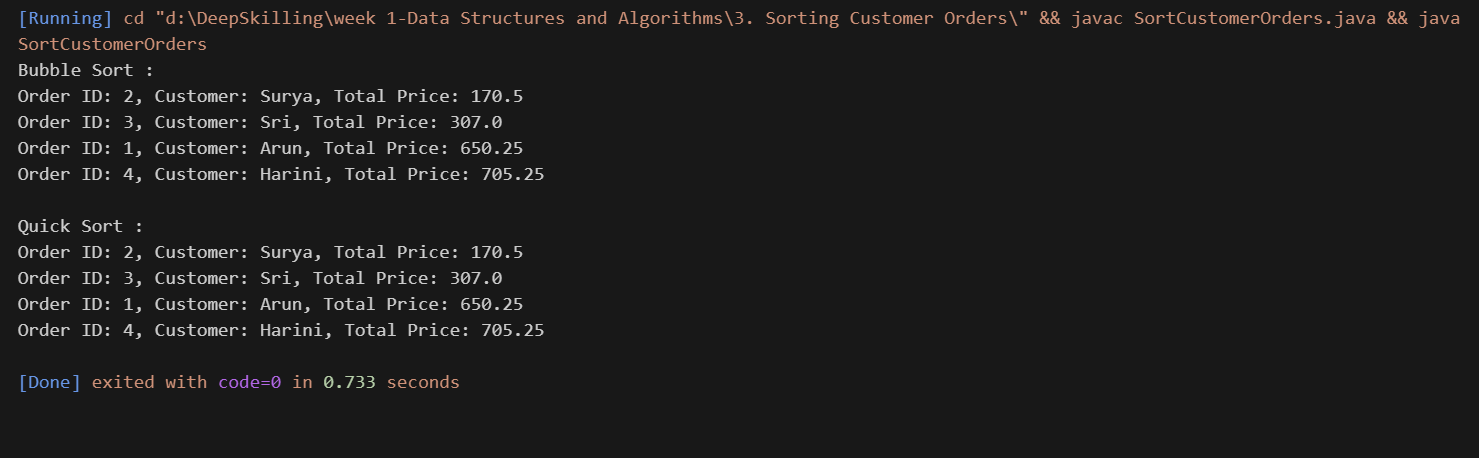
**Exercise 3: Sorting Customer Orders**

**1. Understand Sorting Algorithms:**

* **Bubble Sort**: Repeatedly swaps adjacent elements if they are in the wrong order. Simple but inefficient for large data sets.
* **Insertion Sort**: Builds the final sorted array one item at a time.Efficient for small or nearly sorted data sets.
* **Quick Sort**: Picks a pivot and partitions the array into two subarrays (less than and greater than pivot), then recursively sorts. Very efficient in practice with good pivot selection.
* **Merge Sort**: Divides array into halves, recursively sorts them, and merges. Very stable but uses extra memory.

**Output:**



**4. Analysis:**

* **Time Complexity:**

1. Bubble sort: O(n) [best case], O(n2) [ average case, worst case]
2. Quick Sort: O(n log n) [best case, average case], O(n2) [worst case]

* Why Quick Sort is preferred?
  + **Faster** on average due to divide-and-conquer.
  + **Fewer swaps** and comparisons than Bubble Sort.
  + **Efficient in practice**, especially with large datasets.
  + Bubble Sort is generally avoided for production use due to its poor performance.