The difference between while {} and do {} while lies in **when the condition is checked** and how many times the loop's body is executed:

The difference between while {} and do {} while lies in **when the condition is checked** and how many times the loop's body is executed:

**1. while {} Loop**

* **Condition First**: The condition is checked **before** executing the loop body.
* **Execution**: If the condition is false at the start, the loop body **may not execute even once**.

**Syntax:**

while (condition) {

// Code to execute while the condition is true

}

**Example:**

let x = 0;

while (x < 3) {

console.log(x); // Outputs: 0, 1, 2

x++;

}

If x is initially 3 or greater, the loop will not run.

**2. do {} while Loop**

* **Condition Last**: The loop body is executed **at least once** before checking the condition.
* **Execution**: Even if the condition is false, the loop body runs **once**.

**Syntax:**

do {

// Code to execute at least once

} while (condition);

**Example:**

let y = 5;

do {

console.log(y); // Outputs: 5

y++;

} while (y < 3);

Here, even though y starts as 5 (which doesn't satisfy the condition y < 3), the loop runs once before the condition is checked.

**Key Difference**

| **Feature** | **while {} Loop** | **do {} while Loop** |
| --- | --- | --- |
| **Condition Check** | Before the loop body | After the loop body |
| **Guaranteed Execution** | May not execute at all | Executes at least once |

Use while when you want the condition to determine whether the loop starts. Use do while when you need the loop to run at least once regardless of the condition.