**There are several sorting algorithms, each with its own strengths and weaknesses. Here's a brief overview of the four algorithms mentioned:**

* **Bubble Sort**: Bubble sort is a simple sorting algorithm that repeatedly steps through the list, compares adjacent elements, and swaps them if they are in the wrong order. The pass through the list is repeated until the list is sorted.
* **Insertion Sort**: Insertion sort is a simple sorting algorithm that builds the final sorted array (or list) one item at a time. It is much less efficient on large lists than more advanced algorithms like quicksort, heapsort, or merge sort.
* **Quick Sort**: Quick sort is a divide-and-conquer algorithm that picks an element as a pivot and partitions the given array around the picked pivot. It is one of the most efficient sorting algorithms with an average time complexity of O(n log n).
* **Merge Sort**: Merge sort is a divide-and-conquer algorithm that divides the input array into two halves, recursively calls itself for the two halves, and then merges the two sorted halves.

**Comparing the performance (time complexity) of Bubble Sort and Quick Sort:**

* **Bubble Sort:** The time complexity of Bubble Sort is O(n^2) in the worst case, where n is the number of orders. This is because Bubble Sort uses two nested loops to compare and swap elements.
* **Quick Sort:** The time complexity of Quick Sort is O(n log n) on average, where n is the number of orders. This is because Quick Sort uses a divide-and-conquer approach to sort the array.

**Quick Sort is generally preferred over Bubble Sort because:**

* Quick Sort is generally preferred over Bubble Sort because of its better time complexity. Quick Sort has an average time complexity of **O(n log n**), which is much faster than Bubble Sort's **O(n^2)** time complexity for large datasets. This makes Quick Sort more efficient and scalable for sorting large arrays of orders.
* Additionally, Quick Sort is a more stable sorting algorithm than Bubble Sort, which means that it preserves the order of equal elements. This is important in many applications, including sorting customer orders by their total price.