Question: Given a node, insert at the end of a linked list

Algorithm:

There are two possibilities

1. If linked list is empty

   Make the new node the head and return it

2. If linked list already has nodes

   Traverse the linked list to find the last

   keep moving until Node.next==None, This node is our tail Node

   Prev Make next of this node to be new node

   Make Previous of new node to be tail

Template to build a node

Class Node:

```
def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None
```

```
// Function to implement insertion in tail.
def InsertTail(head, value):
    newNode = Node(value)
    // check if head is None
    if head == None:
        return newNode
    // find the last Node
    else:
        curr = head
        while curr.next != None:
            // at each iteration move curr one step closer to tail
            curr = curr.next
        // by the we exit from the loop, curr becomes tail
        PrevNode = curr.prev
        // Change next of tail to be newNode
        curr.next = newNode
        // change prev of newNode to be Previous tail node (curr)
        newNode.prev = PrevNode
        // return head
        return head
```

Example-1: Insert 10 at the end of Given linked list.