

概述

从本文开始分享requests相关知识，及如何用requests进行接口测试。

requests号称：是唯一的一个非转基因的Python HTTP库，人类可以安全享用。

功能特性

下面我们看看requests的功能特色：

- Keep-Alive & 连接池
- 国际化域名和URL
- 带持久化Cookie的会话
- 浏览器式的SSL认证
- 内容自动解码
- basic/Digest认证
- key/value Cookie管理
- 自动解压
- Unicode响应
- HTTP/HTTPS代理支持
- 文件分块上传
- 流下载
- 连接超时
- 分块请求
- 支持.netrc

看完上述特色，是不是发现掌握了requests库，在http/https处理方面几乎处于无敌状态了？

安装

先看下怎么安装requests, 执行以下命令：

```
pip install requests
```

安装好后如何导入requests模块呢？如下所示：

```
import requests
```

基本示例

下面我们看一个基本的示例，体验下requests的强大，直接上代码演示利用requests访问github的api，具体api说明请参见：

```
https://developer.github.com/v3
```

```
#-*- coding:utf-8 -*-  
  
__author__ = "苦叶子"
```

```
# 导入模块
import requests

if __name__ == "__main__":
    print("开源优测 - requests基本示例")

    # 发送HTTP GET请求, 获取github API列表
    r = requests.get("https://api.github.com")

    # 请求返回码
    status_code = r.status_code

    # 完整的返回头
    headers = r.headers

    # 请求返回头 content-type的值
    content_type = r.headers["content-type"]

    # 返回内容编码类型
    code = r.encoding

    # 返回内容文本
    text = r.text

    # 若返回结果为json格式, 我们可以获取其json格式内容
    json_data = r.json()

    # 打印上述所有获取到的值
    print("状态码: ", status_code)
    print("返回头: ", headers)
    print("content-type: ", content_type)
    print("编码: ", code)
    print("文本内容: ", text)
    print("json串内容: ", json_data)
```

将上述代码保存至requests_basic_demo.py中, 执行下属命令运行:

```
python requests_basic_demo.py
```

运行结果如下图:

```
D:\writing>python d:\writing\Python3接口测试\requests_basic_demo.py
开源优测 - requests基本示例
版本号: 200
返回头: {'Date': 'Tue, 12 Sep 2017 08:08:32 GMT', 'Content-Type': 'application/json; charset=utf-8', 'Transfer-Encoding': 'chunked', 'Server': 'GitHub.com', 'Status': '200 OK', 'X-Ratelimit-Limit': '60', 'X-Ratelimit-Remaining': '50', 'X-Ratelimit-Reset': '1505204501', 'Cache-Control': 'public, max-age=60, s-maxage=60', 'Vary': 'Accept, Accept-Encoding', 'ETag': 'W/"7dc470913f1fe9bb6c7355b50a0737bc"', 'X-Github-Media-Type': 'github.v3; format=json', 'Access-Control-Expose-Headers': 'ETag, Link, X-Github-OTP, X-Ratelimit-Limit, X-Ratelimit-Remaining, X-Ratelimit-Reset, X-OAuth-Scopes, X-Accepted-OAuth-Scopes, X-Poll-Interval', 'Access-Control-Allow-Origin': '*', 'Content-Security-Policy': 'default-src \'none\'', 'Strict-Transport-Security': 'max-age=31536000; includeSubdomains; preload', 'X-Content-Type-Options': 'nosniff', 'X-Frame-Options': 'deny', 'X-XSS-Protection': '1; mode=block', 'X-RunTime-rack': '0.010023', 'Content-Encoding': 'gzip', 'X-Github-Request-Id': '4FFB:ID397:65F0470:767F84A:59879600'}
content-type: application/json; charset=utf-8
编码: utf-8
文本内容: {"current_user_url": "https://api.github.com/user", "current_user_authorizations_html_url": "https://github.com/settings/connections/applications{/client_id}", "authorizations_url": "https://api.github.com/authorizations", "code_search_url": "https://api.github.com/search/code?q={query}{&page,per_page,sort,order}", "commit_search_url": "https://api.github.com/search/commits?q={query}{&page,per_page,sort,order}", "emails_url": "https://api.github.com/user/emails", "emojis_url": "https://api.github.com/emojis", "events_url": "https://api.github.com/events", "feeds_url": "https://api.github.com/feeds", "followers_url": "https://api.github.com/user/followers", "following_url": "https://api.github.com/user/following{/target}", "gists_url": "https://api.github.com/gists{/gist_id}", "hub_url": "https://api.github.com/hub", "issue_search_url": "https://api.github.com/search/issues?q={query}{&page,per_page,sort,order}", "issues_url": "https://api.github.com/issues", "keys_url": "https://api.github.com/user/keys", "notifications_url": "https://api.github.com/notifications", "organization_repositories_url": "https://api.github.com/orgs/{org}/repos?type={type},page,per_page,sort", "organization_url": "https://api.github.com/orgs/{org}", "public_gists_url": "https://api.github.com/gists/public", "rate_limit_url": "https://api.github.com/rate_limit", "repository_url": "https://api.github.com/repos/{owner}/{repo}", "repository_search_url": "https://api.github.com/search/repositories?q={query}{&page,per_page,sort,order}", "current_user_repositories_url": "https://api.github.com/user/repos?type={type},page,per_page,sort", "starred_url": "https://api.github.com/user/starred{/owner}/{repo}", "starred_gists_url": "https://api.github.com/gists/starred", "team_url": "https://api.github.com/teams", "user_url": "https://api.github.com/users/{user}", "user_organizations_url": "https://api.github.com/user/orgs", "user_repositories_url": "https://api.github.com/users/{user}/repos?type={type},page,per_page,sort", "user_search_url": "https://api.github.com/search/users?q={query}{&page,per_page,sort,order}"
JSON内容: {"current_user_url": "https://api.github.com/user", "current_user_authorizations_html_url": "https://github.com/settings/connections/applications{/client_id}", "authorizations_url": "https://api.github.com/authorizations", "code_search_url": "https://api.github.com/search/code?q={query}{&page,per_page,sort,order}", "commit_search_url": "https://api.github.com/search/commits?q={query}{&page,per_page,sort,order}", "emails_url": "https://api.github.com/user/emails", "emojis_url": "https://api.github.com/emojis", "events_url": "https://api.github.com/events", "feeds_url": "https://api.github.com/feeds", "followers_url": "https://api.github.com/user/followers", "following_url": "https://api.github.com/user/following{/target}", "gists_url": "https://api.github.com/gists{/gist_id}", "hub_url": "https://api.github.com/hub", "issue_search_url": "https://api.github.com/search/issues?q={query}{&page,per_page,sort,order}", "issues_url": "https://api.github.com/issues", "keys_url": "https://api.github.com/user/keys", "notifications_url": "https://api.github.com/notifications", "organization_repositories_url": "https://api.github.com/orgs/{org}/repos?type={type},page,per_page,sort", "organization_url": "https://api.github.com/orgs/{org}", "public_gists_url": "https://api.github.com/gists/public", "rate_limit_url": "https://api.github.com/rate_limit", "repository_url": "https://api.github.com/repos/{owner}/{repo}", "repository_search_url": "https://api.github.com/search/repositories?q={query}{&page,per_page,sort,order}", "current_user_repositories_url": "https://api.github.com/user/repos?type={type},page,per_page,sort", "starred_url": "https://api.github.com/user/starred{/owner}/{repo}", "starred_gists_url": "https://api.github.com/gists/starred", "team_url": "https://api.github.com/teams", "user_url": "https://api.github.com/users/{user}", "user_organizations_url": "https://api.github.com/user/orgs", "user_repositories_url": "https://api.github.com/users/{user}/repos?type={type},page,per_page,sort", "user_search_url": "https://api.github.com/search/users?q={query}{&page,per_page,sort,order}"}
```

小结

本文演示了GET方法及如何获取响应状态码、响应头、编码、文本内容、json内容。

扫一扫关注微信公众号：

