Gulp

Automate and enhance your workflow

Try it now

前端技术分享
2016/09/18
尤桂芳

What is Gulp ?

yidakou

- ☐ 合并文件
- ☐ 压缩文件
- ☐ 编译文件
- ☐ 语法检查
- ☐ 监测变化
- ☐ 自动更新
- ☐ 代码测试
- ☐ ……
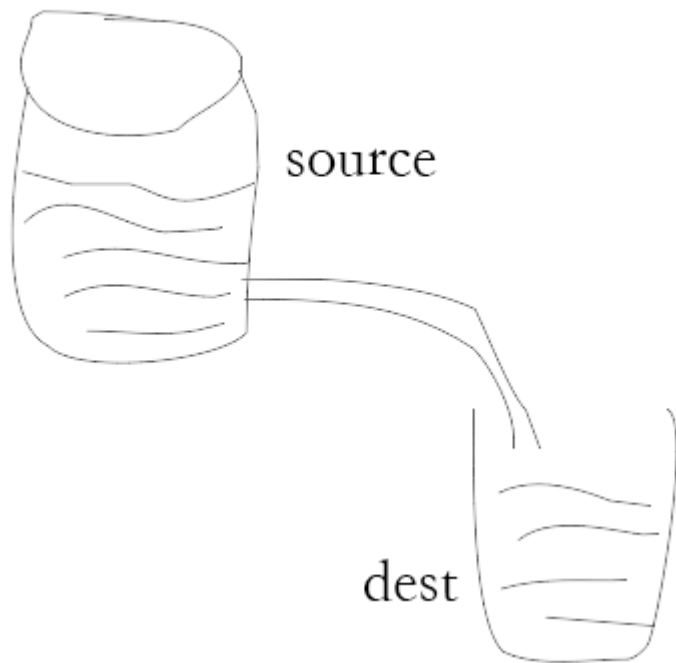
自动化构建工具

重复的事情流程化；

流程的事情标准化；

标准的事情自动化。
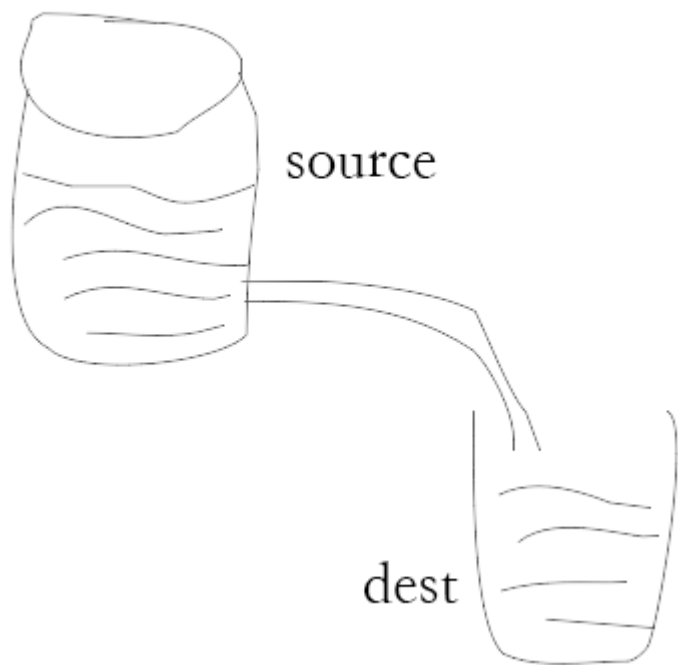
☐ source：源头

☐ dest：终点

☐ pipe：管道

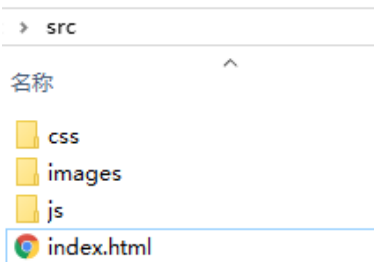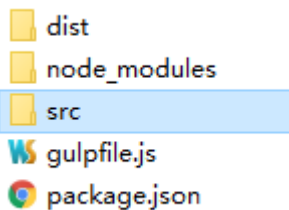# 工作原理：Stream


```
gulp.task('minifyHTML', function(){
    gulp.src('src/*.html')
        .pipe(minifyHTML({collapse
Whitespace: true}))
        .pipe(gulp.dest('dist'));
});
```

Let's try it

- *搭建项目结构*

  dist
  node_modules
  src
  gulpfile.js
  package.json

  > src
  名称
  css
  images
  js
  index.html

- *安装gulp及所需插件*

  npm install gulp gulp-htmlmin gulp-uglify gulp-minify-css gulp-imagemin gulp-concat gulp-css-spriter del

  gulp-less --save

- *配置gulpfile*

压缩*HTML*: gulp-htmlmin

```
//压缩html文件
var gulp = require('gulp');
var minifyHTML = require('gulp-htmlmin');
gulp.task('minifyHTML', function(){
gulp.src('src/*.html')
.pipe(minifyHTML({collapseWhitespace: true}))
.pipe(gulp.dest('dist'));
});
```

压缩前

```
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4       <meta charset="UTF-8">
5       <title>gulp测试</title>
6   </head>
7   <body>
8   <h1>Hello World!</h1>
9   <script src="js/a.js"></script>
10  <script src="js/b.js"></script>
11  </body>
12  </html>
```

压缩后

压缩JS: gulp-uglify

```
//压缩并合并js
var uglify = require('gulp-uglify');
var concat = require('gulp-concat');
gulp.task('script', function () {
    gulp.src('src/js/*.js').pipe(uglify())
        .pipe(concat(timeStamp +
".min.js"))//全部合并到这个文件里
        .pipe(gulp.dest('dist/js'));
});
```

压缩前

t > src > js

名称

a.js
b.js

压缩后

pt > dist > js

名称

1474198742887.min.js

```
!function(){var o="testa";console.log("This is ",o)}();
!function(){var o="testb";console.log("This is ",o)}();
```

```
l<script src="js/1474198742887.min.js"></script></body>
```

压缩*css:* gulp-minify-css

```
//压缩CSS文件
    var gulp = require('gulp');
var minifyCSS = require('gulp-minify-
css ');
gulp.task('css', function () {
    gulp.src('src/css/*.css')
        .pipe(minifyCSS())
        .pipe(concat('all.css'))
        .pipe(gulp.dest('dist/css'));
});
```

压缩前

ot > src > css

名称

test1.css
test2.css

压缩后

pt > dist > css

名称

all.css

;itle>gulp测试</title><link rel="stylesheet" href="css/all.css">;</head><b

*{margin:0;padding:0}body{width:100%;background-color:#08c}
table{margin:0;padding:0}h1{width:100%;background-color:#ccc;margin:100px 0}

合并图片: gulp-imagemin

```
//压缩图片
var imagemin = require('gulp-
imagemin');
gulp.task('images', function () {
   gulp.src('src/images/*.*')
      .pipe(imagemin({
         progressive: true
      }))
      .pipe(gulp.dest('dist/images'));
});
```

压缩前

压缩后

合并icon: gulp-css-spriter

```
//合并icon雪碧图
var spriter = require('gulp-css-spriter');
gulp.task('sprite', function () {
    var spritTimeStamp=new Date().getTime();
    gulp.src('src/css/**.css')//需要合并的图片地址
        .pipe(spriter({
            // 生成的spriter的位置
            'spriteSheet':
'dist/images/sprite/sprite'+spritTimeStamp+'.png',
            // 生成样式文件图片引用地址的路径
            // 如下将生产:
backgound:url(../images/sprite20324232.png)
            'pathToSpriteSheetFromCSS':
'../images/sprite/sprite'+spritTimeStamp+'.png'
        }))
        .pipe(minifyCSS())
        .pipe(concat('all.css'))
        // 产出路径
        .pipe(gulp.dest('dist/css/'));
});
```

压缩前
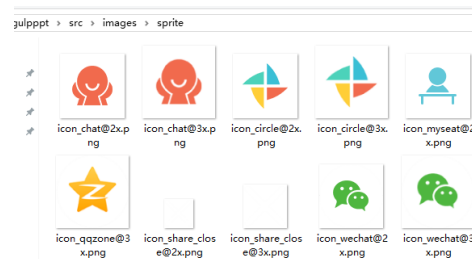
压缩后
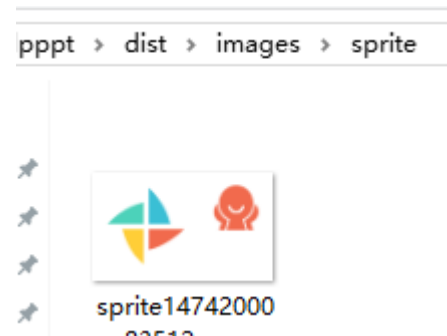
清空文件： del

```
//清空
var del = require('del');
gulp.task('clean', function (cb) {
    del([
        'dist/*'
    ], cb);
});
```

压缩前

▼ 📁 js
　　📄 1474199615848.min.js
　　📄 1474199649551.min.js
　　📄 1474199773571.min.js
　　📄 1474199785842.min.js
　　📄 1474199916648.min.js
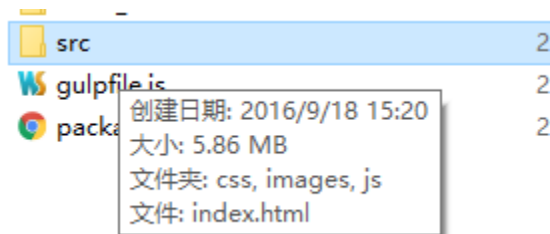　　📄 1474200038713.min.js
　　📄 1474200083029.min.js
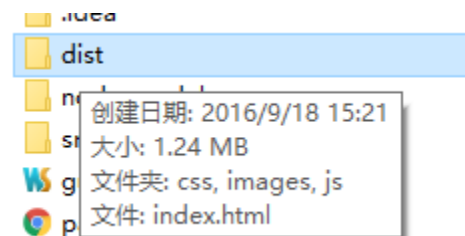
压缩后

▼ 📁 js
　　📄 1474200759856.min.js
　📄 index.html

The End

压缩前：5.86M

📁 src		2
W̲S̲ gulpfile.js		2
🔵 packa		2

创建日期: 2016/9/18 15:20
大小: 5.86 MB
文件夹: css, images, js
文件: index.html

about 2.6ms ……

压缩后：1.24M

📁 .idea
📁 dist
📁 n
📁 sr
W̲S̲ g
🔵 p

创建日期: 2016/9/18 15:21
大小: 1.24 MB
文件夹: css, images, js
文件: index.html

Gulp

Gulp vs Grunt

- 更快
- 更语义化
- 代码优于配置
- 基于流（防爆仓）
- ……

Thanks **Gulp**