Delete Node in a Linked List

Write a function to delete a node (except the tail) in a singly linked list, given only access to that node.

Supposed the linked list is `1 -> 2 -> 3 -> 4` and you are given the third node with value `3`, the linked list should become `1 -> 2 -> 4` after calling your function.

Solution 1

We can't really delete the node, but we can kinda achieve the same effect by instead removing the **next** node after copying its data into the node that we were asked to delete.

C++

```
void deleteNode(ListNode* node) {
    *node = *node->next;
}
```

But better properly delete the next node:

```
void deleteNode(ListNode* node) {
    auto next = node->next;
    *node = *next;
    delete next;
}
```

Java and C#

```
public void deleteNode(ListNode node) {
    node.val = node.next.val;
    node.next = node.next.next;
}
```

Python

```
def deleteNode(self, node):
    node.val = node.next.val
    node.next = node.next.next
```

C

```
void deleteNode(struct ListNode* node) {
    *node = *node->next;
}
```

But better properly free the next node's memory:

```
void deleteNode(struct ListNode* node) {
    struct ListNode* next = node->next;
    *node = *next;
    free(next);
}
```

JavaScript

```
var deleteNode = function(node) {
    node.val = node.next.val;
    node.next = node.next.next;
};
```

Ruby

```
def delete_node(node)
    node.val = node.next.val
    node.next = node.next.next
    nil
end
```

written by StefanPochmann original link here

Solution 2

This question is stupid and should be deleted intermediately.

written by smfwuxiao original link here

Solution 3

```
public class Solution {
    public void deleteNode(ListNode node) {
        if(node != null && node.next != null) {
            node.val = node.next.val;
            node.next = node.next.next;
        }
    }
}
```

written by zwangbo original link here