# **【延伸阅读】tsconfig.json编译器配置文档**

一般来说，项目的TS编译器配置全部存储在项目根目录下的tsconfig.json文件中。

编译器启动时，首先会读取tsconfig.json，以获取有关如何编译项目的说明（例如，要编译哪些源文件，在哪里存储输出等）。 不过，如果编译器配置文档保存在其他特殊位置，我们也可以通过使用-p选项告诉编译器配置文档的具体位置：



现在，我们来研究一下上图出现的比较常见的编译器/tsconfig选项：

## **1. “target”: “es5”**

这个属性定义了你编译后的目标javascript版本，一般来说，我们需要让他编译为es5，这样就可以被主流浏览器解读了。当然，你说我的react代码不是给浏览器看的，比如说，你使用的是react-native做作手机app，那么这里的选项可以选择es6。除了es5和es6，我们还有很多其他常见的选项，ES5, ES6/ES2015, ES2016, ES2017, ES2018, ES2019, ES2020, ESNext，等等等等

## **2. “lib”: [“dom”, “dom.iterable”, “esnext”]**

这个属性列出了编译期间需要被包括进来的库文件，通过这些库文件，告诉typescript编译器可以使用哪些功能。比如说，我们这里有一个dom的库文件，这个文件会告诉编译器dom api的接口，所以当我们在ts代码中使用dom的时候，比如说执行“document.getElementById("root")”这句话的时候，编译器就会知道该如何进行检查。如果我们不设置这个选项，那么编译器也有自己默认的库文件列表，一般来说是["dom", "es6","DOM.Iterable"]等等

## **3. “allowJs”: true**

允许混合编译JavaScript文件

## **4. “esModuleInterop”: true**

这个选项允许我们使用ES6的方式import默认文件。比如说，在没有开启这个选项时，我们需要这样写才能引用react:

import \* as React from 'react'

但是当我们开启了这个选项以后，import方式就与普通的JavaScript没有区别了，可以写为:

import React from 'react'

这样处理项目引入会更自然。

## **5. “module”: “esnext”**

这里配置的是我们代码的模块系统，比较常见的有Node.js的CommonJS系统，ES6标准的esnext系统，以及requirejs的AMD系统。我们这里使用的是ES6标准的esnext系统，不过如果把这里换成CommonJS也是可以的。

## **6. “moduleResolution”: “node”**

这个选项决定了我们编译器的工作方式，也决定了我们各个文件之间调用、import的工作流程。这里曾经有两个选项，"node" and "classic"，但是"classic"这个选项在2019年12月就已经废弃了。

## **7. “isolatedModules”: “node”**

开启这个选项以后，编译器会将每个文件作为单独的模块来使用。

## **“noEmit”: true,**

开启这个选项表示当发生错误的时候，编译器不会生成 JavaScript 代码。

## **9. “jsx”: “react”,**

显而易见，这个选项允许编译器支持编译react代码

## **10. “include”: [“src/\*\*/\*”]**

使用此选项列出我们需要编译的文件， “文件路径”选项需要文件的相对或绝对路径，例如： "\*\*" - 任意子目录 "\*" - 任意文件名 "?" - 只要字符跟随“?”，这个字符就会被视为可忽略字符 (e.g., "src/\*.tsx?"则同时指代"src/\*.tsx"与"src/\*.ts")

## **11. “files”: ["./file1.ts", “./file2.d.ts”, …]**

使用此选项列出编译器应始终包含在编译中的文件。无论是否使用“exclude”选项，都将会编译使用此选项内包括的所有文件。

## **12. “exclude”: [“node\_modules”, “\*\*/*/*.test.ts”]**

此选项将会列出从编译中排除的文件。它与“include”选项采取相同的模式，我们以使用此选项来过滤使用“include”选项指定的文件。 但是, “exclude”选项不会影响“files”选项。

通常，我们会排除node\_modules、测试文件、和编译输出目录。

如果省略此选项，编译器将使用“outDir”选项指定的文件夹。

如果没有同时指定“files”和“include”这两个选项，则编译器将编译根目录和任何子目录中的所有TS文件，但不包括使用“exclude”选项指定的文件。