```java
package usingInnerClassObject;

public class SortedInsertion {

    Node head;

    class Node{
        int data;
        Node next;

        //creating a constructor for assigning value to data and next as null
        Node(int data){
            this.data = data;
            next = null;
        }

    } // closing of class Node

    void sortedInsert(Node new_node) {

        Node curr = head;

        if(head == null || head.data > new_node.data) {
            new_node.next = head;
            head = new_node;
        }
        else {  // we are trying to find out the appropriate place for insertion of the new_node
            while(curr.next!=null && curr.next.data < new_node.data) { // traversing till we find the place to insert
                curr = curr.next; // incrementing by single step
            }
            //insert the new node just after the while loop breaks.
            new_node.next = curr.next;
            curr.next = new_node;
        }
    }


      void printList() {
          Node curr=head;

      while(curr!=null) {
          System.out.print(curr.data + " -> ");
          curr = curr.next;

      }
        System.out.print("NULL");

     }

    public static void main(String[] args) {

        SortedInsertion si = new SortedInsertion();
        SortedInsertion.Node new_node =si.new Node(2);
        si.sortedInsert(new_node);

        //inserting the second value into the list
        new_node =si.new Node(5);
        si.sortedInsert(new_node);

        //3rd value
        new_node =si.new Node(7);
        si.sortedInsert(new_node);

        //4th value
        new_node =si.new Node(10);
        si.sortedInsert(new_node);

        //5th value
        new_node =si.new Node(15);
        si.sortedInsert(new_node);


        // here we are trying to add a new_node as 9
        new_node =si.new Node(9);
        si.sortedInsert(new_node);


        si.printList();








    }

}

```