

Microsoft Typescript 基础

讲师: indeex hsi (史红庆)

部门: 鲸棋项目部 (后端项目组)

职位: web前端工程师 (客户端开发)

职级: P5

Typescript介绍

TypeScript 是一种由微软开发的自由和开源的编程语言。它是 JavaScript 的一个实现（超集），而且本质上向这个语言添加了可选的静态类型和基于类的面向对象编程。

TypeScript 提供最新的和不断发展的 JavaScript 特性，包括那些来自 2015 年的 ECMAScript（2012年被否决的提案）和未来的提案中的特性，比如异步功能和Decorators，以帮助建立健壮的组件。

编译器本身是用TypeScript编写，Typescript可用在浏览器和服务端或其他环境中

Typescript 1.0.0-rc.0

Feature	Status
ES6/ES7	100%
ES8	100%
ES9	100%
ES10	100%
ES11	100%
ES12	100%
ES13	100%
ES14	100%
ES15	100%
ES16	100%
ES17	100%
ES18	100%
ES19	100%
ES20	100%
ES21	100%
ES22	100%
ES23	100%
ES24	100%
ES25	100%
ES26	100%
ES27	100%
ES28	100%
ES29	100%
ES30	100%
ES31	100%
ES32	100%
ES33	100%
ES34	100%
ES35	100%
ES36	100%
ES37	100%
ES38	100%
ES39	100%
ES40	100%
ES41	100%
ES42	100%
ES43	100%
ES44	100%
ES45	100%
ES46	100%
ES47	100%
ES48	100%
ES49	100%
ES50	100%
ES51	100%
ES52	100%
ES53	100%
ES54	100%
ES55	100%
ES56	100%
ES57	100%
ES58	100%
ES59	100%
ES60	100%
ES61	100%
ES62	100%
ES63	100%
ES64	100%
ES65	100%
ES66	100%
ES67	100%
ES68	100%
ES69	100%
ES70	100%
ES71	100%
ES72	100%
ES73	100%
ES74	100%
ES75	100%
ES76	100%
ES77	100%
ES78	100%
ES79	100%
ES80	100%
ES81	100%
ES82	100%
ES83	100%
ES84	100%
ES85	100%
ES86	100%
ES87	100%
ES88	100%
ES89	100%
ES90	100%
ES91	100%
ES92	100%
ES93	100%
ES94	100%
ES95	100%
ES96	100%
ES97	100%
ES98	100%
ES99	100%
ES100	100%



Microsoft Typescript 基础

讲师: indeex hsi (史红庆)

部门: 鲸祺项目部 (后端项目组)

职位: web前端工程师 (客户端开发)

职级: P5

在日常使用中，下列哪些词是**对的**，
那些是**错的**？

A: 一二三三

B: 壹式叁肆

C: 喝沏

D: 丶滴

E: 彦色

F: 公司

总结：历史遗留，时代不同，标准不同

Typescript介绍

TypeScript 是一种由微软开发的自由和开源的编程语言。它是 JavaScript 的一个实现（超集），而且本质上向这个语言添加了可选的静态类型和基于类的面向对象编程。

TypeScript 提供最新的和不断发展的 JavaScript 特性，包括那些来自 2015 年的 ECMAScript（2012 年被否决的提案）和未来的提案中的特性，比如异步功能和 Decorators，以帮助建立健壮的组件。

编译器本身是用 TypeScript 编写，Typescript 可用在浏览器和服务端或其他环境中

TypeScript与JavaScript区别

TypeScript	JavaScript
JavaScript 的超集用于解决大型项目的代码复杂性	一种脚本语言，用于创建动态网页
可以在编译期间发现并纠正错误	作为一种解释型语言，只能在运行时发现错误
强类型，支持静态和动态类型	弱类型，没有静态类型选项
最终被编译成 JavaScript 代码，使浏览器可以理解	可以直接在浏览器中使用
支持模块、泛型和接口	不支持模块，泛型或接口
支持 ES3, ES4, ES5 和 ES6 等	不支持编译其他 ES3, ES4, ES5 或 ES6 功能
社区的支持仍在增长，而且还不是很大	大量的社区支持以及大量文档和解决问题的支持

Javascript发展中的问题和解决方案

1990年：Line Mode浏览器发布，使用指令操作，并获得（W3C）支持

1993年：Arena浏览器发布，支持CSS

1994年：Amaya浏览器，最初只是HTML和CSS编辑器，后支持操作XML、XHTML、MathML，可显示PNG图片、SVG矢量图形和播放动画

1995年：Brendan Eich就职于Netscape，十天开发Mocha，后改名Livescript，与同年出生的Java一样成为web开发者的首选语言，同年12月联合Sun Microsystems改名为Javascript

1996年：8月Microsoft发布IE3.0与Jscript（唯一有离线文档的实现（超集）），此时ScriptEase脚本也拥有大量开发者，与网景展开竞争。同年11月修复了大量bug和完善了大量设计及内置了大量方法和属性的Javascript1.1找到ECMA，并委派第三十九技术委员会（TC39）制定ECMA-262标准

1997年：6月被ECMA组织标准化的ECMA-262标准，命名为ECMAScript，Jscript进入淘汰期

2000年：8月Macromedia发布基于ECMA-262标准的ActionScript 1.0，同年年底W3C发布XHTML1.0

2005年：12月Adobe正式并购Macromedia

2006年：6月Adobe发布Action script3.0，同年Unity发布Unityscript

2007年：杭州天地会成立，Actionscript开发在中国大肆流行，随后全球99%网站被Actionscript技术占领

2008年：在全球开发者的努力下，HTML5发布

2009年：基于Actionscript的技术达到高峰，导致大量其他软件行业从业人员大量失业，包括只会Jquery和CSS样式排版的其他Web从业人员，同年12月，ECMAScript 5.0版正式发布，其他先进草案被第一次否决

2010年：4月由于ActionScript太过强大，为了维护自己的霸主地位和Appstore的生态，乔布斯发布《Flash之我见》（该言论由于言辞过于激烈和大量不实诋毁，已被Apple撤掉）联合其他行业巨头一起干掉Flash，Actionscript进入淘汰期

2011年：3月发布Internet Explorer 9.0，同时发布基于ECMA-262 5th edition的Jscript9.0，同年6月ECMAScript 5.1发布，其他先进草案全部被第二次否决

2012年：Facebook宣布放弃HTML5，同年Microsoft发布Typescript

2013年：React库发布

2014年：由个人实现的小型库Vue库beta开始被部分开发者关注并使用

2015年：6月ECMAScript 6发布，改名为ECMAScript 2015，以后均已ECMAScript+年命名

2017年：8月28日杭州天地会关站

2020年：故事你来写。。。

Typescript优点：

- 1.类型系统，类型系统能确保你的程序不会进入无效状态，编译阶段解决掉低级错误
- 2.可编译为旧浏览器支持的Javascript
- 3.代码可读性和维护性极大提高
- 4.更快和更简洁的编写组件
- 5.更像编程语言

Typescript缺点：

- 1.不健全的类型系统，尤其使用any、unknown、never类型时
官方：应用一个健全或“正确无误”的类型系统（不是我们的目标）。相反，要在正确性和生产力之间取得平衡。
- 2.Typescript非标准，非官方，只是属于某个组织
- 3.复杂（大）项目编译较慢
- 4.浏览器或服务器无法直接支持和使用
- 5.类型推断非严格型，复杂项目需要单元测试
- 6.流行即淘汰，2020年Typescript已经进入淘汰期，相继还有Nodejs

其他问题：

- 1.现有项目迁移工作量、成本巨大，大量第三方库不支持
- 2.学习成本曲线陡峭
- 3.大量web排版人员不接受
- 4.没有完善和全面的学习社区

安装、使用

官网: <https://www.typescriptlang.org/>

安装: 1.(C)NPM ((c)npm install -g typescript)

2.YARN (yarn global add typescript)

3.IDE插件 (如Visual Studio、WebStrom、HbuilderX)

使用: tsc helloIndeex.ts (编译为helloIndeex.js)

引入: 将编译后的js文件引入浏览器环境、服务器环境或其他环境

集成: 1.Rollup

2.Parcel

3.Webpack

4.Grunt

5.Gulp

6.Deno

7.Go

8.C/C++

Typescript语法

1. 类型注释和编译时类型检查

```
function add(left: number, right: number): number {  
    return left + right;  
}
```

2. 头文件（声明文件，类似于C/C++中的头文件）和命名空间

```
declare namespace arithmetics {  
    add(left: number, right: number): number;  
    subtract(left: number, right: number): number;  
    multiply(left: number, right: number): number;  
    divide(left: number, right: number): number;  
}
```

3. 类（ECMAScript 2015移植）

```
class Person {  
    private name: string;  
    private age: number;  
    private salary: number;  
    constructor(name: string, age: number, salary: number) {  
        this.name = name;  
        this.age = age;  
        this.salary = salary;  
    }  
    toString(): string {  
        return `${this.name} (${this.age}) (${this.salary})`;  
    }  
}
```

4.模块（ECMAScript 2015移植）

命名空间namespace利用JavaScript的立即调用函数表达式来封装代码（内部模块）

模块module利用JavaScript库模式来封装代码（AMD或CommonJS，外部模块）

5.接口

早期其他JS实现（超集）定义：接口是定义一个方法组的数据类型，其中的方法必须由实现接口的任何类定义

菜鸟教程定义：接口是一系列抽象方法的声明，是一些方法特征的集合，这些方法都应该是抽象的，需要由具体的类去实现，然后第三方就可以通过这组抽象方法调用，让具体的类执行具体的方法

语法：interface InterfaceName [extends InterfaceName] {}

注意：其他JS实现（超集）必须实现接口使用，而Typescript使用比较灵活

6.枚举

7.泛型

```
function doSomething<T>(arg: T): T {  
    return arg;  
}
```

详细使用：<https://blog.indeex.club>

值类型、引用类型、装饰器、映射、反射等其他语法和相关ECMAScript标准移植语法详见官网

TODOLIST

功能：增加、删除、修改、搜索

预览：todolist

相关技术栈：typescript、HTML、CSS、浏览器存储、HTTP
通讯、Node、Webpack

哇哈哈

哇哈哈2

哇哈哈3

哇哈哈888
