



链滴

# JavaScript 的表达式和语句有什么区别?

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1547453774159>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 回答

JavaScript 有两种主要的语法：表达式和语句。还有一种是表达式语句，即在语句中写入表达式。可述为：

- **表达式**：产生一个值
- **语句**：执行一个操作
- **表达式语句**：产生一个值并执行一个操作

一般的经验法则：

如果他有输出值或者可以把他赋值给一个变量，那么他就是一个表达式。否则就是一个声明。

## 语句

```
let x = 0
function declaration() {}
if (true) {
}
```

语句作为指令可以显示的执行一些操作，但不会产生值。

```
let x
if (y >= 0) {
  x = y
} else {
  x = -y
}
```

以上代码中仅 `y >= 0` 是表达式：他产生了一个值，要么 `true`，要么 `false`。而其他部分的代码不会成值。

## 表达式

表达式会产生一个值，因此他们可以传递给函数。这是因为解释器可以用解析出的值去替换表达式。

```
5 + 5 // => 10
lastCharacter("input") // => "t"
true === true // => true
```

## 表达式语句

当示例二中的 `if` 语句使用条件（三元）运算符简写后就可以成为表达式语句：

```
var x = y >= 0 ? y : -y
```

这既是表达式又是语句，因为我们声明了一个变量 `x`（语句），并将产生值赋予了他（表达式）

## 加分回答

- 分号可以结束一个语句 `foo(); bar()`, 逗号会对表达式求值并且返回第二个表达式的结果 `foo(), bar)`
- JavaScript 有以下五种类型的表达式:
- 算数: 得出一个数字, 如 3.14159
- 字符串: 得出一个字符串, 如 "Fred" 或 "234"
- 逻辑值: 得出 true 或 false
- 基本表达式: JavaScript 中的关键字和一般表达式
- 左值表达式: 分配给左值
- 函数声明 `function foo() {}`; 可以在声明之前使用
- 函数表达式 `let bar = function baz() {}`; 可以省略函数名称且定义时就会被运行

## 返回总目录

每天 30 秒