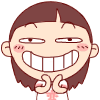
[Nodejs初阶之express](http://www.cnblogs.com/Darren_code/p/node_express.html)

2014-02-28 23:00 by 聂微东, 56881 阅读, 21 评论, [收藏](http://www.cnblogs.com/Darren_code/p/node_express.html), [编辑](http://i.cnblogs.com/EditPosts.aspx?postid=3565918)

　　PS: 2014/09/24 更新《[Express 4.X 启航指南](http://www.cnblogs.com/Darren_code/p/express4.html" \t "_blank)》，欢迎阅读和评论:)

　　老规矩，开头部分都是些自娱自乐的随想，想到哪写到哪...

　　到今天俺已经在俺厂工作俩年零几天了，工作以来头一回在一家企业工作超俩年，对于这俩年确实只有满满的成就感，不管是工作还是生活...写到这句突然又想写篇这俩年的总结，等这篇文章发了之后就着手整理吧，吼吼。



　　那篇《[node.js 初体验](http://www.cnblogs.com/Darren_code/archive/2011/10/31/nodejs.html)》好像才发生在前几月，没想到时间已过两年有多。且看到文章末尾处那句： “这篇文章只会是一个开始” 令俺脸红掩面http://images.cnitblog.com/blog/159097/201402/270046481715808.gif，尼玛真没想到这一开始就开始了两年多，这让老夫情何以堪http://images.cnitblog.com/blog/159097/201402/270002525962790.gif...

　　好吧，反正脸皮厚已不是俩三天，让俺装傻混过去吧，咱往前看才是硬道理不是。俺一定痛改前非、洗心革面、好好改造，今年中心的小心思是把【[前端攻略系列](http://www.cnblogs.com/Darren_code/archive/2013/01/05/fe_series.html" \t "_blank)】折腾点名堂出来，厚积薄发的时刻来鸟。

　　Node不得不写写，没辙呀，太火！看看最近的前端圈子吧，没弄过几行Node代码都不好意思和别人打招呼；不折腾点Node的小平台、小工具，去食堂都不好意思多搞几碗饭，所以俺也就顺大潮把自己的工作笔记整理整理，试试能否写好这篇Nodejs系列。

http://images.cnitblog.com/blog/159097/201402/270010498417459.jpg

　　好吧，不开玩笑哒~~~介绍这篇和接下来几篇Node文章的内容，基本都属于Node的初阶，略偏实战，进阶和高阶内容正在YY中，最终的目录与里程碑下一篇应该差不多能定下来。

　　本篇以介绍Express为话题，延伸出用Node开发小页面，然后其中穿插概述express主要的API、路由的原理、Node模块概念，顺便也会介绍自己使用Node工具等等。默认看官们都具备JavaScript基础，所以过于基础的内容俺就不浪费篇幅咯，GOGO。

　　最后自我爆料一个吧。老婆已回家养胎，所以俺穷的只有时间了。有时间好好加班拼工作，有时间好好写博(客)拼积累...诸如心灵鸡汤什么的就不灌各位啦，喝多了反胃:)

　　一不小心随想就写了那么多，尼玛炫酷狂拽屌炸天有木有！

http://images.cnitblog.com/blog/159097/201402/271125378366371.gif

*-------------------------------------------- 华丽丽的分割线 --------------------------------------------*

　　Node和NPM的安装够便捷了，不细说...有几点基础顺手提一下：

1. 安装命令中的 “-g” 表示全局(global)
2. express的版本不是通常的 “-v” 来查看，而是 “-V”
3. 安装express项目的命令如下  
   **express -e nodejs-product**

-e, --ejs add ejs engine support

-J, --jshtml add jshtml engine support (defaults to jade)

PS：模板引擎之类暂时不必care，不过俺当初学习搭建Node+express时用的是ejs，所以也就顺手一直用着了

**Node的小基友supervisor**

　　每次修改代码后会自动重启。懒程序员就指望这种省事省力的工具活着了:)

　　安装：**npm install -g supervisor**

　　执行：**supervisor app.js**

　　另一个**小基友forever**

　　虚拟机一关node服务就关了，不过forever可以让node服务不停止，介绍如下，安装和执行不细说啦，我懒：

　　forever是一个简单的命令式nodejs的守护进程，能够启动，停止，重启App应用。forever完全基于命令行操作，在forever进程之下，创建node的子进程，通过monitor监控node子进程的运行情况，一旦文件更新，或者进程挂掉，forever会自动重启node服务器，确保应用正常运行。

**express项目目录**

http://images.cnitblog.com/i/159097/201402/272113409312155.png

　　如上图就是一个express项目结构，简单过一下：

* app.js： 项目入口，反正express爱叫app.js没辙，你可以改成index.js或者main.js都成。相当于php项目中的 index.php、index.html
* node\_modules： 存放项目的依赖库
* package.json： 项目依赖配置及开发者信息(这个要说就说多了，还是看文档好，俺就不误人子弟了。下期看看抽个小段单说Node模块)
* public： 静态文件如 css,js,img (PS:俺其实习惯叫static)
* routes： 路由文件(学习的重要攻克对象。尼玛业务好不好，路由是关键)
* Views： 页面文件(Ejs或者jade的模板，默认是jade，俺这用Ejs，在初阶练手最重要，所以都可以试试)

     打开View 文件发现index.ejs比较不习惯，所以对app.js进行小改动：

1. “app.set('view engine', 'ejs');” 变成 “app.engine('.html', ejs.\_\_express);app.set('view engine', 'html');”
2. 上一行出现的ejs变量需要require ejs模块，增加代码“var ejs = require('ejs');”

     最终的app.js如下：



**代码小解：**

     因为针对的是初阶入门，俺们还是继续过一下express的使用与Node的方法哈：

     require() 用于在当前模块中加载和使用其他模块；此方法是模块的基础，使用中大概有路径的概念就行。PS：JS文件可以去掉".js"后缀

　　 exports 表示模块的导出对象，用于导出模块的属性和公共方法。在项目routes文件夹下有index.js和users.js（路由有细说），都使用到exports对象导出对象，如33行的routes.index和34行的user.list;

　　 PS：一个模块的代码只会在模块第一次被使用时执行，不会因require多次而被初始化多次。

**express()**表示创建express应用程序。简单几行代码其实就可以创建一个应用，如下：

[复制代码](javascript:void(0);)

var express = require('express');

var app = express();

app.get('/', function(req, res){

res.send('hello world');

console.log('hello world');

});

app.listen('8808');

[复制代码](javascript:void(0);)

     app.listen() 就是在给定的主机和端口上监听请求，这个和node中http模块的http.createServer(function(){...}).listen()效果一致；

     app.set(name, value)和app.get(name)就是你想的那样，set()为设置 name 的值设为 value，get()为获取设置项 name 的值。如俺app.js的图片16行中的一句“app.set('port', process.env.PORT || 3000)”就是设置项目的port，在下面使用http.createServer时就可以使用app.get('port')来获取，只是俺偷懒没用来着http://images.cnitblog.com/i/159097/201402/280809248922088.gif；

     了解app.engine()方法之前先看看express应用的安装命令:“express -e nodejs-product”，其中的 -e 和 -J 我们一开始已经提到，表示ejs和jade模板。

     如果想把模板后缀改成“.html”时就会用到app.engine方法，来重新设置模板文件的扩展名，比如想用ejs模板引擎来处理“.html”后缀的文件：app.engine('.html', require('ejs').\_\_express);

     app.engine(ext, callback) 注册模板引擎的 callback 用来处理ext扩展名的文件。

　　 PS：\_\_express不用去care，其实就是ejs模块的一个公共属性，表示要渲染的文件扩展名。

     app.use([path], function) 使用中间件 function,可选参数path默认为"/"。使用 app.use() “定义的”中间件的顺序非常重要，它们将会顺序执行，use的先后顺序决定了中间件的优先级(经常有搞错顺序的时候);

     最后介绍个很有用的express API：

     app.render(view, [options], callback) 渲染 view, callback 用来处理返回的渲染后的字符串。

**路由实战：**

     路径代码应该是项目中最本机的一部分了。express中创建一个或者一套新的handle非常简单，先看看express现有的，一会儿我们创建俩个实际的规则。

http://images.cnitblog.com/i/159097/201402/282058558455441.png

     变量 routes 和 user 都是刚才require的模块，他们各自exports了index方法和list方法；其中Response.render()表示渲染view，同时传进对应的数据，Response.send()为发送一个响应；在设置路由时index和list方法作为回调函数最终执行。

　　 流程大概了解啦，俺们也就实际搞一把，最easy的一种方式，简单俩步：

1. 第一种方式就是在当前的routes/index.js或者routes/test.js中加几行代码如下
2. exports.test = function(req, res){
3. res.send('test welcome.');

};

1. 在app.js文件设置路由那块加上app.get('/test', routes.test);

　　 第二种方式就是多了两步，先新建一个模块如test.js文件，输出然后exports对应的方法；在app.js中require这个模块，再加一行设置路由即完成了。

**快速炫起来，集成Bootstrap：**

     JS工程师使用Nodejs上手还是以快速搭建网站为主，所以才会介绍Express，那么为了让网站更快的体面起来，集成使用Bootstrap就是上佳选择，非常喜欢其响应式布局和整体系的脚手架。

     PS：因为Bootstrap的JS插件都依赖jQeury，所以jQuery也一并引入了。

　　 前文已经说到了，静态文件都放在public文件夹中，切文件夹内已经帮我们把类目都分好了，images、 javascripts、 stylesheets。

　　 分别引入bootstrap.min.css文件至stylesheets目录下；jquery-1.x.x.min.js和bootstrap.min.js放到javascripts文件夹下。

　　 然后俺们修改view/index.html把文件引入使用即可，下面放出俺在bootstrap demo的基础改的index.html，大家随意拿去使用和修改。

http://images.cnblogs.com/OutliningIndicators/ContractedBlock.gif View Code

　　 如果样式有问题请检查下bootstrap的路径是否正确引入。

　　 启动项目之后觉得 高大上 很简单，有木有！！

**FAQ&总结：**

     俺们的express项目暂时，且express也并没有涉及到任何数据库，这个事情需要第三方node模块，比如mysql或者MongoDB，后续俺会有一章单独介绍这块。

　　 express也不是Node中web框架的唯一选择，不过由于其文档较全，所以才以其为示例为大家介绍，其原理和实现其实细化之后并不复杂，也希望更多的JS工程师折腾出自己的Web框架。

　　 尼玛还有几个小时就到3月了，一月一篇可是俺的目标，所以全面的demo俺之后再补上哈。

**如果觉得这文章也算用心，请劳驾点右下角的推荐。**

　　 最后，有 北京&上海 的朋友想换工作的，百度移动云 真诚无比的期待【有技术、毁底线、无节操、大聪明、多勤奋】的你...(JD和我邮箱在页面右上角)