

## requests

Requests is an HTTP library , it is easy to use. Similar to Python requests.

## Installation

```
1 go get -u github.com/asmcoss/requests
```

## Start

```
1 package main
2
3 import "github.com/asmcoss/requests"
4
5 func main (){
6
7     resp,err := requests.Get("http://www.zhanluejia.net.cn")
8     if err != nil{
9         return
10    }
11    println(resp.Text())
12 }
```

## Post

```
1 package main
2
3 import "github.com/asmcoss/requests"
4
5
6 func main (){
7
8     data := requests.Datas{
9         "name":"requests_post_test",
10    }
11    resp,_ := requests.Post("https://www.httpbin.org/post",data)
12    println(resp.Text())
13 }
```

---

```
1 Server return data...
```

```
1 {
2   "args": {},
3   "data": "",
4   "files": {},
5   "form": {
6     "name": "requests_post_test"
7   },
8   "headers": {
9     "Accept-Encoding": "gzip",
10    "Connection": "close",
11    "Content-Length": "23",
12    "Content-Type": "application/x-www-form-urlencoded",
13    "Host": "www.httpbin.org",
14    "User-Agent": "Go-Requests 0.5"
15  },
16  "json": null,
17  "origin": "114.242.34.110",
18  "url": "https://www.httpbin.org/post"
19 }
```

## PostJson

```
1 package main
2
3 import "github.com/asmc0s/requests"
4
5
6 func main (){
7
8     jsonStr := "{\"name\":\"requests_post_test\"}"
9     resp,_ := requests.PostJson("https://www.httpbin.org/post",
10     jsonStr)
11     println(resp.Text())
12 }
```

```
1 Server return data...
```

```
1 {
2   "args": {},
3   "data": "",
4   "files": {},
5   "form": {
6     "name": "requests_post_test"
7   },
8   "headers": {
```

---

```
9     "Accept-Encoding": "gzip",
10    "Connection": "close",
11    "Content-Length": "23",
12    "Content-Type": "application/x-www-form-urlencoded",
13    "Host": "www.httpbin.org",
14    "User-Agent": "Go-Requests 0.5"
15  },
16  "json": null,
17  "origin": "114.242.34.110",
18  "url": "https://www.httpbin.org/post"
19 }
```

## Feature Support

- Set headers
- Set params
- Multipart File Uploads
- Sessions with Cookie Persistence
- Proxy
- Authentication
- JSON
- Chunked Requests
- Debug
- SetTimeout

## Set header

### example 1

```
1 req := requests.Requests()
2
3 resp,err := req.Get("http://www.zhanluejia.net.cn",requests.Header{"
4   Referer":"http://www.jeapedu.com"})
5 if (err == nil){
6   println(resp.Text())
7 }
```

### example 2

---

```
1 req := requests.Requests()
2 req.Header.Set("accept-encoding", "gzip, deflate, br")
3 resp,_ := req.Get("http://www.zhanluejia.net.cn",requests.Header{"
    Referer":"http://www.jeapedu.com"})
4 println(resp.Text())
```

### example 3

```
1 h := requests.Header{
2     "Referer":      "http://www.jeapedu.com",
3     "Accept-Language": "zh-CN,zh;q=0.9,en;q=0.8",
4 }
5 resp,_ := req.Get("http://www.zhanluejia.net.cn",h)
6
7 h2 := requests.Header{
8     ...
9     ...
10 }
11 h3,h4 ....
12 // two or more headers ...
13 resp,_ = req.Get("http://www.zhanluejia.net.cn",h,h2,h3,h4)
```

## Set params

```
1 p := requests.Params{
2     "title": "The blog",
3     "name":  "file",
4     "id":    "12345",
5 }
6 resp,_ := req.Get("http://www.cpython.org", p)
```

## Auth

Test with the `correct` user information.

```
1 req := requests.Requests()
2 resp,_ := req.Get("https://api.github.com/user",requests.Auth{"asmcos",
    "password...."})
3 println(resp.Text())
```

github return

---

```
1 {"login":"asmcos","id":xxxxx,"node_id":"Mxxxxxxxxx==".....
```

## JSON

```
1 req := requests.Requests()
2 req.Header.Set("Content-Type","application/json")
3 resp,_ = req.Get("https://httpbin.org/json")
4
5 var json map[string]interface{}
6 resp.Json(&jjson)
7
8 for k,v := range json{
9     fmt.Println(k,v)
10 }
```

## SetTimeout

```
1 req := Requests()
2 req.Debug = 1
3
4 // 20 Second
5 req.SetTimeout(20)
6 req.Get("http://golang.org")
```

## Get Cookies

```
1 resp,_ = req.Get("https://www.httpbin.org")
2 coo := resp.Cookies()
3 // coo is [] *http.Cookies
4 println("*****cookies*****")
5 for _, c:= range coo{
6     fmt.Println(c.Name,c.Value)
7 }
```