

# 必修二10、网络应用开发基础

## 1、网络应用个软件的实现架构

- 1) 客户端/服务器架构(C/S架构)  
比如微信app、淘宝app等
- 2) 浏览器/服务器架构(B/S架构)  
由网页提供功能的软件架构，比如微信网页版、淘宝网网页版。  
B/S架构更加灵活，用户无需安装特定软件，但是服务器压力较大  
C/S架构可以提供更个性化的服务，而且服务器压力相对较小

## 2、利用flask框架编写网络接口

flask是python的一个网络应用程序接口框架。

```
from flask import Flask #导入Flask模块
app=Flask(__name__) #创建一个Flask实例
@app.route("/",methods=['GET']) #装饰app.route方法，指定访问路径为根目录，允许使用GET方法
def welcome(): #自定义函数
    return "welcome flask"
app.run(host='127.0.0.1',port=5000) #监听5000端口，绑定ip地址，127.0.0.1是一个特殊ip，任何时候都是本地计算机

* Serving Flask app '__main__' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off

* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

运行代码后，用浏览器访问127.0.0.1:5000就可以看到返回的文字：

← → ↻ ⓘ 127.0.0.1:5000

welcome flask

## 3、将数据传输给flask服务器

在原来基础上增加/submit接口，通过request对象获得GET请求的name字段和age字段的内容，并进行格式化字符串后输出

```
from flask import Flask,request #导入Flask模块
app=Flask(__name__) #创建一个Flask实例
@app.route("/",methods=['GET']) #装饰app.route方法, 指定访问路径为根目录, 允许使用GET方法
def welcome(): #自定义函数
    return "welcome flask"
@app.route("/submit",methods=['GET'])
def submit():
    n=request.values.get("name")
    a=int(request.values.get("age"))
    return "你好%s, 你的年龄是%d"%(n,a)
app.run(host='127.0.0.1',port=5000) #监听5000端口, 绑定ip地址, 127.0.0.1是一个特殊ip, 任何时候都是本地计算机

* Serving Flask app '__main__' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
```

在浏览器地址栏输入以下内容后, 可以看到:

← → ↻ ⓘ 127.0.0.1:5000/submit?name=james&age=30

你好james, 你的年龄是30

## 4、访问局域网其它的服务器

将app.run中的host字段改成自己电脑网卡的ip地址, 并用局域网内其它电脑来访问此服务器, 他人就可以通过网络访问到本机的内容了。