# 数据库

* 1. MySQL

下载Mysql Installer

安装

Port:3306

添加用户，账户名：terence，密码：terence

添加用户，账户名：wang，密码： wang

* 1. Navicate for Mysql

下载

安装

连接数据库：填写数据库端口号port、用户名、密码

# 后端

* 1. JDK 1.7—java虚拟机

下载安装

配置变量

* 1. IntellJ IDEA—开发环境

下载

安装

激活

* 1. PostMan—接口测试工具

安装postman，以作后端接口调试。

# 前端

前端调试：谷歌浏览器Chrome。

VSCode nodejs npm Webpack

Vue

* 1. Chrome—试工具

安装谷歌浏览器，以作前端开发调试环境

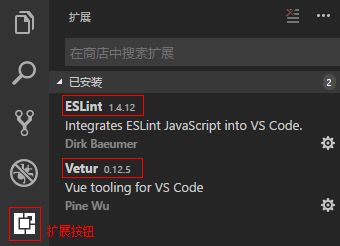
* 1. VSCode—开发环境

1.下载并安装VSCode，比较简单，直接根据安装向导安装即可。

2.安装插件Vetur和Eslint

（1）安装vetur插件和eslint插件，vetur是vue语法的高亮插件，而Eslint可以智能检测错误，类似于eclipse中可以智能提示语法错误、无效变量等。两者安装插件的方法一样：

进入VSCode—查看—扩展（X），或直接点击扩展按钮或快捷键“Ctrl+shift+x”，出现商店扩展搜索界面，输入ESlint或Vetur搜索插件，然后点击安装即可。



（2）插件配置。

进入VSCode，文件—首选项—设置，进入右侧用户设置洁面，看到一对{}，在这对大括号里面粘贴如下代码，其中第一个是对Vutur的配置，后两个是对eslint的配置，配置之后，用户设置界面的代码配置会覆盖左侧的默认配置：

"emmet.syntaxProfiles": {

"vue-html": "html",

"vue": "html"

},

"eslint.validate":[

"javascript",

"javascriptreact",

"html",

"vue"

],

"eslint.options": {

"plugins": [

"html"

]

}

至此，插件安装结束并生效。

* 1. Node.js—开发语言

1. 下载并安装node
2. 测试node

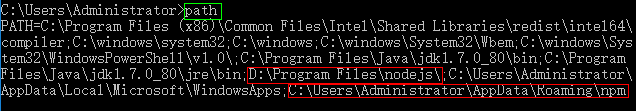
(1)快捷键：win+R，输入命令：cmd

(2)回车进入运行界面，输入命令：path

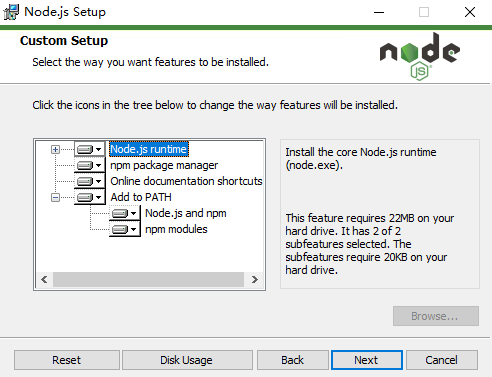
看到系统路径中添加了两个新的路径：

D:\Program Files\nodejs;

C:\Users\Adminstrator\AppData\Roaming\npm;



因为安装node的时候，默认安装了所有模块，包括了node.js和npm package manager 等模块，默认将这两个模块Add to PATH:



这里解释一下npm，NPM是随同NodeJS一起安装的包管理工具，能解决NodeJS代码部署上的很多问题，例如：允许用户从NPM服务器下载别人编写的第三方包到本地使用、允许用户从NPM服务器下载并安装别人编写的命令行程序到本地使用、允许用户将自己编写的包或命令行程序上传到NPM服务器供别人使用。

更多NPM知识见：<https://www.runoob.com/nodejs/nodejs-npm.html>

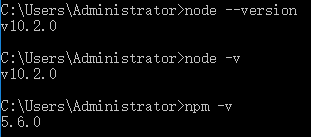
(3)运行环境输入命令：

node –version

node –v

npm –v

可以看到它们的版本号：



1. 配置npm和node

请在计算机管理员的身份下操作如下步骤：

如果遇到错误，说明不是在管理身份下操作，因为Win10或win8在win+R方式下并不是以管理员身份运行，需要win+x，选择以管理员身份打开命令提示符。

**第一步**，修改NPM的缓存目录和全局目录路径，将对应的模块目录改到D盘nodejs的安装目录，以防逐渐增大的node\_modules下的cluster目录（存储安装模块的地方）逐渐增大占用默认磁盘空间。

在nodejs目录下创建node\_cache和node\_global两个目录文件夹，用来放安装过程的缓存文件以及最终的模块配置位置。

执行如下命令，可将全局模块目录和缓存目录配置到这两个文件夹目录。

npm config set prefix "D:\Program Files\nodejs\node\_global"

npm config set cache "D:\Program Files\nodejs\node\_cache"

然后执行如下命令， 安装cluster模块：

npm install cluster –g

**第二步**，是配置npm和nodejs的环境变量，这样nodejs才能正确地调用对应的模块。

配置NPM环境路径，在计算机属性新建环境变量：

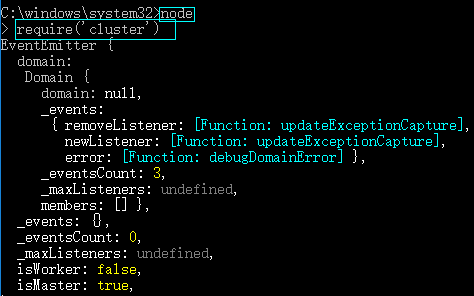
NODE\_PATH： D:\Program Files\nodejs\node\_modules\

配置nodejs的模块调用路径，将原来默认的调用路径C:\Users\Administrator\AppData\Roaming\npm修改为D:\Program Files\nodejs\node\_global\

完毕。

测试：

进入cmd窗口，输入命令：node，可进入nodejs的交互式命令控制台，然后输入命令：require('cluster') ，输出cluster模块的信息则说明配置生效。



1. 更改npm下载仓库

要把npm的模块下载仓库从默认的国外站点改为国内的站点，即淘宝的npm，叫cnpm，以便下载速度更快，输入命令：

npm --registry https://registry.npm.taobao.org install cluster

或 永恒使用命令：

npm install –g cnpm --registry=https://registry.npm.taobao.org

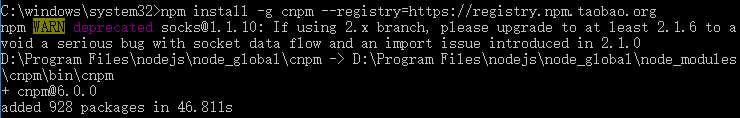
若想一直使用这个npm源的地址，配置命令为：

npm install -g cnpm --registry=registry\_url

registry\_url是指国内的一些npm仓库地址，常用的有：

https://registry.npm.taobao.org

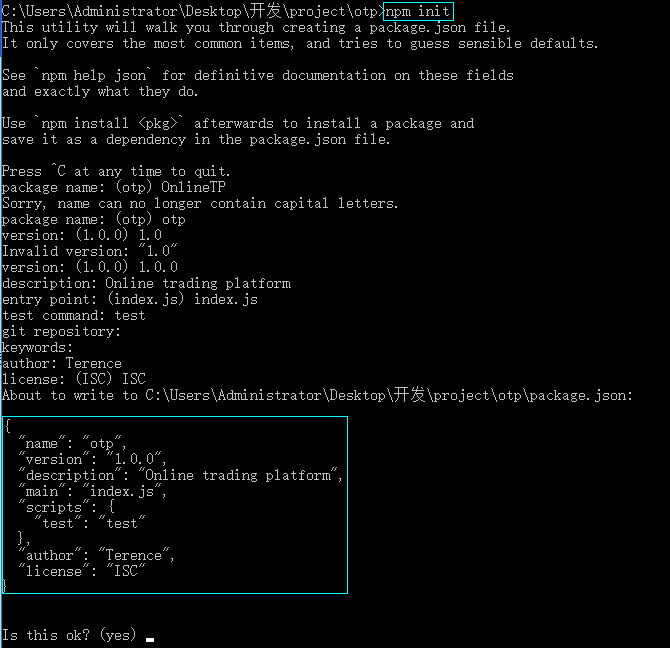
<http://r.cnpmjs.org/>



配置完毕。

* 1. 搭建项目

<http://www.mamicode.com/info-detail-2244111.html>



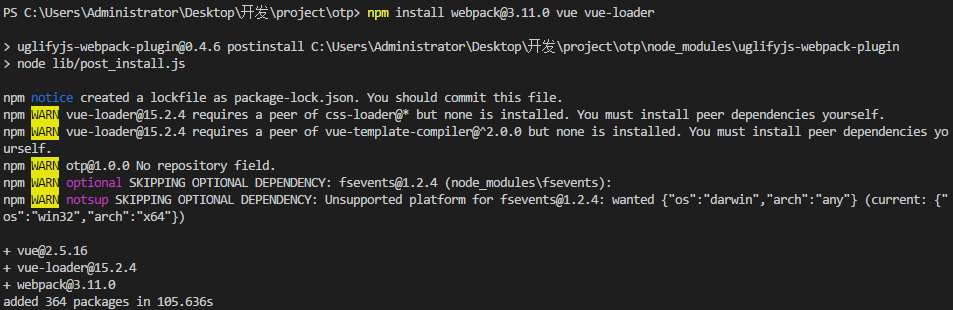
1. 安装vue和webpack。

Vue简介：

WebPack简介：

在终端执行命令安装webpack：

npm install webpack@3.11.0 vue vue-loader



执行如下命令，安装依赖。

npm install css-loader vue-template-compiler

Vue环境搭建+VSCode+Win10：<https://blog.csdn.net/junshangshui/article/details/80376489>

* 1. Gulp

最全gulp构建前端自动化开发环境：<https://blog.csdn.net/lensgcx/article/details/71420058>

gulp详细入门教程：<http://www.ydcss.com/archives/18>

[用less+gulp+requireJs 搭建项目](https://www.cnblogs.com/yaokunlun/p/6434506.html)：<https://www.cnblogs.com/yaokunlun/p/6433220.html>

<https://segmentfault.com/a/1190000003969465>

<http://www.mamicode.com/info-detail-2244111.html>

* 1. gulp和webpack

gulp是工具链、构建工具，可以配合各种插件做js压缩，css压缩，less编译, 自动刷新页面替代手工实现自动化工作, 简单来说就是配置你需要的插件，就可以把以前需要手工做的事情让它帮你做了，所以gulp能够优化前端工作流程。

1.构建工具

2.自动化

3.提高效率用

Gulp和grunt是一种工具，两者类似。

webpack是文件打包工具，可以把项目的各种js文、css文件等打包合并成一个或多个文件，主要用于模块化方案，预编译模块的方案

1.打包工具

2.模块化识别

3.编译模块代码方案用

说到webpack，不得不提到browserlfy/seaJs/requireJs，他们四个都是js模块化的方案。

其中

seaJs和requireJs是一种即时编译模块的方案，浏览器加载页面的时候，会同时加载一个CMD/AMD解释器，用来解释模块代码，这样浏览器就能识别define、exports、module等。

webpack和browserlfy是一种预编译模块的方案，不需要在浏览器中加载解释器来解释js，这种预编译行为可以将AMD/CMD、ES6等各种规范风格的模块代码编译成浏览器能识别的JS。这种预编译行为比在线编译更好一点。

所以定义和用法上来说 都不是一种东西，无可比性 ，更不冲突！

# 命令

Ctrl+D 退出

.exit 退出

1. npm install本地安装  
   （1）将安装包放在 ./node\_modules 下（运行 npm 命令时所在的目录），如果没有 node\_modules 目录，会在当前执行 npm 命令的目录下生成 node\_modules 目录。  
   （2）可以通过 require() 来引入本地安装的包。  
   2.npm install -g全局安装  
   (1) 将安装包放在 /usr/local 下或者你 node 的安装目录。  
   (2)可以直接在命令行里使用。  
   3.npm install --save  
   (1)会把msbuild包安装到node\_modules目录中  
   (2)会在package.json的dependencies属性下添加msbuild  
   (3)之后运行npm install命令时，会自动安装msbuild到node\_modules目录中  
   (4)之后运行npm install --production或者注明NODE\_ENV变量值为production时，会自动安装msbuild到node\_modules目录中  
   4.npm install --save-dev  
   (1)会把msbuild包安装到node\_modules目录中  
   (2)会在package.json的devDependencies属性下添加msbuild  
   (3)之后运行npm install命令时，会自动安装msbuild到node\_modules目录中  
   (4)之后运行npm install --production或者注明NODE\_ENV变量值为production时，不会自动安装msbuild到node\_modules目录中
2. npm install --save-dev xxx是自动把模块和版本号添加到devdependencies部分  
   npm install -g xxx是将模块安装在全局
3. 但是执行了npm install --save-dev，package.json中devDependencies会加入安装的模块，但是在命令行中执行命令时还是出现command not found，还是必须要npm install -g安装在全局才行。
4. npm install --save-dev的作用仅仅是自动把模块和版本号添加到devdependencies部分么，我们还是需要先安装么？还是说我们没配对环境变量？ 为什么很多教程在Installation这流程都是npm install --save-dev xxx就OK了呢

# 框架知识

MTV框架

<https://www.cnblogs.com/ccorz/p/5820497.html>

vue-cli脚手架快速搭建VUE

<https://blog.csdn.net/sinat_29412671/article/details/53690658>

# 前端

Webpack：衡量前端开发能力，架构能力，提高开发效率。

Vue: 前端的组件化的开发框架，开发速度，缺陷解决速度，使用难易程度都更加有优势。

使用webpack优化配置，加快加载速度，达到上线标准。

几个重要的点：

1.如何搭建前端工程？

2.网络优化？

http?加快加载http速度，如何缓存http

3.api定制？

前后端分离，利用api和后端交流，比如，json。

4.nodejs

前端的运行环境，webpack,gulp，grunt等工具都是在nodejs环境下运行的。

至少要知道如何使用nodejs写脚本，在命令行跑起来，运行前端工程。要会使用，学习nodejs可加强api定制能力。

Vue-cli生成的项目是一个模板，自己要学习了解，实现定制。

## 搭建前端工程

1.vue-loader+webpack项目配置

2.webpack静态资源加载配置

如jpg,less等使用modul.rules

1. webpack-dev-server配置(用在开发环境)

npm install webpack-dev-server

npm install cross-env //设置跨平台夸环境时变量命令兼容，package.json中

npm install html-webpack-plugin //将js挂载在html页面上

问题：

C:\Users\Administrator\Desktop\开发\project\demo\node\_modules\vue-loader\lib\index.js:21

loaderContext.emitError(new Error(

^

TypeError: loaderContext.emitError is not a function

at new module.exports (C:\Users\Administrator\Desktop\开发\project\demo\node\_modules\vue-loader\lib\index.js:21:19)

<https://blog.csdn.net/cominglately/article/details/80555210>

<http://www.voidcn.com/article/p-vvrwiyaw-e.html>

## API重点

1.生命周期方法

2.Computed

依赖缓存

Css后处理器

npm install post-css-loader autoprefixer babel-loader babel-core

npm i babel-preset-env babel-plugin-transform-vue-jsx

npm i [babel-helper-vue-jsx-merge-props@^2.0.0](mailto:babel-helper-vue-jsx-merge-props@%5e2.0.0)

vue项目工作流程

