

- **值类型(基本类型):**

字符串 (String) 、 数字(Number)、 布尔(Boolean)、 对空 (Null) 、 未定义 (Undefined) 、 Symbol。

Symbol 是 ES6 引入了一种新的原始数据类型，表示独一无二的值。

JavaScript 变量均为对象。当您声明一个变量时，就创建了一个新的对象。

JavaScript 拥有动态类型。这意味着相同的变量可用作不同的类型：

实例

var x; // x 为 undefined

var x = 5; // 现在 x 为数字

var x = "jam"; // 现在 x 为字符串

- **字符串 (String)**

字符串可以是引号中的任意文本。

实例：

```
var h_name="XC60";
```

- **数字 (Number)**

数字可以带小数点，也可以不带：（没有那么严格）

实例

```
var x1=34.00; //使用小数点
```

```
var x2=34; //不使用小数点
```

```
var y=123e5; // 12300000 (可以使用科学计数法)
```

```
var z=123e-5; // 0.00123
```

- **布尔 (Boolean)**

```
var x=true;
```

```
var y=false; //布尔（逻辑）只能有两个值：true 或 false。
```

- **Undefined 和 Null**

Undefined 这个值表示变量不含有值。

可以通过将变量的值设置为 null 来清空变量。

```
cars=null;  
person=null;
```

- **Symbol**

实例是唯一且不可改变的；一种唯一标识符，可用作对象的唯一属性名，这样其他人就不会改写或覆盖你设置的属性值。

```
let id = Symbol("id");
```

数据类型的特点：唯一性，即使是用同一个变量生成的值也不相等。

```
let id1 = Symbol("id");
```

```
let id2 = Symbol("id");
```

```
console.log(id1 == id2); //false
```

数据类型的特点：隐藏性，for...in, object.keys() 不能访问。

能够访问的方法：Object.getOwnPropertySymbols，会返回一个数组，成员是当前对象的所有用作属性名的 Symbol 值。