变量命名：驼峰命名法，单词间用字母大小写分隔；

下划线命名法，字母全部小写，不同单词用下划线分隔。

字符串str：“6”，整数int：6，浮点数float：6.0

布尔类型bool：true、false，空值类型NoneType：None

+-\*/加减乘除，乘方是\*\*，如2\*\*3是2的3次方

Ctrl+Z是返回，Ctrl+/是整段注释

一个汉字占用3个字节

Type(“hello”)能返回数据类型

函数名跟括号，括号里面加参数，是标准的调用函数的方式

字符串转整数，里面要确实是整数；浮点数转整数，里面可以不是整数，转换后抹掉小数点后面的

条件语句

If [条件]:

[四个空格]执行语句

执行语句

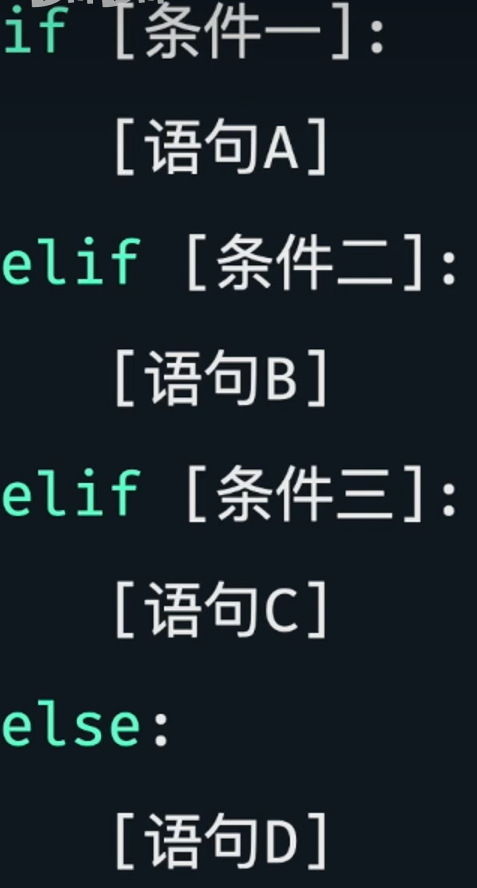
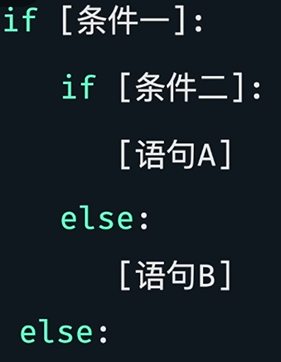
else:

[四个空格]执行语句

执行语句

判断符号：==、!=、＞、＞=、＜、＜=

嵌套条件语句 多条件判断



逻辑运损：and与 or或 not非

shopping\_list = [“键盘”，“鼠标”]

shopping\_list.append(“显示器”)——对象.方法名(…)

len(shopping\_list)——函数名(对象)

append() 方法用于向列表末尾添加一个元素，但它不返回任何值（即返回 None），所以还是应该print(shopping\_list)。列表可以放很多类型的数据。访问列表中的元素用[]。

列表可变，可增加或减少直接改变，其他数值类型不可变。

字典contacts = {}、contacts={“小明”:”18372982637”,”小花”:”18263637283”}。键：值，键值对之间用逗号分隔开。键的类型必须是不可变的，列表不能作为键。于是有了元组，元组很像列表，但是不可变，用圆括号。要引用键就用contacts[“小明”]

循环格式：for 变量名 in 可迭代对象：

#对每个对象做的事情

temperature\_dict = {“111”:”36.2”,”112”:”36.5”,”113”:”37.0”}

temperature\_dict.keys() #所有键

temperature\_dict.values() #所有值

temperature\_dict.items() #所有键值对

range(5,10,1):括号里是起始值和结束值和步长，即5，6，7，8，9.

While 条件A：

行动B

在条件何时结束未知的情况下，while循环比for循环更适合使用。

For循环适用于：有明确循环对象或次数的时候更方便

While循环适用于：循环次数未知。

format方法：

format(year,name)后，{0}代表year里的，{1}代表name里的。

format(year=year,name=name)中，前面的year是关键字，对应花括号里面的关键字，后面的year是参数值。

f-字符串也行，就是year=“虎”，name=“老林”, f”””…{year}…{name}…”””也可以直接替换。

def calculate\_sector\_1(central\_angle, radius):#def是关键词表示开始定义函数,calculate\_sector\_1是函数名，(central\_angle, radius)里面是参数。

return a 可以让我们在函数外继续使用这些变量，没有return语句，会默认返回return None。

引入模块

1. import语句

import statistics

print(statistics.median([1,2,1,3,4,2]))

print(statistics.mean([1,2,1,3,4,2])) 模块名+函数名/变量名

1. from…import…语句

from statistics import median,mean

print(median([1,2,1,3,4,2]))

print(mean([1,2,1,3,4,2]))

1. from…import\*

会类似2，单引入这个模块里所有函数/变量，容易和自己别的命名搞混，不建议经常使用

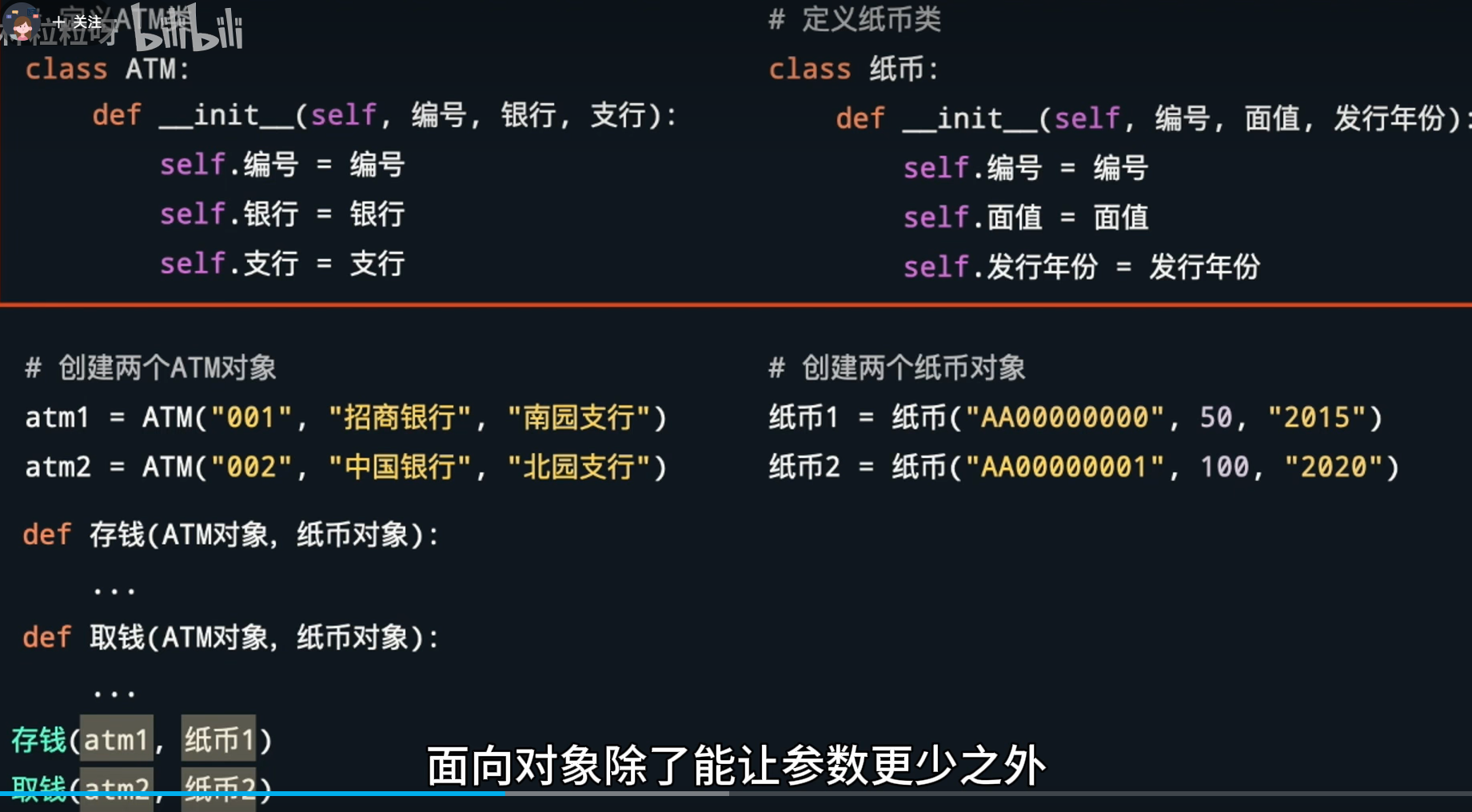
Window系统 按住control+点击函数名查看引入的函数具体是怎么写的

安装语句pip install（第三方库），引入语句import

pypi.org这个网站可以对第三方库进行搜索

面向对象编程（object oriented programming）——还有面对过程编程

对象可以绑定属性和方法



类是创建对象的模板，对象是类的实例。

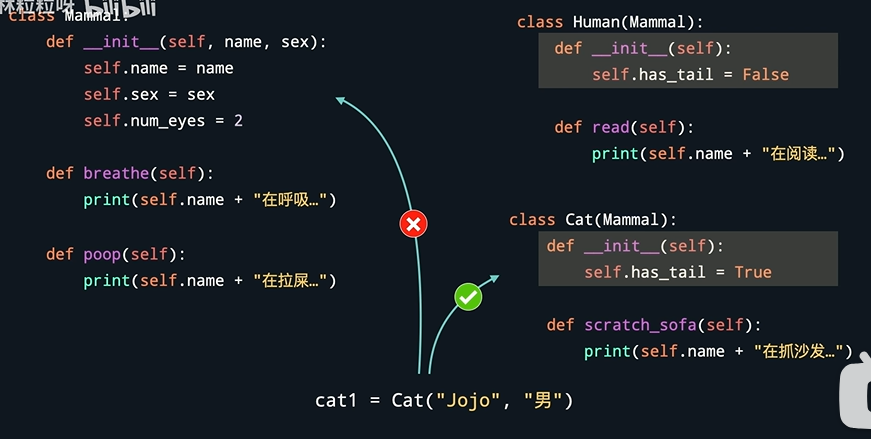
变量名：user\_name

类名：UserAccount

类有一个特殊的方法叫构造函数，主要作用是定义实例对象的属性，被命名为\_\_init\_\_。调用时类名(参数)就可以返回一个对象。要获取对象的属性可以用对象.属性名来获取。如print(cat1.name)。

定义属性用\_init\_(self,…,…): 定义方法用比如speak(self,…,…):

优先看子类有没有该方法，没有的话看同名的父类有没有定义该方法。



class human(mammal)

读文件：

f = open(“./data.txt” , ”r” , encoding = “uft-8”)#“r”是读取，”w”是写入

print(f.read(10))#会读第1-10个字节的文件内容# print(f.read())是读全部

print(f.read(10))#会读第11-20个字节的文件内容

/或者

f = open(“./data.txt” , ”r” , encoding = “uft-8”)

print(f.readline())#会读第一行的文件内容

print(f.readline())#会读第二行的文件内容

#循环读取法

line = f.readline()  
while line !="":  
 print(line)  
 line =f.readline()

/或者

lines = f.readlines() #把每行内容储存到列表里

for line in lines:

print(line)

写文件：

with open(“./data.txt” , ”a” , encoding = “uft-8”) as f:

f.write(“Hello!\n”)

f.write(“Yoooo!”)#”a”是在原文件后继续写的意思，”w”是删空重写

读+写文件：

with open(“./data.txt” , ”r+” , encoding = “uft-8”) as f:

print(f.read())

f.write(“hello!”)#会直接在读取后追加

捕捉异常：

