

Remove Linked List Elements

Remove all elements from a linked list of integers that have value ***val***.

Example

Given: 1 --> 2 --> 6 --> 3 --> 4 --> 5 --> 6, ***val*** = 6

Return: 1 --> 2 --> 3 --> 4 --> 5

Credits:

Special thanks to [@mithmatt](#) for adding this problem and creating all test cases.

Solution 1

```
public ListNode removeElements(ListNode head, int val) {  
    if (head == null) return null;  
    head.next = removeElements(head.next, val);  
    return head.val == val ? head.next : head;  
}
```

written by [renzid](#) original link [here](#)

Solution 2

```
public class Solution {  
    public ListNode removeElements(ListNode head, int val) {  
        ListNode fakeHead = new ListNode(-1);  
        fakeHead.next = head;  
        ListNode curr = head, prev = fakeHead;  
        while (curr != null) {  
            if (curr.val == val) {  
                prev.next = curr.next;  
            } else {  
                prev = prev.next;  
            }  
            curr = curr.next;  
        }  
        return fakeHead.next;  
    }  
}
```

written by [lx223](#) original link [here](#)

Solution 3

Hi guys!

Here's an iterative solution without dummy head. First, we shift a head of a list while its' value equals to val. Then, we iterate through the nodes of the list checking if the next node's value equals to val and removing it if needed.

```
public class Solution {  
    public ListNode removeElements(ListNode head, int val) {  
        while (head != null && head.val == val) head = head.next;  
        ListNode curr = head;  
        while (curr != null && curr.next != null)  
            if (curr.next.val == val) curr.next = curr.next.next;  
            else curr = curr.next;  
        return head;  
    }  
}
```

written by [shpolsky](#) original link [here](#)

From [LeetCoder](#).