

函数

2258次阅读

我们知道圆的面积计算公式为：

$$S = \pi r^2$$

当我们知道半径 r 的值时，就可以根据公式计算出面积。假设我们需要计算3个不同大小的圆的面积：

```
r1 = 12.34
r2 = 9.08
r3 = 73.1
s1 = 3.14 * r1 * r1
s2 = 3.14 * r2 * r2
s3 = 3.14 * r3 * r3
```

当代码出现有规律的重复的时候，你就需要当心了，每次写 $3.14 * x * x$ 不仅很麻烦，而且，如果要把3.14改成3.14159265359的时候，得全部替换。

有了函数，我们就不再每次写 $s = 3.14 * x * x$ ，而是写成更有意义的函数调用 $s = \text{area_of_circle}(x)$ ，而函数`area_of_circle`本身只需要写一次，就可以多次调用。

基本上所有的高级语言都支持函数，Python也不例外。Python不但能非常灵活地定义函数，而且本身内置了很多有用的函数，可以直接调用。

抽象

抽象是数学中非常常见的概念。举个例子：

计算数列的和，比如： $1 + 2 + 3 + \dots + 100$ ，写起来十分不方便，于是数学家发明了求和符号 Σ ，可以把 $1 + 2 + 3 + \dots + 100$ 记作：

100

$$\sum n$$

$n=1$

这种抽象记法非常强大，因为我们看到 Σ 就可以理解成求和，而不是还原成低级的加法运算。

而且，这种抽象记法是可扩展的，比如：

100

$$\sum (n^2+1)$$

$n=1$

还原成加法运算就变成了：

$$(1 \times 1 + 1) + (2 \times 2 + 1) + (3 \times 3 + 1) + \dots + (100 \times 100 + 1)$$

可见，借助抽象，我们才能不关心底层的具体计算过程，而直接在更高的层次上思考问题。

写计算机程序也是一样，函数就是最基本的一种代码抽象的方式。
