

# Node原生服务器

以下代码要求能手写：

```
/* 不借助任何第三方库，去搭建Node原生服务器 */

//1.引入Node内置的http模块
let http = require('http')

//2.创建一个“服务员” ---- 创建服务对象
let server = http.createServer(function (request,response) {
  response.setHeader("content-type","text/html;charset=utf-8")
  response.end('ok')
})

//3.指定服务器运行的端口号(绑定端口监听)
server.listen(3000,function (err) {
  if (!err) console.log('服务器启动成功了')
  else console.log(err)
})
```

# express服务器

express是一个基于Node.js平台的极简、灵活的web应用开发框架，它提供一系列的强大特性，帮助你快速创建各种web和移动设备应用。

简单来说express就是运行在node中的用来搭建服务器的模块。

以下代码要求能手写：

```
//引入express
const express = require('express')

//1.创建app服务对象
const app = express()

//2.配置路由 ----- 对请求的url进行分类，服务器根据分类决定交给谁去处理。
/*
  (1).在Node.js课程中，我们所有说的“路由”，默认都是指【后端路由】
  (2).路由可以理解为：一组一组key-value的组合，key:请求方式 + URI路径，value:回调函数
  (3).根据路由定义的顺序(写代码的顺序)，依次定义好路由，随后放入一个类似数组的结构，当有请求时，依次取出匹配。若匹配成功，不再继续匹配了。
  (4).该URL:http://localhost:3000/meishi 中meishi，叫什么？ 1.URI名字 2.虚拟路径名字
*/
//根路由
app.get('/',function (request,response) {
  console.log(request.query)
  console.log(request.url)
  response.send('ok')
})

//3.指定服务器运行的端口号(绑定端口监听)
app.listen(3000,function (err) {
  if (!err) console.log('服务器启动成功了')
```

```
else console.log(err)  
})
```