).get_name()
       matplotlib.rc('font', family=font_name)
45
46
       plt.xticks(index, VisitYM)
47
48
       plt.plot(index, cnVisit)
       plt.xlabel('방문월')
49
       plt.ylabel('방문객수')
50
51
       plt.grid(True)
52
       plt.show()
53
54
    if __name__ = '__main__':
55
       main()
56
```

```
[2018-12-20 22:33:15.646995] Url Reguest Success
   "response": {
       "body": {
           "items": {
               "item": {
                   "ed": "방한외래관광객",
                   "edCd": "E",
                   "natCd": 275,
                   "natKorNm": "미 국",
                   "num": 44152,
                   "rnum": 1.
                   "ym": 201101
           "numOfRows": 10.
           "pageNo": 1,
           "totalCount": 1
       },
```

