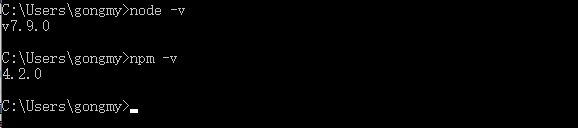
# Node开发相关知识（一个简单的Demo例子）

一、Node.js简介

* Node.js 是一个基于 Chrome V8 引擎的 JavaScript 运行环境。
* Node.js 使用了一个事件驱动、非阻塞式 I/O 的模型，使其轻量又高效。
* Node.js 的包管理器 npm，是全球最大的开源库生态系统。
* Nde.js是运行在服务器端的JavaScript

二、Node.js的搭建（以Webstorm为例）

* 下载NodeJs安装包：安装包地址(<https://nodejs.org/en/>)，根据自身电脑系统下载当前稳定版本即可
* 打开下载的安装包安装
* 安装之后检测是否安装成功，打开cmd命令窗口输入命令node-v（查看node版本），npm-v（查看npm版本），出现以下内容即安装成功



* 在webstorm里配置nodejs环境：File->Settings->在搜索框中输入nodejs,找到Nodejs and NPM -> Node interpreter ,找到下载的nodejs的node.exe路径.再点击应用确认。之后重启Webstorm即可。

三、建立一个简单的Node服务

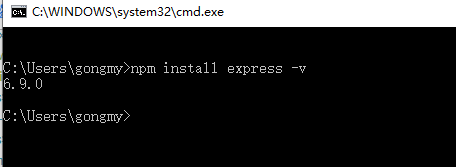
* 在编辑器中建立一个js文件
* 
* 完成上述代码后，运行js文件，任意浏览器网址栏打开主机名+端口号的地址127.0.0.1:3000（或者loaclhost：30000）
* 创建成功，服务运行在127.0.0.1的3000端口上

四、Express 搭建初始化项目

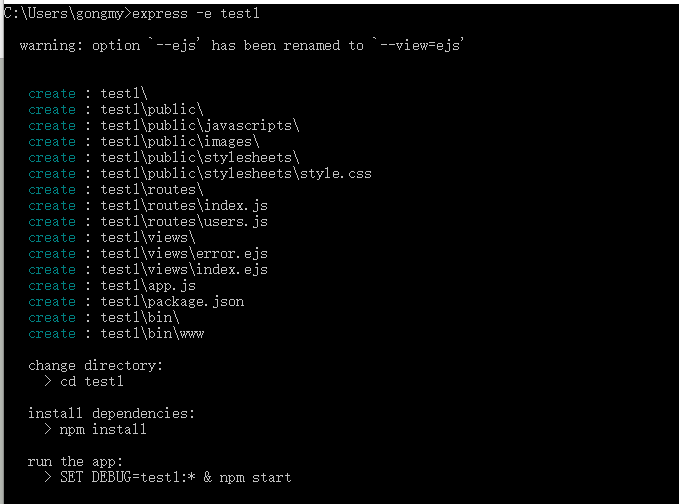
* Express（是一个简洁而灵活的 node.js Web应用框架, 提供了一系列强大特性帮助你创建各种 Web 应用，和丰富的 HTTP 工具）
* 通过上述安装好Node之后
* 全局安装Express及express-generator

命令： **npm install  -g express，——全局安装Express**

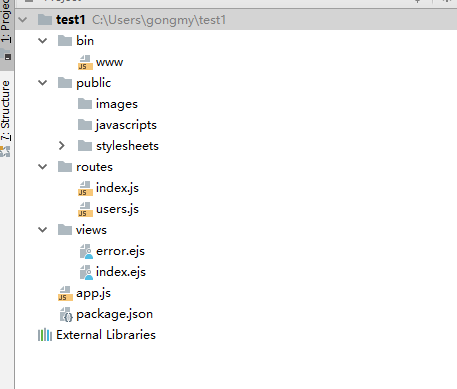
**npm install -g express-generator ——全局安装Express命令工具**



**（Express：版本号v6.9.0）**

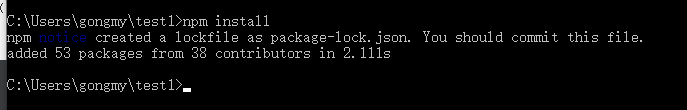
* 安装完成之后，输入命令express -e test1（test1为项目名，可自定义）
* 
* 完成名为test1的express项目，之后webstprm打开该初始化项目

初始化后的test1项目目录结构：



**目录结构简介：**

* app.js:  应用的初始化文件，包括引入应用程序的基础依赖项、设置视图即view的引擎目录以及模板、设置静态资源路径、配置通用的中间件、引入路由和一些错误处理中间件等。
* package.json： 应用的配置文件，文件内包含程序的基础信息、启动脚本和依赖包等。
* bin/www： 应用的启动文件，文件内包含引用要启动的应用、设置应用监听的端口和启动http服务等。
* public/\*\*： 应用的静态资源文件目录，该目录下的文件资源不需要经过文件映射就可以直接访问。
* routes/\*\*： 应用的路由文件，这些路由文件中设置的接口最终会以指定的HTTP请求方式暴露给用户，并在用户请求之后将结果返回。
* views:  应用的视图文件，在app.js中设置好视图引擎和模板之后，该目录即为应用视图的根目录，然后路由文件就会根据app.js中的设置加载并渲染该目录下的视图文件。(demo例子为html模板，默认为ejs模板)
* 此时该项目初始化完成，后续输入命令npm install 安装相关依赖包即可



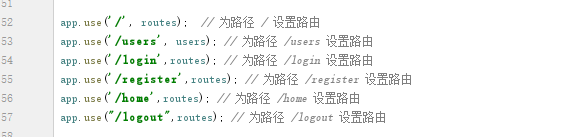
* 完成之后即可输入命令 npm start启动该初始化项目，浏览器输入locallhost：3000（默认端口为3000，可在bin/www文件修改）访问，以下即为成功



五、搭建简单的登录注册demo（test）

利用 Express + node + mongodb搭建

1. 在原始项目之上创建index、login、home、register四个html页面，保留原有的error页面（本demo将项目中的原始ejs改为html模板）
2. 设置路由（页面较少所以直接在app.js中直接定义路由，没有单独创建router.js文件）



3. 路由相对应的前后台js文件

Index登录注册首页



登录页



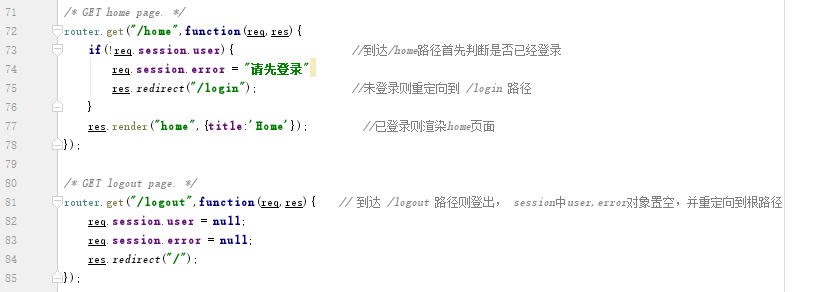


注册页





登入的home页以及点击 ”我要注销” 之后的 ”/logout” 重定向到 ”/” 路径



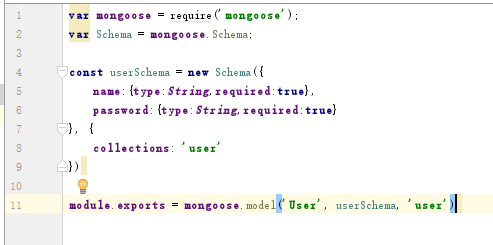
4.数据库的创建（mongodb）



dbHandel.js



User.js



Schema：在 Mongoose 中，所有数据都由一个 Schema 开始创建。每一个 schema 都映射到一个 Mongodb 的集合(collection)，并定义了该集合(collection)中的文档(document)的形式。也就是说相当于建了一个user表映射到数据库中。

六、Demo演示