**Understand Sorting Algorithms**

Sorting algorithms are used to arrange data in a particular order, such as ascending or descending. In an e-commerce platform, sorting orders based on total price helps in identifying high-value transactions and improving service efficiency. Here are some common sorting algorithms:  
  
- **Bubble Sort:** This is a simple algorithm where each pair of adjacent elements is compared, and they are swapped if they are in the wrong order. It's easy to understand but not efficient for large datasets.  
  
- **Insertion Sort:** It builds the final sorted array one item at a time. It works well with small datasets and nearly sorted data.  
  
- **Quick Sort:** This is a divide-and-conquer algorithm. It selects a 'pivot' element and partitions the array into two halves – one with elements less than the pivot and one with elements greater. It's faster and widely used in practice.  
  
- **Merge Sort:** Another divide-and-conquer algorithm. It divides the array into halves, sorts each half, and merges them back. It has consistent performance and is good for large datasets but uses more memory.

**Analysis**

**Performance Comparison (Time Complexity):**

- Bubble Sort:  
 - Best Case: O(n) (if already sorted)  
 - Average Case: O(n^2)  
 - Worst Case: O(n^2)  
  
- Quick Sort:  
 - Best Case: O(n log n)  
 - Average Case: O(n log n)  
 - Worst Case: O(n^2) (if pivot selection is poor)

**Why Quick Sort is Generally Preferred Over Bubble Sort:**

Quick Sort is generally faster and more efficient than Bubble Sort, especially for large lists. It uses fewer comparisons and is better optimized for real-world scenarios. Bubble Sort, although simple, becomes very slow as the number of items increases. This is why Quick Sort is often used in modern applications, including sorting customer orders in e-commerce platforms.