📌 Understanding Operator Precedence in JavaScript

When writing mathematical expressions in JavaScript, the order in which operations are performed **follows standard algebraic precedence rules**.

🔢 Operator Precedence (Default Order)

1. **Parentheses ()** → Highest priority
2. **Multiplication *, Division /, and Modulus %** → Performed before addition and subtraction
3. **Addition + and Subtraction -** → Performed last

✅ Example Without Parentheses:

```
var totalCost = 1 + 3 * 4;
console.log(totalCost); // Output: 13 (Multiplication happens first: 3 * 4 = 12, then 1 + 12 = 13)
```

✅ Example With Parentheses to Change Order:

```
var totalCost = (1 + 3) * 4;
console.log(totalCost); // Output: 16 (Addition happens first: 1 + 3 = 4, then 4 * 4 = 16)
```

📌 Using Parentheses to Control Order

Parentheses **override** the default precedence, making your intention **explicit**.

🔹 *Example 1: Multiplication Before Addition*

```
var result = (2 * 4) * 4 + 2;
console.log(result); // Output: 34 (2*4 = 8 → 8*4 = 32 → 32+2 = 34)
```

🔹 *Example 2: Changing Precedence with Parentheses*

```
var result = (2 * 4) * (4 + 2);
console.log(result); // Output: 48 (2*4 = 8 → 4+2 = 6 → 8*6 = 48)
```

📌 Why Use Parentheses?

✅ **Avoid confusion** – Clearly indicate which operations should be executed first.
✅ **Improve readability** – Easier for you and others to understand the code later.
✅ **Prevent logic errors** – Avoid unintended calculation mistakes.

💡 Quick Summary

✓ Multiplication (*) and Division (/) happen before Addition (+) and Subtraction (-)
✓ Use parentheses () to control the order explicitly
✓ More parentheses = clearer code!
