

Insert a Node at the Tail of a Linked List

This challenge is part of a tutorial track by [MyCodeSchool](#) and is accompanied by a video lesson.

You are given the pointer to the head node of a linked list and an integer to add to the list. Create a new node with the given integer. Insert this node at the tail of the linked list and return the head node of the linked list formed after inserting this new node. The given head pointer may be null, meaning that the initial list is empty.

Function Description

Complete the *insertNodeAtTail* function in the editor below.

insertNodeAtTail has the following parameters:

- *SinglyLinkedListNode* pointer *head*: a reference to the head of a list
- *int* *data*: the data value for the node to insert

Returns

- *SinglyLinkedListNode* pointer: reference to the head of the modified linked list

Input Format

The first line contains an integer *n*, the number of elements in the linked list.
The next *n* lines contain an integer each, the value that needs to be inserted at tail.

Constraints

- $1 \leq n \leq 1000$
- $1 \leq list_i \leq 1000$

Sample Input

STDIN Function ----- 5 size of linked list n = 5 141 linked list data values 141..474 302 164 530 474

Sample Output

```
141
302
164
530
474
```

Explanation

First the linked list is NULL. After inserting 141, the list is 141 -> NULL.
After inserting 302, the list is 141 -> 302 -> NULL.

After inserting 164, the list is 141 -> 302 -> 164 -> NULL.

After inserting 530, the list is 141 -> 302 -> 164 -> 530 -> NULL. After inserting 474, the list is 141 -> 302 -> 164 -> 530 -> 474 -> NULL, which is the final list.