

Mathematical Physics II

Insertion Sort using Python

- **Insertion Sort Process:**

Here we shall explain this process pictorially with an example. Let we have a set of 6 numbers in random order and we want to arrange them in ascending order. The given set of number is:

3	5	4	1	0	2
---	---	---	---	---	---

In the insertion method of sorting one has to choose a key number at each step, which has been marked in red colour here. The choice has to start from the second place. At each particular step this key number searches its proper position by comparing its own value consecutively with the numbers located on its left side, marked in green colour, as per the requirement of ordering (ascending or descending). In consecutive steps the position of the key number shifts towards right and the numbers at the left of the key number to be chosen in the next step get ordered. In this way in the final step the position of the key number reaches the right most position and with its proper positioning we get the ordered number set fulfilling the requirement.

- **Step1:**

initial	3	5	4	1	0	2	No Change
final	3	5	4	1	0	2	

- **Step2:**

initial	3	5	4	1	0	2	4 has been inserted in between 3 and 5
final	3	4	5	1	0	2	

- **Step3:**

initial	3	4	5	1	0	2	1 has been placed in front of 3
final	1	3	4	5	0	2	

- **Step4:**

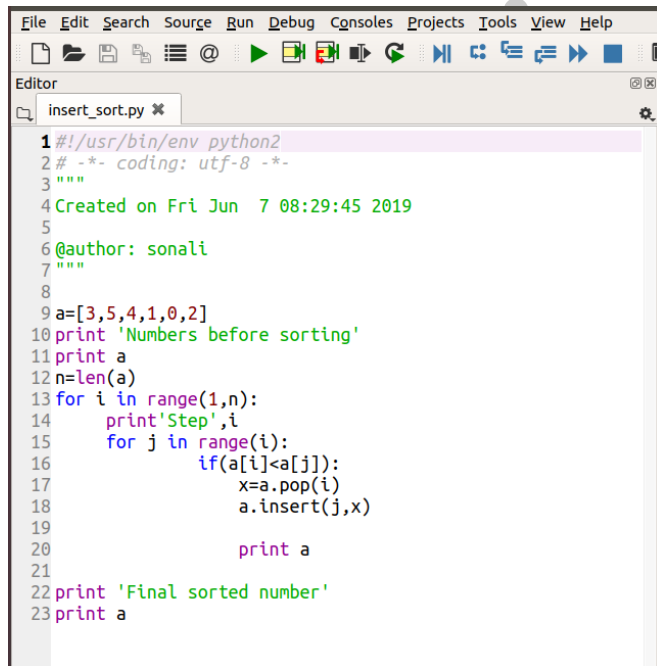
initial	1	3	4	5	0	2	0 has been placed in front of 1
final	0	1	3	4	5	2	

- **Step5:**

initial	0	1	3	4	5	2	2 has been inserted in between 1 and 3
final	0	1	2	3	4	5	

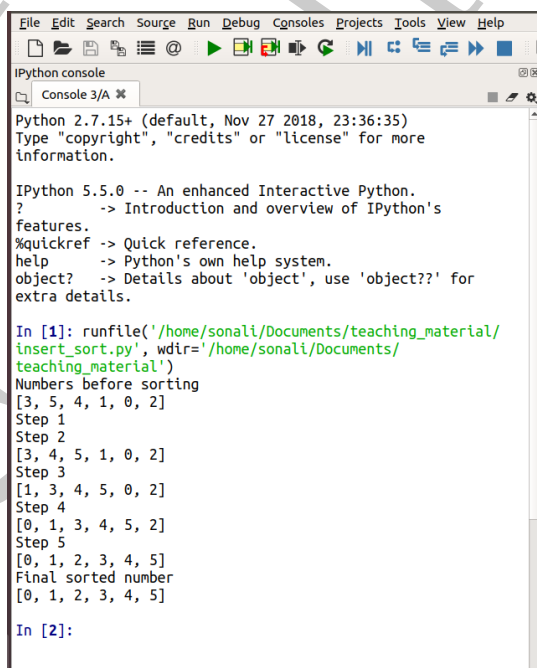
From the above pictorial presentation it is clear that in insertion sort method how the key number is placed in proper position by comparing its value with the numbers located in its leftside. The choice of key number in each successive step shifts towards right and with its proper positioning finally we get an ordered number set as per our requirement.

Sample python program for the above example and the output is shown in the next page.



```
File Edit Search Source Run Debug Consoles Projects Tools View Help
Editor
insert_sort.py x
1#!/usr/bin/env python2
2# -*- coding: utf-8 -*-
3"""
4Created on Fri Jun 7 08:29:45 2019
5
6@author: sonali
7"""
8
9a=[3,5,4,1,0,2]
10print 'Numbers before sorting'
11print a
12n=len(a)
13for i in range(1,n):
14    print'Step',i
15    for j in range(i):
16        if(a[i]<a[j]):
17            x=a.pop(i)
18            a.insert(j,x)
19
20        print a
21
22print 'Final sorted number'
23print a
```

Figure 1: Python program for insertion sort.



```
File Edit Search Source Run Debug Consoles Projects Tools View Help
iPython console
Console 3/A x
Python 2.7.15+ (default, Nov 27 2018, 23:36:35)
Type "copyright", "credits" or "license" for more
information.

iPython 5.5.0 -- An enhanced Interactive Python.
?          -> Introduction and overview of iPython's
features.
%quickref -> Quick reference.
help       -> Python's own help system.
object?    -> Details about 'object', use 'object??' for
extra details.

In [1]: runfile('/home/sonali/Documents/teaching_material/
insert_sort.py', wdir='/home/sonali/Documents/
teaching_material')
Numbers before sorting
[3, 5, 4, 1, 0, 2]
Step 1
Step 2
[3, 4, 5, 1, 0, 2]
Step 3
[1, 3, 4, 5, 0, 2]
Step 4
[0, 1, 3, 4, 5, 2]
Step 5
[0, 1, 2, 3, 4, 5]
Final sorted number
[0, 1, 2, 3, 4, 5]

In [2]:
```

Figure 2: Output of the Python code for insertion sort.