

Gulp 使用教程

一、Gulp 简介

gulp 是前端开发过程中一种基于流的代码构建工具，是自动化项目的构建利器；她不仅能对网站资源进行优化，而且在开发过程中很多重复的任务能够使用正确的工具自动完成；使用她，不仅可以很愉快的编写代码，而且大大提高我们的工作效率。

gulp 是基于 Nodejs 的自动任务运行器，它能自动化地完成 javascript、coffee、sass、less、html/image、css 等文件的测试、检查、合并、压缩、格式化、浏览器自动刷新、部署文件生成，并监听文件在改动后重复指定的这些步骤。在实现上，她借鉴了 Unix 操作系统的管道（pipe）思想，前一级的输出，直接变成后一级的输入，使得在操作上非常简单。



gulp 官方网址: <http://gulpjs.com>

gulp 插件地址: <http://gulpjs.com/plugins>

gulp 官方 API: <https://github.com/gulpjs/gulp/blob/master/docs/API.md>

gulp 中文 API: <http://www.ydcss.com/archives/424>

gulp 常用插件: <http://www.jianshu.com/p/98db023b5b89>

易于使用

通过代码优于配置的策略，gulp 让简单的任务简单，复杂的任务可管理。

构建快速

利用 Node.js 流的威力，你可以快速构建项目并减少频繁的 IO 操作。

易于学习

通过最少的 API，掌握 gulp 毫不费力，构建工作尽在掌握：如同一系列流管道。

插件高质

gulp 严格的插件指南确保插件如你期望的那样简洁高质得工作。

二、安装 nodejs

1、说明：gulp 是基于 nodejs，理所当然需要安装 nodejs；

2、安装：打开 nodejs 官网，点击硕大的绿色 Download 按钮，它会根据系统信息选择对应版本（.msi 文件）。然后像安装 QQ 一样安装它就可以了（安装路径随意）。

3、使用命令行（如果你熟悉命令行，可以直接跳过）

node -v 查看安装的 nodejs 版本，出现版本号，说明刚刚已正确安装 nodejs。（PS：未能出现版本号，请尝试重启电脑重试。）

npm -v 查看 npm 版本号，nodejs 安装后自动安装了 npm 命令，我们需要通过 npm 命令来安装 gulp

cd 定位到目录，用法：cd + 路径；

dir 列出文件列表；

D: 进入 D 盘

cls 清空命令提示符窗口内容。

```
管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.2.9200]
(c) 2012 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>node -v 查看nodejs版本号
v0.12.0

C:\Users\Administrator>npm -v 查看npm版本号
2.5.1

C:\Users\Administrator>cd D:\wamp\www\project 定位目录
C:\Users\Administrator>D: 因为我们位于不同盘符，所以还需再输入D:才能正确定位目录
也可先输入D: 回车定位到D盘，再cd wamp\www\project
D:\wamp\www\project>dir 查看当前定位目录下的文件

D:\wamp\www\project 的目录
附：cd .. 定位至上层目录

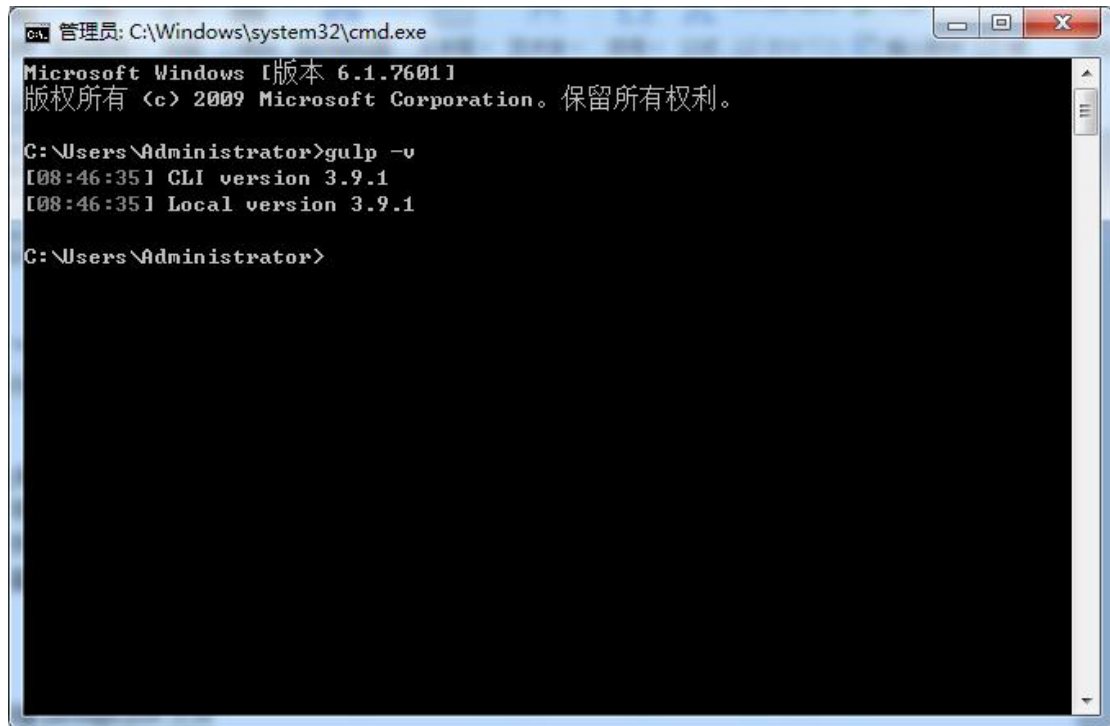
2015/02/24 21:01 <DIR>      .
2015/02/24 21:01 <DIR>      ..
2015/02/23 23:14 <DIR>      dist
2015/02/23 23:19      378 gulpfile.js
2015/02/24 21:01 <DIR>      node_modules
2015/02/24 21:01      393 package.json
2015/02/24 19:41      4 README.md
2015/02/23 22:56 <DIR>      src
                3 个文件      775 字节
                5 个目录  48,377,171,968 可用字节

D:\wamp\www\project>cls 清空内容

搜狗拼音输入法 半：
```

三、全局安装 gulp

- 1、说明：全局安装 gulp 目的是为了通过她执行 gulp 任务；
- 2、安装：命令提示符执行 `npm install gulp -g`；安装需要一定的时间。
- 3、查看是否正确安装：命令提示符执行 `gulp -v`，出现版本号即为正确安装。



```
管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>gulp -v
[08:46:35] CLI version 3.9.1
[08:46:35] Local version 3.9.1

C:\Users\Administrator>
```

四、新建 package.json 文件

1. 说明: package.json 是基于 nodejs 项目必不可少的配置文件, 它是存放在项目根目录的普通 json 文件;
2. 进入工程根目录, 使用 npm init 创建 package.json 文件 (以下截图中的 cnpm 即为 npm, 实操中请输入 npm 代替 cnpm, 文档最后有解释原因)



```
管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.2.9200]
(c) 2012 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>cd D:\wamp\www\project
C:\Users\Administrator>D:
D:\wamp\www\project>cnpm init 创建package.json文件
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sane defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg> --save` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
name: <project> test 项目名称(必须)
version: <1.0.0> 1.0.0 项目版本(必须)
description: this is a test 项目描述(必须)
entry point: <gulpfile.js> 入口文件
test command: 测试命令
git repository: git地址
keywords: 关键字
author: 作者信息
license: <ISC> 许可协议
About to write to D:\wamp\www\project\package.json:

{
  "name": "test",
  "version": "1.0.0",
  "description": "this is a test",
  "main": "gulpfile.js",
  "dependencies": {},
  "devDependencies": {},
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}

Is this ok? <yes> y
D:\wamp\www\project>
搜狗拼音输入法 半 :
```

五、创建 gulpfile.js 文件

1. gulp 也需要一个文件作为它的主文件，在 gulp 中这个文件叫做 **gulpfile.js**。新建一个文件名为 **gulpfile.js** 的文件，然后放到你的项目目录中。之后要做的事情就是在 **gulpfile.js** 文件中定义我们的任务了。

2. gulp 的使用流程一般是：首先通过 **gulp.src()** 方法获取到想要处理的文件流，然后把文件流通过 **pipe()** 方法导入到 gulp 的插件中，最后把经过插件处理后的流再通过 **pipe** 方法导入到 **gulp.dest()** 中，**gulp.dest()** 方法则把流中的内容写入到文件中。

gulp.src() 用于找到要处理的文件。支持 “*” 语法，如：“main/*”，“js/*.js”，“***” 代表所有文件夹。

pipe() 返回处理结构以便后面操作的衔接。node 函数。

gulp.dest() 被处理文件的输出路径

```
var gulp = require("gulp");
gulp.src("src/js/*.js") //获取文件
.pipe(gulp.dest('dest/js')); //生成文件的目录
```

gulp.task() 方法用来定义任务，内部使用的是 **Orchestrator** (用于排序、执行任务和最大并发依赖关系的模块)，其语法为：

```
gulp.task(name[, deps], fn)
```

name 为任务名；

deps 是当前定义的任务需要依赖的其他任务，为一个数组。当前定义的任务会在所有依赖的任务执行完毕后才开始执行。如果没有依赖，则可省略这个参数；

fn 为任务函数，我们把任务要执行的代码都写在里面。该参数也是可选的。当你定义一个简单的任务时，需要传入任务名字和执行函数两个。

默认任务

```
//默认任务
var gulp = require("gulp");
gulp.task("default", function(){
  console.log("default任务");
});
```

任务依赖

可以通过任务依赖来实现。例如我想要执行 **one,two,three** 这三个任务，那我们就可以定义一个空的任務，然后把那三个任务当做这个空的任务的依赖就行了：

//只要执行 **default** 任务，就相当于把 **one,two,three** 这三个任务执行了

```
gulp.task('default',['one','two','three']);
```

(Ps:如果任务相互之间没有依赖，任务就会按你书写的顺序来执行，如果有依赖的话则会先

执行依赖的任务)

但是如果某个任务所依赖的任务是异步的,就要注意了, gulp 并不会等待那个所依赖的异步任务完成,而是会接着执行后续的任务。例如:

```
var gulp = require("gulp");
gulp.task("one", function(){
  console.log("one task");
  setTimeout(function(){ //异步
    console.log("one task 2");
  }, 1000);
});
//two依赖于one任务,one先执行
gulp.task('two', ['one'], function(){
  console.log("two task");
})
```

六、安装 gulp 插件

1. 安装 gulp 插件使用命令 `npm install --save-dev 插件名称`
2. 本示例以 html 插件 `gulp-htmlmin` 为例,执行 `npm install --save-dev gulp-htmlmin`
安装插件(注意要先 cd 到项目的根目录)

3. 使用 html 插件 `gulp-htmlmin`

在 gulpfile.js 中输入代码:

```
var gulp = require("gulp");
var htmlmin = require("gulp-htmlmin");
gulp.task('htmlTask', function(){
  gulp.src('src/html/*')
    .pipe(htmlmin(obj))
    .pipe(gulp.dest('dest/html'));
});
gulp.task('default', ['htmlTask']);
```

参数(可选):

```
var obj = {
  removeComments: true, //清除 HTML 注释
  collapseWhitespace: true, //压缩 HTML
  collapseBooleanAttributes: true, //省略布尔属性的值<input checked="true"/> ==>
  <input checked/>
  removeEmptyAttributes: true, //删除所有空格作属性值 <input id="" /> ==> <input />
  removeScriptTypeAttributes: true, //删除<script>的 type="text/javascript"
  removeStyleLinkTypeAttributes: true, //删除<style>和<link>的 type="text/css"
  minifyJS: true, //压缩页面 JS
  minifyCSS: true //压缩页面 CSS
}
```

4. 使用其他插件

压缩 js 插件 **gulp-uglify**
解决中文乱码插件 **gulp-bom**
压缩 css 插件 **gulp-minify-css**
压缩 html 插件 **gulp-minify-html**
压缩图片插件 **gulp-imagemin**
压缩 png 图片插件 **imagemin-pngquant**
重命名插件 **gulp-rename**

执行 `npm install --save-dev 插件名称` 安装上面的插件

在 `gulpfile.js` 中输入代码：

```
var gulp = require('gulp');
var uglify = require('gulp-uglify'); //js 压缩插件
var bom = require('gulp-bom'); //解决中文乱码插件
var minifyCss = require('gulp-minify-css'); //css 压缩插件
var minifyHtml = require('gulp-minify-html'); //html 压缩插件
var imagemin = require('gulp-imagemin'); //图片压缩相关插件
var pngquant = require('imagemin-pngquant'); //png 图片压缩插件
var rename = require('gulp-rename'); //重命名插件
```

```
//压缩 js
gulp.task('jsTask', function(){
  gulp.src('src/js/*')
  // .pipe(bom())
  .pipe(uglify())
  .pipe(gulp.dest('dest/js'));
});
```

```
//压缩 css
gulp.task('cssTask', function(){
  gulp.src('src/css/*')
  .pipe(minifyCss())
  .pipe(gulp.dest('dest/css'));
});
```

```
//压缩 html
gulp.task('htmlTask2', function(){
  gulp.src('src/html/*')
  .pipe(minifyHtml())
  .pipe(gulp.dest('dest/html'));
});
```



```
//压缩图片
gulp.task('imgTask', function(){
  gulp.src('src/img/*')
    .pipe(imagemin({
      progressive: true,
      use: [pngquant()] //使用 pngquant 插件来压缩 png 图片
    }))
    .pipe(gulp.dest('dest/img'));
});

//重命名
gulp.task('renameTask', function(){
  gulp.src('src/js/bullet.js')
    .pipe(uglify())
    .pipe(rename('bullet.min.js'))
    .pipe(gulp.dest('dest/js'));
});

gulp.task('default', ['jsTask', 'cssTask', 'htmlTask2', 'imgTask',
  'renameTask']);
```

七、执行 gulp

1. 输入命令：gulp 来执行 gulp

```
C:\Users\ijeff\Desktop\深圳1703\bk\31-35\day33\gulp_bk>gulp
[00:07:04] Using gulpfile ~\Desktop\深圳1703\bk\31-35\day33\gulp_bk\gulpfile.js
[00:07:04] Starting 'default' ...
default任务
[00:07:04] Finished 'default' after 2.06 ms
```

淘宝镜像

ps: 因为 npm 安装插件是从国外服务器下载，受网络影响大，可能出现异常，如果 npm 的服务器在中国就好了，所以我们乐于分享的淘宝团队干了这事。这是一个完整 npmjs.org 镜像，你可以用此代替官方版本(只读)，同步频率目前为 10 分钟 一次以保证尽量与官方服务同步。

官方网址：<http://npm.taobao.org>;

安装：命令提示符执行 `npm install cnpm -g --registry=https://registry.npm.taobao.org`； 注意：安装完后最好查看其版本号 `cnpm -v` 或关闭命令提示符重新打开，安装完直接使用有可能会出错；

注：cnpm 跟 npm 用法完全一致，只是在执行命令时将 npm 改为 cnpm（以下操作将以 cnpm 代替 npm）。