

title: Typescript 类型声明文件打包

date: 2019-09-24

tag: typescript

description: 组件打包 Typescript 类型文件

Tsc 命令生成声明文件

Tsc 通过 `declaration` 可以解析项目中的 ts 文件，得到 js 文件和 `*.d.ts` 声明文件。

```
// square.ts
function squar(a: number): number {
  return a * a
}
```

执行命令 `tsc --declaration square.ts` 可以得到两个文件

```
// square.js
function squar(a) {
  return a * a;
}
```

和

```
// square.d.ts
declare function squar(a: number): number;
```

配置 tsconfig.json 生成声明文件

```
{
  "compilerOptions": {
    "module": "commonjs",
    "outDir": "lib",
    "declaration": true,
  }
}
```

- `module` 打包模式
- `outDir` 输出目录
- `declaration` 是否生成声明文件

还有几个跟声明文件相关的选项

- `declarationDir` 设置生成 `*.d.ts` 文件的目录
- `declarationMap` 每个 `*.d.ts` 文件生成对应的 `*.d.ts.map(sourceMap)` 文件
- `emitDeclarationOnly` 仅仅生成 `*.d.ts` 文件

组件生成声明文件

声明文件放在 `types` 目录下

```
tsc --declaration -p ./ -t es2015 --emitDeclarationOnly --outDir types
```

确识别声明文件

1. tsc 自动生成，无须做其他配置，只需跟编译组件代码一起发布到 npm，就可以获取到类型提示；
2. 手动编写的声明文件
 - 通过 package.json 中的 types 指定

```
{  
  "types": "types/index.d.ts",  
}
```

指定 types 后，引用会找到 types/index.d.ts 作为此库的声明文件；

- 没有指定，会在跟目录下寻找 index.d.ts 文件，将此 作为此库的声明文件。

@types 管理

DefinitelyTyped

初始化

```
Generate these by running npx dts-gen --dt --name wizard-ui--template module
```