**package** linkedlist;

//import linkedlist.linkedlist.Node;

**public** **class** linkedlist {

Node head; // head of list

// Linked list Node.

// This inner class is made static

// so that main() can access it

**static** **class** Node {

**int** data;

Node next;

// Constructor

Node(**int** d)

{

data = d;

next = **null**;

}

}

// Method to insert a new node

**public** **static** linkedlist insert(linkedlist list, **int** data)

{

// Create a new node with given data

Node new\_node = **new** Node(data);

new\_node.next = **null**;

// If the Linked List is empty,

// then make the new node as head

**if** (list.head == **null**) {

list.head = new\_node;

}

**else** {

// Else traverse till the last node

// and insert the new\_node there

Node last = list.head;

**while** (last.next != **null**) {

last = last.next;

}

// Insert the new\_node at last node

last.next = new\_node;

}

// Return the list by head

**return** list;

}

// Method to print the LinkedList.

**public** **static** **void** printList(linkedlist listA)

{

Node currNode = listA.head;

System.***out***.print("LinkedList: ");

// Traverse through the LinkedList

**while** (currNode != **null**) {

// Print the data at current node

System.***out***.print(currNode.data + " ");

// Go to next node

currNode = currNode.next;

}

}

// Driver code

**public** **static** **void** main(String[] args)

{

/\* Start with the empty list. \*/

linkedlist listA = **new** linkedlist();

//

// \*\*\*\*\*\*INSERTION\*\*\*\*\*\*

//

// Insert the values

listA = *insert*(listA, 1);

listA= *insert*(listA, 2);

listA= *insert*(listA, 3);

listA= *insert*(listA, 4);

listA= *insert*(listA, 5);

// Print the LinkedList

*printList*(listA);

}

}