

[登录](#)

2019年04月30日 阅读 1367

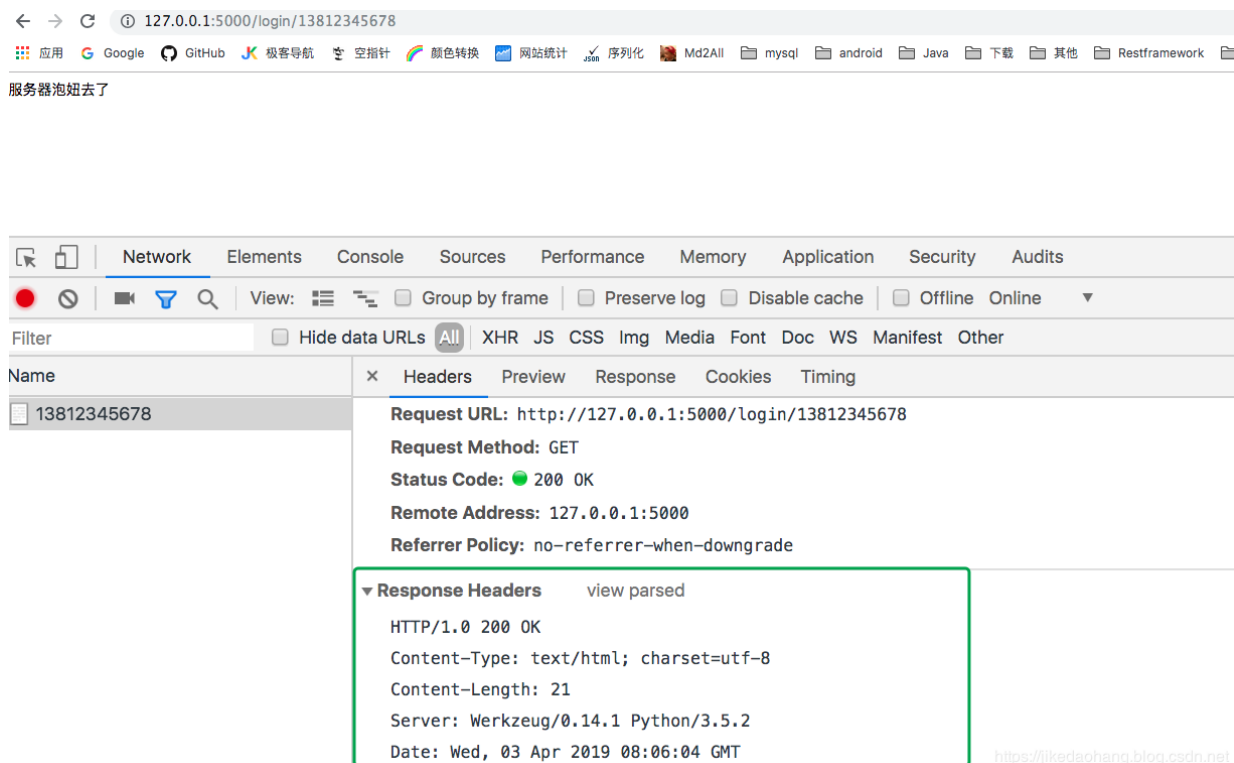
[关注](#)

Flask框架从入门到精通之自定义Response(八)

知识点: 1、自定义响应信息 2、返回Json

一、概况

我们都知道当浏览器发起一个请求时,服务器会给一个响应。这个响应包含了返回的内容类型,状态码,服务器版本等一些。如下图:



如果我们不进行修改这里的响应信息,它会返回默认信息。

二、自定义响应信息

如果我们要自定义响应信息,必须返回的是元组,并且是(response,status,headers)这样的数据结构,至少要有response。status指定的是HTTP状态码,可以是HTTP中状态码,也可以是自定义的状态码。headers指的是额外的响应头。

```
from flask import Flask, abort, Response
```

```
app = Flask(__name__)
```

[复制代码](#)

```

@app.route('/')
def index():
    # return ('自定义响应信息', 502, {"name": "xiaosong", "age": 12})

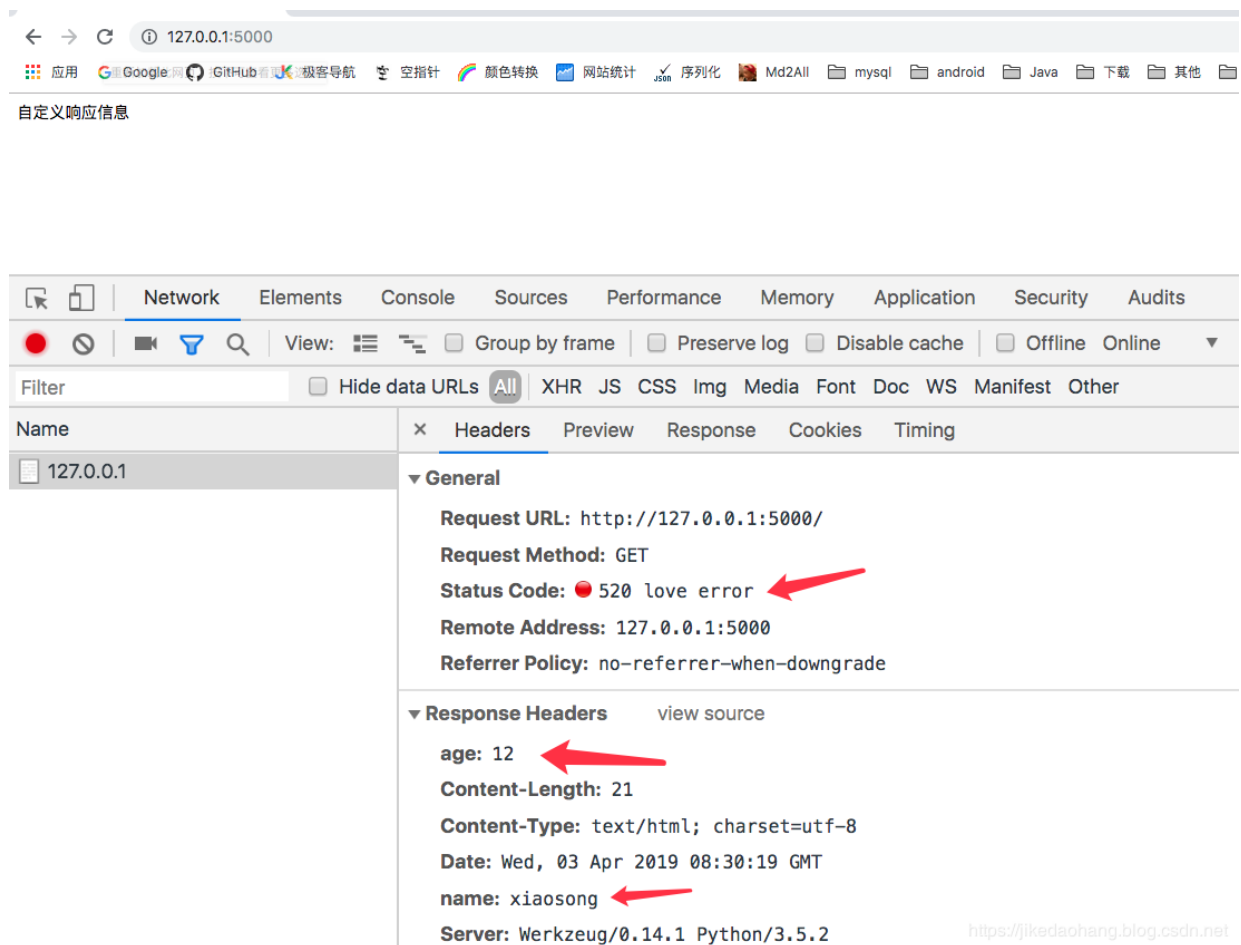
    # 可以不加括号, 会自动组装成元组
    # return '自定义响应信息', 502, {"name": "xiaosong", "age": 12}

    # 自定义状态码 可以加上描述信息
    return '自定义响应信息', 520 love error, {"name": "xiaosong", "age": 12}

if __name__ == '__main__':
    # 0.0.0.0代表任何能代表这台机器的地址都可以访问
    app.run(host='0.0.0.0', port=5000, debug=True) # 运行程序

```

我们在浏览器调试一下:



三、make_response

上面返回自定义响应信息, 可以用make_response函数。make_response(), 相当于Django中的HttpResponse, 效果是一样的。

```

from flask import Flask, abort, Response, make_response

app = Flask(__name__)

@app.route('/')
def index():
    # return ('自定义响应信息', 502, {"name": "xiaosong", "age": 12})

```

复制代码

```
# 可以不加括号, 会自动组装成元组
# return '自定义响应信息', 502, {"name": "xiaosong", "age": 12}

# 自定义状态码 可以加上描述信息
# return '自定义响应信息', '520 love error', {"name": "xiaosong", "age": 12}
resp = make_response()
resp.headers['name'] = 'xiaosong'
resp.status = '520 love error'
return resp

if __name__ == '__main__':
    # 0.0.0.0代表任何能代表这台机器的地址都可以访问
    app.run(host='0.0.0.0', port=5000, debug=True) # 运行程序
```

四、jsonify

当我们做前后端分离项目, 会前端使用ajax请求的时候。我们都会使用JSON作为前后端交互的数据格式, 在Django中有个JsonResponse可以返回Json, 在Flask中我们用jsonify返回Json。

```
from flask import Flask, jsonify
```

复制代码

```
app = Flask(__name__)
```

```
@app.route('/')
def index():
```

```
    data = {
        'name': 'xiaosong',
        'age': 12,
        'gender': '女'
    }
```

```
    return jsonify(data)
```

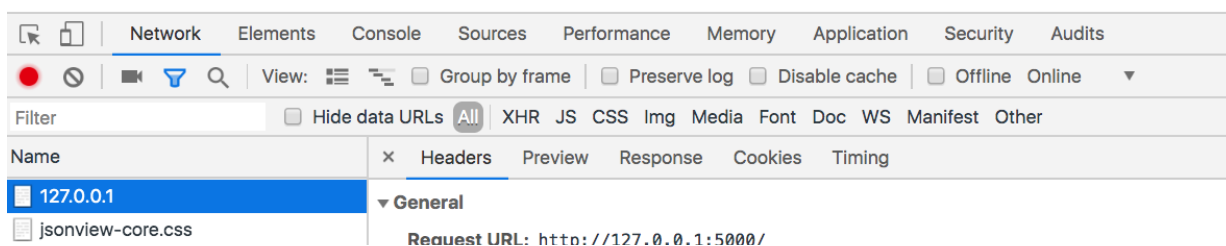
```
if __name__ == '__main__':
```

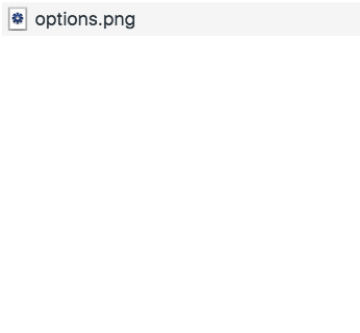
```
    # 0.0.0.0代表任何能代表这台机器的地址都可以访问
    app.run(host='0.0.0.0', port=5000, debug=True) # 运行程序
```

我们在浏览器调试一下：



```
{
  age: 12,
  gender: "女",
  name: "xiaosong"
}
```





Request Method: GET	
Status Code: 200 OK	
Remote Address: 127.0.0.1:5000	
Referrer Policy: no-referrer-when-downgrade	
▼ Response Headers	view source
Content-Length: 62	
Content-Type: application/json	
Date: Wed, 03 Apr 2019 08:40:16 GMT	
Server: Werkzeug/0.14.1 Python/3.5.2	

<https://jjkedaohang.blog.csdn.net>

欢迎关注我的公众号：



文章分类

后端

文章标签

Flask

相关推荐

-
-
-
-