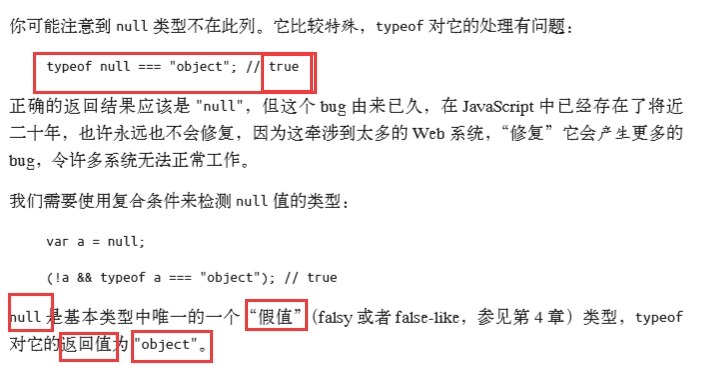
第一章 类型

1.2内置类型

1. JavaScript有7种内置类型：空值（null），未定义（undefined），布尔型（Boolean）

数字（number），字符串（string），对象（object），符号（symbol，ES6中新增）

2. 除对象外，其他统称为“基本类型”，可以用typeof 查看值的类型，它返回的是类型的字符串值



4. 数组也是一个对象，是“object”的一个子类型，typeof [1,2,3] === "object"; // true

5. 函数也是一个对象，是“object”的一个子类型，还可以拥有属性，

typeof function a(){ /\* .. \*/ } === "function"; // true

1.3 值和类型

JavaScript 中的变量是没有类型的，只有值才有。变量可以随时持有任何类型的值。

typeof 运算符总是会返回一个字符串：typeof typeof 42; // "string"

1.3.1 　undefined 和 undeclared（未声明）

1）变量在未持有值的时候为 undefined。此时 typeof 返回 "undefined"：

2）已在作用域中声明但还没有赋值的变量，是 undeﬁned 的。相反，还没有在作

用域中声明 过的变量，是 undeclared 的。

3）对于 undeclared（或者 not deﬁned）变量，typeof 照样返回 "undefined"。请

注意虽然 b 是 一个 undeclared 变量，但 typeof b 并没有报错。这是因为 typeof 有一个特殊的安全防范 机制。

1.3.2 　typeof Undeclared

1）通过 typeof 的安全防范机制（阻止报错）来检查 undeclared 变量，有时是个不错的办法