**Flask操作session：**

**1. 设置session：**

通过`flask.session`就可以操作session了。操作`session`就跟操作字典是一样的。

session['uname']='momo'。

**2. 获取session：**

也是类似字典，

session.get(key)。

**3. 删除session中的值：**

也是类似字典。可以有2种方式删除session中的值。

\* session.pop(key)。

\* session.clear()：删除session中所有的值。

**4. 设置session的有效期：**

如果没有设置session的有效期。那么默认就是浏览器关闭后过期。

如果设置session.permanent=True，那么就会默认在31天后过期。

如果不想在31天后过期，按如下步骤操作

1：session.permanent=True

2：可以设置`app.config['PERMANENT\_SESSION\_LIFETIME'] = timedelta(hour=2)`在两个小时后过期。

**代码演示如下：**

|  |
| --- |
| **from** flask **import** Flask,session,Response  **import** os  **from** datetime **import** timedelta  app = Flask(\_\_name\_\_)  #设置SECRET\_KEY  app.config[**'SECRET\_KEY'**] = os.urandom(24)  # 5.设置session的有效期方式2【指session可以往后活多长时间】  app.config[**'PERMANENT\_SESSION\_LIFETIME'**] = timedelta(hours=2)  # 1.设置session  @app.route(**'/'**)  **def** index():  session[**'uname'**] = **'momo'**  session[**'pwd'**] = **'123'**  #底层  # resp = Response()  # resp.set\_cookie('session')  #4.设置session的有效期方式1【持久化 31天】  session.permanent = **True**  # print(type(session))  **return 'Hello World!'**  #2.获取session  @app.route(**'/getSession/'**)  **def** getSession():  uname = session.get(**'uname'**)  pwd = session.get(**'pwd'**)  print(pwd)  **return** uname **or '没有session'**  #3.删除session  @app.route(**'/deleteSession/'**)  **def** deleteSession():  #删除指定的key的session  # session.pop('uname')  #删除session中的所有的key 【删除所有】  session.clear()  **return '删除成功'**  **if** \_\_name\_\_ == **'\_\_main\_\_'**:  app.run(debug=**True**) |