

# 53. eval、with 等为何不推荐使用?

## 1. 核心概念 (Core Concept)

`eval()` 和 `with` 是 JavaScript 中两种特殊的语言构造。`eval()` 函数可以将字符串作为 JavaScript 代码执行，而 `with` 语句则可以将一个对象的属性添加到当前作用域链的顶部。

## 2. 为什么不推荐使用它们? (The "Why")

不推荐使用 `eval()` 和 `with` 的主要原因在于：

- **安全性问题 (Security Issues):** `eval()` 可以执行任意字符串，如果字符串内容来自不受控制的外部输入，可能导致安全漏洞（如跨站脚本 XSS 攻击）。
- **性能问题 (Performance Issues):** `eval()` 和 `with` 会改变作用域链，使得 JavaScript 引擎难以进行静态分析和优化。这会导致 JIT (Just-In-Time) 编译器难以生成高效的代码，从而影响性能。
- **代码可读性与维护性差 (Poor Readability & Maintainability):** 它们使得代码的行为难以预测和理解。`eval()` 使代码更难调试，因为它可以在运行时动态地引入新的变量和函数。`with` 语句则可能导致变量名冲突，分不清某个属性是来自当前作用域还是 `with` 所指定对象的属性。

## 3. API 与用法 (API & Usage)

### 3.1 `eval()`

- **定义:** `eval()` 函数会将一个字符串作为 JavaScript 代码进行解析和执行。
- **语法:** `eval(string)`
- **参数:** `string`: 一个包含 JavaScript 表达式或语句的字符串。
- **返回值:** 字符串中代码执行后的返回值。如果字符串为空或不包含可返回值的東西，则返回 `undefined`。

典型（但不推荐）用法示例：

```
const x = 10;
const y = 20;
let result = eval('x + y + 5'); // result is 35

const codeString = "function sayHello() { console.log('Hello from eval!'); } sayHello();";
eval(codeString); // Output: Hello from eval!
```

### 3.2 `with`

- **定义:** with 语句用于扩展一个语句的作用域链。
- **语法:** with (object) { statement }
- **参数:** object: 表达式计算结果为一个对象, 该对象的属性将被添加到当前作用域链的前端。statement: 需要执行的语句或语句块。

#### 典型 (但不推荐) 用法示例:

```
const person = {
  name: 'Alice',
  age: 30,
  city: 'New York'
};

// 不使用 with
console.log(person.name);
console.log(person.age);

// 使用 with (不推荐!)
with (person) {
  console.log(name); // 直接访问 person.name
  console.log(age);  // 直接访问 person.age
  // 如果在此处使用了一个变量名, 且该变量在当前作用域和 person 对象中都存在,
  // 将优先访问 person 对象的属性。
  // console.log(city); // 直接访问 person.city
}
```

## 4. 关键注意事项 (Key Considerations)

- **严格模式下的限制:** 在 JavaScript 的严格模式 ('use strict';) 下, with 语句是禁止使用的, 会抛出 SyntaxError。在严格模式下使用直接 eval (即不是通过其他名字引用的 eval, 例如 const directEval = eval; directEval(...)) 这种形式不会在严格模式下引入新的局部变量) 时, 其中声明的变量和函数不会污染当前作用域, 而是在 eval 内部形成一个独立的作用域, 但这并不能解决其核心的安全和性能问题。
- **替代方案:** 对于需要动态执行代码的场景, 通常有更安全和高效的替代方案。例如, 使用 JSON.parse() 解析数据而不是 eval(); 使用属性访问器 (obj.prop 或 obj['prop']) 而不是 with。
- **现代代码规范:** 现代几乎所有的代码风格指南和 Linting 工具 (如 ESLint) 都将使用 eval() 和 with 列为警告或错误, 强制开发者避免使用它们。

## 5. 参考资料 (References)

- [MDN Web Docs - eval\(\)](#)
- [MDN Web Docs - with](#)
- [ECMAScript® 2024 Language Specification - 19.2.1 eval \(x\)](#)
- [ECMAScript® 2024 Language Specification - 14.12 The with Statement](#)

- [JavaScript: The Definitive Guide, 7th Edition \(Relevant sections on eval and with\)](#) (作为业界公认的权威参考, 可印证其观点)