Two methods of sorting

1. **In-place sorting** using **sort()**
2. **sorted()** that is **not an in-place sorting**.

The **difference** is that

|  |  |
| --- | --- |
| **sort()** | **sorted()** |
| It will change the original list | It will return a new list without changing the original list |
| Works only on lists | Work on any iterable  such as lists, tuples, dictionaries, str and others. |
| If you want to save space and memory, then you should use sort(). | If you want to keep the original record, then you should use sorted(). |
| list.sort(reverse=True|False, key=myFunc) | sorted(iterable, key, reverse) |

**Note**: No matter what iterable is passed in to the sorted() function, it always returns a list.

***Python internally used TIM SORT algorithm.***

Tim Sort is a hybrid sorting algorithm that uses **Insertion Sort** for small subarrays and **Merge Sort** for merging sorted runs, making it efficient for real-world data.