**难度: 简单**

义一个函数，输入一个链表的头节点，反转该链表并输出反转后链表的头节点。

示例:

输入: 1->2->3->4->5->NULL

输出: 5->4->3->2->1->NULL

限制：

0 <= 节点个数 <= 5000

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/fan-zhuan-lian-biao-lcof

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

**思路(遍历节点,然后使用头插法重新插入): 代码有注释**

**注:** 头插法就是每次把节点插入到第一个

|  |
| --- |
| //把每个节点都插到节点的前面(头插法)  public ListNode reverseList(ListNode head) {  //当前节点  ListNode nowNode = head;  //当前节点的下一个节点  ListNode nextNode;  //反转节点  ListNode reverseNode = null;  //判断当前节点是否为空  while(nowNode != null) {  //把下个节点先给到nextNode  nextNode = nowNode.next;  //把当前的反转节点的第一个节点给到临时节点  ListNode temp = reverseNode;  //把当前的节点给到反转节点  reverseNode = nowNode;  //反转节点的下一个节点连接temp  reverseNode.next =temp;  //把刚刚保存的下一个节点给到当前节点  nowNode = nextNode;  }  //返回节点  return reverseNode;  } |