复习

fs模块

stat/mkdir/rmdir/readdir/writeFile/appendFile/readFile/

unlink/existsSync

http协议

通用头信息 请求头信息 响应头信息 请求的主体

http模块

http.get(url,(res)=>{

statusCode

res.on('data', (buf)=>{ })

})

web服务器

var server=http.createServer()

server.listen(3000)

server.on('request', (req,res)=>{  
 req url/method/headers

res write/writeHead(302,{ Location:'/detail' })/end()

})

express框架

创建web服务器

var server=express()

server.listen(3000);

创建路由

server.get('/login', (req,res)=>{

res.send()/sendFile(\_\_dirname+'/login.html')/redirect('')

});

1.路由中的请求对象

req.method 获取请求的方法

req.url 获取请求的URL

req.headers 获取请求的头信息

req.query 获取请求时以查询字符串形式传递的数据，返回格式为对象。

练习：创建文件03\_post.js，创建web服务器，新建路由(get /reg)，响应一个注册文件(reg.html)

2.post和get请求

get请求以查询字符串的形式传递数据，服务器端使用 req.query获取数据，结果是对象

post请求是通过表单提交(现阶段)的方式传递数据，服务器端通过事件形式获取数据(后期会有简单的方法)

|  |
| --- |
| req.on('data', (buf)=>{  获取的结果是buffer数据，需要使用查询字符串解析为对象  }) |

3.使用路由传递数据——路由传参

设置路由中接收的名称

|  |
| --- |
| server.get('/detail/:lid',(req,res)=>{  req.params //获取路由传递的数据，格式为对象  }) |

浏览器请求方式

http://127.0.0.1/detail/5

5就是传递的数据，使用lid来接收

练习：创建购物车的路由，请求的URL：/shopping，请求的方法get，传递商品的价格(price)和名称(pname)。

用户模块

用户列表 /list 详情 /detail 删除 /delete

/user/list /user/detail /user/delete

商品模块

商品列表 /list 详情 /detail 删除 /delete

/product/list /product/detail /product/delete

4.路由器

路由在使用过程中，不同的模块可能出现相同的URL，把同一个模块下的路由挂载到特定的前缀。

例如：商品模块下的路由挂载到/product，访问形式/product/list，用户模块下的路由挂载到/user，访问形式/user/list

路由器就是一个js文件，把同一模块下的路由放到一起。

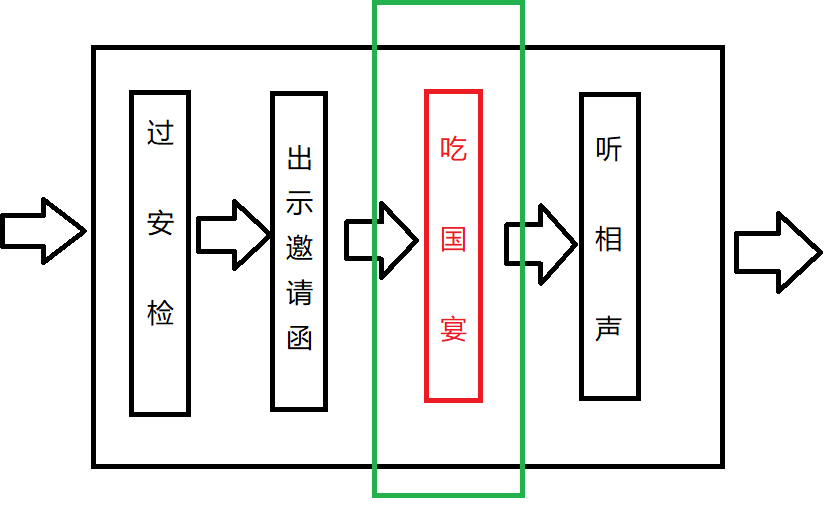
|  |
| --- |
| const express=require('express');  var router=express.Router(); //创建空的路由器对象  router.get('/list', (req,res)=>{ }); //往路由器中添加路由  module.exports=router; |

在web服务器下使用路由器

|  |
| --- |
| const userRouter=require('./user.js'); //引用路由器模块  server.use('/user', userRouter);//把路由器挂载到/user下，访问形式 /user/list |

练习：创建商品模块路由器(product.js)，添加路由商品列表(list)、商品删除(delete)、商品添加(add)，在web服务器引入，并挂载到/product

5.中间件



中间件的作用为主要的业务逻辑所服务。

**分为5个**

应用级中间件、路由级中间件、内置中间件、第三方中间件、错误级中间件

(1)应用级中间件

每一个中间件就是一个函数，需要配合其他的中间件或者路由使用。

server.use(回调函数) 拦截所有的路由

server.use('/detail', 回调函数) 拦截特定的路由

练习：创建路由(get，/view)响应当前的浏览次数，每次请求，响应的次数加1。

在函数外初始化一个变量，设置值为0; 在中间件中实现变量加1，在路由中响应变量。

(2)路由级中间件

用于在服务器中将路由器挂载到特定的URL

server.use('/user', userRouter);

(3)内置中间件

在express中只有一个内置的中间件

server.use( express.static('要托管的目录') )

托管静态资源到某一个目录，如果浏览器端要请求静态资源，则自动到这个目录下查找。

静态资源：html、css、客户端js、图像...

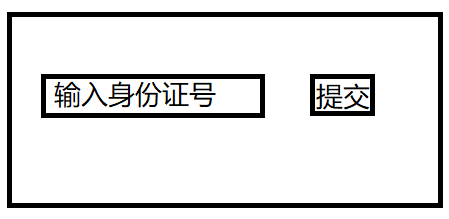
练习：将静态资源托管到files目录下，查看如果两个静态目录下有相同名称的文件，显示哪一个？

课后任务

(1)复习今天内容，复习MySQL中的SQL语句

(2)练习：

创建web服务器，托管静态文件（如下图），点击查询，服务器端获取输入的身份证号(中间件)，截取出生的年月日和性别；在路由中响应给浏览器。



(3)预习mysql模块的使用

https://www.npmjs.com/package/mysql