变量、类型与运算符

Python Reference

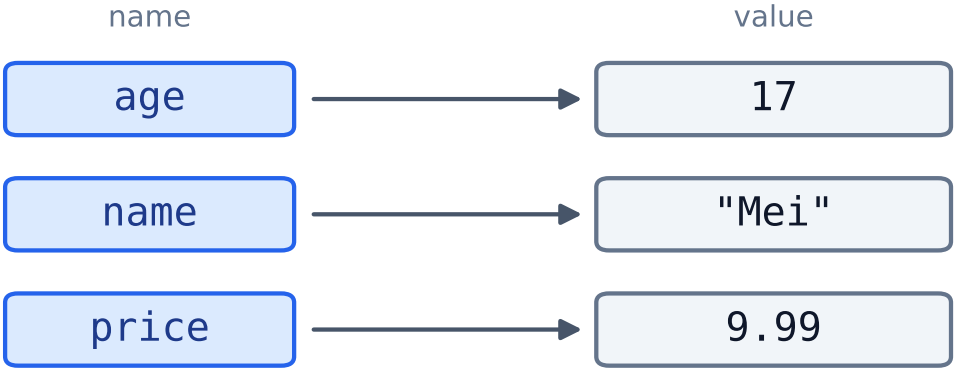
变量与赋值

变量 (variable) 是一个值 (value) 的名字。你用 = 来创建它, 这叫作赋值 (assignment)。名字写在左边, 值写在右边。

```
age = 17
name = "Mei"
price = 9.99
print(age, name, price)
```

现在 age 保存着 17。在任何需要这个值的地方都可以用这个名字, 以后也能改变它:

```
age = 17
age = age + 1   # age is now 18
print(age)
```



variable: name → value in memory

每个变量名都指向内存中的一个值

- = 不表示” 相等”, 它的意思是” 把右边的值存到左边这个名字下”。
- 要判断两个值是否相等, 用 ==(见下文)。

数字: 整数与浮点数

Python 有两种主要的数字类型。整数 (integer, `int`) 是像 `17` 这样的整数。浮点数 (float) 带有小数点, 例如 `9.99`。

下面这些运算符 (operator) 用于数字:

运算符	含义	例子	结果
`+`	加	`3 + 2`	`5`
`-`	减	`3 - 2`	`1`
`*`	乘	`3 * 2`	`6`
`/`	除 (总是浮点数)	`7 / 2`	`3.5`
`//`	整除	`7 // 2`	`3`
`%`	取余 (取模)	`7 % 2`	`1`
`**`	幂	`2 ** 3`	`8`

- `/` 总是得到浮点数, 所以 `4 / 2` 是 `2.0`。
- `//` 和 `%` 常一起用:`17 // 5` 是 `3`,`17 % 5` 是 `2`。

表达式与类型转换

表达式 (expression) 是任何有值的东西, 例如 `3 + 4 * 2`。Python 按通常的数学顺序计算 (`*` 和 `/` 先于 `+` 和 `-`); 加括号能让顺序更清楚。

`input()` 返回字符串, 所以做数学前要先转换。把一个值从一种类型变成另一种, 叫作类型转换 (type conversion):

```
age = int("17")         # text "17" -> number 17
price = float("9.99")   # text -> 9.99
label = str(17)         # number -> text "17"
print(age, price, label)
```

- `int("abc")` 会失败, 所以只转换看起来像数字的文本。
- 混用类型也会失败:`"age: " + 17` 是错误的; 要写成 `"age: " + str(17)`。

布尔值与比较

布尔值 (Boolean) 只有两个取值之一:`True` 或 `False`。一次比较 (comparison) 会返回一个布尔值。

运算符	含义
`==`	等于
`!=`	不等于
`< >`	小于 / 大于
`<= >=`	小于等于 / 大于等于

```
print(7 > 2)         # True
print(3 == 3.0)      # True
age = 20
print(age >= 18)     # True
```

用 and、or、not 连接多个比较:

```
age = 20
print(age >= 13 and age <= 19)   # True only for a teenager
```

常见错误

- / 总是得到浮点数 (float), 连 4 / 2 也是 2.0。想要整数时用 //。
- 把 = 当成 ==:= 是赋值,== 才是判断两个值是否相等。
- 把字符串和数字相加:"age: " + 5 会抛出 TypeError。先用 str(5) 转换。
- 浮点数并不精确,0.1 + 0.2 不正好等于 0.3— 不要对浮点结果用 ==。