Understanding Array Indexing and Loop Ranges

Understanding Array Indexing and Loop Ranges in Java (and C++/Python)

F	Arrays are	0-indexed	in mos	t languages	like Java,	C++, P	ython.

If an array has 'n' elements, the valid indices are:

Loop Analysis:

Case 1: for (int i = 0; i < nums.length; i++)

- Covers all elements from index 0 to length 1.
- Total iterations = length.
- This is the standard loop.

Case 2: for (int i = 0; $i \le nums.length$; i++)

- When i = nums.length, it's out of bounds.
- This causes a runtime error.

Case 3: for (int i = 1; i < nums.length; i++)

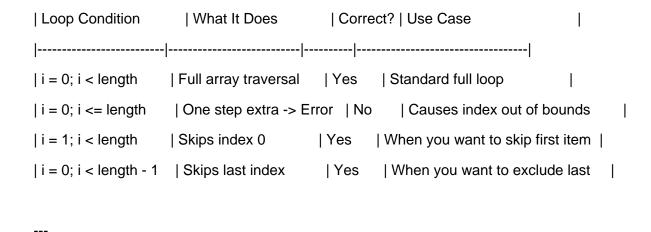
- Skips the first element (index 0).
- Valid, but only when you want to skip the first item.

Case 4: for (int i = 0; i < nums.length - 1; i++)

- Skips the last element (index n 1).
- Valid, but only when you want to exclude the last item.

Understanding Array Indexing and Loop Ranges

Final Summary:



TL;DR:

i = 0; i < length is the correct way to loop over the full array safely.