**Exercise 6: Library Management System**

**Scenario:**

You are developing a library management system where users can search for books by title or author.

**1. Understand Search Algorithms**

**Linear Search**

Linear search is a straightforward algorithm that checks each element in a list sequentially until the target element is found or the list ends.

**Algorithm:**

* Start from the first element.
* Compare each element with the target.
* If the target matches an element, return that element.
* If the target is not found by the end of the list, return null.

**Time Complexity:** O(n), where n is the number of elements in the list. This is because, in the worst case, the algorithm may need to check every element.

**Binary Search**

Binary search is an efficient algorithm for finding an element in a sorted list by repeatedly dividing the search interval in half.

**Algorithm:**

* Ensure the list is sorted.
* Initialize two pointers: left at the start and right at the end of the list.
* Find the middle element.
* If the middle element matches the target, return that element.
* If the target is smaller than the middle element, repeat the search on the left sublist.
* If the target is larger than the middle element, repeat the search on the right sublist.
* If the search interval is empty, return null.

**Time Complexity:** O(log n), where n is the number of elements in the list. This is because the algorithm reduces the search interval by half each time.

**Analysis**

**Time Complexity Comparison:**

* **Linear Search:** O(n)
* **Binary Search:** O(log n)

**Use Cases:**

* **Linear Search:**
  + Use when the list is small or unsorted.
  + Suitable for real-time applications where sorting the list is not feasible due to frequent updates.
* **Binary Search:**
  + Use when the list is large and sorted.
  + Ideal for static or rarely modified lists where sorting overhead is manageable.