
Python

공공데이터수집하기

김선녕(sykim.lecture@gmail.com)

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

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

데이터셋

제공신청

활용사례

정보공유

이용안내

데이터셋 / 통합검색

출입국관광통계서비스



상세검색

전체(1)

파일데이터(0)

오픈API(1)

표준데이터(0)

전체 1건을 찾았습니다.

파일데이터 [0건]

정확도

날 짜

제 목

조회수 ▾

다운로드



검색결과가 없습니다.

오픈API [1건]

정확도

날 짜

제 목

조회수 ▾

활용신청

출입국관광통계서비스

조회수 : 377

활용신청건수 : 1,335

수정일 : 2018.09.13

기관 : 한국문화관광연구원

서비스유형 : REST

출입국관광통계와 관광수지에 대한 정보를 조회하기 위한 서비스로서 출입국관광통계, 국민해외관광통계, 방한외래관광객...

XML

공공행정

표준데이터 [0건]

정확도

날 짜

제 목

조회수 ▾



검색결과가 없습니다.

인기검색어

1 날씨

2 미세먼지

3 범죄

4 경제

5 연속지적도

국가중점데이터

서비스유행필터

서비스유행필터(OPENAPI)

제공기관필터

분류체계필터

이용허락범위필터

확장자



공공행정

·제공기관

한국문화관광연구원

·관리부서명

통계평가센터,통계평가센터

·관리부서 전화번호

02-2669-8473

·등록일

2012-12-06

·키워드

·보유근거

행정업무

·수집방법

정보시스템

출입국관광통계서비스

ENGLISH

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

활용신청 (바로그기) 건수 : 1,333

※ 서비스 오류가 있을시 오류신고 버튼을 이용해주세요.

XML

출입국관광통계서비스

활용신청

닫기

오류신고

출입국관광통계서비스

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

참고문서

한국문화관광연구원 출입국관광통계서비스 활용가이드_v1.1.docx

미리보기

관리부서명	통계평가센터,통계평가센터	관리부서 전화번호	02-2669-8473
-------	---------------	-----------	--------------

연관데이터셋

<div>오픈 API</div>	<div>청양통계</div>	<div>충청남도 청양군</div>	<div>활용신청 건수(바로그기 횟수) : 17</div>
<div>오픈 API</div>	<div>부산시 통계정보</div>	<div>부산광역시</div>	<div>활용신청 건수(바로그기 횟수) : 33</div>

마이페이지

오픈API

- 개발계정
- 활용현황
- 운영계정
- 인증키 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

OPEN API > 개발계정 신청

개발계정 신청



> 기본정보

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

> 시스템유형

1

☒ 일반 ☐ 서버 구축

* 시스템 유형

- 일반 : OpenAPI 서비스를 호출하여 응답받은 결과값을 서버에 저장하지 않고 사용할 경우 (서버 미구축)

* 서버 구축 : OpenAPI 서비스를 호출하여 응답받은 결과값을 서버에 저장하거나 DB화 하여 사용할 경우

> 상세기능정보 필수 입력 정보입니다.

*자동승인 상세기능은 신청과 동시에 활용 가능합니다.

<input type="checkbox"/>	상세기능	설명	일일 트래픽
<input type="checkbox"/>	관광수지조회	월별 관광 수입과 지출, 1인당 평균 소비액을 제공하는 기능	1000
<input type="checkbox"/>	방한외래관광객평균체재 일조회	월별, 국적의 검색 조건에 따라 방한외래관광객 평균체재일수를 제공하는 기능	1000
<input type="checkbox"/>	방한외래관광객통계조회	월별, 국적, 성별, 연령대, 여행목적, 입국항의 검색 조건에 따라 방한외래관광객수를 제공하는 기능	1000
<input type="checkbox"/>	국민해외관광객통계조회	월별, 성별, 연령대, 출국항의 검색 조건에 따라 국민해외관광객수를 제공하는 기능	1000
<input checked="" type="checkbox"/>	출입국관광통계조회	기간, 국가의 검색조건에 따라 관광출입국자수를 제공하는 기능	1000

2

> 라이선스표시

이용허락범위	저작권표시	(사유 :) <input checked="" type="checkbox"/> 동의합니다
--------	-------	---

3

(사유 :)

4

▶ 신청

▶ 취소

System Message

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

확인

마이페이지

오픈API >

- 개발계정
- 활용현황
- 운영계정
- 인증키 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

홈 > 마이페이지 > OPEN API > 개발계정

개발계정



활용자가 개발계정을 신청중인 단계	활용자의 개발계정신청이 승인되어 활용중인 단계	활용자가 중지신청하여 운영이 중지된 단계
<div>신청</div> <div>0건</div>	<div>활용</div> <div>1건</div>	<div>중지</div> <div>0건</div>
<ul style="list-style-type: none"> · 반려 0건 · 보류 0건 	<ul style="list-style-type: none"> · 변경신청 0건 · 반려 0건 · 보류 0건 	

상세검색 ☐

총 1 건

[승인] 출입국관광통계서비스

신청일:2018-12-20 [활용신청] 만료예정 :2020-12-20

서비스유형 :REST 분류 :일반공공행정 > 정부자원관리 제공기관 :한국문화관광연구원

오픈API >

- 개발계정
- 활용현황
- 운영계정
- 인증키 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

[▶ 운영계정 신청](#)
[▶ 연장 신청](#)
[▶ 변경 신청](#)
[▶ 중지 신청](#)
[▶ 일반 인증키 받기](#)
[▶ 목록](#)

> 기본정보

서비스명	출입국관광통계서비스 상세설명				
서비스 유형	REST	일일트래픽	0	평균응답속도(초)	0
심의여부	자동승인	신청유형	개발계정 활용신청	처리상태	승인
활용기간	2018-12-20 ~ 2020-12-20				

> 서비스정보

일반 인증키 (UTF-8)	
End Point	http://openapi.tour.go.kr/openapi/service
데이터포맷	XML
참고문서	한국문화관광연구원 출입국관광통계서비스 활용가이드_v1.1.docx

> 상세기능정보

[개발가이드](#)

NO	상세기능	설명	활용제한 여부	일일 트래픽	심의결과	미리보기 다운로드
1	출입국관광통계조회	기간, 국가의 검색조건에 따라 관광출입국자수를 제공하는 기능	-	1000	승인	실행

오픈API

- 개발계정
- 활용현황
- 운영계정
- 인증키 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

▶ 운영계정 신청

▶ 연장 신청

▶ 변경 신청

▶ 중지 신청

▶ 일반 인증키 재발급

▶ 목록

> 기본정보

서비스명	출입국관광통계서비스 <small>상세설명</small>				
서비스 유형	REST	일일트래픽	0	평균응답속도(초)	0
심의여부	자동승인	신청유형	개발계정 활용신청	처리상태	승인
활용기간	2018-12-20 ~ 2020-12-20				

> 서비스정보

일반 인증키 (UTF-8)	VM1liwkP6fkbqeg%2BBNrwMhb8nTeo5R%2FIfQOO73WkDxDu%2Fj9x1RGmqqJA%2BH eNKVcc%2BeWUdbihGCvrcmNx%2BzcNx%3D%3D
End Point	http://openapi.tour.go.kr/openapi/service
데이터포맷	XML
참고문서	한국문화관광연구원 출입국관광통계서비스 활용가이드_v1.1.docx

> 상세기능정보

개발가이드

NO	상세기능	설명	활용제한 여부	일일 트래픽	심의결과	미리보기 다운로드
1	출입국관광통계조회	기간, 국가의 검색조건에 따라 관광출입국자수를 제공하는 기능	-	1000	승인	실행

마이페이지

오픈API >

- 개발계정
- 활용현황
- 운영계정
- 인증기 발급현황

DATA

나의 문의

나의 관심

회원정보

공공데이터 제공신청

홈 > 마이페이지 > OPEN API > 개발계정

개발계정



활용자가 개발계정을 신청중인 단계	활용자의 개발계정신청이 승인되어 활용중인 단계	활용자가 중지신청하여 운영이 중지된 단계
<div>신청</div> <div>0건</div> <ul style="list-style-type: none"> · 반려 0건 · 보류 0건 	<div>활용</div> <div>2건</div> <ul style="list-style-type: none"> · 변경신청 0건 · 반려 0건 · 보류 0건 	<div>중지</div> <div>0건</div>

상세검색 ☐

총 2 건

[승인] 관광자원통계서비스

추가

신청일:2018-12-20 [활용신청] 만료예정 :2020-12-20

서비스유형 :REST

분류 :일반공공행정 > 정부자원관리

제공기관 :한국문화관광연구원

[승인] 출입국관광통계서비스

신청일:2018-12-20 [활용신청] 만료예정 :2020-12-20

서비스유형 :REST

분류 :일반공공행정 > 정부자원관리

제공기관 :한국문화관광연구원

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

[활용신청](#)
[뒤로가기](#)

> 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	OK	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

```
1 import os
2 import sys
3 import urllib.request
4 import datetime
5 import time
6 import json
7 import math
8
9 access_key="VM1HhWkPGfKbqeg%2BDHwMhbsCmTeo5F%2Ff@0078WkBXDu%2Fj0x1fQmqqJA%2BHcNKYce%2BdellUdbHhGOvrcmX%2BzcnXA%8B%3D"
10
11 def get_request_url(url):
12
13     req = urllib.request.Request(url)
14
15     try:
16         response = urllib.request.urlopen(req)
17         if response.getcode() == 200:
18             print("[%s] URL Request Success" % datetime.datetime.now())
19             return response.read().decode('utf-8')
20     except Exception as e:
21         print(e)
22         print("[%s] Error for URL : %s" % (datetime.datetime.now(), url))
23         return None
```

```
1  #[CODE 1]
2  def getTourPointVisitor(yyyyymm, sido, gungu, nPagenum, nItems):
3
4      end_point = "http://openapi.tour.go.kr/openapi/service/TourismResourceStatsService/getPchrgTrrsrtVisitorList"
5
6      parameters = "?_type=json&serviceKey=" + access_key
7      parameters += "&YM=" + yyyyymm
8      parameters += "&SIDO=" + urllib.parse.quote(sido)
9      parameters += "&GUNGU=" + urllib.parse.quote(gungu)
10     parameters += "&RES_NM=&pageNo=" + str(nPagenum)
11     parameters += "&numOfRows=" + str(nItems)
12
13     url = end_point + parameters
14
15     retData = get_request_url(url)
16
17     if (retData == None):
18         return None
19     else:
20         return json.loads(retData)
21
22  #[CODE 2]
23  def getTourPointData(item, yyyyymm, jsonResult):
24
25     addrCd = 0 if 'addrCd' not in item.keys() else item['addrCd']
26     gungu = '' if 'gungu' not in item.keys() else item['gungu']
27     sido = '' if 'sido' not in item.keys() else item['sido']
28     resNm = '' if 'resNm' not in item.keys() else item['resNm']
29     rnum = 0 if 'rnum' not in item.keys() else item['rnum']
30     ForNum = 0 if 'csForCnt' not in item.keys() else item['csForCnt']
31     NatNum = 0 if 'csNatCnt' not in item.keys() else item['csNatCnt']
32
33     jsonResult.append({'yyyyymm': yyyyymm, 'addrCd': addrCd,
34                       'gungu': gungu, 'sido': sido, 'resNm': resNm,
35                       'rnum': rnum, 'ForNum': ForNum, 'NatNum': NatNum})
36
37     return
```

```

1 def main():
2
3     jsonResult = []
4
5     sido = '서울특별시'
6     gungu = ''
7     nPagenum = 1
8     nTotal = 0
9     nItems = 100
10
11     nStartYear = 2011
12     nEndYear = 2019
13
14     for year in range(nStartYear, nEndYear):
15         for month in range(1, 13):
16
17             yyyyymm = "{0}{1:0>2}".format(str(year), str(month))
18             nPagenum = 1
19
20             while True:
21                 jsonData = getTourPointVisitor(yyyyymm, sido, gungu, nPagenum, nItems)
22
23                 if (jsonData['response']['header']['resultMsg'] == 'OK'):
24                     nTotal = jsonData['response']['body']['totalCount']
25
26                     if nTotal == 0:
27                         break
28
29                     for item in jsonData['response']['body']['items']['item']:
30                         getTourPointData(item, yyyyymm, jsonResult)
31
32                     nPage = math.ceil(nTotal / 100)
33                     if (nPagenum == nPage):
34                         break
35
36                     nPagenum += 1
37
38             else:
39                 break
40
41     with open('%s_관광지입장정보_%d_%d.json' % (sido, nStartYear, nEndYear-1), 'w', encoding='utf8') as outfile:
42         retJson = json.dumps(jsonResult, indent=4, sort_keys=True, ensure_ascii=False)
43         outfile.write(retJson)
44
45     print ('%s_관광지입장정보_%d_%d.json SAVED' % (sido, nStartYear, nEndYear-1))

```

```
1 if __name__ == '__main__':  
2     main()
```

```
[2018-12-20 22:21:03.525624] Url Request 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 Request Success  
[2018-12-20 22:21:05.351705] Url Request Success  
[2018-12-20 22:21:05.624242] Url Request Success  
[2018-12-20 22:21:05.874234] Url Request Success  
[2018-12-20 22:21:06.128480] Url Request Success  
[2018-12-20 22:21:06.395198] Url Request Success  
[2018-12-20 22:21:06.661475] Url Request 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 Request 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 Request 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

활용신청

뒤로가기

> 요청변수(Request Parameter)

항목명(국문)	항목명(영문)	항목크기	항목구분	샘플데이터	항목설명
연월	YM	12	필수	201201	연월
국가코드	NAT_CD	6	옵션	112	국가코드
출입국구분코드	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	OK	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 구분코드	edCd	2	필수	E	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

```
1 import os
2 import sys
3 import urllib.request
4 import datetime
5 import time
6 import json
7
8 import matplotlib.pyplot as plt
9 import matplotlib
10 from matplotlib import font_manager, rc
11
12 access_key="VM+I+WRP6fkbuqeg%2BBN+wmib8nTe55%2F+F00073WkBXDu%2F+9x+FRmiquA%2BHeNkY+cc%2BwUdb+lg6v+cmlx%2BzcNxA%3B%3D"
13
14 def get_request_url(url):
15
16     req = urllib.request.Request(url)
17
18     try:
19         response = urllib.request.urlopen(req)
20         if response.getcode() == 200:
21             print("[%s] URL Request Success" % datetime.datetime.now())
22             return response.read().decode('utf-8')
23     except Exception as e:
24         print(e)
25         print("[%s] Error for URL : %s" % (datetime.datetime.now(), url))
26         return None
27
```

```
1  #[CODE 1]
2  def getNatVisitor(yyyyymm, nat_cd, ed_cd):
3
4      end_point = "http://openapi.tour.go.kr/openapi/service/EdrcntTourismStatsService/getEdrcntTourismStatsList"
5
6      parameters = "?_type=json&serviceKey=" + access_key
7      parameters += "&YM=" + yyyyymm
8      parameters += "&NAT_CD=" + nat_cd
9      parameters += "&ED_CD=" + ed_cd
10
11     url = end_point + parameters
12
13     retData = get_request_url(url)
14
15     if (retData == None):
16         return None
17     else:
18         return json.loads(retData)
```

```

1 def main():
2
3     jsonResult = []
4
5     #중국: 112 / 일본: 130 / 미국: 275
6     national_code = "275"
7     ed_cd = "E"
8
9     nStartYear = 2011
10    nEndYear = 2017
11
12    for year in range(nStartYear, nEndYear):
13        for month in range(1, 13):
14
15            yyyyymm = "{0}{1:0>2}".format(str(year), str(month))
16
17            jsonData = getNatVisitor(yyyyymm, national_code, ed_cd)
18
19            print (json.dumps(jsonData, indent=4, sort_keys=True, ensure_ascii=False))
20
21            if (jsonData['response']['header']['resultMsg'] == 'OK'):
22                krName = jsonData['response']['body']['items']['item']['natKorNm"]
23                krName = krName.replace(' ', '')
24                iTotalVisit = jsonData['response']['body']['items']['item']['num"]
25                print('%s_%s : %s' %(krName, yyyyymm, iTotalVisit))
26                jsonResult.append({'nat_name': krName, 'nat_cd': national_code, 'yyyyymm': yyyyymm, 'visit_cnt': iTotalVisit})
27
28    cnVisit = []
29    VisitYM = []
30    index = []
31    i = 0
32    for item in jsonResult:
33        index.append(i)
34        cnVisit.append(item['visit_cnt'])
35        VisitYM.append(item['yyyyymm'])
36        i = i + 1
37
38    with open('%s(%s)_해외방문객정보_%d_%d.json' % (krName, national_code, nStartYear, nEndYear-1), 'w', encoding='utf8') as outfile:
39        retJson = json.dumps(jsonResult, indent=4, sort_keys=True, ensure_ascii=False)
40        outfile.write(retJson)

```

```

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

```

[2018-12-20 22:33:15.646995] Url Request 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
    },
  },
}

```

