**测试题：**

1. 下边的列表分片操作会打印什么内容？

答：打印[2,9,7]

>>> list1 = [1, 3, 2, 9, 7, 8]

>>> list1[2:5]

* 1
* 2

1. 请问 list1[0] 和 list1[0:1] 一样吗？

答：不一样，前者是获得list这个列表的索引值为0的哪个元素，后者是获得list这个列表0到1的元素。

1. 如果你每次想从列表的末尾取出一个元素，并将这个元素插入到列表的最前边，你会怎么做？

答：可以先用pop函数将末尾的值取出来，并赋值给一个新的变量，然后用insert函数在最开始位置插入这个变量。

批改：list1.insert(0, list1.pop())

1. 有些鱼油比较调皮，他说我想试试 list1[-3:-1] 会不会报错，怎么知道一试居然显示 [9, 7]，这是怎么回事呢？

答：这和正数索引相反，整数索引是从左到右，从下标0开始数，负数索引是从右往左数，从下标-1开始数。

1. 在进行分片的时候，我们知道分片的开始和结束位置需要进行指定，但其实还有另外一个隐藏的设置：步长。  
   在普通的分片操作中，步长默认设置为1，表示逐个遍历元素。其实我们可以人为调整步长以达到不可告人的秘密。

>>> list1[0:6:2]

[1, 2, 7]

* 1
* 2

那么依你推测，关于步长的知识点还有哪些（很多知识点都是通用的）？

1. 课堂上小甲鱼说可以利用分片完成列表的拷贝 list2 = list1[:]，那事实上可不可以直接写成 list2 = list1 更加简洁呢？

批改：为了大家都可以弄明白这个【必须理解】的概念，我啰嗦一下：  
上节课我们提到使用分片创建列表的拷贝，例如：

>>> list1 = [1, 3, 2, 9, 7, 8]

>>> list2 = list1[:]

>>> list2

[1, 3, 2, 9, 7, 8]

>>> list3 = list1

>>> list3

[1, 3, 2, 9, 7, 8]

* 1
* 2
* 3
* 4
* 5
* 6
* 7

看上去貌似一样的，对吧？但事实上呢？我们利用列表的一个小伙伴做下修改，大家看一下差别：

>>> list1.sort()

>>> list1

[1, 2, 3, 7, 8, 9]

* 1
* 2
* 3

看到吧，我们的list1已经从小到大排了个序，那list2和list3呢？

>>> list2

[1, 3, 2, 9, 7, 8]

* 1
* 2

可以看到我们使用分片的方式得到的list2很有原则、很有格调，并不会因为list1的改变而改变，这个原理我待会儿跟大家说，我们接着看下list3：

>>> list3

[1, 2, 3, 7, 8, 9]

* 1
* 2

大家可以看到了，真正的汉奸、墙头草是list3，Ta跟着list1改变了

1. 请写下这一节课你学习到的内容：格式不限，回忆并复述是加强记忆的好方式！

总结：这节课我懂了如何从列表中获取元素以及如何从列表中删除元素以及列表分片。

从列表中获取元素可以用：变量名[索引值]。这种方法，注意：所有索引值是从0开始的。

删除列表中的元素可以用remove，del，pop三种函数。

remove是：变量名.remove(‘需要删除的元素名’)

del是：del 变量名[索引值]

pop是：变量名.pop(索引值)，这里pop是可以输出值的，即可以name=x.pop[0]，name就可以获得x第0个元素的值了

列表分片：变量名[0:5]，中间用冒号隔开，表示从哪里到哪里，要是开头不写数字，则默认从第0个元素开始；要是末尾不写数字，则默认到最后一个元素为止；要是两边都不写，则表示从开头到结尾。这样可以用于复制这个列表。