

数据类型和运算符

1. 类型判断

我们可以使用type来返回数据类型和用isinstance判断数据类型

type用法: type(变量或常量) 返回数据类型名

示例:

```
a = 10
print(type(a))
print(type('明明'))
```

isinstance用法:

isinstance(变量或常量,数据类型名) 如果是指定数据类型返回True, 否则返回False

```
age = 10
name = 'hello'
print(isinstance(age,int))
```

2. 运算符和表达式

在数学上为了计算结果, 我们会写一些式子计算, 例如:

30 + 5

这是数学中的算式, python中为了计算也有类似的式子, 叫表达式。在表达式中30和5称作操作数, +称之为运算符。表达式的目的在于计算结果。表达式构成:

- 最基本的表达式就是一些数值, 变量、函数, 例如: 3, a(变量) print('hello')
- 由基本表达式加上适当的运算符构成复杂的表达式: 3 + a

综合上述, 所谓表达式就是由操作数和运算符组成并符合python语法的式子。要写表达式首先要学习运算符。关于运算符要知道两件事情

- 计算顺序: 是从左向右算还是从右向左算
- 优先级: 比如先乘除后加减

2.1 算术运算符

a = 20

b = 10

运算符	说明	示例
-	负号，取原数的相反数	a = 10 print(-a) #-10
+ - * /	加减乘除，同数学上一样	a + b = 30 a - b = 10 a * b = 200 a / b = 2
%	模运算，即求 a整除以b的 余数	a % 10 = 0
//	整除	a // b = 2
**	对运算符进行指数(幂)计算	a ** b 表示20的10次方

模运算说明

- 用途：判断奇偶、判断是否能够整除、最大公约数、水仙花数

数学函数

函数名	函数的说明	示例
abs	取绝对值	abs(-10)
pow(x,y)	x的y次方	pow(10,2)求10的平方
round(x,[n])	浮点数的4舍5入， n代表保留小数的位数	round(3.456)
max()	求给定参数的最大值	max(21,43,65,75,86,32,3,45)
min()	求给定参数的最小值	min(21,43,65,75,86,32,3,45)
math.ceil()	需要导入import math库 向上取整	math.ceil(18.1) #19
math.floor()	需要导入import math库 向下取整	math.floor(18.1) #18
math.sqrt	需要导入import math库 求平方根	math.sqrt(100)

python3.5.2中文帮助文档 https://yiyibooks.cn/xx/python_352/index.html

随机函数

获取随机数，需要引入random库。

```
import random
```

函数名	函数说明
random.randrange(start,stop,step)	start 指定范围的起始值 包含本身，默认是0； stop 指定范围的结束值 不包含本身； step 步长，默认步长是1。该函数返回一个整数
random.randint(start,end)	返回[start end]之间的一个随机整数，start必须小于end
random.random()	返回一个[0.0,1.0)之间的随机小数

注意

- 优先级： $** > \text{正负号} > // \% * / > + -$
- 从左向右算
- 在幂运算和一元运算符联合计算时，从右向左算，例如： $-1 ** 2 = -1$

2.2 赋值运算符

`a = 5; b = 10; c = 20`

运算符	说明	示例
<code>=</code>	简单赋值, 将b+c的结果赋给a	<code>a = b + c #a=30</code>
<code>+=</code>	<code>a += b</code> 等价于 <code>a = a + b</code>	<code>a = 15</code>
<code>-=</code>	<code>a -= b</code> 等价于 <code>a = a - b</code>	<code>a = -5</code>
<code>*=</code>	<code>a *= b</code> 等价于 <code>a = a * b</code>	<code>a = 50</code>
<code>/=</code>	<code>a /= b</code> 等价于 <code>a = a / b</code>	<code>a = 0.5</code>
<code>%=</code>	<code>a %= b</code> 等价于 <code>a = a % b</code>	<code>a = 5</code>
<code>//=</code>	<code>a //= b</code> 等价于 <code>a = a // b</code>	<code>a = 0</code>
<code>**=</code>	<code>a **= b</code> 等价于 <code>a = a ** b</code>	

注意：

- 赋值运算符左边必须是变量
- 从右向左算
- 没有赋值表达式
- 所有赋值运算符优先级相同，赋值运算符优先级非常低，仅高于成员运算符和身份运算符、逻辑运算符
- 注意在复合赋值中

2.3 关系运算

关系运算就是比较运算，如果表达式成立，返回True，否则返回False。关系运算的结果是布尔值。

运算符	示例	说明
==	a == b	a和b值相等，结果是True，a和b值不相等结果为False
!=	a != b	a不等于b 结果为True， 否则结果为True
>	a > b	a大于b结果为True， 否则为False
>=	a >= b	a大于等于b结果为True， 否则为False
<	a < b	a小于b结果为True， 否则为False
<=	a <= b	a小于等于b结果为True， 否则为False

注意：

- 优先级： 比较运算符优先级相同
- 从左向右算
- 可以这样算： `1 < a < 3` 等价于 `a > 1 and a < 3`