[Daily Question 2024-03-12]

1171. Remove Zero Sum Consecutive Nodes from Linked List

Given the head of a linked list, we repeatedly delete consecutive sequences of nodes that sum to 0 until there are no such sequences.

After doing so, return the head of the final linked list. You may return any such answer.

(Note that in the examples below, all sequences are serializations of ListNode objects.)

Input: head = [1,2,-3,3,1]
Output: [3,1]
Note: The answer [1,2,1] would also be accepted.

Example 2:

Input: head = [1,2,3,-3,4]
Output: [1,2,4]

Example 3:

Input: head = [1,2,3,-3,-2]
Output: [1]

Constraints:

- The given linked list will contain between 1 and 1000 nodes.
- Each node in the linked list has -1000 <= node.val <= 1000.

[답]

```
class ListNode(object):
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

class Solution(object):

    def removeZeroSumSublists(self, head):
        start = ListNode(0, head)
        prefix = start
        current = start.next
        sum = 0
        ht = {0:head}

        while current:
            sum += current.val
            ht[sum] = current.next
            current = current.next


        sum = 0
        current = start

        while current:
            sum += current.val
            current.next = ht[sum]
            current = current.next

        return start.next
```

- 해시 테이블(key = sum, value = next)의 업데이트 기능을 활용한다. 즉 합이 같아질 때, 다음을 가리키는 next를 변경한다. 이후 start.next부터 call하면 업데이트된 next를 따라 출력된다.