

# Python

- Python
  - 注释
  - 数据类型
    - 1. Number （数值）
    - 2. String （字符串）
      - 编码
      - 操作
      - 常用函数
    - 3. List （列表）
      - 基本使用
      - 内置函数
      - 列表操作
      - 列表函数
    - 4. Tuple （元组）
    - 5. Set （集合）
    - 6. Dictionary （字典）
  - 运算符和算数运算
    - 运算符
      - 算数运算符
      - 位运算符
      - 赋值运算符
      - 比较运算符
      - 逻辑运算符
      - 其它运算符

## 注释

```
# 单行注释
# 单独的字符串也可充当注释
```

```
'''
    单引号行注释
'''
```

```
"""
    双引号多行注释
"""
```

## 数据类型

### 1. Number（数值）

各种数，支持 4 种数据类型： `int` `float` `bool` `complex`

- 整数 `int`
  - 二进制 `0b` 开头
  - 八进制 `0o` 开头
  - 十六进制 `0x` 开头
  - 无前缀是十进制
- 小数 `float`

```
>>> 15.707
15.707
>>> -3.14
-3.17
```

- 布尔值 `bool`

```
>>> True
True
>>> False
False
```

- 复数 `complex`

```
>>> 39j
39j
>>> -3.14e-159j
(-0-3.14e-15j)
```

### 2. String（字符串）

编码

ord() 转编码; chr() 转字符

```
>>> ord('羊')
32650
>>> chr(32650)
羊
```

操作

截取、拼接和重复

- 截取

```
>>> item = 'hihihi'
>>> item[0:2]
'hi'
>>> item[1:3]
'ih'
>>> item[2:]
'hihi'
>>> item[:]
'hihihi'
```

- 拼接

```
>>> item += ',hello!'
>>> item
'hihihi,hello'
```

- 重复

```
>>> item = 'hi'
>>> item *= 3 # item = item * 3
>>> item
'hihihi'
```

常用函数

函数名	说明
len()	返回长度
capitalize()	字符串首字母大写
center(width, fillchar=' ')	填补居中
count(str, beg=0, end=len(string))	计数

函数名	说明
<code>find(str, beg=0, end=len(string))</code>	找，返回起始下标
<code>isdigit()</code>	是否全数字
<code>isalpha()</code>	是否全字母
<code>replace(old, new[, max])</code>	替换，最多替换 <b>max</b> 个
<code>strip()</code>	去除两端空格
<code>title()</code>	标题化

### 1. len()

```
>>> item = '例子'
>>> len(item)
2
```

### 2. capitalize()

```
>>> 'hello'.capitalize()
'Hello'
```

### 3. center(width, fillchar=' ')

```
>>> 'name'.center(20, '*')
'*****name*****'
>>> 'name'.center(20)
'          name          '
```

### 4. count(str, beg=0, end=len(string))

```
>>> 'hihihi'.count('h')
3
>>> 'hihihi'.count('hi')
3
>>> 'hihihi'.count('hi', 0, 3)
1
```

### 5. find(str, beg=0, end=len(string))

```
>>> '123'.find('1')
0
>>> '123'.find('2')
1
```

## 6. isdigit()

```
>>> '123'.isdigit()
True
>>> '123j'.isdigit()
False
>>> '0123'.isdigit()
True
```

## 7. isalpha()

```
>>> '123'.isalpha()
False
>>> '123a'.isalpha()
False
>>> 'ab1'.isalpha()
False
```

## 8. replace(old, new[, max])

```
>>> 'hi'.replace('hi', 'hello')
'hello'
>>> item = 'hi'
>>> item.replace('hi', 'hello')
'hello'
>>> item                # item 保持不变
'hi'
```

## 9. strip()

```
>>> '   去除两端空格   '.strip()
'去除两端空格'
>>> item = '   去除两端空格   '
>>> item.strip()
'去除两端空格'
>>> item
'   去除两端空格   '
```

## 10. title()

```
>>> item = 'this is a title'
>>> item.title()
'This Is A Title'
>>> item
'this is a title'
```

# 3. List （列表）

## 基本使用

```
>>> item0 = 1
>>> item1 = '2'
>>> item2 = '第三'
>>> item3 = [1, 2, 3, 4]
>>> item = [item0, item1, item2, item3]
>>> item
[1, '2', '第三', [1, 2, 3, 4]]
```

内置函数

函数	说明
len(list)	求列表长度
max(list)	求列表中的 max
min(list)	求列表中的 min
list(seq)	将序列转为列表

列表操作

方法	说明
list.append(obj)	在列表末尾添加新元素
list.count(obj)	count 列表中 obj 的个数
list.extend(seq)	在列表末尾追加另一个序列的元素
list.index(obj)	返回列表元素 obj 的下标
list.insert(index, obj)	插入
list.pop(index)	移除第 index 个元素，缺省时移除最后一个元素
list.remove(obj)	列表移除第一个匹配项
list.reverse()	列表反向
list.sort([func])	排序，参数为可选的哦哎嘘函数
list.clear()	清空列表
list.copy()	复制列表

列表函数

4. Tuple （元组）

5. Set （集合）

6. Dictionary （字典）

运算符和算数运算

运算符

算数运算符

+ - \* / // % \*\*

+	-	*	/	//	%	**
加	减	乘	除	整除	取模	幂

位运算符

&		^	~	<<	>>
与	或	异或	反	左移	右移

赋值运算符

= 、 += .....

比较运算符

== != > < >= <=

逻辑运算符

x and y  
x or y  
not x

其它运算符

1. 成员运算符

in not in

2. 身份运算符

id() 获取地址、id 相同则身份相同

is is not

