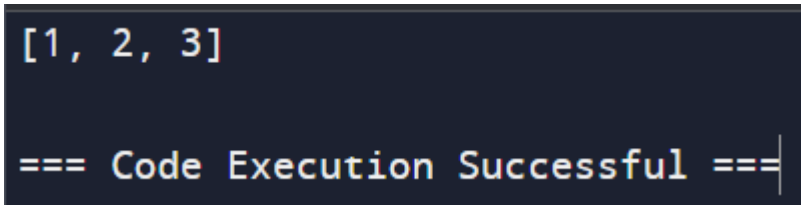
47. Remove Duplicates from Sorted List Given the head of a sorted linked list, delete all duplicates such that each element appears only once. Return the linked list sorted as well.
Example 2
Input: head = [1,1,2,3,3] Output: [1,2,3

PROGRAM:-

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

def deleteDuplicates(head):
    current = head

    while current and current.next:
        if current.val == current.next.val:
            current.next = current.next.next
        else:
            current = current.next

    return head

# Helper function to create a linked list from a list of values.
def create_linked_list(arr):
    if not arr:
        return None
    head = ListNode(arr[0])
    current = head
    for val in arr[1:]:
        current.next = ListNode(val)
        current = current.next
    return head

# Helper function to print the linked list.
def print_linked_list(head):
    result = []
    while head:
        result.append(head.val)
        head = head.next
    print(result)

# Example usage:
head = create_linked_list([1, 1, 2, 3, 3])
head = deleteDuplicates(head)
print_linked_list(head)  # Output: [1, 2, 3]
```

OUTPUT:-



TIME COMPLEXITY:-O(n)