



黑马程序员™
www.itheima.com

传智播客旗下
高端IT教育品牌

变量和数据类型

目录 Contents

- ◆ 什么是变量
- ◆ 变量的使用
- ◆ 类型注解
- ◆ 变量的命名规范
- ◆ 数据类型概述
- ◆ 基本数据类型

1. 什么是变量

通俗的讲：变量，是用来存储数据的容器，并且是可以变化的。

保温杯里泡枸杞，变量里面存数据。

枸杞难挡岁月催，杯里再换点当归：枸杞 -> 当归。



目录

Contents

- ◆ 什么是变量
- ◆ 变量的使用
- ◆ 类型注解
- ◆ 变量的命名规范
- ◆ 数据类型概述
- ◆ 基本数据类型

2. 变量的使用

2.1 基本使用

变量的使用分为两步：1 声明变量并指定类型 2 给变量赋值。

第一步：声明变量并指定类型

```
let age: number;
```

解释：

- `let` 是 TS 的关键字，用来声明变量。
- `age` 是程序员自己定义的变量名称。
- `: number` 用来指定变量 `age` 为数值类型。
- 注意：分号是可选的，可省略。

2. 变量的使用

2.1 基本使用

变量的使用分为两步：1 声明变量并指定类型 2 给变量赋值。

第一步：声明变量并指定类型

```
let age: number;
```

第二步：给变量赋值

```
age = 18
```

解释：

- 使用等号 (=) 来给变量 age 赋值。

2. 变量的使用

2.2 简化形式

- 声明变量的同时就赋值（变量的初始化）：

```
let age: number = 18
```



2. 变量的使用

变量的使用总结：

- 变量的两种使用方式：1 先声明后赋值 2 声明的同时并赋值（推荐）
- 第一种：先声明后赋值

```
// 1 声明变量并指定类型  
let age: number  
  
// 2 赋值  
age = 18
```

第二种：声明的同时并赋值（推荐）

```
let age: number = 18
```

注意：声明变量的时候要指定变量的类型。

目录 Contents

- ◆ 什么是变量
- ◆ 变量的使用
- ◆ 类型注解
- ◆ 变量的命名规范
- ◆ 数据类型概述
- ◆ 基本数据类型

■ 3. 类型注解

3.1 什么是类型注解

示例代码：

```
let age: number = 18
```

代码中的 `: number` 就是类型注解。

3. 类型注解

3.2 作用

类型注解：是一种为变量添加类型约束的方式。

程序员 和 TS 有个约定。

```
// 程序员 和 TS 约定：变量 age 的类型为 number (数值类型)  
let age: number = 18
```

- **重要**：约定了什么类型，就只能给变量赋什么类型的值。

```
// ✖ 错误演示  
age = 'Hello TS'
```

额外说明：VSCode 中代码下方的红色波浪线表示代码报错，出 Bug 了。

目录 Contents

- ◆ 什么是变量
- ◆ 变量的使用
- ◆ 类型注解
- ◆ 命名规则和规范
- ◆ 数据类型概述
- ◆ 基本数据类型



4. 命名规则和规范

4.1 变量的命名规则

- 变量名称只能出现：数字、字母、下划线（_）、美元符号（\$），并且不能以数字开头。

```
let age: number = 18
```

以下示例中变量名称是否正确：

```
let 2b ✖  
let $name  
let first_name  
let @email ✖
```

注意：变量名称区分大小写。

```
// age 和 Age 是两个不同的变量  
let age: number = 18  
let Age: number = 20
```



4. 命名规则和规范

4.2 变量的命名规范

- 代码就是程序员的脸面，规范的代码，能让人觉得专业、优雅。
- 推荐：变量名称要有意义，顾名思义。

```
// 有意义，推荐  
let age: number = 18  
  
// 无意义，不推荐  
let a: number = 18
```

- 推荐：使用驼峰命名法（首字母小写，后面每个单词首字母大写）。

```
let cityName  
let yourAge
```



4. 命名规则和规范

总结：

- 规则：变量名称只能出现 数字、字母、下划线（_）、美元符号（\$），并且不能以 数字 开头。

注意：变量名称区分大小写。

- 规范：变量名称要有意义，顾名思义。

推荐：使用驼峰命名法（首字母小写，后面每个单词首字母大写）。



交换变量的值

- 需求：交换两个变量的值（提示：借助于第三个变量）。

```
// 变量1:  
let num1: number = 33  
  
// 变量2:  
let num2: number = 2  
  
// 目标: num1 的值为2, num2 的值为33。
```



目录

Contents

- ◆ 什么是变量
- ◆ 变量的使用
- ◆ 类型注解
- ◆ 命名规则和规范
- ◆ 数据类型概述
- ◆ 基本数据类型



5. 数据类型概述

平时浏览的网页中都有什么形式的内容啊？ 文字、图片、视频、音乐等

问题：计算机可以正确的处理这些内容，它是如何区分这些不同形式的内容呢？

回答：数据类型。

- TypeScript 中的数据类型分为两大类：1 原始类型（基本数据类型） 2 对象类型（复杂数据类型）。

5. 数据类型概述

平时浏览的网页中都有什么形式的内容啊？ 文字、图片、视频、音乐等

问题：计算机可以正确的处理这些内容，它是如何区分这些不同形式的内容呢？

回答：**数据类型**。

- TypeScript 中的数据类型分为两大类：1 **原始类型（基本数据类型）** 2 对象类型（复杂数据类型）。
- 常用的基本数据类型有 5 个：**number** / **string** / **boolean** / **undefined** / **null**。

```
// 变量 age 的类型是 number (数字类型)  
let age: number = 18
```

```
// 此处的 'Hello TS' 是 string (字符串类型)  
console.log('Hello TS')
```

目录

Contents

- ◆ 什么是变量
- ◆ 变量的使用
- ◆ 类型注解
- ◆ 命名规则和规范
- ◆ 数据类型概述
- ◆ 基本数据类型

6. 基本数据类型

6.1 数字类型

- 数字类型：包含整数值和浮点型（小数）值。

```
// 数字类型：整数
let age: number = 18
// 数字类型：小数
let score: number = 99.9
```

- 当然，从另一个维度来看，也可以包含：正数和负数。

```
// 正数，+直接省略即可
let salary: number = +10000
// 负数
let salaryWithGirlfriend: number = -2000
```

6. 基本数据类型

6.2 字符串类型

- 字符串：由零个或多个字符串联而成的，用来表示文本信息。

```
console.log('Hello TS')  
console.log('') // 空字符串
```

- 字符串可以使用单引号 (') 或双引号 (")，推荐：使用单引号。

```
'Hello TS' // 推荐  
"Hello TS"
```

- 字符串类型的类型注解为：string，声明变量时要添加类型注解。

```
let food: string = '糖葫芦'
```





6. 基本数据类型

6.3 布尔类型

布尔类型，用来表示真或假。

只有两个值，分别是：`true` 和 `false`。`true` 表示真，`false` 表示假。

- 布尔类型的类型注解为：`boolean`。

```
// 真
let isStudying: boolean = true
// 假
let isPlayingGame: boolean = false
```



6. 基本数据类型

6.4 undefined、null

共同特点：只有一个值，值为类型本身。

`undefined` 类型的值为： `undefined`。

`null` 类型的值为： `null`。

```
// 类型注解为: undefined
let u: undefined = undefined

// 类型注解为: null
let n: null = null
```


6. 基本数据类型

6.4 undefined、null

- undefined: 表示声明但未赋值的变量值（找不到值）。

```
let u: undefined  
console.log(u) // 变量u的值为 undefined
```

- null: 表示声明了变量并已赋值，值为 null（能找到，值就是 null）。



6. 基本数据类型

总结

- TypeScript 中常用的基本数据类型有 5 个。
- 分别是：number（数字类型） / string（字符串类型） / boolean（布尔类型） / undefined / null。
- 重点是：number / string / boolean。

补充概念：这些类型的值，也叫做字面量，也就是从字面上就能看出来它是什么。

```
18           // 数字字面量
'保温杯里泡枸杞' // 字符串字面量
true / false   // 布尔字面量
undefined
null
```



传智播客旗下高端IT教育品牌