# TypeScript

## 安装Typescript

npm I –g typescript

## 第一个typescript

### greeter.ts

class User {

fullName: string

firstName: string

lastName: string

constructor(firstName: string, lastName: string) {

this.firstName = firstName

this.lastName = lastName

this.fullName = firstName + ' ' + lastName

}

}

interface Person {

firstName: string,

lastName: string

}

function greeter(person: Person) {

return 'hello ' + person.firstName+ ' '+ person.lastName

}

let user = new User('yan', 'qiang')

console.log(greeter(user))

### 运行

cmd 中

tsc greeter.ts

运行后会生成一个greeter.js

var User = /\*\* @class \*/ (function () {

function User(firstName, lastName) {

this.firstName = firstName;

this.lastName = lastName;

this.fullName = firstName + ' ' + lastName;

}

return User;

}());

function greeter(person) {

return 'hello ' + person.firstName + ' ' + person.lastName;

}

var user = new User('yan', 'qiang');

console.log(greeter(user));

# 初识 TypeScript

TypeScript 作为 JavaScript 语言的超集，它为 JavaScript 添加了可选择的类型标注，大大增强了代码的可读性和可维护性。同时，它提供最新和不断发展的 JavaScript 特性，能让我们建立更健壮的组件。

## [#](http://localhost:8080/ts-axios/chapter1/#typescript-%E7%9A%84%E7%89%B9%E7%82%B9)TypeScript 的特点

TypeScript 主要有 3 大特点：

* **始于JavaScript，归于JavaScript**

TypeScript 可以编译出纯净、 简洁的 JavaScript 代码，并且可以运行在任何浏览器上、Node.js 环境中和任何支持 ECMAScript 3（或更高版本）的JavaScript 引擎中。

* **强大的工具构建大型应用程序**

类型允许 JavaScript 开发者在开发 JavaScript 应用程序时使用高效的开发工具和常用操作比如静态检查和代码重构。

类型是可选的，类型推断让一些类型的注释使你的代码的静态验证有很大的不同。类型让你定义软件组件之间的接口和洞察现有 JavaScript 库的行为。

* **先进的 JavaScript**

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 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。

这些特性为高可信应用程序开发时是可用的，但是会被编译成简洁的 ECMAScript3（或更新版本）的JavaScript。

## [#](http://localhost:8080/ts-axios/chapter1/" \l "%E6%80%BB%E7%BB%93)总结

TypeScript 在社区的流行度越来越高，它非常适用于一些大型项目，也非常适用于一些基础库，极大地帮助我们提升了开发效率和体验。都 2019 年了，如果你还没有开始学习 TypeScript，那么你可能要落后了哟，所以还等什么，快来和我一起学习并使用 TypeScript 吧，来感受一下它为我们带来的奇妙体验。

# 安装 TypeScript

命令行运行如下命令，全局安装 TypeScript：

npm install -g typescript

安装完成后，在控制台运行如下命令，检查安装是否成功(3.x)：

tsc -V