Understanding `while` vs `do...while` Loops in Simple Terms

Both `while` and `do...while` loops **repeat code** multiple times, but there is one key difference:

✅ **`while` loop** → Checks the condition **first** before running the code.
✅ **`do...while` loop** → Runs the code **at least once**, then checks the condition.

1️⃣ `while` Loop (Condition First)

A `while` loop **only runs if the condition is true** at the start.

Example:

```
var i = 0;
while (i <= 3) {
  alert(i);
  i++;
}
```

🔹 How It Works:

1. **Check condition:** `i <= 3` (true for `0, 1, 2, 3`).
2. If true → Run `alert(i);` and then increase `i`.
3. When `i` becomes `4`, the condition is false, so it stops.

2️⃣ `do...while` Loop (Runs at Least Once)

A `do...while` loop **always runs once before checking the condition**.

Example:

```
var i = 0;
do {
  alert(i);
  i++;
} while (i <= 3);
```

🔹 How It Works:

1. **First, run the code** inside `{ ... }`, even if `i` is already too big.
2. Then, check `while (i <= 3)`.
3. If true → Run again. If false → Stop.

3️⃣ The Key Difference

Imagine we start with `i = 5` instead:

Using `while` (Code Never Runs):

```
var i = 5;
while (i < 0) {  // ❌ Condition is false at the start
  alert(i);  // This never runs!
  i++;
}
```

✅ The loop **never runs** because `i < 0` is false.

Using `do...while` (Runs Once Anyway):

```
var i = 5;
do {
  alert(i);  // ✅ Runs at least once!
  i++;
} while (i < 0);
```

✅ **Even though `i < 0` is false, the alert still runs once** before checking the condition.

When to Use Each Loop

- ✅ Use while **if you want to check the condition first** (so the code might never run).
- ✅ Use do...while **if you want to ensure the code runs at least once** (e.g., asking for user input).