```
int linklist_create(student **head);
int linklist_traverse(student *head);
int linklist_destroy(student *head);
```

```
int main()

f student *head = NULL;

if (linklist_create(&head) == OK) {
    linklist_traverse(head);
    linklist_destroy(head);
}

else
    cout << "LinkList Create failed." << endl;

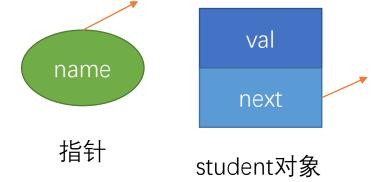
return 0;
}</pre>
```

- 1、链表的建立是否正确?
- 2、为什么遍历不成功?
- 3、链表的销毁是否成功了?
- 4、程序是否有内存丢失情况发生?如果有,发生在哪个函数被调用的阶段?
- 5、只允许修改某个函数的参数类型/该函数的声明,并在该函数内部改动一个地方,main 函数调用处改动一个地方,使程序正确,应该如何改动?(一共修改四处,且**不允许用引用**,不允许改动函数返回类型)
 - 1. 建立过程是正确的。(除了未释放之前节点那一部分)
 - 2. linklist_create 中head 是形参,单向传值,函数里改动并不会修改main 中 head 的值,导致没有将头指针信息传给 main,之后的遍历也就失败了。
 - 3. 失败了,因为头指针传的都是 NULL进去。
 - 4. 有, linklist_create 那里
 - 5. 改动: 见左边截图

```
int linklist create(student **head)
28 - {
        student *p = NULL, *q = NULL;
        int i:
32
        for (i = 0; i < 5; i++) {
           if (i > 0)
               q = p;
           p = new(nothrow) stude c; //思考: 为什么不能用malloc
           if (p == NULL)
               return ERRO: //注:此处未释放之前的链表节点,就借助操作系统来释放(非标用法)
               *head = p; //head 指向第1个结点
               q->next = p;
           cout << "请输入第" << i + 1 << "个人的基本信息" << endl;
           cin >> p->name >> p->num >> p->sex; //键盘输入基本信息
           p->next = NULL;
        return OK;
```

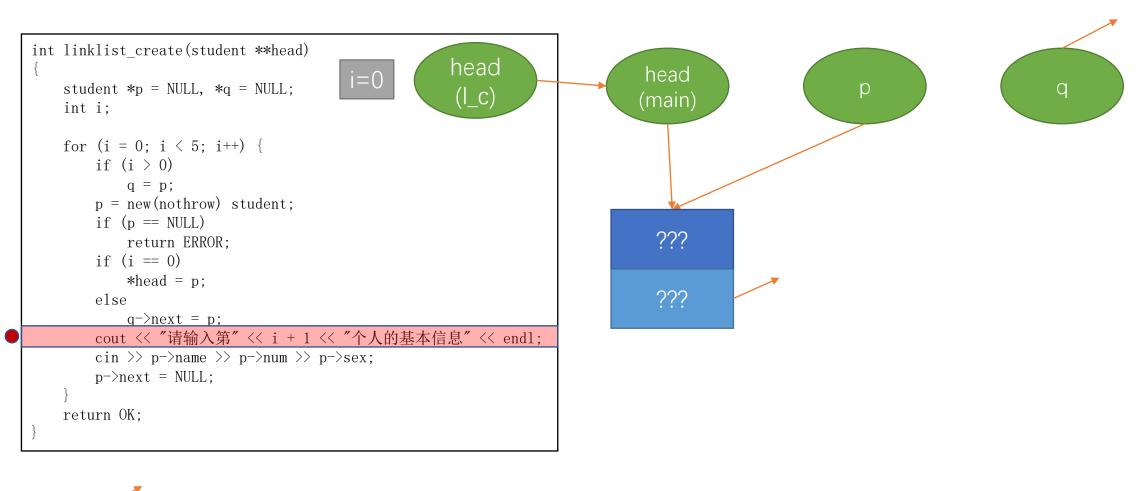
```
int linklist_create(student **head)
                                                  head
                                                                          head
   student *p = NULL, *q = NULL;
                                                   (I_c)
                                                                         (main)
   int i;
   for (i = 0; i < 5; i++)
       if (i > 0)
           q = p;
       p = new(nothrow) student;
       if (p == NULL)
           return ERROR;
       if (i == 0)
           *head = p;
       else
           q->next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << endl;
       cin >> p->name >> p->num >> p->sex;
       p->next = NULL;
   return OK;
```

q



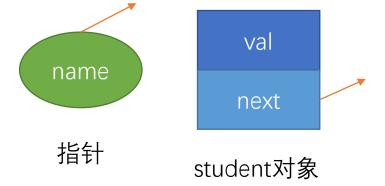
```
int linklist_create(student **head)
                                                 head
                                                                         head
                                    i=0
   student *p = NULL, *q = NULL;
                                                                                                  р
                                                                                                                         q
                                                  (l_c)
                                                                        (main)
   int i;
   for (i = 0; i < 5; i++) {
       if (i > 0)
           q = p;
       p = new(nothrow) student;
       if (p == NULL)
                                                                          ???
           return ERROR;
       if (i == 0)
           *head = p;
                                                                          ???
       else
           q->next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << end1;
       cin >> p->name >> p->num >> p->sex;
       p->next = NULL;
   return OK;
```

val name next student对象



val next https://www.next student对象

```
int linklist_create(student **head)
                                                 head
                                                                        head
                                    i=0
   student *p = NULL, *q = NULL;
                                                                                                 р
                                                                                                                        q
                                                 (l_c)
                                                                       (main)
   int i;
   for (i = 0; i < 5; i++) {
       if (i > 0)
           q = p;
       p = new(nothrow) student;
       if (p == NULL)
                                                                      第一人
           return ERROR;
       if (i == 0)
           *head = p;
                                                                         ???
       else
           q->next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << end1;
       cin \gg p-name \gg p-num \gg p-sex;
       p- next = NULL;
   return OK;
```

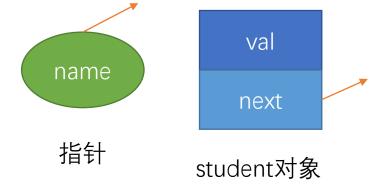


```
int linklist_create(student **head)
                                                 head
                                                                        head
                                    i=1
   student *p = NULL, *q = NULL;
                                                                                                р
                                                 (I_c)
                                                                       (main)
   int i;
   for (i = 0; i < 5; i++) {
       if (i > 0)
           q = p;
       p = new(nothrow) student;
       if (p == NULL)
                                                                      第一人
          return ERROR;
       if (i == 0)
           *head = p;
                                                                       NULL
       else
           q->next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << end1;
       cin >> p->name >> p->num >> p->sex;
       p->next = NULL;
   return OK;
```

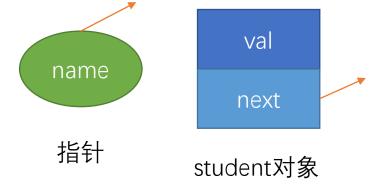
q

val name next student对象

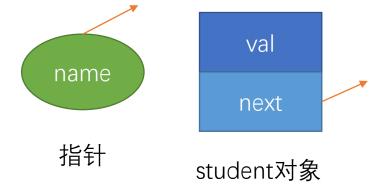
```
int linklist_create(student **head)
                                                head
                                                                       head
                                    i=1
   student *p = NULL, *q = NULL;
                                                                                                р
                                                                                                                      q
                                                 (l_c)
                                                                      (main)
   int i;
   for (i = 0; i < 5; i++) {
       if (i > 0)
           q = p;
       p = new(nothrow) student;
       if (p == NULL)
                                                                      第一人
                                                                                                    ???
          return ERROR;
       if (i == 0)
           *head = p;
                                                                       NULL
                                                                                                    ???
       else
           q->next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << end1;
       cin >> p->name >> p->num >> p->sex;
       p->next = NULL;
   return OK;
```

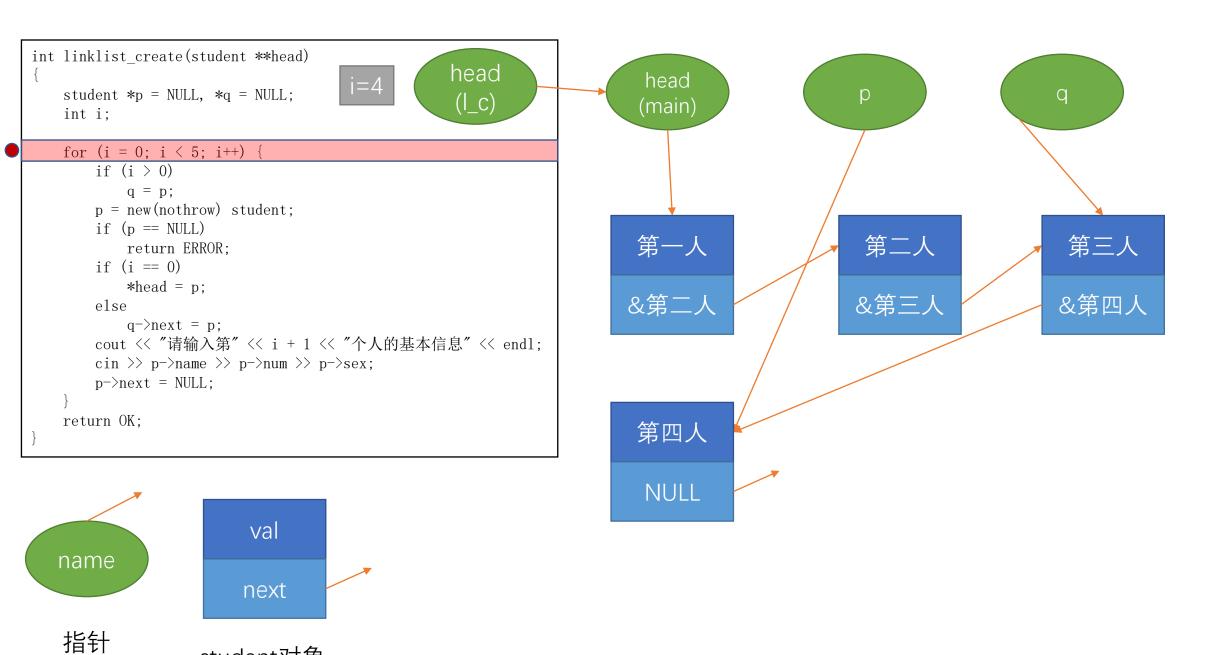


```
int linklist_create(student **head)
                                                head
                                                                      head
                                   i=1
   student *p = NULL, *q = NULL;
                                                                                               p
                                                                                                                     q
                                                 (I_c)
                                                                      (main)
   int i;
   for (i = 0; i < 5; i++) {
       if (i > 0)
          q = p;
       p = new(nothrow) student;
       if (p == NULL)
                                                                     第一人
                                                                                                   ???
          return ERROR;
       if (i == 0)
           *head = p;
                                                                    &第二人
                                                                                                   ???
       else
          q- next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << end1;
       cin >> p->name >> p->num >> p->sex;
       p->next = NULL;
   return OK;
```

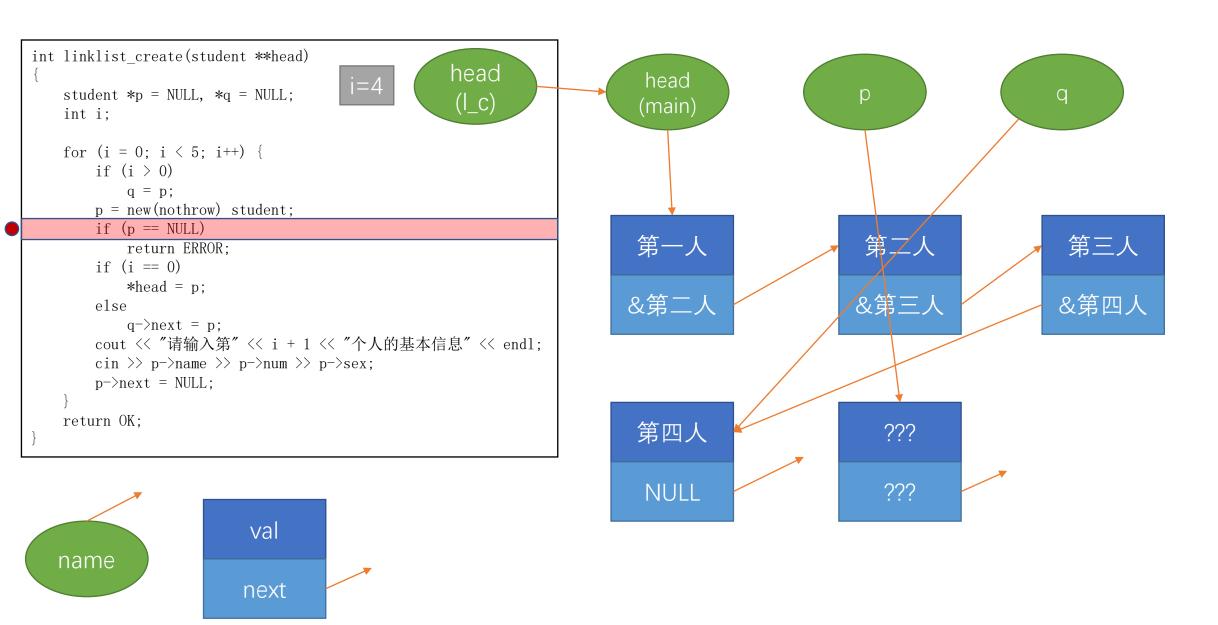


```
int linklist create(student **head)
                                               head
                                                                     head
                                   i=1
   student *p = NULL, *q = NULL;
                                                                                             р
                                                                                                                   q
                                                (I_c)
                                                                     (main)
   int i;
   for (i = 0; i < 5; i++)
       if (i > 0)
          q = p;
       p = new(nothrow) student;
       if (p == NULL)
                                                                                               第二人
                                                                    第一人
          return ERROR;
       if (i == 0)
          *head = p;
                                                                   &第二人
                                                                                                NULL
       else
          q->next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << end1;
       cin >> p->name >> p->num >> p->sex;
       p->next = NULL;
   return OK;
```



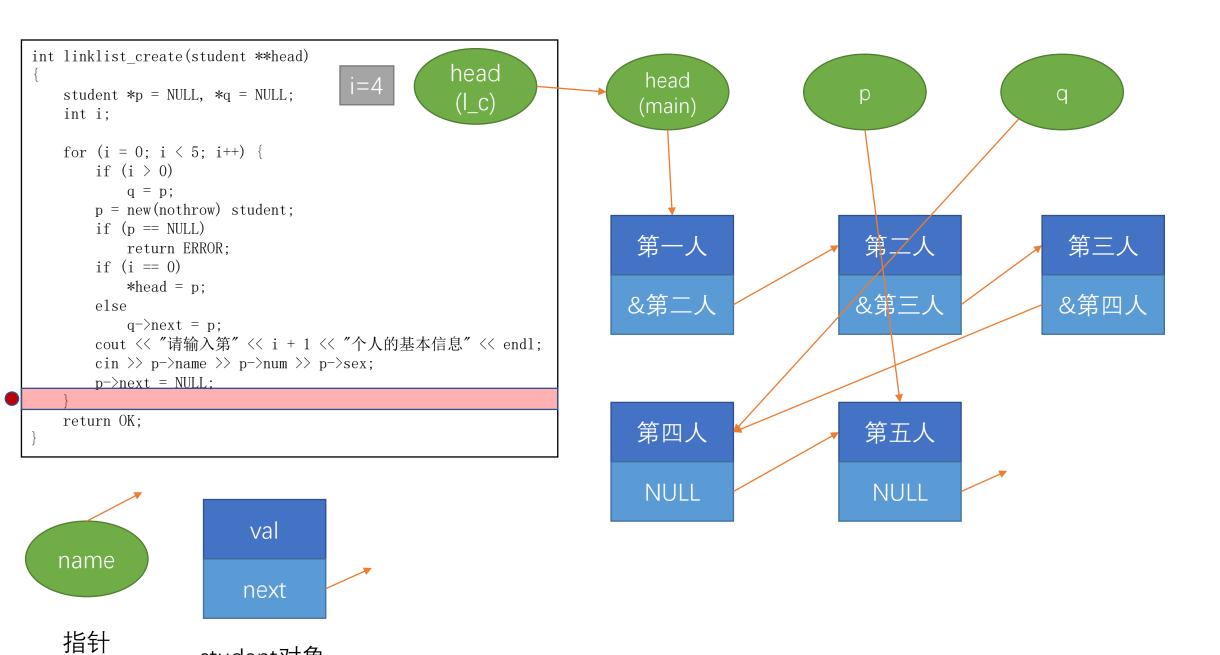


student对象



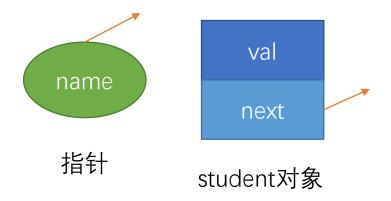
指针

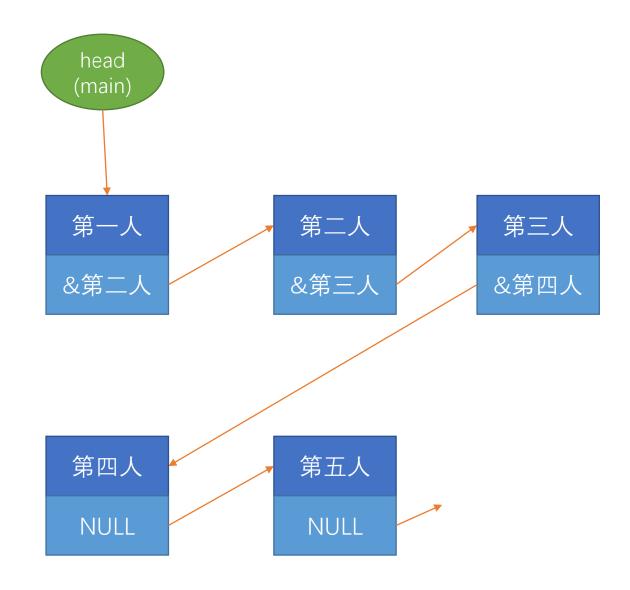
student对象



student对象

```
int linklist create(student **head)
   student *p = NULL, *q = NULL;
   int i;
   for (i = 0; i < 5; i++) {
       if (i > 0)
           q = p;
       p = new(nothrow) student;
       if (p == NULL)
           return ERROR;
       if (i == 0)
           *head = p;
       else
           q->next = p;
       cout << "请输入第" << i + 1 << "个人的基本信息" << end1;
       cin >> p->name >> p->num >> p->sex;
       p->next = NULL;
   return OK;
```

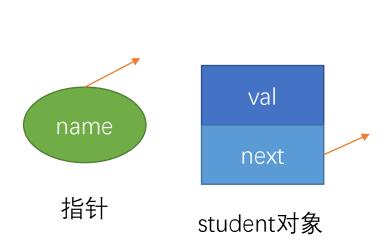


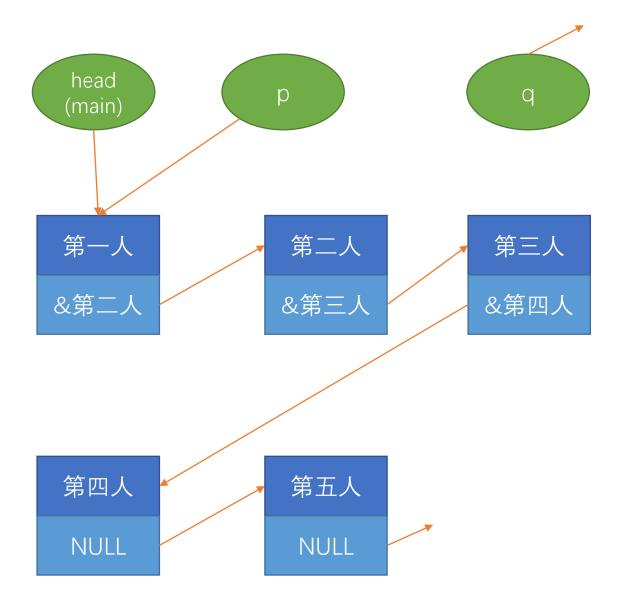


```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放

        q = p->next;
        delete p;
        p = q;
    }
    return OK;
}
```



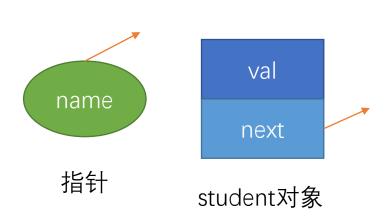


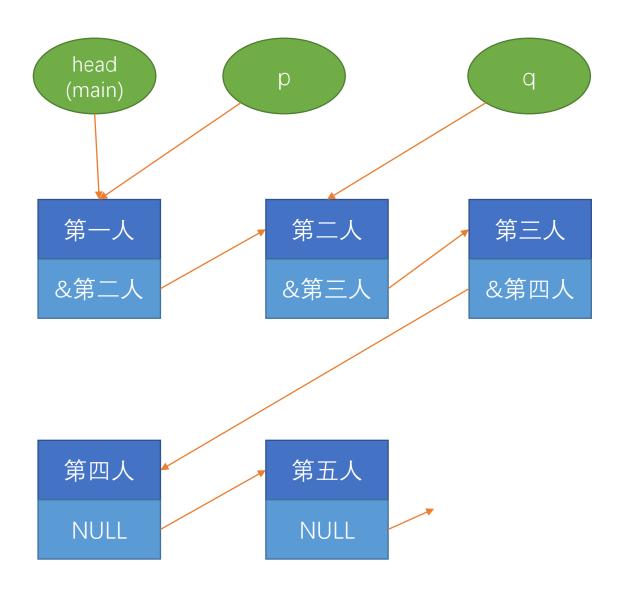
```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放
        q = p->next;

        delete p;

        p = q;
    }
    return OK;
}
```

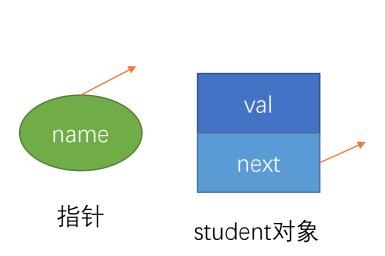


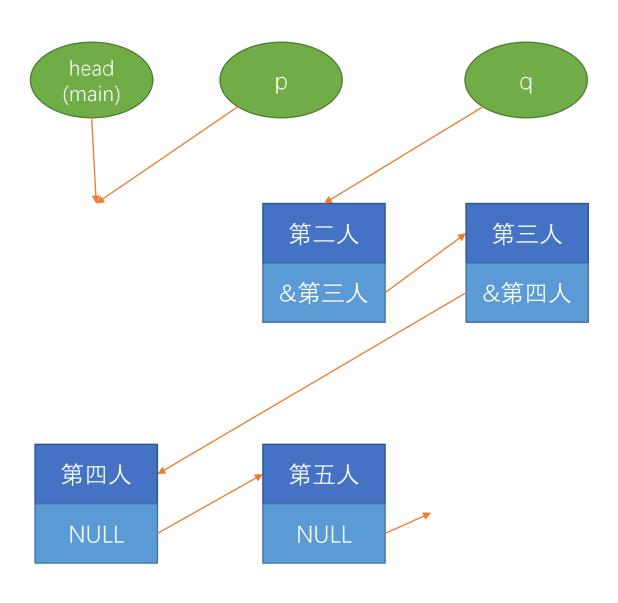


```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放
        q = p->next;
        delete p;

    p = q;
}
    return OK;
}
```

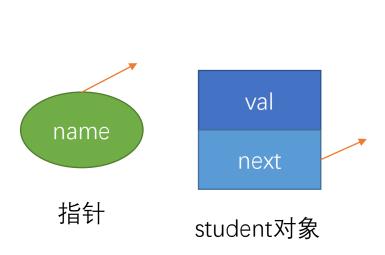


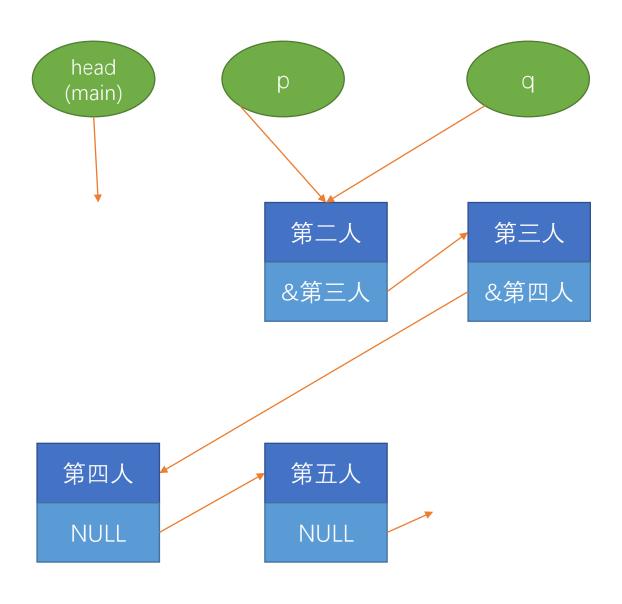


```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放

        q = p->next;
        delete p;
        p = q;
    }
    return OK;
}
```

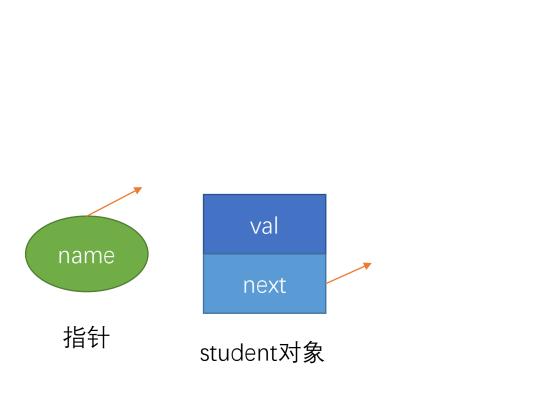


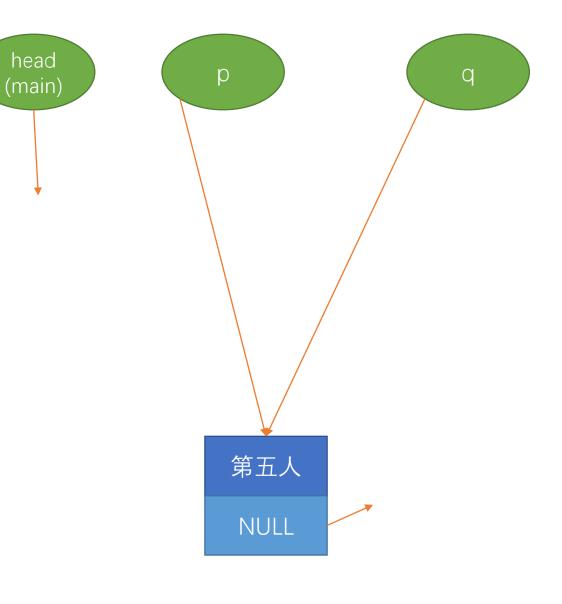


```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放

        q = p->next;
        delete p;
        p = q;
    }
    return OK;
}
```



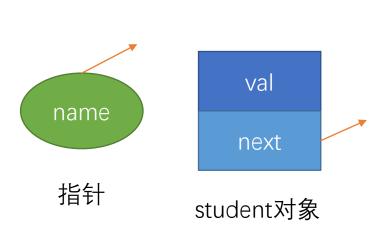


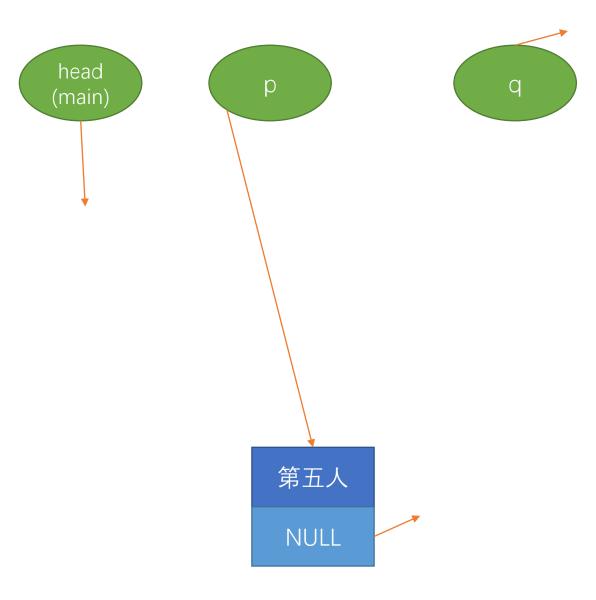
```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放

        q = p->next;

        delete p;
        p = q;
    }
    return OK;
}
```

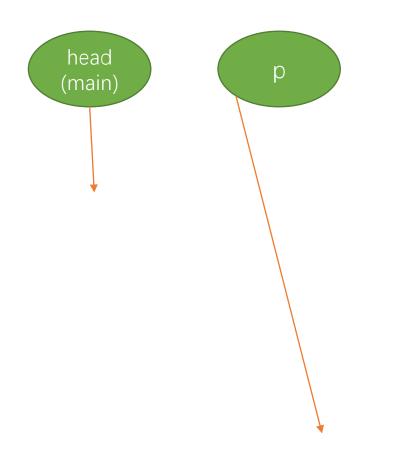


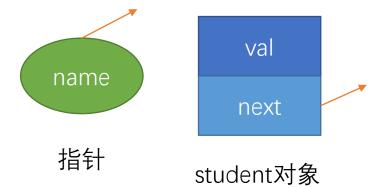


```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放
        q = p->next;
        delete p;

    p = q;
}
    return OK;
}
```

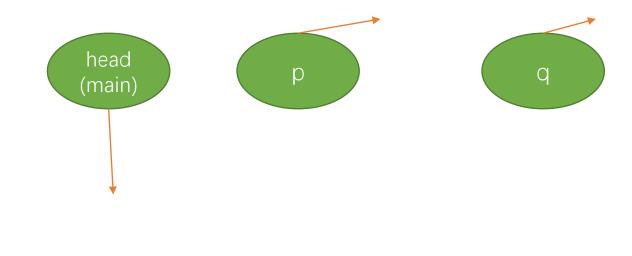




```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放

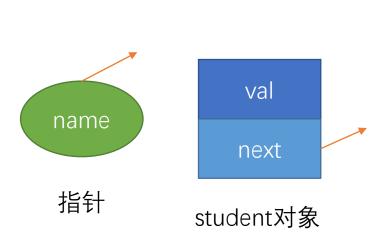
        q = p->next;
        delete p;
        p = q;
    }
    return OK;
}
```

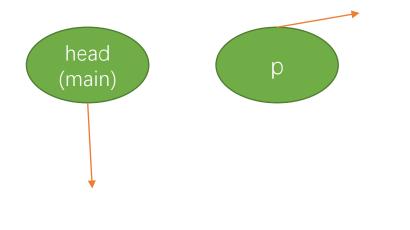


val name next student对象

```
int linklist_destroy(student *head)
{
    student *p, *q;

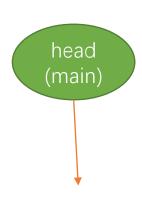
    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放
        q = p->next;
        delete p;
        p = q;
    }
    return OK;
}
```





```
int linklist_destroy(student *head)
{
    student *p, *q;

    p = head; //p复位, 指向第1个结点
    while (p) { //循环进行各结点释放
        q = p->next;
        delete p;
        p = q;
    }
    return OK;
}
```



(遍历过程类似)

