

Python flask变量及方法

Flask变量规则

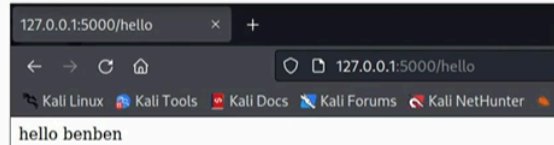
通过向规则参数添加变量部分，可以动态构建URL。

```
from flask import Flask
app = Flask(__name__)

@app.route('/hello')
def hello():
    return "hello benben"

if __name__ == '__main__':
    app.run(debug=True)
```

127.0.0.1:5000/hello



Flask变量规则

通过向规则参数添加变量部分，可以动态构建URL。

```
from flask import Flask
app = Flask(__name__)

@app.route('/hello')
def hello():
    return "hello benben"

if __name__ == '__main__':
    app.run(debug=True)
```

127.0.0.1:5000/hello

```
from flask import Flask
app = Flask(__name__)

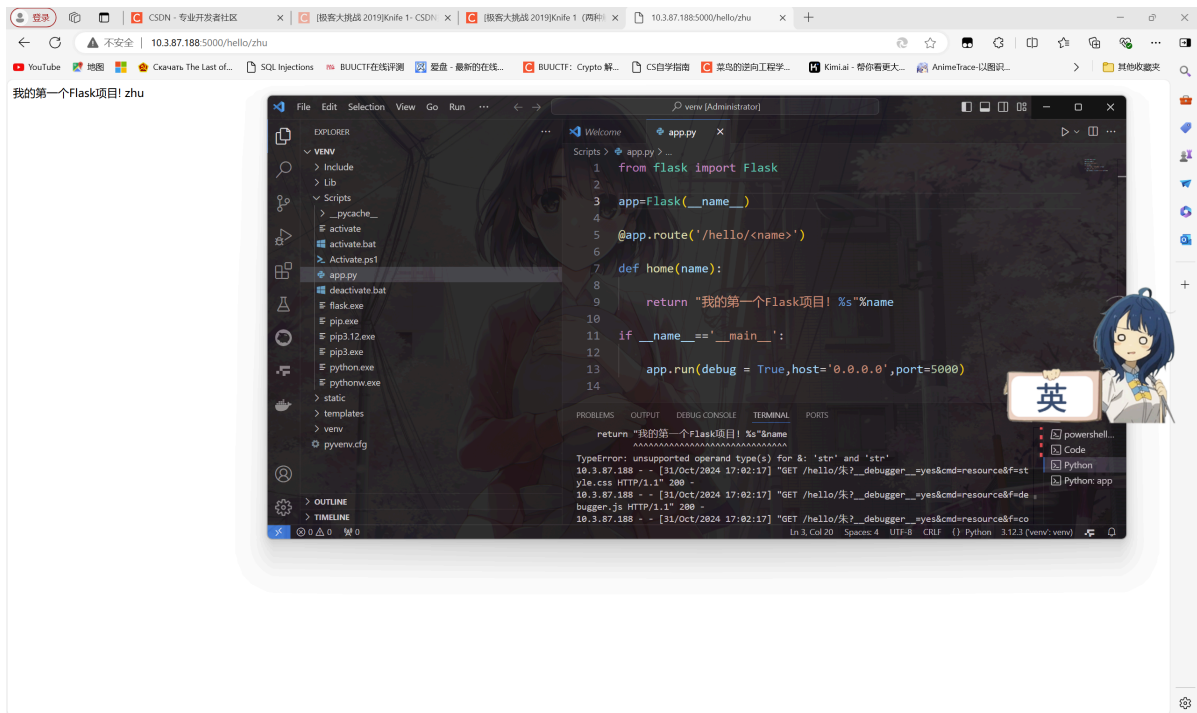
@app.route('/hello/<name>')
def hello(name):
    return "hello %s" % name

if __name__ == '__main__':
    app.run(debug=True)
```

Diagram illustrating variable rules in Flask routes. Red arrows point from the variable part of the route (`<name>`) to the function parameter (`name`). A blue arrow points from the text '变量' (variable) to the variable part of the route. The text '传参' (passing parameter) is written near the function parameter.

动态构造url的过程：app('/hello/),这个name是个写在url里面的变量，具体值多少由我们补充，接下来就会传参到def hello()这个方法中，return"hello %s" %name，%s是格式化字符串，而name是唯一的变量，变量的值就会传递到格式化字符串中输出(至于多个格式化字符串和多个变量只需要编号即可)

成功构造动态URL!!!



当然除了字符串以外还可以用其他形式

Flask变量规则

通过向规则参数添加变量部分，可以动态构建URL。

%s
格式化字符串
%d
接受整数
%f
对于浮点值

```
from flask import Flask
app = Flask(__name__)
```

```
@app.route('/hello/<name>')
def hello(name):
    return "hello %s" % name

if __name__ == '__main__':
    app.run(debug=True)
```

变量
传参

127.0.0.1:5000/hello/动态变量

作为关键字参数传递给与规则相关联的函数。

Flask HTTP方法

Flask HTTP方法

Http协议是万维网中数据通信的基础。
在该协议中定义了从指定URL检索数据的不同方法。

方法	描述
GET	以未加密的形式将数据发送到服务器。最常见的方法。
HEAD	和GET方法相同，但没有响应体。
POST	用于将HTML表单数据发送到服务器。POST方法接收的数据不由服务器缓存。
PUT	用上传的内容替换目标资源的所有当前表示。
DELETE	删除由URL给出的目标资源的所有当前表示。

接下来讲讲如何在python flask中定义提交的方式 (get or post)

将以下脚本另存为index.html用来做login页面POST提交：

```
<html>
<body>
  <form action = "http://localhost:5000/login" method = "post">
    <p>Enter Name:</p>
    <p><input type = "text" name = "ben" /></p>
    <p><input type = "submit" value = "submit" /></p>
  </form>
</body>
</html>
```

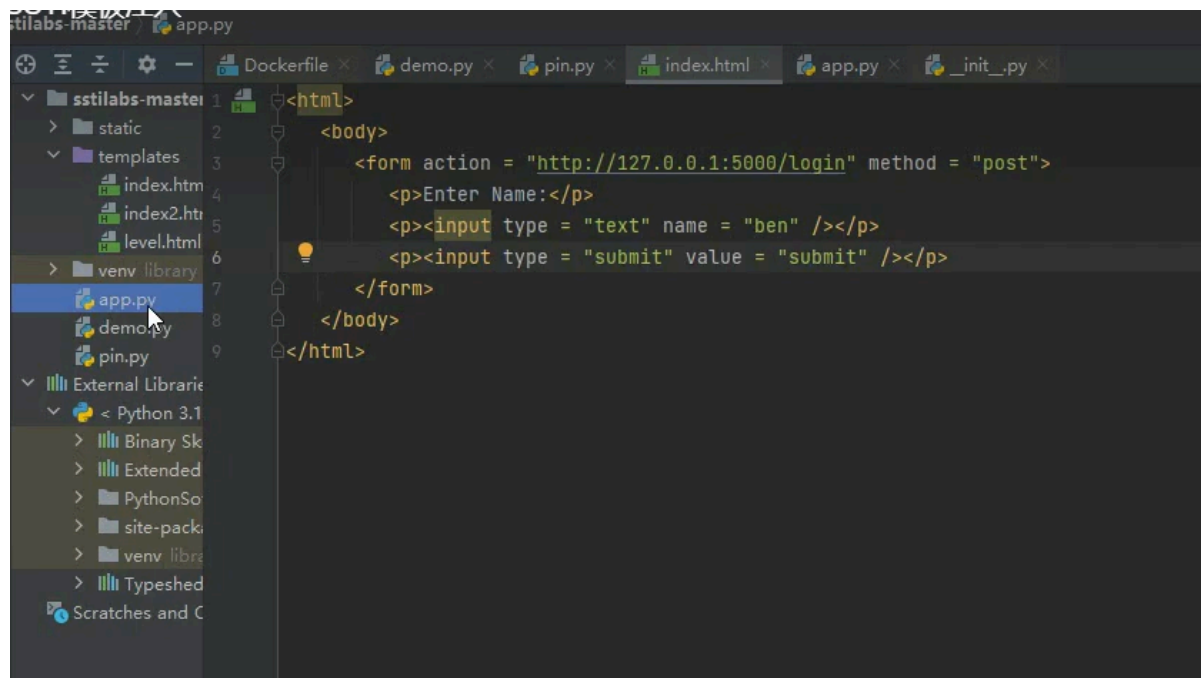
Flask HTTP方法

在Python demo.py中输入以下脚本：

```
from flask import Flask, redirect, url_for, request, render_template
app = Flask(__name__)
@app.route('/')
def index():
    return render_template("index.html")
@app.route('/success/<name>')
def success(name):
    return 'welcome %s' % name
if __name__ == '__main__':
    app.run()
```

127.0.0.1:5000

注：这里有个地方很容易踩坑，index.html文件一定要和flask程序app.py放在同级文件夹下（index.html放在templates下,app.py放在venv下）



```
from flask import Flask,redirect,request,url_for,render_template

app=Flask(__name__)

@app.route('/')
def index():
    return render_template('index1.html')    //渲染index1.html页面
@app.route('/success/<name>')                //路由，在/success/<name>页面下
def success(name):
    return 'welcome %s'%name                //传参name传进来什么就显示'welcome 传参'
@app.route('/login',methods=['GET','POST'])//路由
def login():
    if request.method=='POST':
        print(1)
        user = request.form['ben']
        return redirect(url_for('success',name=user))
    else:
        print(2)
        user = request.form['ben']
        return redirect(url_for('success',name=user))

if __name__=='__main__':

    app.run()
```