姐姐教你学gulp

准备工作：

1. 安装nodejs

2. 全局安装 gulp：

npm install --global gulp

(gulp –v 可以检查是否安装成功)

实战：

1. 进入项目根目录
2. npm init

//在项目目录下可以看到一个多出来的package.json文件

1. npm i --save-dev gulp gulp-less gulp-uglify gulp-minify-css gulp-minify-html

//安装和添加依赖

1. 在项目根目录下创建gulpfile.js文件
2. 看demo里的js代码
3. 控制台输入gulp watch 就大功告成啦~

Ps:

四个命令

gulp.src(globs[, options])globs匹配文件，写类似正则

gulp.dest(path[,options]) 写文件(指定要生成的文件的目录)

gulp.task(name[, deps], fn)//deps依赖

gulp.watch(glob[, opts], tasks)//用来监视文件的变化

一般流程

首先通过gulp.src()方法获取到我们想要处理的文件流，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

**一些常用的gulp插件**

gulp的插件数量虽然没有grunt那么多，但也可以说是应有尽有了，下面列举一些常用的插件。

**1 .自动加载插件**

使用[gulp-load-plugins](https://www.npmjs.com/package/gulp-load-plugins)  
安装：npm install --save-dev gulp-load-plugins  
要使用gulp的插件，首先得用require来把插件加载进来，如果我们要使用的插件非常多，那我们的gulpfile.js文件开头可能就会是这个样子的：

var gulp = require('gulp'),

//一些gulp插件,abcd这些命名只是用来举个例子

a = require('gulp-a'),

b = require('gulp-b'),

c = require('gulp-c'),

d = require('gulp-d'),

e = require('gulp-e'),

f = require('gulp-f'),

g = require('gulp-g'),

//更多的插件...

z = require('gulp-z');

虽然这没什么问题，但会使我们的gulpfile.js文件变得很冗长，看上去不那么舒服。gulp-load-plugins插件正是用来解决这个问题。  
gulp-load-plugins这个插件能自动帮你加载package.json文件里的gulp插件。例如假设你的package.json文件里的依赖是这样的:

{

"devDependencies": {

"gulp": "~3.6.0",

"gulp-rename": "~1.2.0",

"gulp-ruby-sass": "~0.4.3",

"gulp-load-plugins": "~0.5.1"

}

}

然后我们可以在gulpfile.js中使用gulp-load-plugins来帮我们加载插件：

var gulp = require('gulp');

//加载gulp-load-plugins插件，并马上运行它

var plugins = require('gulp-load-plugins')();

然后我们要使用gulp-rename和gulp-ruby-sass这两个插件的时候，就可以使用plugins.rename和plugins.rubySass来代替了,也就是原始插件名去掉gulp-前缀，之后再转换为驼峰命名。  
实质上gulp-load-plugins是为我们做了如下的转换

plugins.rename = require('gulp-rename');

plugins.rubySass = require('gulp-ruby-sass');

gulp-load-plugins并不会一开始就加载所有package.json里的gulp插件，而是在我们需要用到某个插件的时候，才去加载那个插件。  
最后要提醒的一点是，因为gulp-load-plugins是通过你的package.json文件来加载插件的，所以必须要保证你需要自动加载的插件已经写入到了package.json文件里，并且这些插件都是已经安装好了的。

**2. 重命名**

使用[gulp-rename](https://www.npmjs.com/packages/gulp-rename)  
安装：npm install --save-dev gulp-rename  
用来重命名文件流中的文件。用gulp.dest()方法写入文件时，文件名使用的是文件流中的文件名，如果要想改变文件名，那可以在之前用gulp-rename插件来改变文件流中的文件名。

var gulp = require('gulp'),

rename = require('gulp-rename'),

uglify = require("gulp-uglify");

gulp.task('rename', function () {

gulp.src('js/jquery.js')

.pipe(uglify()) //压缩

.pipe(rename('jquery.min.js')) //会将jquery.js重命名为jquery.min.js

.pipe(gulp.dest('js'));

//关于gulp-rename的更多强大的用法请参考https://www.npmjs.com/package/gulp-rename

});

**3. js文件压缩**

使用[gulp-uglify](https://www.npmjs.com/packages/gulp-uglify)  
安装：npm install --save-dev gulp-uglify  
用来压缩js文件，使用的是uglify引擎

var gulp = require('gulp'),

uglify = require("gulp-uglify");

gulp.task('minify-js', function () {

gulp.src('js/\*.js') // 要压缩的js文件

.pipe(uglify()) //使用uglify进行压缩,更多配置请参考：

.pipe(gulp.dest('dist/js')); //压缩后的路径

});

**4. css文件压缩**

使用[gulp-minify-css](https://www.npmjs.com/packages/gulp-minify-css)  
安装：npm install --save-dev gulp-minify-css  
要压缩css文件时可以使用该插件

var gulp = require('gulp'),

minifyCss = require("gulp-minify-css");

gulp.task('minify-css', function () {

gulp.src('css/\*.css') // 要压缩的css文件

.pipe(minifyCss()) //压缩css

.pipe(gulp.dest('dist/css'));

});

**5. html文件压缩**

使用[gulp-minify-html](https://www.npmjs.com/packages/gulp-minify-html)  
安装：npm install --save-dev gulp-minify-html  
用来压缩html文件

var gulp = require('gulp'),

minifyHtml = require("gulp-minify-html");

gulp.task('minify-html', function () {

gulp.src('html/\*.html') // 要压缩的html文件

.pipe(minifyHtml()) //压缩

.pipe(gulp.dest('dist/html'));

});

**6. js代码检查**

使用[gulp-jshint](https://www.npmjs.com/packages/gulp-jshint)  
安装：npm install --save-dev gulp-jshint  
用来检查js代码

var gulp = require('gulp'),

jshint = require("gulp-jshint");

gulp.task('jsLint', function () {

gulp.src('js/\*.js')

.pipe(jshint())

.pipe(jshint.reporter()); // 输出检查结果

});

**7. 文件合并**

使用[gulp-concat](https://www.npmjs.com/packages/gulp-concat)  
安装：npm install --save-dev gulp-concat  
用来把多个文件合并为一个文件,我们可以用它来合并js或css文件等，这样就能减少页面的http请求数了

var gulp = require('gulp'),

concat = require("gulp-concat");

gulp.task('concat', function () {

gulp.src('js/\*.js') //要合并的文件

.pipe(concat('all.js')) // 合并匹配到的js文件并命名为 "all.js"

.pipe(gulp.dest('dist/js'));

});

**8. less和sass的编译**

less使用[gulp-less](https://www.npmjs.com/packages/gulp-less),安装：npm install --save-dev gulp-less

var gulp = require('gulp'),

less = require("gulp-less");

gulp.task('compile-less', function () {

gulp.src('less/\*.less')

.pipe(less())

.pipe(gulp.dest('dist/css'));

});

sass使用[gulp-sass](https://www.npmjs.com/packages/gulp-sass),安装：npm install --save-dev gulp-sass

var gulp = require('gulp'),

sass = require("gulp-sass");

gulp.task('compile-sass', function () {

gulp.src('sass/\*.sass')

.pipe(sass())

.pipe(gulp.dest('dist/css'));

});

**9. 图片压缩**

可以使用[gulp-imagemin](https://github.com/sindresorhus/gulp-imagemin)插件来压缩jpg、png、gif等图片。  
安装：npm install --save-dev gulp-imagemin

var gulp = require('gulp');

var imagemin = require('gulp-imagemin');

var pngquant = require('imagemin-pngquant'); //png图片压缩插件

gulp.task('default', function () {

return gulp.src('src/images/\*')

.pipe(imagemin({

progressive: true,

use: [pngquant()] //使用pngquant来压缩png图片

}))

.pipe(gulp.dest('dist'));

});

gulp-imagemin的使用比较复杂一点，而且它本身也有很多[插件](https://www.npmjs.com/browse/keyword/imageminplugin)，建议去它的[项目主页](https://github.com/sindresorhus/gulp-imagemin)看看文档

**10. 自动刷新**

使用[gulp-livereload](https://github.com/vohof/gulp-livereload)插件，安装:npm install --save-dev gulp-livereload。  
当代码变化时，它可以帮我们自动刷新页面  
该插件最好配合谷歌浏览器来使用，且要安装[livereload chrome extension](https://chrome.google.com/webstore/detail/livereload/jnihajbhpnppcggbcgedagnkighmdlei)扩展插件,不能下载的请自行FQ。

var gulp = require('gulp'),

less = require('gulp-less'),

livereload = require('gulp-livereload');

gulp.task('less', function() {

gulp.src('less/\*.less')

.pipe(less())

.pipe(gulp.dest('css'))

.pipe(livereload());

});

gulp.task('watch', function() {

livereload.listen(); //要在这里调用listen()方法

gulp.watch('less/\*.less', ['less']);

});