

将具有相互关系的数据/操作封装成对象，以对象的角度去处理问题，让对象来完成相应处理

- 面向对象编程 (OOP, Object Oriented Programming)
- 面向过程编程 (Procedure Oriented Programming, POP, 面向过程程序设计)
- 按照业务逻辑 从上到下 设计程序的方式

8.面向对象

下箭头调试程序 (F7 Step Into)，从上往下执行，不利于运行函数

斜右下的箭头 (F8 Step Over)，专门调试函数的。

7.调试

可变量类型，值可以改变：

- 列表 list
- 字典 dict

不可变量类型，值不可以改变：

- 数值类型 int, long, bool, float
- 字符串 str
- 元组 tuple

6.1类型

自动组包：
当把多个值赋值给一个变量的情况下（又不是列表，又不是元组），自动组成一个元组的数据格式。a = 1, 2

自动解包：
当把元组赋值给多个变量，若变量的个数与元组中元素的个数一致，会自动解（拆）包。a, b = (1, 2)

一个变量对应多个值，就会组成元组；
多个变量，对应多个元组，就会拆（解）包

元组的应用场景：
组包和解包，给各位存储值

6.2元组

例：数据交换
解包，分别赋值
a, b = b, a

元组中只包含一个元素 时，需要在元素后面添加逗号
例如：tuple = (50,)

列表可以相加。例如：list1 + list2 + ... + listn
打印被新列表字典中某个键的值 dict[""]

6.3列表

字典是python中最灵活的，可读性最强的一个存储数据的格式

没有索引值
key:value (键:值) 键值对

定义字典，通过键值对，每一条数据用逗号隔开；键的名字只能出现一次

pop() 是带有返回值的，能够返回被删除元素的值
update() 能够添加多个字典，数据存在则覆盖（修改），不存在则添加

6.4字典

dict[key] 根据键取值，键值对不存在会报错

dict.items() 获取所有的键和值，以元组的形式表示，可以通过for循环解包得到每个键和值

字典用得最多的情况，列表嵌套字典，再通过遍历输出每一条某信息

程序是用来处理数据的，变量是用来存储数据的
在 PyCharm 中，可以使用 SHIFT + F6 统一修改变量名

函数内赋值变量时，默认为定义并赋值局部变量，赋值后获取的也是局部变量的值
如果在函数中修改全局变量，那么就需要使用global进行声明，否则出错

定义：函数给调用方提供的结果

一个函数可以有多个return语句，但是只要有一个return语句被执行，那么整个函数就会结束

当返回多个数据时，python会自动将数据 组包 成元组

如果使用多个变量接收返回值，python会自动将元组 拆包 成单个数据

5.1返回值

print()函数自带带有返回数据的功能的

1.能够把函数里的数据带出去

2.会停止程序的运行

return后面不要直接写代码

return

return可以一次返回多个值，或一个值都不返回，返回多个值时，会组包元组

一个函数中可以有多个return，但只会执行其中的一个

调用函数时，如果没有传入默认参数对应的实参，则实参使用默认值。

注意：默认参数一定要位于参数列表的最后面。

调用函数时，实参可以指定对应的形参

用于接收任意数量的参数

前边需要添加*，用于提示python解释器该参数为可变参数

直接用args即可（不需要加*）

本质是 将传递的参数包装成了元组

可以接收不存在的关键字参数

需要在变量名前添加两个*

将 不存在的关键字参数包装成字典

带关键字的元组 -> 字典

不同函数中可以有同名的形参，但形参最好不要和变量同名

id函数查看引用的是否为同一个内存空间，如果返回相同，说明引用相同

5.2参数

值是索引用来传递的

保存某些功能，或者某段程序代码（块），能够保存任意数据

定义函数时，函数上下空出2行

定义好函数之后，只表示这个函数封装了一段代码而已，如果不主动调用函数，函数是不会主动执行的

在函数内部定义的变量，只能被当前函数所使用

5.3引用

1.增加代码可读性

2.提升开发效率

3.增加复用性

5.4函数的作用

放在函数定义内部的第一行，用多行注释的方法，即用三个双引号，在函数调用处按住ctrl键鼠标放在函数名上就会有提示

5.5函数的文档注释

格式：
lambda [arg1 [,arg2,...,argn]] expression

Lambda函数能接收任何数量的参数但只能返回一个表达式的值

匿名函数不能直接调用print，因为lambda需要一个表达式

5.6匿名函数lambda

python基础

1.命名规范

1. 不能用中文
 2. 不能用特殊符号 ~|@#%&'*()
 3. 不能以数字、下划线开头
 4. 不能与关键字重名 True = 18
- 最好用：纯英文单词
- 小驼峰：出现2个单词以上时使用
- 大驼峰：出现2个单词以上时，面向对象时命名使用

2.while 循环

Ctrl + d 快速复制当前行到下一行

while 语句基本语法

死循环：一个很大的BUG绝对不能出现的，重复一直的去执行

增量：主要是为了解决死循环的

break关键字：终止循环继续运行，配合判断使用

continue作用：跳过这一步循环，直接进入下一步循环

外循环执行一次，内循环会执行多次

tip 一个print()是占一行的

end()的作用：可以改变print结束的换行问题

\t 横向制表符，能够让文本垂直对齐

看代码的运行流程，可以打断点，debug

列表的作用：保存大量的数据

若把pop放在程序中，返回被删除数据的值

函数：直接一个单词带着小括号就能用的 例如：len(列表)

方法：必须前面有个对象，才能使用的 例如：列表.append(数据)

函数、方法、关键字的区别

关键字：例如：del[索引]=数据

写一个登陆系统包含很多功能时：可以利用while True然后利用continue及break组合使用

pass是空语句，用作占位语句

是为了保持程序结构的完整性

无限循环 while True: 语句

for 变量 in 集合:
循环体代码
else:
没有通过 break 退出循环，循环结束后，会执行的代码，只会执行一次

for.....else格式 行一次

for循环只完成遍历的功能，并不能修改或更改变量的值

3.for 循环

切片格式：
字符串[开始索引:结束索引:步长]

random.randint(1, 10) : 表示[1,10]

if 字符串[-1] not in "男女":
判断字符串最后一个是不是男女

"" in 字符串: 用于判断字符串中是否有空格

字符串也是有索引值的

4.字符串

find 找不到返回-1

replace("src","des",4) 4为替换的次数

split("") 拆分，默认以空格为分割字符，拆成列表

string.join(list)
通过string这个分隔符将list表里的所有字符串连接成一个字符串

str[::-1] 将字符串倒序，反转，如果步长为负数，则从右开始往左截取切片

切片
最右边的索引值是 -1，依次递减