Python

Difficulty: **Category:** **Successful Submissions:** 17,373+

Shift Linked List

Write a function that takes in the head of a Singly Linked List and an integer `k`, shifts the list in place (i.e., doesn't create a brand new list) by k positions, and returns its new head.

Shifting a Linked List means moving its nodes forward or backward and wrapping them around the list where appropriate. For example, shifting a Linked List forward by one position would make its tail become the new head of the linked list.

Whether nodes are moved forward or backward is determined by whether `k` is positive or negative.

Each `LinkedList` node has an integer `value` as well as a `next` node pointing to the next node in the list or to `None` / `null` if it's the tail of the list.

You can assume that the input Linked List will always have at least one node; in other words, the head will never be `None` / `null`.

Sample Input

```
head = 0 -> 1 -> 2 -> 3 -> 4 -> 5 // the head node with value 0
k = 2
```

Sample Output

```
4 -> 5 -> 0 -> 1 -> 2 -> 3 // the new head node with value 4
```

Hints

Hint 1

Hint 2

Hint 3

Hint 4

Prompt | Your Solutions | Custom Output | •••