# Python基础

## 数据类型和变量

### 转义字符 ’\’ ：

\n 换行；\t 制表符；\\ 字符\；

r’ ‘ 表示‘’内部字符不转义；‘’’…’’’ 表示多行内容；

### 空值

None

### 变量

变量本身类型不固定的语言称之为动态语言，与之对应的是静态语言。

### 除法运算

/ 除法计算结果是浮点数，即使是两个整数恰好整除，结果也是浮点数；

// 除法只取结果的整数部分；

% 余数运算。

无论整数做 // 除法还是取余数，结果永远是整数，所以，整数运算结果永远是精确的。

## 字符串和编码

### 字符编码

ord()函数获取字符的整数表示，chr()函数把编码转换为对应的字符

以Unicode表示的str通过encode()方法可以编码为指定的bytes

把bytes变为str，就需要用decode()方法，如果bytes中只有一小部分无效的字节，可以传入errors='ignore'忽略错误的字节：

# -\*- coding: utf-8 -\*- 为了告诉Python解释器，按照UTF-8编码读取源代码，否则，你在源代码中写的中文输出可能会有乱码。

| **占位符** | **替换内容** |
| --- | --- |
| %d | 整数 |
| %f | 浮点数 |
| %s | 字符串 |
| %x | 十六进制整数 |

## 3. 使用list和tuple

### 1. list

往list中追加元素到末尾：classmates.append(' ')

把元素插入到指定的位置，比如索引号为1的位置：classmates.insert(1, 'Jack')

删除list末尾的元素：classmates.pop()

要删除指定位置的元素，用pop(i)方法：classmates[1] = 'Sarah'

### Tuple

tuple和list非常类似，但是tuple一旦初始化就不能修改。因为tuple不可变，所以代码更安全。如果可能，能用tuple代替list就尽量用tuple。

classmates = ('Michael', 'Bob', 'Tracy')

只有1个元素的tuple定义时必须加一个逗号,，来消除歧义：t = (1,)

## 4. 条件判断

If语句注意不要少写了冒号:

if语句的完整形式就是：

if <条件判断1>:

<执行1>

elif <条件判断2>:

<执行2>

elif <条件判断3>:

<执行3>

else:

<执行4>

## 循环

1. for x in ...循环就是把每个元素代入变量x，然后执行缩进块的语句

range()函数，可以生成一个整数序列，比如range(5)生成的序列是从0开始小于5的整数：

1. while循环
2. break 退出循环
3. continue 跳过当前循环，开始下一循环。

## 6. Dict和set

1. dict

>>> d = {'Michael': 95, 'Bob': 75, 'Tracy': 85}

>>> d['Michael']

95

要删除一个key，用pop(key)方法，对应的value也会从dict中删除：

>>> d.pop('Bob')

75

>>> d

{'Michael': 95, 'Tracy': 85}

请务必注意，dict内部存放的顺序和key放入的顺序是没有关系的。

和list比较，dict有以下几个特点：

1.查找和插入的速度极快，不会随着key的增加而变慢；

2.需要占用大量的内存，内存浪费多。

而list相反：

1.查找和插入的时间随着元素的增加而增加；

2.占用空间小，浪费内存很少。

所以，dict是用空间来换取时间的一种方法。

**要牢记的第一条就是dict的key必须是不可变对象。**

2. set

通过add(key)方法可以添加元素到set中，可以重复添加，但不会有效果：

>>> s.add(4)

>>> s

{1, 2, 3, 4}

>>> s.add(4)

>>> s

{1, 2, 3, 4}

通过remove(key)方法可以删除元素：

>>> s.remove(4)

>>> s

{1, 2, 3}

set和dict的唯一区别仅在于没有存储对应的value，但是，set的原理和dict一样，所以，同样不可以放入可变对象，

# 2. 函数

## 1. 调用函数