

1.cookie简介

Cookie以文本文件的形式存储在客户端的计算机上。其目的是记住和跟踪与客户使用相关的数据，以获得更好的访问者体验和网站统计信息。

Request对象包含**Cookie**的属性。它是所有**cookie**变量及其对应值的字典对象，客户端已传输。除此之外，**cookie**还存储其网站的到期时间，路径和域名。

response对象的 **set_cookie** 方法来设置 **Cookies**。

request对象的 **cookies** 属性是客户端提交过来的所有**cookie**键值对，字典类型。

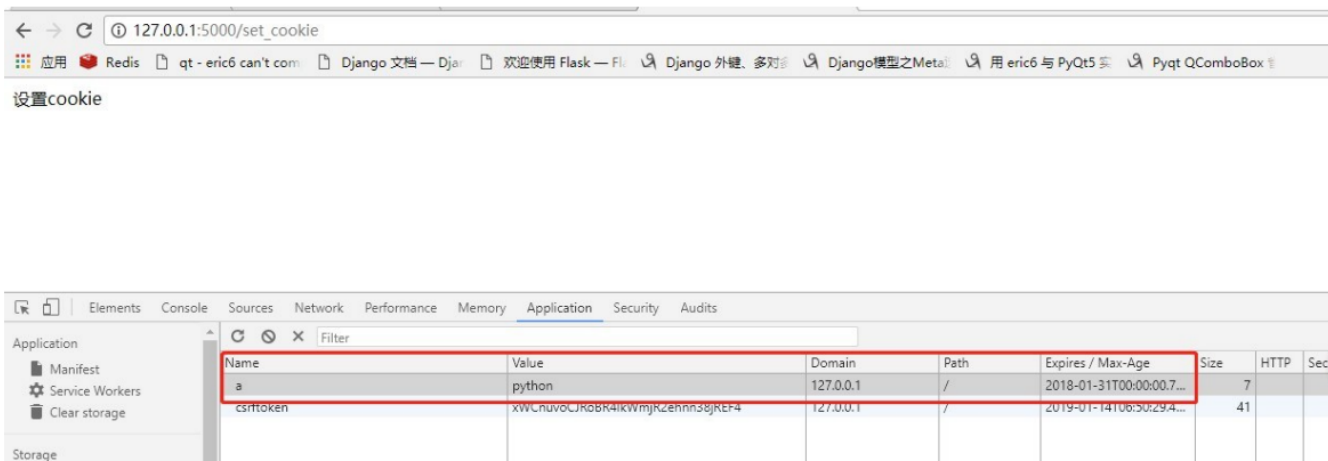
2.设置cookie

cookie 是以键值对的形式保存在浏览器中。

设置**cookie**我们比较关心的三个参数：

- **key** **cookie**的键
- **value** **cookie**的值
- **max_age=None** 超时时间，单位是秒
- **expires=None** 超时时间，**datetime**对象

```
@app.route('/set_cookie')
def set_cookie():
    """设置cookie"""
    resp = make_response('设置cookie')
    # 向浏览其中写入一个key为a,值为python, 超时时间在2018,1,31日
    #resp.set_cookie('a', 'python', expires=datetime.datetime(2018, 1, 31))
    # 向浏览其中写入一个key为a,值为python, 超时时间是60*60秒之后
    resp.set_cookie('a', 'python', max_age=60*60)
    return resp
```



3. 获取cookie

`request` 对象中的`cookie`属性是包含了所有浏览器上传到服务器的`cookie`。
`get` 方法获取其中的键值，可以传一个默认值。

```
@app.route('/get_cookie')
def get_cookie():
    """获取cookie"""
    # 通过键获取cookie值，如果没有这个键返回None
    a = request.cookies.get('a')
    print(a)
    # 可以传一个默认值，如果不存在这个键，将使用默认值。
    b = request.cookies.get('b', 'not found')
    print(b)
    return '获取到的cookie是: a: %s b:%s' % (a, b)
```

4. 删除cookie

```
@app.route('/del_cookie')
def del_cookie():
    """删除cookie"""
    response = make_response('删除cookie')
    # 将key为a的cookie 删除
    response.delete_cookie('a')
    return response
```