```
//#1 - Sort 10 schools around your house by distance:
// insertion sort because it is fast with smaller data. It has space complexity
of O(1)

//#2 - eBay sorts listings by the current Bid amount:
// radix or counting sort
// because of bidding amount with be fixed somewhere between $1 to $50,000
probably even less
// one dollar to 100 dollar
// because of it is fixed length
// to beat O(nlogn) time complexity

//#3 - Sport scores on ESPN
// scores can be of any sports such as tennis, cricket, football, volleyball and
values can have decimal places and obviously it is not going to be sorted
already. So, to avoid space in Memory Quick sort would be better option to get
O(nlogn) time complexity. Worst case can be avoided by picking the pivot
randomly. Space complexity is O(logn)

//#4 - Massive database (can't fit all into memory) needs to sort through past
year's user data
// as the data cannot be fit into memory, we need to sort externally and since
the data is massive choosing the sorting algorithm should be efficient once in
performance. So Merge sort could be wise choise here as the time complexity is
always O(nlogn) in best/average and worst case. But the space requires O(n)...

// quick sort because since it's massive database, we do not need to store
additional memory for Quick sort. it is inplace sorting algorithm needs constant
space O(1)

//#5 - Almost sorted Udemy review data needs to update and add 2 new reviews
// insertion sort. because it is good for pre sorted data

//#6 - Temperature Records for the past 50 years in Canada
// if the temperature values going to have only integers from -30 to 50, the
range is small so choosing radix or counting would be a good choice.
```

```
// if decimal places are includes, then quick sort can be best as it can do in memory sorting. Radix or counting sort can only be done for integers and not decimal numbers.

//#7 - Large user name database needs to be sorted. Data is very random.
// merge sort if we have enough memory and memory is not too expensive on the machine or system we are storing the data
// OR quick sort if the username db is not that big. just to save on that memory space. I can pick the good pivot

//#8 - You want to teach sorting for the first time
// bubble sort / selection sort
```