Insertion Sort

It builds the sorted array one element at a time by **picking an element and inserting it into its correct position** in the sorted part of the array.

How it Works:

Let’s take an example:
Input: [5, 3, 4, 1, 2]

1. Start from index 1 → pick 3 → compare with 5 → insert before → [3, 5, 4, 1, 2]
2. Next 4 → insert between 3 and 5 → [3, 4, 5, 1, 2]
3. Next 1 → insert at the beginning → [1, 3, 4, 5, 2]
4. Last 2 → insert between 1 and 3 → [1, 2, 3, 4, 5]

Key Points:

- **Time Complexity**:
  - Worst: **$O(n^2)$** (reverse sorted)
  - Best: **$O(n)$** (already sorted)
  - Average: **$O(n^2)$**
- **Space Complexity**: **$O(1)$**
- **Stable Sort**
- **In-place** sorting algorithm
- Good for **small or nearly sorted datasets**