

# 12/18 Node2

trim(): 首尾去空格

window.location.href : 跳转页面的路径。

node 运行的js文件，分为get和post请求两种。今天讲了两两种的不同写法，详情见课件。

## Express

管理员窗口：（中操作）

npm init : 创建一个文件 package.json

www.baidu.com:80 / == http : // localhost : 80 /

res.query : 直接把查询信息转成信息。

res.json : 等于 : res.write(JSON.stringify(obj) ) res.end( )

app.get ( url, (req,res)=>{ } )

app.post ( url, (req,res)=>{ } )

// 静态文件：

app.use(express.static('www'))

通过use给app注入一个，管理静态文件的新功能。

```
let express = require('express') // 引入 express
app.use(express.static('www'),(req,res,next)=>{ // static() 是什么方法?
    res.write(fs.readFileSync('./www/404.html')); // 读取这个路径上的这个文件，并把读取的内容 发送出去。
    res.end(); // 发送结束
    next(); // ?
});
```

## 安装express 组件

npm init 项目初始化（无脑回车即可）\*name 属性名字不要取某些工具名字，

产生一个 package.json 的文件。

安装包：（后台管理输入）

- npm install body-parser cookie-parser express multer --save (此处这是一次性安装了 body-parser、multer、express、三个组件，会出现对应的三个文件)

引包：

- require('express') -> 引入express。其他的一样引入。

\_\_dirname 当前文件目录

```
const express = require('express');
const app = express();
const fs = require('fs');
const bodyParser = require('body-parser');
const multer = require('multer'); // 引入'multer' 获取中间件

app.use(express.static('www')); 静态资源管理

app.post('/file_upload', (req, res)=>{
  // 写入的路径和名字
  let des_file = __dirname + "/uploads" + req.files[0].originalname;
  try{
    let data = fs.readFileSync(req.files[0].path);
    fs.writeFileSync(des_file, data);
    res.json({code:0, msg:'成功'});
  }catch(error){
    res.json({code:1, msg:'失败'});
  }
});

app.listen(80, ()=>{
  console.log('启动成功');
})
```

~~：取整。

express multer -> 中间件；

\_\_dirname：当前文件夹的目录