Binary Search: Iterative vs Compact Approach

# 🧠 Introduction

Binary Search is an efficient algorithm to search for an element in a sorted array. It repeatedly divides the search interval in half.

# ✅ Method 1: Basic Iterative Version

const binarySearch = (arr, target) => {  
 let start = 0;  
 let end = arr.length - 1;  
  
 while (start <= end) {  
 let mid = Math.floor((start + end) / 2);  
  
 if (arr[mid] === target) return mid;  
 else if (arr[mid] < target) start = mid + 1;  
 else end = mid - 1;  
 }  
  
 return -1;  
};

# ⚡ Method 2: Compact Professional Version

const binarySearch = (arr, target) => {  
 for (let start = 0, end = arr.length - 1; start <= end;) {  
 let mid = Math.floor((start + end) / 2);  
 if (arr[mid] === target) return mid;  
 arr[mid] < target ? (start = mid + 1) : (end = mid - 1);  
 }  
 return -1;  
};

# 🔍 Why Ternary Operator is Used?

Ternary operator is a short form of if...else. It helps in writing concise and clean code, commonly used by professionals when the condition is simple.

Syntax: condition ? true : false

# 📊 Comparison

| Method | Code Length | Readability | Best For |  
|---------------|-------------|-------------|------------------|  
| Iterative | Medium | Easy | Beginners |  
| Compact | Short | Moderate | Professionals |

Generated by ChatGPT on 2025-05-31