

13.数据类型阶段总结

数据类型分类

基础数据类型：

- 整数
 - int
 - 进行数学运算
 - 逻辑运算
 - 比较运算
 - 位运算（不需要掌握）
- 浮点数
- 布尔类型
 - True
 - False
- 字符串
 - 去除前后的空白字符
 - 字符串替换
 - 字符串拼接 `join`
 - 字符串格式化
 - 原始字符串
- 列表
 - 容器，可以存放任何数据类型
 - 增删改查
 - 推导式
- 元组
 - 不可变
 - 元组解包
- 集合
 - 无序的
 - 元素或者是成员不重复
- 字典
 - key-value的键值对，索引速度非常快，主要就是通过键名来索引
- None

- 表示什么都没有，在其它编程语言里面表示null，nil

按是否可变分类

可变与否在于是否能在原来数值的基础上进行修改，一个最简单的方法就是看看该类型是否可哈希，也就是是否可以使用 `hash` 函数来计算得到结果。

可变数据类型：

- 列表
- 可变集合
- 字典

不可变数据类型

- 整数
- 浮点数
- 元组
- 布尔类型

序列类型和非序列类型

符号表示：sequence

序列就是有顺序的队列，而且顺序都是从0开始的，然后往后递增

序列类型可以进行的操作：

- 可以通过下标进行访问里面的元素
- 可以进行切片
- 可以使用循环来遍历里面的内容

序列类型：

- 字符串
- 列表
- 元组

非序列类型：

- 整数
- 浮点数
- 布尔类型
- 集合
- 字典

可迭代数据类型

符号表示：iter、iterable

- 可迭代就是可以通过循环来一个个的遍历里面的元素
- 都可以通过推导式来初始化对应的类型

可迭代数据类型：

- 字符串
- 列表
- 元组
- 集合
- 字典

条件判断语句

如果 `bool(obj)` 为 `True` 那么条件就成立，否则就不成立

各个数据类型的零值默认的布尔条件就是假 `False`

利用类型的不为零值就是True的特点，可以简化条件表达式，提升效率。

只有需要判断元素个数作为条件的时候，才需要使用len方法

对于多个逻辑判断的，需要加括号来明确优先级

循环语句

for循环是有次数限制，while循环可以是死循环。但是要避免死循环。

for循环可以直接遍历类型里面的值，而while循环则需要通过下标或内置的方法才能获取到里面的值

如果需要获取类型里面的值，大部分情况下推荐使用for循环的。

需要注意break和continue的区别，以及循环嵌套的时候他们作用范围是只针对当前所在的循环