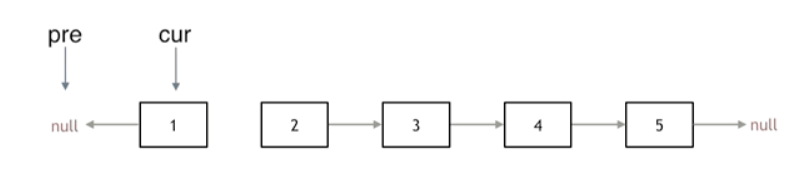
翻转链表最直接的想法就是暴力，即再新建一个链表，但是这个方法会消耗一定空间。另一个方法可以在原链表基础直接进行翻转，进而节省空间。

### 双指针法：



首先定义一个cur指针，指向头结点，再定义一个pre指针，初始化为null。然后就要开始反转了，首先要把 cur->next 节点用tmp指针保存一下，也就是保存一下这个节点。为什么要保存一下这个节点呢，因为接下来要改变 cur->next 的指向了，将cur->next 指向pre ，此时已经反转了第一个节点了。接下来，就是循环走如下代码逻辑了，继续移动pre和cur指针。最后，cur 指针已经指向了null，循环结束，链表也反转完毕了。 此时我们return pre指针就可以了，pre指针就指向了新的头结点。



### 递归法：

递归的思想同双指针差不多，主要差别在于一个基于循环，一个基于递归操作。

