# 1 Cms前端开发项目

## 首先先安装node.js

<https://nodejs.org/en/download/>

选择LTS版本，一路选择默认即可，默认安装在C:\Program Files\nodejs

## 设置npm

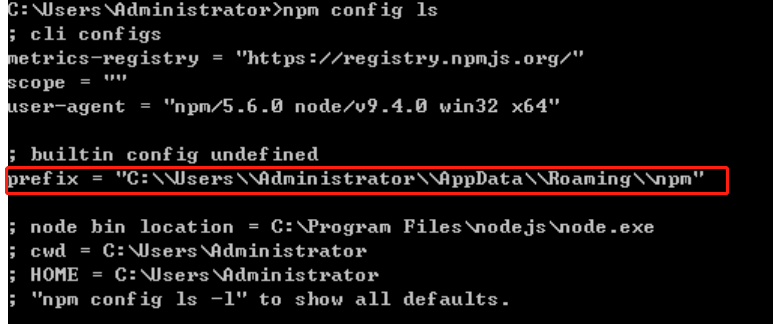
1）Npm把它想象成maven，可以从远程仓库下载js库

node.js已经集成了npm工具，在命令提示符输入 **npm -v** 可查看当前npm版本



2）设置npm的包路径（为了方便管理）

先查看npm默认的包路径 **npm config ls**



然后进入nodejs的安装路径，创建npm\_modules和npm\_cache文件夹，执行下方命令，注意要用管理员运行cmd，要不然会不能创建文件夹

**npm config set prefix "C:\Program Files\nodejs\npm\_modules"**

**npm config set cache "c:\Program Files\nodejs\npm\_cache"**

3) 设置淘宝镜像，npm下载的库都是从境外来的，所以要设置淘宝镜像来加快下载库的速度

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

执行完毕后可以查看cnpm的版本 **cnpm -v，**查看版本的时候，可能会出现cnpm命令不可用，原因是因为上面更改了modules等的路径，环境变量中需要更改，

在电脑的环境变量中打开，修改用户的环境变量的path更改为上方npm\_modules的路径

|  |
| --- |
|  |

命令行 **nrm ls** 可以查看镜像指向，如果没有安装nrm可以使用 **cnpm install -g nrm**来进行安装

|  |
| --- |
|  |

使用**nrm use taobao**切换到taobao镜像

# 安装webpack

不同的应用可以使用不同的webpack，如果不同应用需要使用不同的webpack版本，可以使用本地安装，这里我们使用全局安装

**npm install webpack -g 或 cnpm install webpack -g**

指定版本全局安装

**npm install webpack@3.6.0 -g或 cnpm install webpack@3.6.0 -g**

**注意：本地安装可以覆盖全局安装（比如全局安装了webpack3.6， 然后在一个应用下面本地安装了之后，显示的是本地安装的版本）**

使用**webpack**命令来查看

|  |
| --- |
|  |

# 了解webpack打包使用

1. 创建测试

|  |
| --- |
|  |

1. 进行打包

在项目路径下打开cmd，使用命令行**webpack main.js build.js (表示将main.js打包成build.js ，一般来说都要打包模块，但是main引入了其他模块，所以只打包main.js就行)**

然后在项目中就会出现build.js文件

1. 引入打包后的文件，

在html文件中引入build.js文件，即可完成

|  |  |
| --- | --- |
|  |  |

# 安装webpack-dev-server

在上面的测试中，我们需要在项目里进行打包等的操作，现在需要使用webpack-dev-server来进行自动打包

1）本地安装webpack及webpack-dev-server及 html-webpack-plugin，安装完成之后会自动创建package.json文件

使用 webpack-dev-server需要安装webpack、 webpack-dev-server和 html-webpack-plugin三个包。

**cnpm install webpack@3.6.0 webpack-dev-server@2.9.1 html-webpack-plugin@2.30.1 --save-dev**

全局安装

**cnpm install** [**webpack-dev-server@2.9.1**](mailto:webpack-dev-server@2.9.1) **-g html-webpack-plugin@2.30.1 -g --save-dev**

2）修改测试目录结构，第二行是webpack.config.js文件的内容，第三行是package.json的内容

|  |  |
| --- | --- |
|  |  |
|  | var htmlwp = require('html‐webpack‐plugin');  module.exports={      entry:'./src/main.js',  //指定打包的入口文件      output:{          path : \_\_dirname+'/dist',  // 注意：\_\_dirname表示webpack.config.js所在目录的绝对路径          filename:'build.js'    //输出文件      },      plugins:[          new htmlwp({              title: '首页',  //生成的页面标题<head><title>首页</title></head>              filename: 'index.html', //webpack‐dev‐server在内存中生成的文件名称，自动将build注入到这  个页面底部，才能实现自动刷新功能              template: 'vue\_02.html' //根据index1.html这个模板来生成(这个文件请程序员自己生成)          })      ]  } |
| 注意版本要和自己本地的版本号一致 | {    "scripts": {      "dev": "webpack‐dev‐server ‐‐inline ‐‐hot ‐‐open ‐‐port 5008"    },    "devDependencies": {      "html‐webpack‐plugin": "^2.30.1",      "webpack": "^3.6.0",      "webpack‐dev‐server": "^2.9.1"    }  } |

1. 启动

|  |
| --- |
|  |

# Cms前端工程项目结构讲解

## 大体介绍

|  |
| --- |
|  |

## 重要部分介绍

|  |
| --- |
|  |
|  |

# 启动

|  |  |
| --- | --- |
|  |  |

记得在webstorm启动前要先设置node配置

|  |
| --- |
|  |
|  |