

任务2-1 变量和数据类型

变量和赋值

超市购物

现实生活中，大家去超市买东西的时候，往往都需要一个菜篮子，用来进行存储物品，等到所有的物品都购买完成后，在收银台进行结账即可。



变量和赋值

Python中的**变量用来存储数据**，变量可以理解为去超市购物的菜篮子，其类型和值在赋值的那一刻被初始化。

```
num1 = 100
```

```
num2 = 87
```

```
result = num1 + num2
```

num1和num2变量就好比一个小菜篮子，它们分别存储的数据是100和87。result变量存储的数据是num1和num2这两个“菜篮子”中的数据累计之和。

标识符

举例：水果名称

现实生活中，人们常用一些名称来标记事物，例如，现实生活中**每种水果都有一个名称来标识。**



ORANGE



APPLE



LEMON



KIWI



MELON



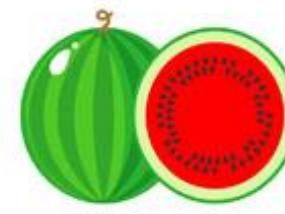
POMEGRANATE



FIGS



PEACH



WATERMELON

标识符

若希望在程序中表示一些事物，需要开发人员自定义一些符号和名称，这些符号和名称叫做**标识符**。

命名规则

- 标识符由字母、下划线和数字组成，且数字不能开头。
- Python中的标识符是区分大小写的。
- python中的标识符不能使用关键字。

关键字

关键字指的是具有特殊功能的标识符。

```
>>> help()           # 进入帮助系统
help> keywords       # 查看所有关键字列表
help> return         # 查看return这个关键字的说明
help> quit           # 退出帮助系统
```

变量的类型

1. 数字类型
 - 整型
 - 复数
2. 布尔类型
 - 布尔类型
3. 字符串类型
4. 列表类型
5. 元组类型
6. 字典类型



过渡页



- 01 基本语法
- 02 变量和数据类型
- 03 标识符和关键字
- 04 简单数值类型**
- 05 运算符
- 06 位运算
- 07 运算符优先级

整型

整数类型 (int) 简称整型, 它用于表示整数。

- 二进制: 0b10100

- 八进制: 010111

- 十进制: 64

- 十六进制: 0x14

整型

Python的整型可以表示的**范围是有限的**，**它和系统的最大整型一致**，例如，32位上的整型是32位，可以表示的数的范围是 $-2^{31} \sim 2^{31}-1$ 。在64位机器上的整型是64位的，可以表示的数的范围是 $-2^{63} \sim 2^{63}-1$ 。

浮点型

- 浮点型 (Float) 用于表示**实数**。
- 浮点型字面值可以用十进制或科学计数法表示。

<实数>E或者e<整数>

E或e表示基是10，后面的整数表示指数，指数的正负使用+或-表示。

浮点型

Python的浮点型遵循的是IEEE754双精度标准，每个浮点数占8个字节，能表示的数的范围是-1.8308~1.8308

布尔类型

- 布尔型其实是整型的子类型。
- 布尔型数据只有两个取值：True和False，分别对应整型的1和0。

布尔类型

– 以下对象的布尔值都是False:

- None
- False (布尔型)
- 0 (整型0)
- 0L (长整型0)
- 0.0 (浮点型0)
- 0.0+0.0j (复数0)
- "" (空字符串)
- [] (空列表)
- () (空元组)
- {} (空字典)

复数类型

复数类型，用于表示数学中的复数，例如， $5+3j$ 。

两大特点

- (1) 复数由实数部分和虚数部分构成，表示为：
 $\text{real}+\text{imag}j$ 或 $\text{real}+\text{imag}J$
- (2) 复数的实数 real 和虚数 imag 都是浮点型

数字类型转换

| 函数 | 说明 |
|-------------------------------------|------------|
| <code>int(x [,base])</code> | 将x转换为一个整数 |
| <code>float(x)</code> | 将x转换到一个浮点数 |
| <code>complex(real [,imag])</code> | 创建一个复数 |