**Insertion Sort**

The task is to complete the **insert()** function which is used to implement Insertion Sort.

**Example 1:**

**Input**:

N = 5

arr[] = { 4, 1, 3, 9, 7}

**Output**:

1 3 4 7 9

**Example 2:**

**Input**:

N = 10

arr[] = {10, 9, 8, 7, 6, 5, 4, 3, 2, 1}

**Output**:

1 2 3 4 5 6 7 8 9 10

**Expected Time Complexity:** O(N\*N).  
**Expected Auxiliary Space:** O(1).

**Constraints:**  
1 <= N <= 1000  
1 <= arr[i] <= 1000

**Company Tags**

[**Microsoft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Microsoft) [**MAQ Software**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=MAQ%20Software) [**Juniper Networks**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Juniper%20Networks) [**Cisco**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Cisco) [**Accenture**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Accenture) [**Dell**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Dell)[**Veritas**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Veritas) [**Grofers**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Grofers)

### Java Code

class Solution

{

static void insert(int arr[],int i)

{

// Your code here

}

//Function to sort the array using insertion sort algorithm.

public void insertionSort(int arr[], int n)

{

//code here

for(int i=0;i<n;i++)

{

for(int j=i;j<n;j++)

if(arr[i]>arr[j])

{

int temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

}