# Selection Sort

* ## Selection Sort :

→ It is a simple sorting algorithm.
→ In this algorithm, we select an element and put it at its correct position.
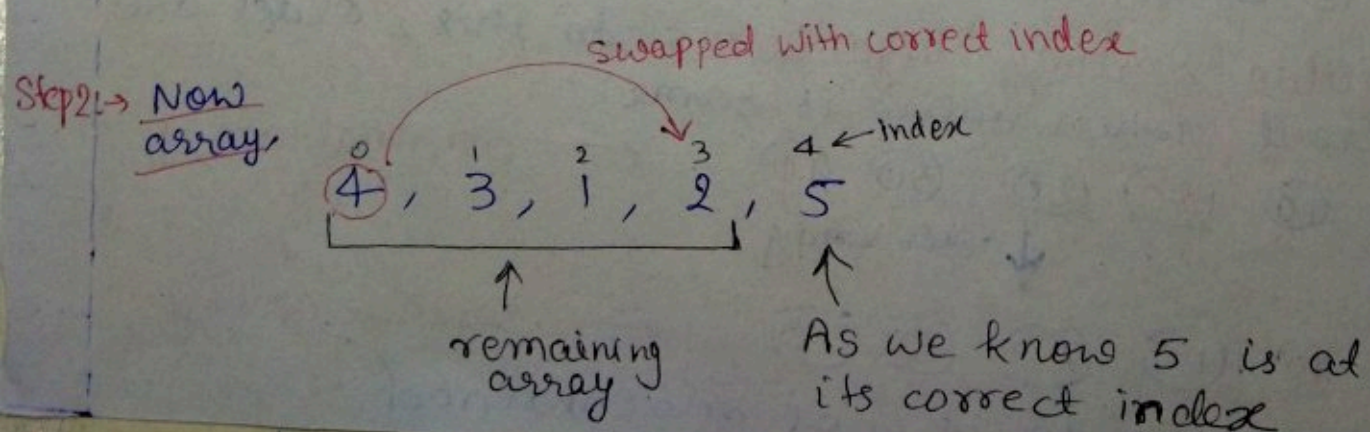
>> In this algorithm, we have to find the maximum value in the list, then put it in the correct position
Then select 2nd largest element from the remaining array then put it at its correct position.... And repeat this step till we got the sorted array.
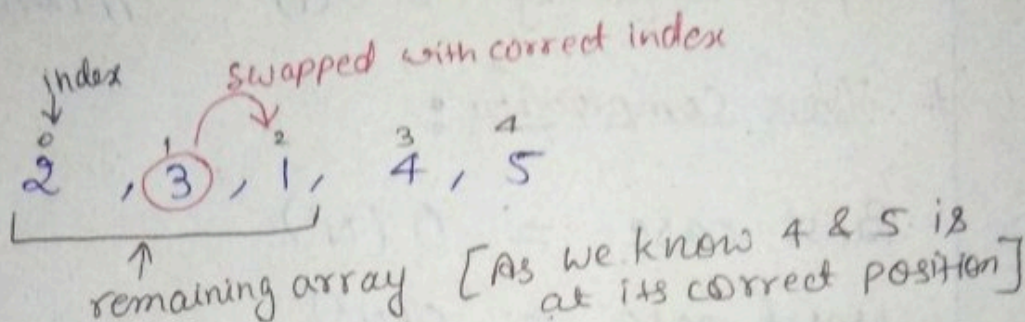
Example :-   4, 5, 1, 2, 3

Step 1 :→

index
↓
swapped with correct index

| 0 | 1 | 2 | 3 | 4 |
4 , ⑤ , 1 , 2 , 3
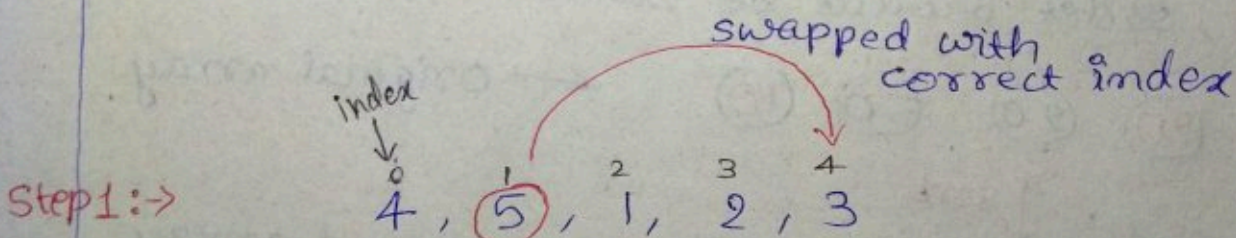
>> Select the largest element from the above array (Here, 5). So, put 5 at its correct index i·e at index 4.

Step 2 :→ Now array,

swapped with correct index

④ , 3 , 1 , 2 , 5
0   1   2   3   4 ← index

⟵ remaining array

↑
As we know 5 is at its correct index

>> Now, find the maximum item from the remaining array (Here, 4 is the largest). Now swap 4 with 2 (i.e put 4 at its correct or index, i.e index 3)
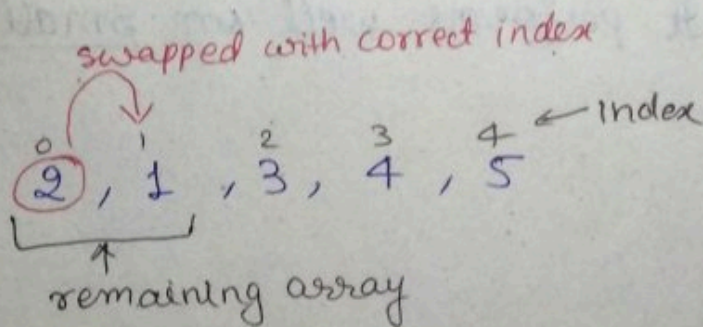
Now array is

index   swapped with correct index

$\overset{0}{2}$, $\overset{1}{③}$, $\overset{2}{1}$, $\overset{3}{4}$, $\overset{4}{5}$

Step 3 :->

remaining array   [As we know 4 & 5 is at its correct position]

>>> Select the largest element from remaining array (Here, 3 is largest). Put 3 at its correct index (i.e, at index 2)

Now array is

swapped with correct index

Step 4 :->

$\overset{0}{②}$, $\overset{1}{1}$, $\overset{2}{3}$, $\overset{3}{4}$, $\overset{4}{5}$ <- index

remaining array
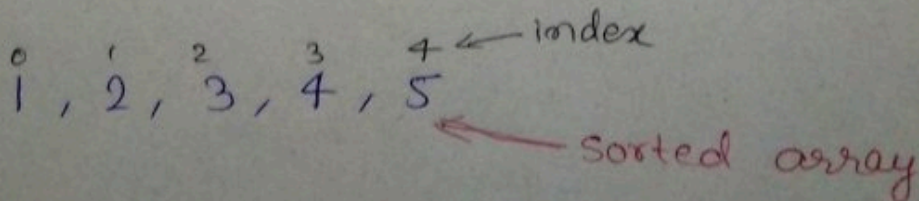
>>> Select the largest element from remaining array (Here, 2 is largest). Put 2 at its correct index (i.e, at index 1)

Now array is:-

$\overset{0}{1}$, $\overset{1}{2}$, $\overset{2}{3}$, $\overset{3}{4}$, $\overset{4}{5}$ <- index

<- Sorted array

* NOTE *

Here, I am selecting the maximum element & putting at its correct index. You can select the minimum element and do the same process.

* Complexity of Selection Sort :

Space Complexity $= O(1)$ // constant

Time Complexity :

  Best case $= O(N^2)$
  worst case $= O(N^2)$

Stable $= NO$

* Use case :
  $\Rightarrow$ It performs well on small lists/arrays.