Insert a node at the head of a linked list

This challenge is part of a tutorial track by MyCodeSchool and is accompanied by a video lesson.

Given a pointer to the head of a linked list, insert a new node before the head. The $next$ value in the new node should point to $head$ and the $data$ value should be replaced with a given value. Return a reference to the new head of the list. The head pointer given may be null meaning that the initial list is empty.

Function Description

Complete the function $insertNodeAtHead$ with the following parameter(s):

- $SinglyLinkedListNode\ llist$: a reference to the head of a list
- $data$: the value to insert in the $data$ field of the new node

Input Format

The first line contains an integer n, the number of elements to be inserted at the head of the list.
The next n lines contain an integer each, the elements to be inserted, one per function call.

Constraints

- $1 \leq n \leq 1000$
- $1 \leq list[i] \leq 1000$

Sample Input

```
STDIN   Function
-----   --------
5       n = 5
383     data items to insert 383 ... 321
484
392
975
321
```

Sample Output

```
321
975
392
484
383
```

Explanation

Intially the list in NULL. After inserting 383, the list is 383 -> NULL.
After inserting 484, the list is 484 -> 383 -> NULL.
After inserting 392, the list is 392 -> 484 -> 383 -> NULL.

After inserting 975, the list is 975 -> 392 -> 484 -> 383 -> NULL.
After inserting 321, the list is 321 -> 975 -> 392 -> 484 -> 383 -> NULL.