



---

Python

# Naver API 연동

---

[참고문헌]

파이썬을 이용한 빅데이터 수집, 분석과 시각화  
비팬북스, 이원화지음

김선녕(sykim.lecture@gmail.com)

---

# Naver API등록- <https://developers.naver.com>

2

**NAVER** Developers Products Documents Application NAVER D2 Support API 상태

API 이용 안내	Clova	네이버 아이디로 로그인	지도	파파고	서비스 API
<b>API 소개</b>	Clova Platform	네이버 아이디로 로그인 API		Papago NMT 번역	데이터랩
운영 정책	Clova A.I. APIs	블로그		Papago SMT 번역	검색
FAQ		카페		언어 감지	단축URL
BI 가이드		캘린더		한글인명-로마자 변환	캡차
이용약관		적용 가이드			네이버 공유하기
상표사용 가이드		적용 사례			모바일앱 연동
					네이버 오픈메인

오픈소스 SW개발에 관심 있는 대학(원)생이라면 도전하세요!

프로젝트 제출 <http://d2campusfest.kr>  
19년 1월 3일(목) 23:59 까지

  
Clova

  
네이버 아이디로 로그인

  
Maps

  
papago

  
서비스 API

  
네이버 클라우드 플랫폼

Products > 서비스 API > 데이터랩

API 이용 안내

Clova

네이버 아이디로 로그인

지도

파파고

서비스 API

데이터랩

검색

단축URL

캡차

네이버 공유하기

모바일앱 연동

네이버 오픈메인

## 데이터랩 [↗](#)

[N](#) [트윗](#) [공유하기 1개](#)

### 통합검색어 트렌드 [↗](#)

통합검색어 트렌드는 네이버 통합검색에서 발생하는 검색어를 연령별, 성별, 기기별(PC, 모바일)로 세분화해서 조회할 수 있는 API입니다.

#### 분석하고 싶은 주제군을 설정합니다 [↗](#)

궁금한 주제어를 설정하고, 하위 주제어에 해당하는 검색어를 쉼표(,)로 구분해 입력합니다. 입력한 단어의 검색 추이를 하나로 합산해 해당 주제가 네이버에서 얼마나 검색되는지 조회할 수 있습니다.

예) 주제어 캠핑 : 캠핑, Camping, 캠핑용품, 겨울캠핑, 캠핑장, 글램핑, 오토캠핑, 캠핑카, 텐트, 캠핑요리

#### 세분화된 정보를 확인할 수 있습니다 [↗](#)

설정한 주제군에 대해 기간을 일간, 주간, 월간 단위로 조회할 수 있습니다. 그리고 5살 단위로 연령을 세분화해 조회할 수 있습니다. 연령을 5살 간격으로 설정한 이유는 연령 이외에 직업군 유추도 어느 정도 가능할 것이라 보기 때문입니다. 성별도 남성, 여성으로 세분화해 조회할 수 있습니다. 마지막으로 PC와 모바일을 각각 구분해서 조회할 수 있습니다.

#### 상대적 값으로만 제공됩니다 [↗](#)

검색어 트렌드는 요청된 기간 중 검색 횟수가 가장 높은 시점을 100으로 두고 나머지는 상대적 값으로 제공하고 있습니다. 검색 횟수의 절대값 제공은 아직 고려하고 있지 않습니다. 상대적 값만으로도 특정 주제, 검색어의 트렌드를 파악하는 데는 충분할 것입니다.

오픈 API 이용 신청

개발 가이드 보기

내 애플리케이션

애플리케이션 등록




CLOVA Platform Console  $\beta$

API 제휴 신청

계정 설정

## 애플리케이션 등록 (API 이용신청)

애플리케이션의 기본 정보를 등록하면, 좌측 **내 애플리케이션** 메뉴의 서브 메뉴에 등록하신 애플리케이션 이름으로 서브 메뉴가 만들어집니다.

애플리케이션 이름 $\leftrightarrow$	<div>Search </div> <p>네이버 아이디로 로그인할 때 사용자에게 표시되는 이름이므로 가급적 10자 이내의 간결한 이름을 사용해주세요.</p> <ul style="list-style-type: none"> <li>40자 이내의 영문, 한글, 숫자, 공백문자, "-", "_"만 입력 가능합니다.</li> </ul>
사용 API $\leftrightarrow$	<div> <div>선택하세요.  </div> <div> <div>검색</div> <div>네아로 (네이버 아이디로 로그인)</div> <div>단축 URL</div> <div>데이터랩 (검색어트렌드)</div> <div>데이터랩 (쇼핑인사이드)</div> <div>카페</div> <div>캘린더</div> <div>캡차 (음성)</div> <div>캡차 (이미지)</div> <div>한글인명-로마자변환</div> <div>Clova Face Recognition</div> <div>Papago 번역</div> <div>Papago 언어감지</div> </div> </div>

• [사용 API] 설정을 확인해 주세요.

내 애플리케이션

애플리케이션 등록












CLOVA Platform Console  $\beta$

API 제휴 신청

계정 설정

## 애플리케이션 등록 (API 이용신청)

애플리케이션의 기본 정보를 등록하면, 좌측 [내 애플리케이션](#) 메뉴의 서브 메뉴에 등록하신 애플리케이션 이름으로 서브 메뉴가 만들어집니다.

애플리케이션 이름 $\leftrightarrow$	<input type="text" value="Search"/>  <ul style="list-style-type: none"> <li>네이버 아이디로 로그인할 때 사용자에게 표시되는 이름이므로 가급적 10자 이내의 간결한 이름을 사용해주세요.</li> <li>40자 이내의 영문, 한글, 숫자, 공백문자, "-", "_" 만 입력 가능합니다.</li> </ul>				
사용 API $\leftrightarrow$	<div> <div>선택하세요. </div>  </div> <table> <tr> <td>검색</td> <td></td> </tr> <tr> <td>Clova Face Recognition</td> <td></td> </tr> </table>	검색		Clova Face Recognition	
검색					
Clova Face Recognition					
비로그인 오픈 API 서비스 환경	<div> <div>환경 추가 </div>  </div> <div> <div>Android 설정</div> <div>iOS 설정</div> <div>WEB 설정</div> </div>				

- [비로그인 오픈 API 서비스 환경] 설정을 확인해 주세요.

등록하기

취소

내 애플리케이션

애플리케이션 등록

CLOVA Platform Console  $\beta$

API 제휴 신청

계정 설정

애플리케이션 이름  $\text{↗}$

- 네이버 아이디로 로그인할 때 사용자에게 표시되는 이름이므로 가급적 10자 이내의 간결한 이름을 사용해주세요.
- 40자 이내의 영문, 한글, 숫자, 공백문자, "-", "\_"만 입력 가능합니다.

사용 API  $\text{↗}$

선택하세요.  $\blacktriangledown$   $\checkmark$

검색  $\times$

Clova Face Recognition  $\times$

비로그인 오픈 API  
서비스 환경

환경 추가  $\blacktriangledown$

WEB 설정  $\times$   $\wedge$

웹 서비스 URL (최대 10개)

http://localhost  $+$   $\checkmark$

- 텍스트 폼 우측 끝의 '+' 버튼을 누르면 행이 추가되며, '-' 버튼을 누르면 행이 삭제됩니다.
- http와 https는 구분하지 않습니다.
- www는 빼고 입력해 주세요. 예) http://naver.com
- 서브 도메인이 있으면 대표 도메인명만 입력해 주세요. (예: http://naver.com)
- 하이브리드 앱은 location.href 객체 출력 값을 입력하면 됩니다. (예: file://로컬 URI)

등록하기

취소

## 내 애플리케이션

FaceRecognition

crawler

crawler

Search

## 애플리케이션 등록

CLOVA Platform Console  $\beta$

API 제휴 신청

계정 설정

## Search

개요

API 설정

멤버관리

로그인 통계

API 통계

Playground (Beta)

## 애플리케이션 정보

Client ID

KbF0ZpOTDX738D64bAWZ

Client Secret

.....

보기

## API 호출 안내

지도 API 인증실패나 네이버 로그인 이용 제한이 걸렸다면 [API 설정] 탭에서 URL 관련 설정을 수정하시면 정상 이용 가능합니다 !!!

## 비로그인 오픈 API 당일 사용량

API호출량/일일허용량

검색

0/25000

Clova Face Recognition

0/1000

## API 공통 가이드

### 네이버 오픈API 종류

#### 로그인 방식 오픈 API

#### 비로그인 방식 오픈 API

#### 사전 준비 사항

#### 내 애플리케이션 관리

#### 용어 정리

#### 샘플 코드

#### 오류 코드

## 검색 [↗](#)

다음은 검색 API에서 사용하는 주요 요청 URL과 메서드, 응답 형식입니다.

요청 URL	메서드	응답 형식	설명
<code>https://openapi.naver.com/v1/search/news</code>	GET	JSON, XML	네이버 검색의 뉴스 검색 결과를 반환합니다.
<code>https://openapi.naver.com/v1/search/encyc</code>	GET	JSON, XML	네이버 검색의 백과사전 검색 결과를 반환합니다.
<code>https://openapi.naver.com/v1/search/blog</code>	GET	JSON, XML	네이버 검색의 블로그 검색 결과를 반환합니다.
<code>https://openapi.naver.com/v1/search/shopping</code>	GET	JSON, XML	네이버 검색의 쇼핑 검색 결과를 반환합니다.
<code>https://openapi.naver.com/v1/search/movie</code>	GET	JSON, XML	네이버 검색의 영화 검색 결과를 반환합니다.
<code>https://openapi.naver.com/v1/search/web</code>	GET	JSON, XML	네이버 검색의 웹 문서 검색 결과를 반환합니다.
<code>https://openapi.naver.com/v1/search/image</code>	GET	JSON, XML	네이버 검색의 이미지 검색 결과를 반환합니다.



## API 공통 가이드

### 네이버 오픈API 종류

#### 로그인 방식 오픈 API

#### 비로그인 방식 오픈 API

#### 사전 준비 사항

#### 내 애플리케이션 관리

#### 용어 정리

#### 샘플 코드

#### 오류 코드

## Clova Face Recognition [↗](#)

다음은 Clova Face Recognition API에서 사용하는 주요 요청 URL과 메서드, 응답 형식입니다.

요청 URL	메서드	응답 형식	설명
<code>https://openapi.naver.com/v1/vision/face</code>	POST	JSON	입력된 사진에서 얼굴 윤곽, 부위, 표정을 반환합니다.
<code>https://openapi.naver.com/v1/vision/celebrity</code>	POST	JSON	입력된 사진과 닮은 유명인의 이름과 닮은 정도를 반환합니다.

## Papago 번역 [↗](#)

다음은 Papago 번역 API에서 사용하는 주요 요청 URL과 메서드, 응답 형식입니다.

요청 URL	메서드	응답 형식	설명
<code>https://openapi.naver.com/v1/papago/n2mt</code>	POST	JSON	인공 신경망 기반의 기계 번역 결과(영어, 중국어(간체))를 반환합니다.

Documents > 서비스 API > 검색 > 뉴스

API 공통 가이드

SDK & Tools

CLOVA

네이버 아이디로 로그인

파파고

서비스 API

데이터랩

검색

- 블로그
- 뉴스
- 책
- 성인 검색어 판별
- 백과사전
- 영화
- 카페글
- 지식iN
- 지역
- 오타변환
- 웹문서
- 이미지
- 쇼핑
- 전자자료

단축 URL

## 검색 > 뉴스

네이버 뉴스 검색 결과를 출력해주는 REST API입니다. 비로그인 오픈 API이므로 GET으로 호출할 때 HTTP Header에 애플리케이션 등록 시 발급받은 [Client ID](#)와 [Client Secret](#) 값을 같이 전송해 주시면 활용 가능합니다.

[오픈 API 이용 신청 >](#)

### 1. 준비사항

- 애플리케이션 등록: 네이버 오픈 API로 개발하시려면 먼저 '[Application-애플리케이션 등록](#)' 메뉴에서 애플리케이션을 등록하셔야 합니다.  
[\[자세한 방법 보기\] >](#)
- 클라이언트 ID와 secret 확인: '[내 애플리케이션](#)'에서 등록한 애플리케이션을 선택하면 Client ID와 Client Secret 값을 확인할 수 있습니다.
- API 권한 설정: '[내 애플리케이션](#)'의 'API 권한관리' 탭에서 사용하려는 API가 체크되어 있는지 확인합니다. 체크되어 있지 않을 경우 403 에러(API 권한 없음)가 발생하니 주의하시기 바랍니다.

### 2. API 기본 정보

메서드	인증	요청 URL	출력 포맷
GET	-	<a href="https://openapi.naver.com/v1/search/news.x">https://openapi.naver.com/v1/search/news.x</a>	XML

```
1 import os
2 import sys
3 import urllib.request
4 import datetime
5 import time
6 import json
7
8 #[CODE 1]
9
10 client_id="KbF0Zp0TDX738D64bAWZ"
11 client_secret="Y00U03F03C"
12 # Request 헤더부분에 ID와 Secret을 함께 전송함으로써 REST API를 사용
13 def get_request_url(url):
14
15     req = urllib.request.Request(url)
16     req.add_header("X-Naver-Client-Id", client_id)
17     req.add_header("X-Naver-Client-Secret", client_secret)
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
```

28 *#[CODE 2]*

29  
30 `def getNaverSearchResult(sNode, search_text, page_start, display):`

31  
32  `base = "https://openapi.naver.com/v1/search"`

33  `node = "/%s.json" % sNode`

34 *#display = page\_start == 901 and 99 or display*

35  `parameters = "?query=%s&start=%s&display=%s" % (urllib.parse.quote(search_text), page_start, display)`

36  `url = base + node + parameters`

37  
38  `retData = get_request_url(url)`

39  
40  `if (retData == None):`

41  `return None`

42  `else:`

43  `return json.loads(retData)`

44

```
45 #[CODE 3]
46
47 def getPostData(post, jsonResult):
48
49     title = post['title']
50     description = post['description']
51     org_link = post['originallink']
52     link = post['link']
53
54     #Tue, 14 Feb 2017 18:46:00 +0900
55
56     pDate = datetime.datetime.strptime(post['pubDate'], '%a, %d %b %Y %H:%M:%S +0900')
57     pDate = pDate.strftime('%Y-%m-%d %H:%M:%S')
58
59     jsonResult.append(
60         {'title':title, 'description': description, 'org_link':org_link, 'link': org_link, 'pDate':pDate})
61     return
62
```

```
63 def main():
64
65     jsonResult = []
66
67     sNode = 'news'
68     search_text = '크리스마스'
69     display_count = 100
70
71     jsonSearch = getNaverSearchResult(sNode, search_text, 1, display_count)
72     while ((jsonSearch != None) and (jsonSearch['display'] != 0)):
73         for post in jsonSearch['items']:
74             postData(post, jsonResult)
75
76         nStart = jsonSearch['start'] + jsonSearch['display']
77         jsonSearch = getNaverSearchResult(sNode, search_text, nStart, display_count)
78
79     with open('%s_naver_%s.json' % (search_text, sNode), 'w', encoding='utf8') as outfile:
80         retJson = json.dumps(jsonResult, indent=4, sort_keys=True, ensure_ascii=False)
81         outfile.write(retJson)
82
83     print ('%s_naver_%s.json SAVED' % (search_text, sNode))
84
85 if __name__ == '__main__':
86     main()
```

```
[2020-12-20 22:54:01.448701] Url Request Success  
[2020-12-20 22:54:01.594309] Url Request Success  
[2020-12-20 22:54:01.715984] Url Request Success  
[2020-12-20 22:54:01.849647] Url Request Success  
[2020-12-20 22:54:01.969308] Url Request Success  
[2020-12-20 22:54:02.095997] Url Request Success  
[2020-12-20 22:54:02.244359] Url Request Success  
[2020-12-20 22:54:02.369028] Url Request Success  
[2020-12-20 22:54:02.490673] Url Request Success  
[2020-12-20 22:54:02.750739] Url Request Success
```

HTTP Error 400: Bad Request

```
[2020-12-20 22:54:02.865721] Error for URL : https://openapi.naver.com/v1/search/news.json?query=%ED%81%AC%EB%A6%AC%EC%8A%A4%EB%A7%88%EC%8A%A4&start=1001&display=100
```

크리스마스\_naver\_news.json SAVED

```
jupyter 크리스마스_naver_news.json✓ 10분 전 Logout
File Edit View Language JSON

1 [
2   {
3     "description": "식사를 마친 후 한지민은 세 사람을 위한 깜짝 <b>크리스마스</b> 트리와 손편지
4     를 선물해 훈훈함을 안겼다. 마지막으로 한지민은 &quot;일상의 소중함을 누리지 못하는 지금이기 때문에
5     저 또한 저도 사랑하는 언니와 조카를 못 만나고... ",
6     "link": "http://sports.chosun.com/news/ntype.htm?id=202012210100156290009591&servicedate=20201220",
7     "org_link": "http://sports.chosun.com/news/ntype.htm?id=202012210100156290009591&servicedate=20201220",
8     "pDate": "2020-12-20 22:52:00",
9     "title": "한지민, 미모의 언니 공개 &quot;내가 제일 못생겼다고...난 그냥 까칠한 동생&quot;
10    ('더...'
11    },
12    {
13      "description": "<b>크리스마스</b> 마켓 홍보모델로 선정된 아역배우 김준 군. 중기부 제공 중소
14      벤처기업부는 앞서 영화 '나홀로 집에'를 패러디한 바이럴 홍보 영상을 통해 익살스러운 모습으로 <b>크
15      리스마스</b> 마켓을 널리 알리고 있는...",
16      "link": "http://www.busan.com/view/busan/view.php?code=2020122022484384097",
17      "org_link": "http://www.busan.com/view/busan/view.php?code=2020122022484384097",
18      "pDate": "2020-12-20 22:50:00",
19      "title": "인공지능(AI)이 <b>크리스마스</b> 마켓 홍보모델 선정"
20    },
21  ],
22 ]
```



```
1 import os
2 import sys
3 import requests
4 client_id = "A8b0ura19UXecv5Sh9Ue"
5 client_secret = "L4jEps8T7s"
6 # url = "https://openapi.naver.com/v1/vision/face" // 얼굴감지
7 url = "https://openapi.naver.com/v1/vision/celebrity" # 유명인 얼굴인식
8 files = {'image': open('./data/suzy.jpg', 'rb')}
9 headers = {'X-Naver-Client-Id': client_id, 'X-Naver-Client-Secret': client_secret }
10 response = requests.post(url, files=files, headers=headers)
11 rescode = response.status_code
12 if(rescode==200):
13     print (response.text)
14 else:
15     print("Error Code:" + rescode)
```

```
{"info":{"size":{"width":600,"height":833},"faceCount":1},"faces":[{"celebrity":{"value":"수지","confidence":1.0}]}}
```