模版·链表的实现

```
class Node{
    int val;
    Node next;

    public Node(int val){
        this.val=val;
        this.next=null;
    }
}

class LinkedList{
    private Node head;

    public LinkedList(){
        this.head=null;
    }

    //插入操作：在第一个值为x的结点之前插入y
    public void insert(int x,int y){
        Node prev=null;
        Node curr=head;

        while(curr!=null&&curr.data!=x){
            prev=curr;
            curr=curr.next;
        }

        if(curr!=null){//找到值为x的结点
            Node newNode=new Node(y);
            newNode.next=curr;
```

```
            if(prev==null){//x是第一个结点
                head=newNode;
            }else{
                prev.next=newNode;
            }
        }else{//链表中不存在值为x的结点，插入在末尾
            insertAtTail(y);
        }
    }

    //插入操作：在链表末尾插入y
    public void insertAtTail(int y){
        Node newNode=new Node(y);
        if(head==null){
            head=newNode;
        }else{
            Nodecurr=head;
            while(curr.next!=null){
                curr=curr.next;
            }
            curr.next=newNode;
        }
    }

    //删除操作：删除第一个值为x的结点
    public void delete(int x){
        Node prev=null;
        Node curr=head;

        while(curr!=null&&curr.data!=x){
            prev=curr;
            curr=curr.next;
            }
```

```
            if(curr!=null){//找到值为x的结点
                if(prev==null){//x是第一个结点
                    head=curr.next;
                }else{
                    prev.next=curr.next;
                }
            }
        }

        //遍历链表
        public void printList(){
            Nodecurr=head;
            while(curr!=null){
                System.out.print(curr.data+"");
                curr=curr.next;
            }
            System.out.println();
        }
}
```