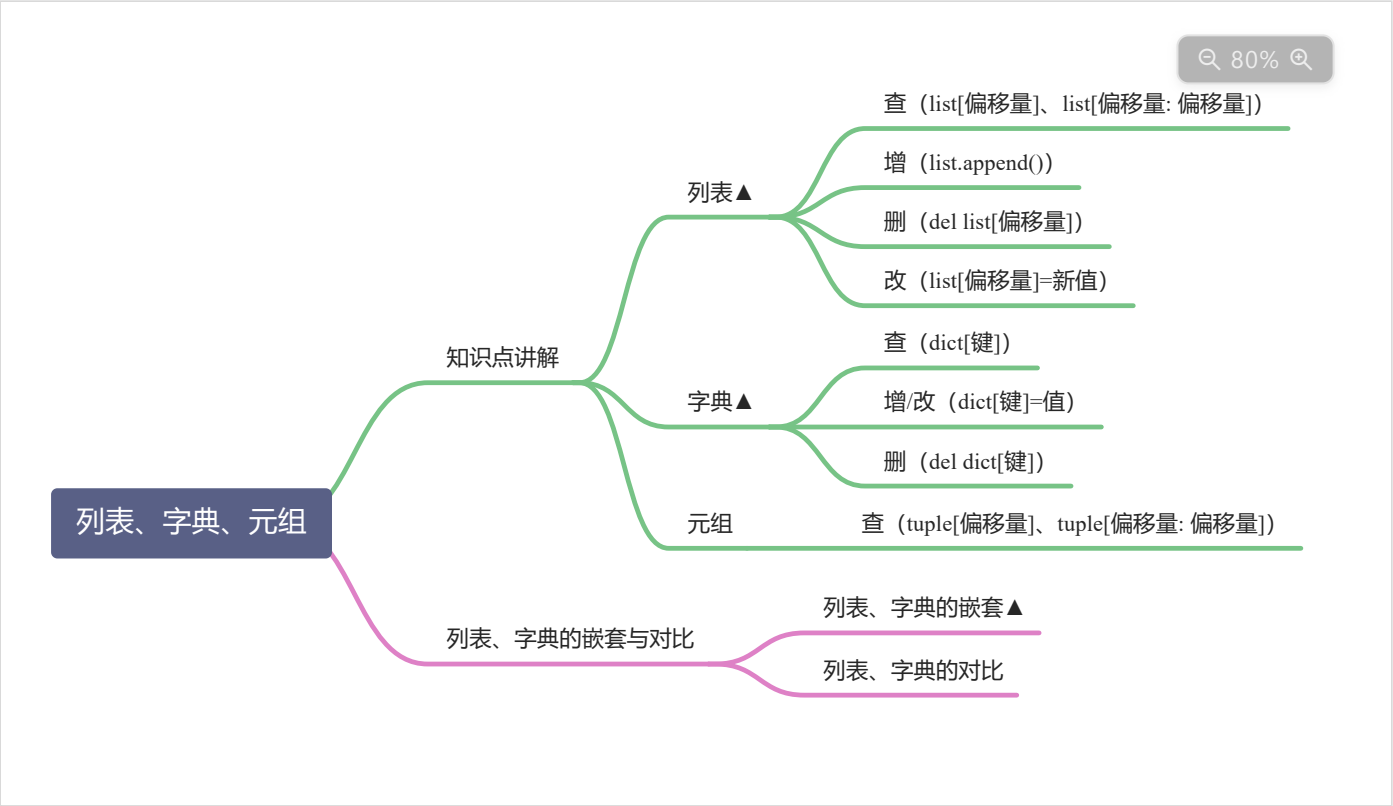


第4课 列表、字典、元组

一、课程结构导图

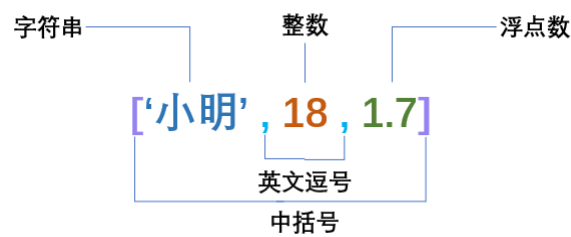


注：▲为重点知识点。

二、知识点讲解

2.1 列表

概念： 列表是一个任意类型的对象的位置相关的有序集合¹。形如：[元素1, 元素2, 元素3,], 使用中括号框起来，各个元素之间使用英文逗号隔开。注意：元素可以是任意类型。



风变科技

风变科技

风变科技

用法： 列表中的元素是有顺序的，每个元素都有两个编号，一个是正向的编号，一个是负向的编号，这些编号都称为列表元素的偏移量或者索引值，索引值（偏移量）主要用于列表取值。正向偏移量从0开始数起，负向偏移量从-1开始数起。

示例：

列表：['小明', 18, 1.7]

| 元素 | '小明' | 18 | 1.7 | 注： |
|------|------|----|-----|--------------|
| 正向索引 | 0 | 1 | 2 | 正向从0开始，往右递增 |
| 负向索引 | -3 | -2 | -1 | 负向从-1开始，往左递减 |

风变科技

风变科技

2.1.1 查（查找元素）

用法1： 列表名[偏移量]，使用该语法取出列表中的元素。

示例1：

```
1 # 取 18
2 list = ['小明', 18, 1.7]
3 print(list[1]) # 正向取
4 print(list[-2]) # 负向取
```

用法2： 切片：使用 [左偏移量 : 右偏移量] 的方式取值，这种取值方式叫做切片。冒号可以理解为一直取，切片的取值规则简单概括为：**左取右不取，空则取到头。**

风变科技

示例2:

| | | | | | |
|--------------------------------------|-----|-----|-----|-----|-----|
| 列表: students = ['a','b','c','d','e'] | | | | | |
| 元素 | 'a' | 'b' | 'c' | 'd' | 'e' |
| 正向索引 | 0 | 1 | 2 | 3 | 4 |

列表: students = ['a','b','c','d','e']

student[1:3] → 非空, 左取右不取 → ['b','c']

student[:3] → 左空, 空取到头, 右不取 → ['a','b','c']

student[3:] → 右空, 空取到头, 左取 → ['d','e']

student[:] → 左右空, 空取到头 → ['a','b','c','d','e']

2.1.2 增 (增加元素)

用法: 使用append()函数可以给列表增加元素, 新增的元素放在append()括号中, 并添加到列表的最后。语法: **列表名.append(新增元素)**。

示例:

```

1 students = ['a', 'b', 'c', 'd', 'e']
2 students.append(2)
3 print(students)
4 # 结果: ['a', 'b', 'c', 'd', 'e', 2]
5
6 students.append([1,2])
7 print(students)
8 # 结果: ['a', 'b', 'c', 'd', 'e', 2, [1,2]]

```

注意: 使用append()函数只能传入一个元素, 不能同时传入多个元素, 如第6行代码, 改为students.append(1,2)则会报错。

2.1.3 删（删除元素）

用法：通过指定列表中的偏移量，对列表中的某个元素进行删除，语法：**del 列表名[偏移量]**。

示例：

```
1 students = ['a', 'b', 'c', 'd', 'e']
2 del students[2]      # 删除'c'
3 print(students)
4 # 结果: ['a', 'b', 'd', 'e']
```

2.1.4 改（修改元素）

用法：通过指定列表中的偏移量，对列表中的某个元素进行修改，语法：**列表名[偏移量]=改后元素**。

示例：

```
1 students = ['a', 'b', 'c', 'd', 'e']
2 students[1] = 6      # 把'b'改为整数6
3 print(students)
4 # 结果: ['a', 6, 'c', 'd', 'e']
```

2.2 字典

概念：字典是一个含有一对对数据，且各个数据对的值可以是任意类型的无序集合。形如：{键1: 值1, 键2: 值2, 键3: 值3,}，字典使用大括号框起来，大括号中使用逗号隔开每一键值对，每对键值对再使用英文冒号隔开。



注：键和值都可以是字符串、整数、浮点数；
键值对间使用逗号隔开，键值对内使用冒号隔开。

注意：键有唯一性，必须保证不可变，像列表是可变的对象，可以进行增删改，故不能作为字典的键。

2.2.1 查（提取键值）

用法：通过键来取值，语法：**字典名[键]**。

示例：

```
1 scores = {'小明':95,'小红':92,'小刚':90}
2 print(scores['小红'])    # 查看小红的分数
3 # 结果：92
```

注意：可以通过for循环来遍历提取字典的键，第5关会涉及。

2.2.2 增/改（增加/修改值）

用法：使用语法 **字典名[键]=值**，通过搜索字典的键，对字典的键值进行增加/修改。当增加字典键值对时，会自动添加到字典的最后，作为字典新的键值对。

示例：

```
1 scores = {'小明':95,'小红':92,'小刚':90}
2
3 # 修改字典键值
4 scores['小明'] = 100
5 print(scores)
```

```
6 # 结果: {'小明': 100, '小红': 92, '小刚': 90}
7
8 # 添加字典键值对
9 scores['小薇'] = 99
10 print(scores)
11 # 结果: {'小明': 100, '小红': 92, '小刚': 90, '小薇': 99}
```

2.2.3 删 (删除键值对)

用法: 使用del语句, 语句后面加一个空格, 再加上字典名及偏移量, 语法为: **del 字典名[键]**。

示例:

```
1 scores = {'小明':95,'小红':92,'小刚':90}
2 del scores['小明']
3 print(scores)
4 # 结果: {'小红': 92, '小刚': 90}
```

2.3 元组

概念: 元组也是一个任意类型的对象的位置相关的有序集合。形如: (元素1, 元素2, 元素3,), 它和列表很相似, 不同点在于元组创建使用小括号, 而且只能进行“查”, 不能“增”、“删”、“改”。



用法： 元组中的元素是有顺序的，它们都有2个编号，一个是正向的编号，一个是负向的编号，这些编号都称为列表元素的偏移量或者索引值，它们主要用于列表取值。

示例：

风变科技

元组：('小明', 18, 1.7)

风变科技

| 元素 | '小明' | 18 | 1.7 | 注： |
|------|------|----|-----|--------------|
| 正向索引 | 0 | 1 | 2 | 正向从0开始，往右递增 |
| 负向索引 | -3 | -2 | -1 | 负向从-1开始，往左递减 |

2.3.1 查（提取元素）

用法1： 使用偏移量取值，语法为：元组名【偏移量】

示例1：

风变科技

风变科技

```
1 # 取 18
2 tuple = ('小明',18,1.7)
3 print(tuple[1])      # 正向取
4 print(tuple[-2])     # 负向取
```

用法2： 切片（同列表）。

示例2：

风变科技

风变科技

风变科技

风变科技

元组: `students = ('a','b','c','d','e')`

`student[1:3]` — 非空, 左取右不取 —→ `['b','c']`
`student[:3]` — 左空, 空取到头, 右不取 —→ `['a','b','c']`
`student[3:]` — 右空, 空取到头, 左取 —→ `['d','e']`
`student[:]` — 左右空, 空取到头 —→ `['a','b','c','d','e']`

三、列表、字典的嵌套与对比

3.1 列表、字典嵌套

概念: 字典中含有字典或列表、列表中含有字典或列表从而形成一个复杂的数据结构。

用法1: 字典中加入字典或列表。

示例1:

```
1 scores = {  
2     '第一组': {'小明': [95, 96], '小红': [90, 92]},  
3     '第二组': {'小强': [99, 93], '小兰': [89, 98]}  
4 }  
5 print(scores['第二组']['小兰'][0])  
6 # 结果: 89
```

代码解析: 在取值的过程中, 分别对每一层单独进行取值, 比如: `scores`的最外层是字典, 可以简化为 `{'第一组': xx, '第二组': xx}`, 往里一层也是字典, 内层字典可以简化为 `{'小明':xx, '小红':xx}`和 `{'小强':xx, '小兰':xx}`, 再进一层就是列表。由此从外及里一层层剥开查看数据的结构。

用法2: 列表中加入字典或列表。

示例2:

```
1 scores = [  
2     {'小明':[95,96], '小红':[90,92]},  
3     {'小强':[99,93], '小兰':[89,98]}  
4 ]  
5 print(scores[1]['小兰'][0])  
6 # 结果: 89
```

代码解析：同用法1。在取值的过程中，分别对每一层单独进行取值，比如：scores的最外层是列表，可以简化为[元素1, 元素2]，往里一层是字典，字典可以简化为{'小明':xx,'小红':xx}和{'小强':xx,'小兰':xx}，'xx'也是一层列表。由此从外及里一层层剥开查看数据的结构。

3.2 列表、字典对比

| | 列表 | 字典 |
|------|--|---|
| 基本元素 | 中括号、逗号，元素； 如['小明',17,5.2] | 大括号、键值对(冒号)、逗号； 如{'小明':95,'小红':90,'小刚':90} |
| 查找元素 | 格式：列表名[偏移量]。 偏移量：正向从0开始计算，负向从-1开始。 切片取值：左取右不取，空则取到头。 | 格式：字典名[键] |
| 增加元素 | 列表名.append() | 字典名[键] = 值 |
| 修改元素 | 列表名[旧值的偏移量]=新元素值 | 字典名[键] = 新值（同字典增加元素） |
| 删除元素 | del 列表名[元素的偏移量] | del 字典名[键] |
| 区别 | 列表有偏移量，通过偏移量对元素值进行操作 | 字典有键值对，通过键对值进行操作 |

注意：所有的符号都是英文符号！

四、巩固练习

1.（实操题）两个小组的同学，语文和数学学科在期中和期末的考试成绩的记录如下：

```
1 scores = {  
2     '第一组': {'小明': {'语文': [94, 96], '数学': [88, 96]}, '小红': {'语文': [89, 93], '数学': [91, 89]}},  
3     '第二组': {'小强': {'语文': [92, 95], '数学': [90, 91]}, '小兰': {'语文': [91, 95], '数学': [86, 98]}}  
4 }
```

注：列表第一个元素是期中考试成绩，第二个元素是期末考试成绩，例如：小明语文学科期中考试成绩是94，期末考是96分。

要求1：小强想查一下自己两个科目在期末的成绩，请你来帮小强查一下，并打印出来；

要求2：经核实，小红的期末数学成绩登记错误，实际是92分，请你来帮小红改一下成绩；

要求3：由于老师录入数据时疏忽，把“小蓝”的名字，录成了“小兰”，请你帮忙更正。

2. (单项选择题) 地摊文化兴起，小华购进了6种花，如下：flowers = ['风信子', '富贵竹', '勿忘我', '迷迭香', '栀子花', '绿萝']。经过一段时间的观察，小华发现，多肉市场不错，而风信子滞销，于是她决定采购多肉，下架风信子，现在需要对列表进行增、删，请你来帮小华选出正确的做法（ ）

A、flowers.append('多肉')、del.flowers[0]

B、flowers.append('多肉')、del flowers[0]

C、flowers = flowers.append('多肉')、del flowers[0]

D、flowers = flowers.append('多肉')、del.flowers[0]

3. (单项选择题) 字典{'小明': 100, '小红': 92, '小明': 90, '小薇': 99}有多少个键值对？

A、1 B、2 C、3 D、4

参考文献：

1. Mark Lutz著，李军 刘宏伟等译，《Python学习手册》，第98页。