

## 基础案例实战

### 1. 打包压缩合并js文件成一个文件 压缩成 一个JS

```
// 打包压缩合并js文件成一个文件 压缩JS

let gulp = require("gulp");
let uglify = require("gulp-uglify");
let concat = require("gulp-concat");

gulp.task("default", function() {
  // 将你默认的任务代码放在这里
  console.log("Ok");

  return gulp.src("./js/*.js")
    .pipe(uglify()) //压缩 获取到的js 不写不压缩
    .pipe(concat("all.min.js")) //将该目录下所有的js文件合并到一个名为all.min.js的文件
    .pipe(gulp.dest("./dist/js/")); //将all.min.js文件输出到dist/js/目录下
});
```

### 2. 打包合并css文件

```
let gulp = require("gulp");
let uglify = require("gulp-uglify"); // 压缩 js
let concat = require("gulp-concat"); // 合并文件
let uglify_css = require("gulp-minify-css") //压缩css

gulp.task("default", function() {
  // 将你默认的任务代码放在这里
  console.log("Ok");

  return gulp.src("./css/*.css")
    .pipe(uglify_css()) //压缩 获取到的css 不写不压缩
    .pipe(concat("all.min.css")) //将该目录下所有的css文件合并到一个名为all.min.css的文件
    .pipe(gulp.dest("./dist/css/")); //将all.min.css文件输出到dist/js/目录下
});
```

### 3. 在打包合并js文件之后 才打包合并css文件

```
let gulp = require("gulp");
let uglify = require("gulp-uglify"); // 压缩 js
let concat = require("gulp-concat"); // 合并文件
let uglify_css = require("gulp-minify-css") //压缩css

gulp.task("scripts", function() {
  return gulp.src("./js/*.js")
    .pipe(uglify()) //压缩 获取到的js 不写不压缩
    .pipe(concat("all.min.js")) //将该目录下所有的js文件合并到一个名为all.min.js的文件
    .pipe(gulp.dest("./dist/js/")); //将all.min.js文件输出到dist/js/目录下
});

gulp.task("css", function() {
  return gulp.src("./css/*.css")
    .pipe(uglify_css()) //压缩 获取到的css 不写不压缩
    .pipe(concat("all.min.css")) //将该目录下所有的css文件合并到一个名为all.min.css的文件
    .pipe(gulp.dest("./dist/css/")); //将all.min.css文件输出到dist/js/目录下
});

gulp.task("my-tasks", gulp.series("scripts", "css", function() {
  console.log("js、css打包完毕!")
}));
```

### 4. 监听文件改变 根据改变的内容 做出对应的处理

```

let gulp = require("gulp");
let uglify = require("gulp-uglify"); // 压缩 js
let concat = require("gulp-concat"); // 合并文件
let uglify_css = require("gulp-minify-css") //压缩css

gulp.task("watch_js",function(done){
  let watcher = gulp.watch(["./js/*.js"])

  //文件被修改
  /*
    'add', 'addDir', 'change', 'unlink', 'unlinkDir', 'ready', 'error', 'all'
  */
  watcher.on("add", (path, stats) => {
    console.log(`File ${path} was added`); // File js\add.js was added
  })

  watcher.on("change", (path, stats) => {
    console.log(`File ${path} was changed`);
    console.log(stats)
  })
  //fs
  })
  // watcher.close()
  done()
})

/*gulp.task("watch_js",function(){
  return gulp.watch(["./js/*.js","./css/*.css","!./css/style.css"], gulp.series("scripts", "css"),
function(cb){
  console.log("js、css修改文件被修改，已经自动打包压缩完成!")
  cb()
  })
}) */

```

## 5. 使用新版gulp4 写法

```

const { task } = require('gulp');

const clean = function(cb) {
  // body omitted
  console.log("clean")
  cb();
};
clean.displayName = 'clean:all';

task(clean);

function build(cb) {
  // body omitted
  console.log("build")
  cb();
}
build.description = 'Build the project';
build.flags = { '-e': 'An example flag' };

task(build);

/*

D:\me\gulp\gulp-test>gulp --tasks
[16:35:32] Tasks for D:\me\gulp\gulp-test\gulpfile.js
[16:35:32] └─ clean:all
[16:35:32] └─ build      Build the project
[16:35:32]     -e        ...An example flag

*/

```

## 6. 编译 sass

1. 首先安装[ruby \(https://rubyinstaller.org/\)](https://rubyinstaller.org/)
2. 然后安装sass: `gem install sass`
3. 最后安装gulp-sass: `cnpm install --save-dev gulp-sass`

```
let gulp = require("gulp");
let babel = require("gulp-babel")
let uglify = require("gulp-uglify"); // 压缩 js
let concat = require("gulp-concat"); // 合并文件
let uglify_css = require("gulp-minify-css") //压缩css
let sass = require("gulp-ruby-sass")

gulp.task("task-sass", (done) => {
  sass("./scss/*.scss", {style: "compressed"})
    .pipe(concat("scss.min.css"))
    .pipe(gulp.dest("./dist/scss-css/"))
  done()
})
```