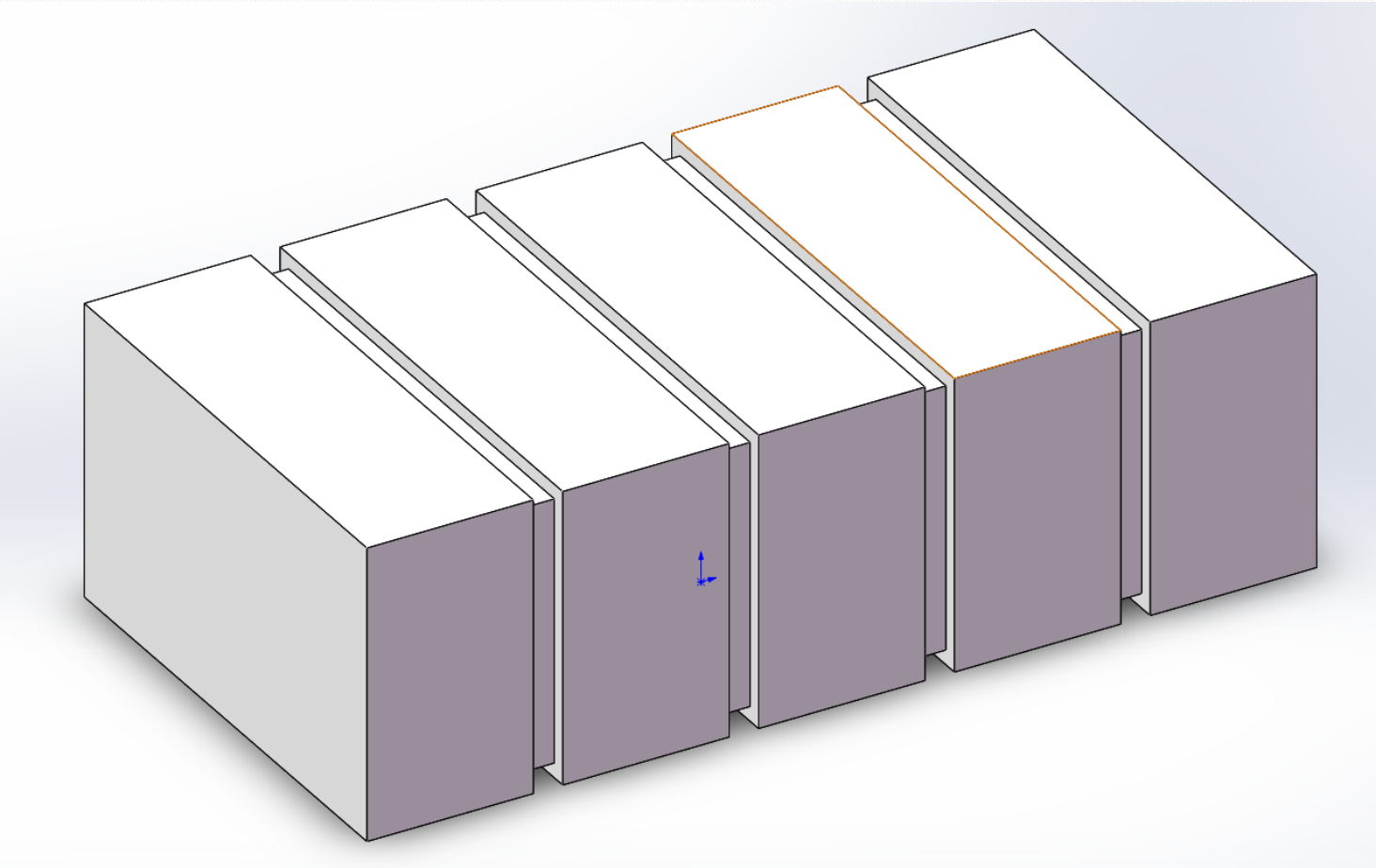
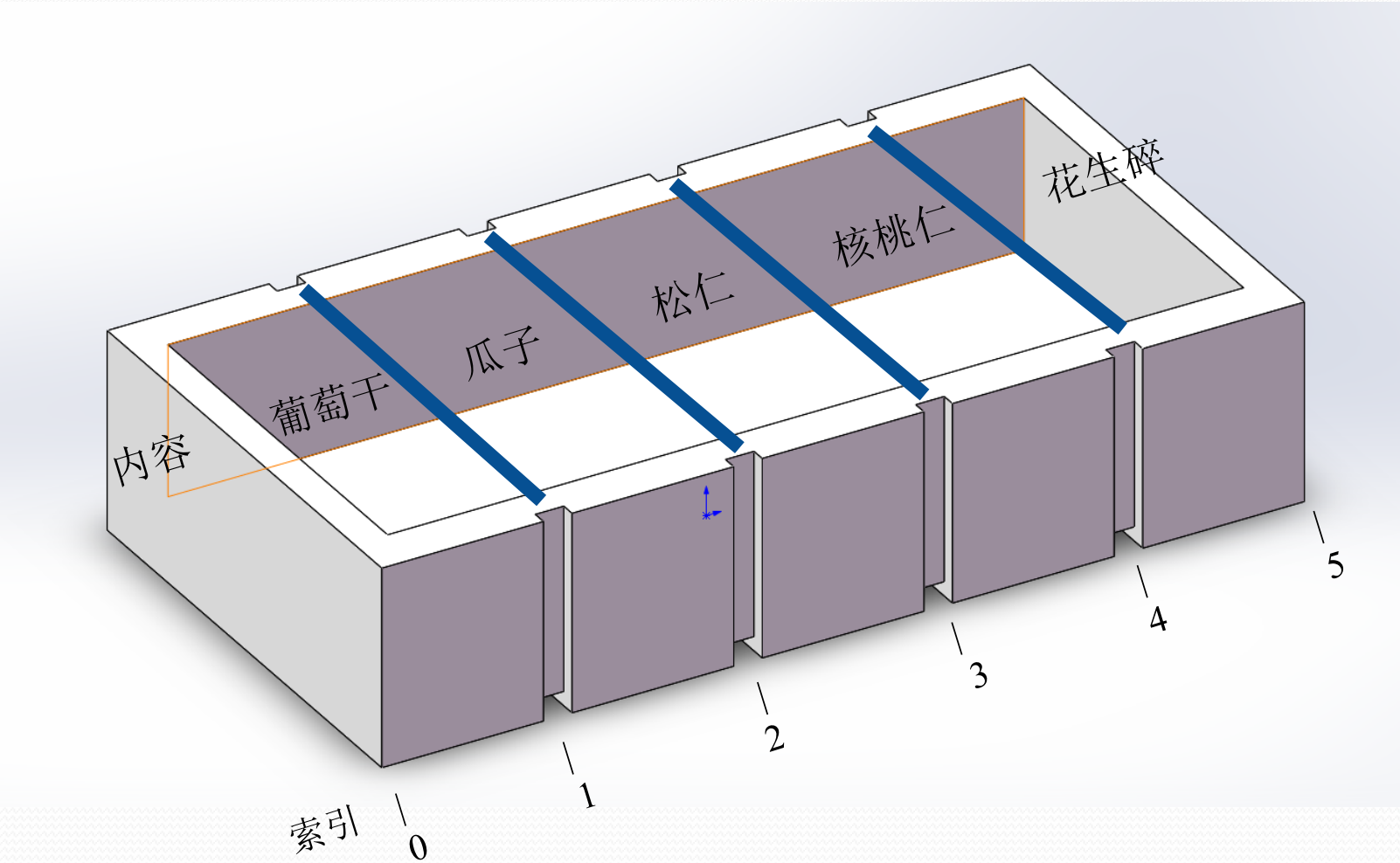
# 少儿编程

21工作室出品

- 收集起来-列表
  - 有一类集合叫列表list
    - 什么是列表
      - family = ['Mom', 'Dad', 'Junior', 'Baby']
      - luckyNumbers = [2, 7, 14, 26, 30]
      - 列表中的单个元素叫做项或者元素（item）
      - 用逗号分隔列表内的各项
    - 如何创建列表
      - 为变量赋不同类型的值
        - 数
        - 字符串
        - 列表
          - 像不像我们以前学过的嵌套？
        - 其他对象
          - 一条鱼
      - 创建一个空列表
        - newList = []
    - 如何使用列表
      - 取
        - 从列表获取元素
          - print(lettersp[0])
          - 为什么索引从0而不是1开始？
          - 索引（index）表示某个东西的位置
        - 列表“分片”
          - 面包里面有很多提子（葡萄干），我可以切片的时候，多获得几个提子
          - print(letters[1:4])
          - 与for循环中的range()类似
          - 对列表分片时，取回的是另一个（通常更小的）列表。这个更小的列表称为原列表的一个分片（slice）。
          - 原来的列表并没有改变。这个分片是原列表的部分副本（copy）。
        - 分片简写
          - print(letters[:2])
          - print(letters[2:])
          - print(letters[:])
      - 增
        - 向列表增加元素
          - friends = [] # 手机通讯录，增加一个人名<br>friends.append('David') # 附加、增补的意思<br>print(friends)
          - append继续增加
          - 向列表增加元素之前，必须先创建列表（可以时空列表，也可以是非空列表）
          - 做蛋糕之前一定要有一个容器
          - 列表的变量名就是一个对象，点号代表后面是对象的一个方法，一种操作、某种处理
          - 列表可以包含Python能存储的任何类型的数据，例如数字、字符串、对象，甚至可以包含其他列表。并不要求列表中的元素是同种类型或同一种东西。
          - letters = ['a', 'b', 'c', 'd', 'e']
        - 向列表增加元素的其他方法
          - append()
          - extend()
          - insert()
          - append()和extend()的区别
      - 删
        - 从列表删除元素
          - remove()
          - del
          - pop()
      - 改
        - 修改元素
          - print(letters)
          - letters[2] = 'z'
      - 查
        - 搜索列表
          - 查找元素是否在列表中
          - 查找元素在列表中的哪个位置（获得元素的索引号）
      - 处理
        - 循环处理列表
          - 程序
          - 循环迭代处理（循环处理）列表中的所有值
        - 列表排序
          - 列表是一种有顺序（ordered）的集合
          - 对列表排序-sort()
        - 可改变和不可改变
          - 真正改变一个数或字符串是做不到的，你能改变的只是把一个名字指派（assign）到哪个数或字符串（移动标签）
          - 可改变的类型、对象、变量
          - 不可改变的列表
        - 双重列表-嵌套
          - 数据结构（data structure）



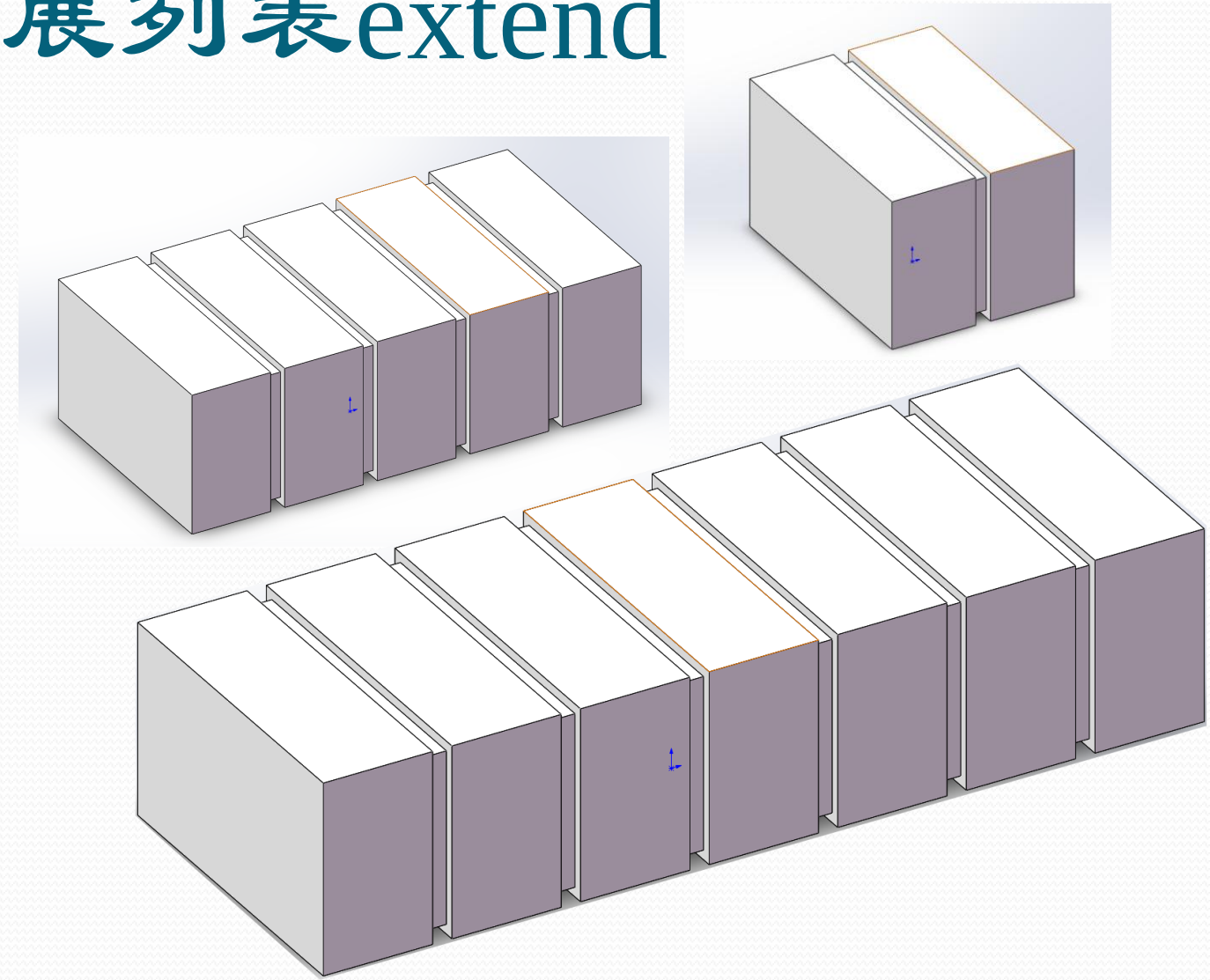
# 切片（面包）



# 从列表中获取元素

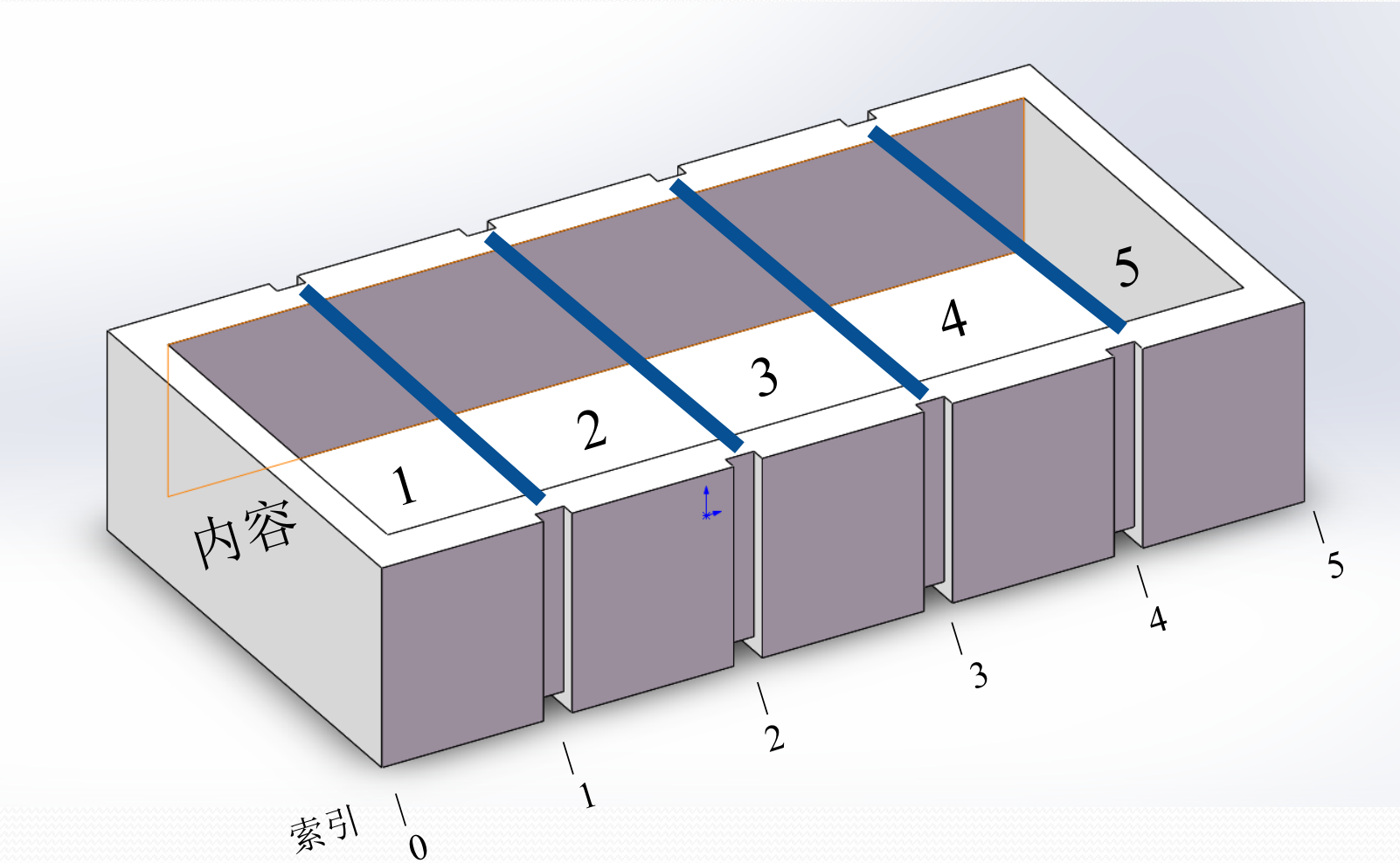
- >>> mianbao=["葡萄干", "瓜子", "松仁", "核桃仁", "花生碎“]
- >>> print(mianbao)
- ['葡萄干', '瓜子', '松仁', '核桃仁', '花生碎']
- >>> print(索引号:索引号加个数)
- >>> print(mianbao[2:4])
- ['松仁', '核桃仁']
- >>>

# 扩展列表extend



- >>> mianbao=["葡萄干", "瓜子", "松仁", "核桃仁", "花生碎“]
- >>> mianbao.extend(["芝麻", "腰果"])
- >>> print(mianbao)
- ['葡萄干', '瓜子', '松仁', '核桃仁', '花生碎', '芝麻', '腰果']

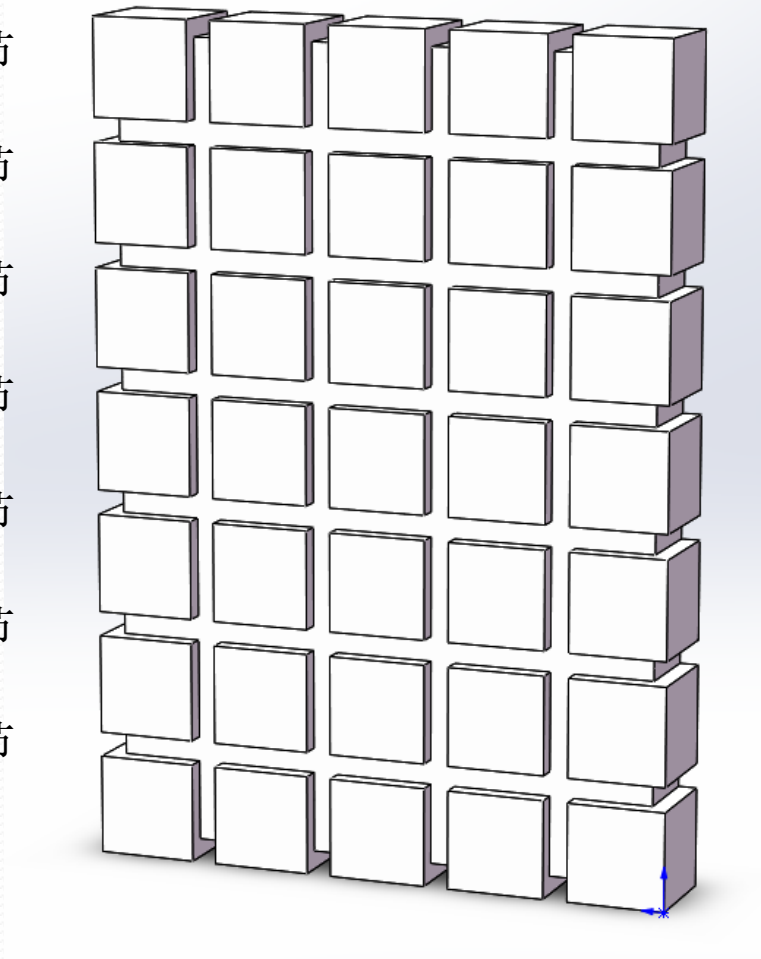
# 索引号——下标——位置信息



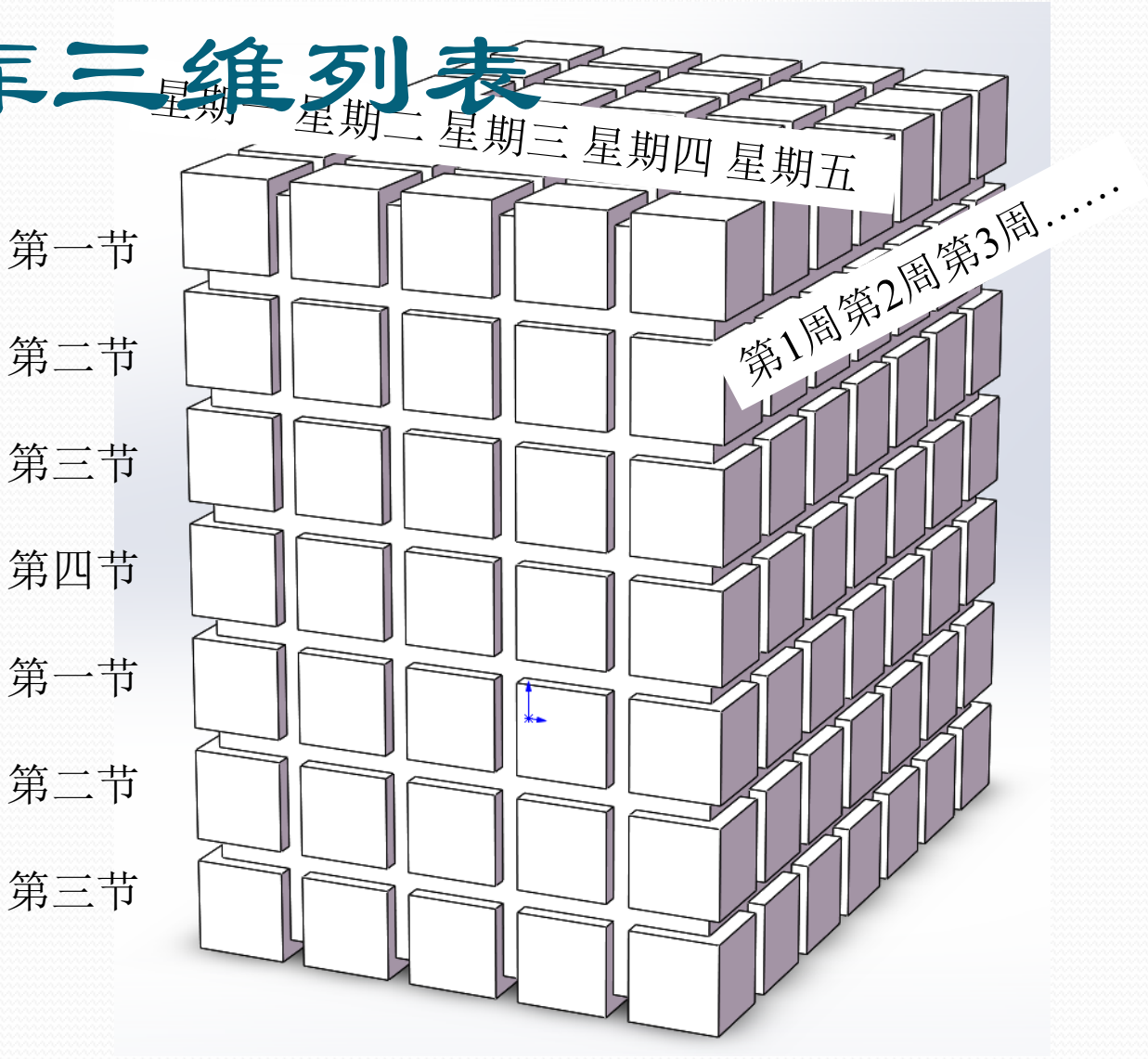
- >>> letters = ['a', 'b', 'c', 'd', 'e', 'f']
- >>> print(letters)
- ['a', 'b', 'c', 'd', 'e', 'f']
- >>> print(letters[1:4])
- ['b', 'c', 'd']
- >>> print(letters[:2])
- ['a', 'b']
- >>> print(letters[2:])
- ['c', 'd', 'e', 'f']
- >>> print(letters[:]) # 头也不限制、尾也不限制
- ['a', 'b', 'c', 'd', 'e', 'f']
- >>>

# 课程表二维列表

| | 星期一 | 星期二 | 星期三 | 星期四 | 星期五 |
|---|---|---|---|---|---|
| 第一节 | | | | | |
| 第二节 | | | | | |
| 第三节 | | | | | |
| 第四节 | | | | | |
| 第一节 | | | | | |
| 第二节 | | | | | |
| 第三节 | | | | | |



# 数据库三维列表



# 你学到了什么

- 列表是什么。
- 如何向列表中增加元素。
- 如何从列表删除元素。
- 如何确定列表是否包含某个值。
- 如何对列表排序。
- 如何建立列表的副本。
- 元组。
- 双重列表。

# 测试题

1. 向列表增加元素有哪些方法？
2. 从列表删除元素有哪些方法？
3. 要得到一个列表的有序副本，但又不能改变原来的列表，有哪两种方法？
4. 怎样得出某个值是否在列表中？
5. 如何确定某个值在列表中的位置？
6. 什么是元组？
7. 如何建立双重列表？
8. 如何从一个双重列表中得到一个值？

# 动手试一试

1. 写一个程序，让用户提供 5 个名字。程序要把这 5 个名字保存在一个列表中，最后打印出来。就像这样：

```
Enter 5 names:
Tony
Paul
Nick
Michel
Kevin
The names are Tony Paul Nick Michel Kevin
```

2. 修改第 1 题的程序，要求不仅显示原来的名字列表，还要显示出排序后的列表。
3. 修改第 1 题的程序，要求只显示用户键入的第 3 个名字，就像这样：

```
The third name you entered is: Nick
```

4. 修改第 1 题的程序，让用户替换其中一个名字。用户应该能选择要替换哪个名字，然后键入新名字。最后显示这个新的列表：

```
Enter 5 names:
Tony
Paul
Nick
Michel
Kevin
The names are Tony Paul Nick Michel Kevin
Replace one name. Which one? (1-5): 4
New name: Peter
The names are Tony Paul Nick Peter Kevin
```