



学习目标

1. 掌握列表常用方法;
2. 使用 for 循环遍历列表;
3. 掌握公共方法的使用;
4. 了解元组与列表的不同;

传智播客-黑马程序员



目录

第1章 变量进阶-----变量类型.....	3
第2章 变量进阶-----列表.....	3
第3章 变量进阶-----公共方法.....	7
第4章 变量进阶-----元组.....	8

传智播客-黑马程序员



第 1 章 变量进阶-----变量类型

Python 中数据类型可以分为数字型和非数字型。

- 数字型
 - 整型 (int) ;
 - 浮点型 (float) ;
 - 布尔型 (bool)
 - ◆ 真 True 非 0 数 —— 非零即真 ;
 - ◆ 假 False 0 ;
- 非数字型
 - 列表;
 - 元组;
 - 集合;
 - 字典;
 - 字符串。

在 Python 中，所有非数字型变量都支持以下特点：

1. 都是一个序列 ;
2. 通过 变量名[索引] 方式取值 ;
3. 通过 for in 遍历;
4. 可以计算长度、最大和最小值 。

第 2 章 变量进阶-----列表

List (列表) 是 Python 中使用最频繁的数据类型，在其他语言中通常叫做数组，专门用于存储 一串信息 。

- 列表用 [] 定义，列表中的数据之间使用 , 分隔 ;
- 列表的索引从 0 开始 ;
- 索引就是数据在列表中的位置编号，索引又可以被称为下标 ;



- 注意：从列表中取值时，如果超出索引范围程序会报错。

```
# 定义一个空列表变量，名字叫 a
a = []
# 定义一个列表变量，名字叫 list1，有三个元素
list1 = ["刘备", "关羽", "张飞"]
# 显示列表第一个元素的值
print(list1[0])
# IndexError: list index out of range
# 错误，列表没有[3]这个值
print(list1[3])
```

一、查看列表所有方法

通过 dir() 函数查看某类型中定义的方法

```
# 定义一个列表变量，名字叫 list1，有三个元素
list1 = ["刘备", "关羽", "张飞"]
# 通过 dir 函数显示列表所有的方法
print(dir(list1))
```

二、列表常用方法

分类	方法	说明
增加	insert(索引,数据)	在指定位置插入数据
	append(数据)	在末尾追加数据
	extend(列表)	追加另一个列表的值
修改	列表[索引] = 值	修改指定索引的数据
删除	del(列表[索引])	删除指定索引的数据
	remove(数据)	删除第一次出现的指定数据
	pop()	删除末尾数据
	pop(索引)	删除指定索引数据
	clear()	清空列表
统计	count(数据)	返回指定数据在列表中出现的次数
	index(数据)	返回数据在列表中的索引值，如果找不到抛出异常
排序	sort()	升序排序



	<code>sort(reverse=True)</code>	降序排序
	<code>reverse()</code>	逆置，反转

1. 课堂练习---

定义一个空列表变量
向列表内添加 5, 9, 13 这三个数字

2. 课堂练习---

定义一个列表变量，内容如下
["张飞", "刘备", "关羽", "刘邦", "刘老二", "曹操"]
把"刘老二"修改为"周瑜"

三、循环遍历列表

遍历就是从头到尾依次从列表中获取数据

```
# 定义一个列表变量，名字叫list1，有三个元素
list1 = ["刘备", "关羽", "张飞"]
# 通过for 循环遍历list1 中所有的元素
for n in list1:
    print(n)
```

在 Python 中，可以使用 for 循环遍历所有非数字型类型的变量，包括：列表、元组、字典以及字符串。

1. 课堂练习---

定义一个列表变量，内容如下
[0, 3, 3, 9, 10, 3, 5]



使用 for 循环遍历计算列表中一共有多少数字

2. 课堂练习---

定义一个列表变量，内容如下

[0, 3, 3, 9, 10, 3, 5]

计算列表中所有数字相加的总和

四、拆包

- 拆包就是把一个列表中每个值拆出来;
- 拆包操作同样适用于元组，集合和字典;

```
# 定义一个列表
list1 = ["张三", 30, 4.5]
# 通过对列表进行拆包方式获取列表中每个元素的值
a, b, c = list1
print(a, b, c)
```

五、列表推导式

- 所谓的列表推导式，就是指轻量级的循环创建列表的方法。

1. 基本的方式

```
# a 的内容为[0, 1, 2, 3]
a = [x for x in range(4)]
# a 的内容为[2, 3]
a = [x for x in range(2, 4)]
# a 的内容为[3, 5, 7, 9]
a = [x for x in range(3, 10, 2)]
```



2. 在推导过程中使用if

```
# a 的内容为[4, 6, 8]
a = [x for x in range(3, 10) if x % 2 == 0]
# a 的内容为[3, 5, 7, 9]
a = [x for x in range(3, 10) if x % 2 != 0]
```

3. 课堂练习---用列表推导式，创建一个列表，内容为[0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

第3章 变量进阶-----公共方法

分类	方法	说明
统计	len(列表)	返回列表元素个数
	max(列表)	返回列表中最大的数据
	min(列表)	返回列表中最小的数据
成员运算符	值 in 列表	判断值是否在列表中，存在返回 True，不存在返回 False
	值 not in 列表	判断值是否在列表中，存在返回 False，不存在返回 True

公共方法同样适用于元组，集合，字典，字符串等类型

1. 课堂练习---

定义一个列表变量，内容如下
["张飞", "刘备", "关羽", "刘邦", "刘老二", "曹操"]
查找列表中是否有刘备，如果有将其删除

2. 课堂练习---

定义一个列表变量，内容如下
[3, 5, 67, 2, 34, 12, 5, 11]
显示列表中最大值



第4章 变量进阶-----元组

- Tuple（元组）与列表类似，不同之处在于元组的元素不能修改；
- 元组用 () 定义；

```
# 定义一个元组，名字叫tuple1，有三个元素
tuple1 = ("孙权", 28, 1.75)
# 定义一个空元组，名字叫tuple2
tuple2 = ()
```

- 元组只有一个元素时，元素后面需要添加逗号

```
# 元组中只包含一个元素时，需要在元素后面添加逗号
tuple3 = (50, )
```

- 定义元组时，()可以省略

```
tuple4 = "张飞", 30
tuple5 = "吕布",
```

- 元组只有一个元素，尝试后面加和不加逗号的差别

```
a = ("hello", )
a = ("hello")
```

一、元组常用方法

元组的常用方法与列表类似，元组的内容不可修改，所以没有增加、修改和删除相关方法。

二、循环遍历元组

与遍历列表类似。



三、元组和列表之间的转换

- 把列表转化为元组目的是让列表不可以被修改，以保护数据安全；
- 使用 list 函数可以把元组转换成列表；
- 使用 tuple 函数可以把列表转换成元组。

```
list1 = ["刘备", "关羽", "张飞"]  
# 通过 tuple 函数把 list1 转换为元组  
tuple1 = tuple(list1)  
  
tuple2 = ("孙权", "周瑜", "鲁肃")  
# 通过 list 函数把 tuple2 转换为列表  
list2 = list(tuple2)
```

1. 课堂练习---

```
list1 = ["刘备", "关羽", "张飞"]  
tuple1 = ("曹操", "周瑜")  
# 将元组 tuple1 的元素追加到 list1 元素后面
```