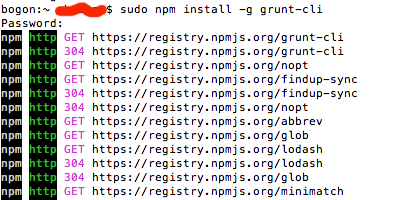
## 安装node

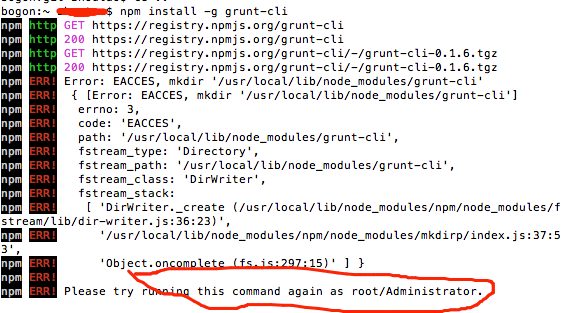
参考[nodejs入门](http://www.cnblogs.com/snandy/archive/2012/03/03/2377380.html" \t "_blank) （最新的node会自动安装npm）

## 安装grunt命令行工具grunt-cli

使用-g全局安装，这样可以在任何一个目录里使用了。命令: npm install -g grunt-cli



需要注意的是在linux或mac下有时会报没有权限的错误，这时须在前面加一个[sudo](http://baike.baidu.com/view/1138183.htm" \t "_blank)，

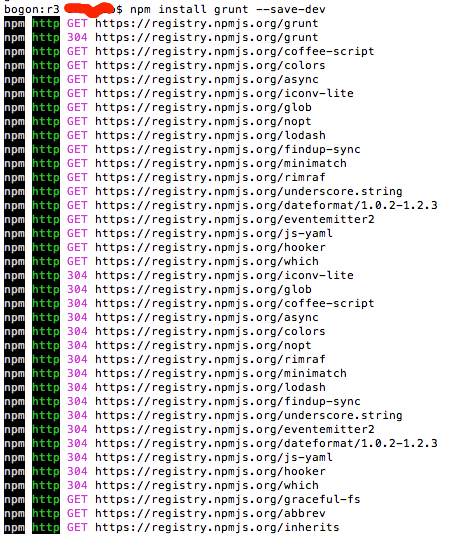


安装后，可以查看改工具的版本。命令: grunt -version

http://images.cnitblog.com/blog/114013/201303/06215601-5a18b7fbf68249ee97f82442aa02223a.png

## 安装grunt及其插件

进入到某项目根目录，使用命令: npm install grunt --save-dev





此时，再查看grunt版本会多了一个4.0，如下

http://images.cnitblog.com/blog/114013/201303/06220501-0fbb816a4db54082a7d999f8a637e860.png

至此，安装完毕。

## 安装依赖项 – 安装插件

比如你到底要做什么，如果压缩则安装压缩需要的插件，

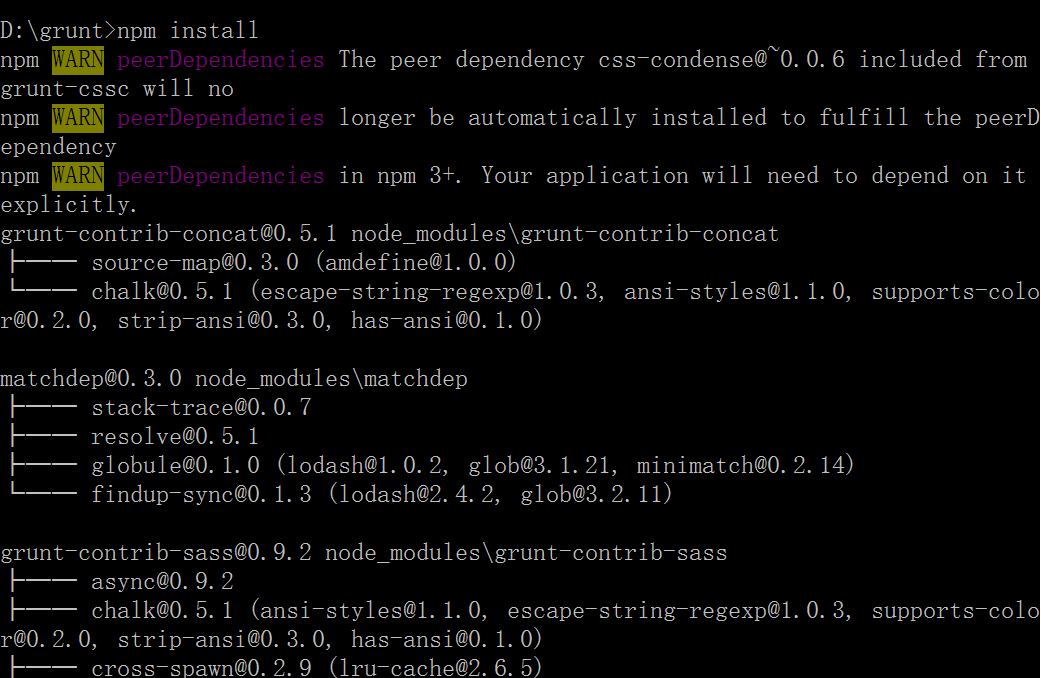
如何安装

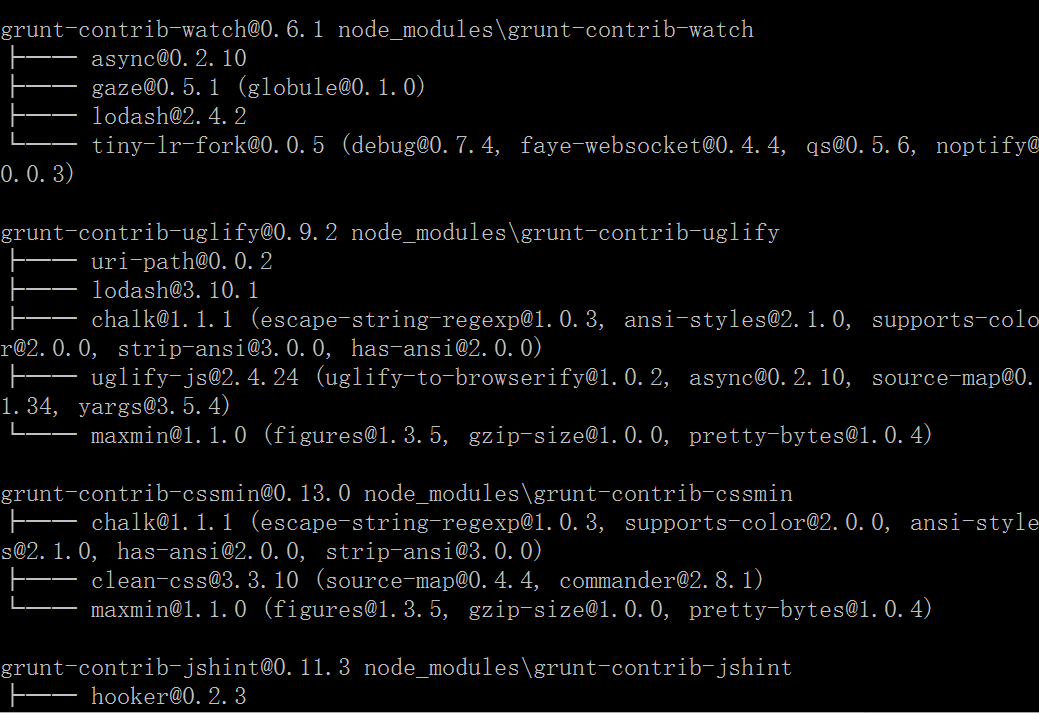
首先配置依赖项，然后利用命令安装

$ npm install

相关插件

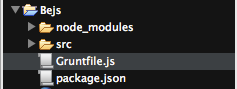
* grunt-contrib-jshint js语法检查
* grunt-contrib-uglify 压缩，采用[UglifyJS](http://www.cnblogs.com/snandy/archive/2012/03/03/2377390.html" \t "_blank)
* grunt-contrib-concat 合并文件





## 新建文件Gruntfile.js

Gruntfile.js也是放在项目根目录下，几乎所有的任务都定义在该文件中，它就是一个普通的js文件，里面可以写任意js代码而不仅局限于JSON。和package.json一样它也要和源码一样被提交到svn或git。



Gruntfile.js由以下内容组成

1. wrapper函数，结构如下，这是Node.js的[典型写法](http://www.cnblogs.com/snandy/archive/2012/03/08/2378441.html)，使用exports公开API

|  |  |
| --- | --- |
| 1  2  3 | module.exports = function(grunt) {    // Do grunt-related things in here  }; |

1. 项目和任务配置
2. 载入grunt插件和任务
3. 定制执行任务

该示例完成以下任务

1. 合并src下的文件（ajax.js/selector.js）为domop.js
2. 压缩domop.js为domop.min.js
3. 这两个文件都放在dest目录下

最终的Gruntfile.js如下

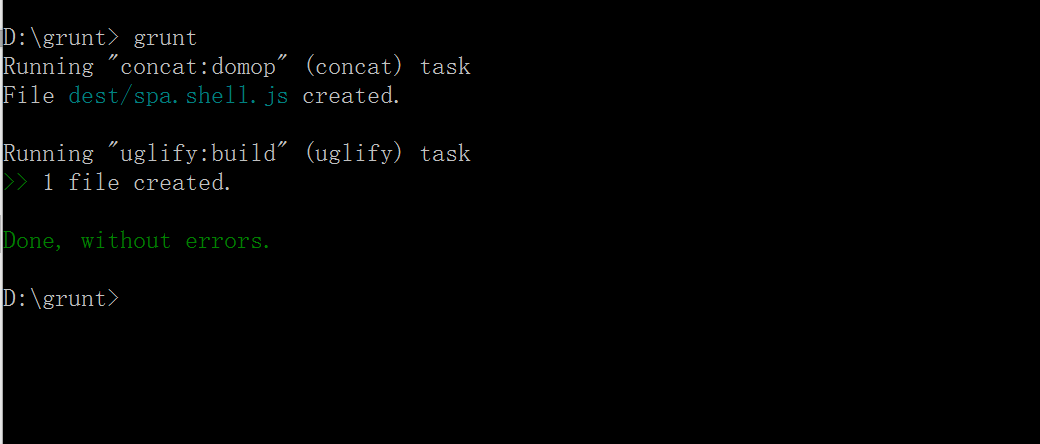
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26 | module.exports = function(grunt) {      // 配置      grunt.initConfig({          pkg : grunt.file.readJSON('package.json'),          concat : {              domop : {                  src: ['src/ajax.js', 'src/selector.js'],                  dest: 'dest/domop.js'              }          },          uglify : {              options : {                  banner : '/\*! <%= pkg.name %> <%= grunt.template.today("yyyy-mm-dd") %> \*/\n'              },              build : {                  src : 'dest/domop.js',                  dest : 'dest/domop.min.js'              }          }      });      // 载入concat和uglify插件，分别对于合并和压缩      grunt.loadNpmTasks('grunt-contrib-concat');      grunt.loadNpmTasks('grunt-contrib-uglify');      // 注册任务      grunt.registerTask('default', ['concat', 'uglify']);  }; |

我的grunt文件

module.exports = function(grunt) {  
 // 配置  
 grunt.initConfig({  
 **pkg** : grunt.**file**.readJSON('package.json'),  
 **concat** : {  
 **domop** : {  
 **src**: ['src/spa.shell.js','src/spa.chat.js', 'src/spa.header.js','src/spa.logo.js'],  
 **dest**: 'dest/spa.shell.js'  
 }  
 },  
 **uglify** : {  
 **options** : {  
 **banner** : '/\*! <%= pkg.name %> <%= grunt.template.today("yyyy-mm-dd") %> \*/\n'  
 },  
 **build** : {  
 **src** : 'dest/spa.shell.js',  
 **dest** : 'dest/spa.shell.min.js'  
 }  
 }  
 });  
 // 载入concat和uglify插件，分别对于合并和压缩  
 grunt.loadNpmTasks('grunt-contrib-concat');  
 grunt.loadNpmTasks('grunt-contrib-uglify');  
 // 注册任务  
 grunt.registerTask('default', ['concat', 'uglify']);  
};

## 执行grunt，合并压缩

将命令切换到项目当前目录，执行



## 自动执行

## 常见任务

CSS合并和压缩

JS合并

JS压缩