

# 1 baidu-serp-api

## 1.1 安装

```
pip install baidu-serp-api
```

## 1.2 使用

```
from baidu_serp_api import BaiduPc
import time

pc_serp = BaiduPc()
max_page = 10
max_retry = 5
page = 1
retry = 0

while page <= max_page:
    try:
        results = pc_serp.search('python',
                                  pn=page,
                                  exclude=['recommend', 'match_count'])

        if len(results['data']['results']) == 0:
            if retry < max_retry:
                retry += 1
                time.sleep(5)
                print(f'Retrying... ({retry}/{max_retry})')
                continue
            else:
                print('Max retries reached. Exiting...')
                break

        print(f'Page {page}')
        for result in results['data']['results']:
            print(result['title'])

        if results['data']['last_page']:
            break
        page += 1
        retry = 0
        time.sleep(2)
    except Exception as e:
        retry += 1
        print(f'Attempt {retry}/{max_retry} failed: {e}')
        if retry < max_retry:
            time.sleep(5)
        else:
            print('Max retries reached.')
            break
```

### 1.3 参数

- keyword: 搜索关键词
- date\_range(可选): 搜索日期范围, 格式为 '20250501,20250507'
- pn(可选): 搜索结果页码
- proxies(可选): 代理设置
- exclude(可选): 排除的结果类型

### 1.4 返回结果

返回格式为 `dict[str, Any] | Any`

- code: 请求状态
- msg: 请求信息
- data: 返回数据
  - results: 搜索结果列表
  - recommend: 推荐相关搜索词
  - last\_page: 是否为最后一页

### 1.5 注意事项

- 请求速率  
QPS 约为 0.1154, 即每次请求间隔约为 8.67 秒. 经常性返回空数据导致额外耗时
- 语言设置  
默认为中文, 无法设置其他语言
- 代理设置  
`pc_serp.search(...proxies={'http': 'http://yourproxy:port'})`

### 1.6 官方文档

<https://github.com/ohblue/baidu-serp-api>

## 2 duckduckgo-search

### 2.1 安装

```
pip install duckduckgo-search
```

## 2.2 使用

```
from duckduckgo_search import DDGS
import time

ddgs = DDGS(timeout=20)
max_retry = 5

retry = 0
while retry < max_retry:
    try:
        results = ddgs.text('python', max_results=10)
        for result in results:
            print(result['title'])
            print(result['href'])
            print(result['body'])
        time.sleep(2)
        break
    except Exception as e:
        retry += 1
        print(f"Attempt {retry}/{max_retry} failed: {e}")
        time.sleep(5)
        continue
else:
    print("Max retries reached.")
```

## 2.3 参数

- keyword: 搜索关键词
- max\_results(可选): 最大返回结果数, 若为None, 则仅返回第一个响应的结果, 默认为None
- safe\_search(可选): 安全搜索设置, 默认为'moderate'
- region(可选): 地区, 默认为'wt-wt'
- proxy(可选): 代理设置

## 2.4 返回结果

返回格式为list[dict[str, Any]]

- title: 搜索结果标题
- href: 搜索结果链接
- body: 搜索结果摘要

## 2.5 注意事项

- 请求速率

QPS 约为 0.2277, 即每次请求间隔约为 4.39 秒. 频繁使用会触发速率限制

- 语言设置

- region='wt-wt': 默认语言
- region='us-en': 英文
- region='cn-zh': 中文

- 代理设置

```
ddgs = DDGS(proxy='http://yourproxy:port')
```

## 2.6 官方文档

[https://github.com/deedy5/duckduckgo\\_search](https://github.com/deedy5/duckduckgo_search)

## 3 googlesearch

### 3.1 安装

```
pip install googlesearch-python
```

### 3.2 使用

```
from googlesearch import search
import time

max_retry = 3
retry = 0

while retry < max_retry:
    try:
        results = search("python",
                          num_results=10,
                          advanced=True)
        for result in results:
            print(result.title)
            print(result.url)
            print(result.description)
        break
    except Exception as e:
        retry += 1
        print(f"Attempt {retry}/{max_retry} failed: {e}")
        if retry < max_retry:
            time.sleep(5)
        else:
            print("Max retries reached. Exiting...")
```

### 3.3 参数

- keyword: 搜索关键词
- num\_results(可选): 最大返回结果数, 默认为 10
- lang(可选): 语言设置, 默认为"en"
- region(可选): 地区设置, 默认为None
- safe(可选): 安全搜索设置, 默认为"active"
- advanced(可选): 是否获取更多信息, 若为False则仅返回url, 默认为False
- proxy(可选): 代理设置
- ssl\_verify(可选): 是否验证 SSL 证书, 默认为False

### 3.4 返回结果

返回结果为Generator[SearchResult | str, Any, None]

- title: 搜索结果标题
- url: 搜索结果链接
- description: 搜索结果摘要

### 3.5 注意事项

- 请求速率

QPS 约为 1.0882, 即每次请求间隔约为 0.92 秒.

- 语言设置

- lang='en': 英文
- lang='zh': 中文

也可以修改地区设置

- region='us': 美国
- region='cn': 中国

- 代理设置

search('python', proxy='http://yourproxy:port')

### 3.6 官方文档

<https://github.com/Nv7-GitHub/googlesearch>