

JS作用域（全局作用域+局部作用域）

在 JavaScript 中，您可以在任意位置声明变量，但不同的位置会影响变量的可用范围，这个范围称为作用域。作用域可以大致分为两种类型，分别是全局作用域和局部作用域。下面就来分别介绍一下。

JS 全局作用域

全局作用域是指变量可以在当前脚本的任意位置访问，拥有全局作用域的变量也被称为“全局变量”，一般情况下拥有以下特征的变量具有全局作用域：

- 最外层的函数和在最外层函数外面定义的变量拥有全局作用域；
- 所有未定义直接赋值的变量拥有全局作用域；
- 所有 window 对象的属性拥有全局作用域，例如 window.name、window.location、window.top 等。

示例代码如下：

```
01. var str = "Hello World!";
02.
03. function myFun(){
04.     document.write(str);    // 输出: Hello World!
05. }
06.
07. myFun();
08. document.write(str);      // 输出: Hello World!
```

实际情况下，所有具有全局作用域的变量都会被绑定到 window 对象中，成为 window 对象的一个属性，如下例所示：

```
01. var str = "JavaScript";
02. document.write(str);          // 输出: JavaScript
03. document.write(window.str);   // 输出: JavaScript
04. document.write(str === window.str); // 输出: true
```

JS 局部作用域

在函数内部声明的变量具有局部作用域，拥有局部作用域的变量也被称为“局部变量”，局部变量只能在其作用域中（函数内部）使用。示例代码如下：

```
01. function myFun(){
02.     var str = "Hello World!";
03.     document.write(str);    // 输出: Hello World!
04. }
05.
06. document.write(str);      // 报错: str is not defined
```

在函数内定义的局部变量只有在函数被调用时才会生成，当函数执行完毕后会立即销毁。