Insertion Sort

Builds a sorted sublist on the left by picking an element to sort, shifting larger elements to the right and then inserting the sort element into its correct position relative to the already sorted part of the list.

```python
def insertion(arr: list) -> list:

    for i in range(1, len(arr)):

        sort_value = arr[i]
        compare_idx = i - 1

        while compare_idx >= 0 and arr[compare_idx] > sort_value:

            arr[compare_idx + 1] = arr[compare_idx]
            compare_idx -= 1

        arr[compare_idx + 1] = sort_value

    return arr
```

Property	Details
How it works	Selects an element to sort and stores it temporarily. Then, it compares each element with the stored value. If the stored value is smaller, the larger element is shifted to the right. A variable keeps track of the position of the last sorted element. At the end of the iteration, the stored value is inserted into its correct position, right after the last sorted element. This process gradually builds a sorted sublist on the left, while the unsorted portion remains on the right.
Time Complexity	$O(n^2)$ in the worst and average case, $O(n)$ in the best case (already sorted)
Space Complexity	$O(1)$
Stable	Yes
Type	in-place, iterative

- Follow on Bluesky
- Join Discord