

编译选项

选项	类型	默认值	
--allowJs	boolean	false	允许编译javascript
--allowSyntheticDefaultImports	boolean	module === "system"	允许从没有设置默认导出的模块中， 为了类型检查。
--allowUnreachableCode	boolean	false	不报告执行不到的代码
--allowUnusedLabels	boolean	false	不报告未使用的标签
--alwaysStrict	boolean	false	以严格模式解析并检查
--baseUrl	string		解析非相对模块名时， Resolution.md#base-url
--charset	string	"utf8"	输入文件的字符集
--declaration -d	boolean	false	生成相应的'.d.ts'文件
--declarationDir	string		生成声明文件的输出目录
--diagnostics	boolean	false	显示诊断信息。
--disableSizeLimit	boolean	false	禁用JavaScript引擎的 在输出文件的开头添加
--emitBOM	boolean	false	给源码里的装饰器添加 信息。
--emitDecoratorMetadata ^[1]	boolean	false	
--experimentalDecorators ^[1]	boolean	false	启用实验性的ES装饰器
--forceConsistentCasingInFileNames	boolean	false	不允许不一致包装器
--help -h			打印帮助信息。
--importHelpers	string		从tslib导入生成器 生成单个sourcemap
--inlineSourceMap	boolean	false	将代码与sourcemap inlineSourceMap或
--inlineSources	boolean	false	初始化TypeScript项 将每个文件作为单独
--init			在.tsx文件里支持 指定生成目标为rea
--isolatedModules	boolean	false	React.createElement
--jsx	string	"Preserve"	编译过程中需要引入 可能的值为：
--jsxFactory	string	"React.createElement"	

- ▶ ES5
- ▶ ES6
- ▶ ES2015
- ▶ ES7
- ▶ ES2016

--lib

string[]

- ▶ ES2017
- ▶ DOM
- ▶ DOM.Iterable
- ▶ WebWorker
- ▶ ScriptHost
- ▶ ES2015.Core
- ▶ ES2015.Collect
- ▶ ES2015.Generat
- ▶ ES2015.Iterabl
- ▶ ES2015.Promise
- ▶ ES2015.Proxy
- ▶ ES2015.Reflect
- ▶ ES2015.Symbol
- ▶ ES2015.Symbol.
- ▶ ES2016.Array.I
- ▶ ES2017.object
- ▶ ES2017.SharedM

--listEmittedFiles

boolean false

--listFiles

boolean false

--locale

string *(platform specific)*

--mapRoot

string

--maxNodeModuleJsDepth

number 0

--module

-m

string target === "ES6" ?
"ES6" : "commonjs"

注意：如果--lib没

▶ 针对于--target

▶ 针对于--target

打印出编译后生成

编译过程中打印文

显示错误信息时使

为调试器指定指定

当.map文件是在运

指定的路径会嵌入

node_modules依赖的

allowJs。

指定生成哪个模块

码："None"，"Com

▶ 只有"AMD"和"Sy

▶ "ES6"和"ES2015"

--moduleResolution

string `module === "AMD"

"System"

--newLine

string *(platform specific)*

--noEmit

boolean false

--noEmitHelpers

boolean false

当生成文件时指定

不生成输出文件。

不在输出文件中生

<code>--noEmitOnError</code>	boolean	false	报错时不生成输出
<code>--noFallthroughCasesInSwitch</code>	boolean	false	报告switch语句的fallthrough
<code>--noImplicitAny</code>	boolean	false	在表达式和声明上推断any类型
<code>--noImplicitReturns</code>	boolean	false	不是函数的所有返回语句都返回
<code>--noImplicitThis</code>	boolean	false	当this表达式的值不为any时，报告错误
<code>--noImplicitUseStrict</code>	boolean	false	模块输出中不包含'use strict'
<code>--noLib</code>	boolean	false	不包含默认的库文件
<code>--noResolve</code>	boolean	false	不把/// <reference> 语句放入输出
<code>--noUnusedLocals</code>	boolean	false	若有未使用的局部变量，报告错误
<code>--noUnusedParameters</code>	boolean	false	若有未使用的参数，报告错误
<code>--out</code>	string		弃用。使用 <code>--outFile</code>
<code>--outDir</code>	string		重定向输出目录。
<code>--outFile</code>	string		将输出文件合并为一个文件。和 <code>///<reference>`</code> 了解详情。
<code>paths</code> ^[2]	Object		模块名到基于 <code>baseUrl</code> 的 <code>Resolution.md#path-resolution</code>
<code>--preserveConstEnums</code>	boolean	false	保留 <code>const</code> 和 <code>enum</code> 声明
<code>--pretty</code> ^[1]	boolean	false	给错误和消息设置漂亮的格式
<code>--project</code>	string		编译指定目录下的 <code>.ts</code> 文件
<code>-p</code>	string		编译。查看 tsconfig 。
<code>--reactNamespace</code>	string	"React"	当目标为生成"react"时，指定命名空间
<code>--removeComments</code>	boolean	false	删除所有注释，除非 <code>--sourceMap</code>
<code>--rootDir</code>	string	<i>(common root directory is computed from the list of input files)</i>	仅用来控制输出的根目录
<code>rootDirs</code> ^[2]	string[]		根 (<i>root</i>) 文件夹列表
<code>--skipDefaultLibCheck</code>	boolean	false	文档](./Module Resolution.md#lib) 忽略[库的默认声明
<code>--skipLibCheck</code>	boolean	false	libtrue)的类型检查。
<code>--sourceMap</code>	boolean	false	忽略所有的声明文件
<code>--sourceRoot</code>	string		生成相应的 <code>.map</code> 文件
<code>--strictNullChecks</code>	boolean	false	指定TypeScript源文件
<code>--stripInternal</code> ^[1]	boolean	false	在运行时指定时使用
<code>--suppressExcessPropertyErrors</code> ^[1]	boolean	false	在严格的null检查时
<code>--suppressImplicitAnyIndexErrors</code>	boolean	false	许用它们自己和any

<code>--target</code> <code>-t</code>	<code>string</code> <code>"ES3"</code>	注意: "ESNext"最新
<code>--traceResolution</code>	<code>boolean</code> <code>false</code>	生成模块解析日志
<code>--types</code>	<code>string[]</code>	要包含的类型声明 详细信息。
<code>--typeRoots</code>	<code>string[]</code>	要包含的类型声明 解详细信息。
<code>--version</code> <code>-v</code>		打印编译器版本号
<code>--watch</code> <code>-w</code>		在监视模式下运行

- ^[1] 这些选项是试验性的。
- ^[2] 这些选项只能在`tsconfig.json`里使用，不能在命令行使用。

相关信息

- 在[tsconfig.json](#)文件里设置编译器选项。
- 在[MSBuild工程](./Compiler Options in MSBuild.md)里设置编译器选项。