[选取日期]

|  |
| --- |
| Python学习笔记 | Wang Jingqi |



|  |  |
| --- | --- |
| MyOwn | Python学习心得 |

[1. 基础 3](#_Toc99051204)

[1.1. 缩进 3](#_Toc99051205)

[1.2. 数据结构 3](#_Toc99051206)

[1.3. 函数 4](#_Toc99051207)

[1.4. 控制流 4](#_Toc99051208)

[1.5. 模块 7](#_Toc99051209)

[2. 应用 9](#_Toc99051210)

[2.1. Time 9](#_Toc99051211)

[2.2. Pip 9](#_Toc99051212)

[2.3. Class 9](#_Toc99051213)

[2.4. Schedule定时方法 10](#_Toc99051214)

[2.5. 收发邮件 10](#_Toc99051215)

[2.6. 分段计税 12](#_Toc99051216)

[2.7. Excel应用 13](#_Toc99051217)

# 基础

## 缩进

放置在一起的语句必须拥有相同的缩进:block

Python始终对block使用 缩进

使用 4个空格来缩进

## 数据结构

### 列表list:保存一系列有序项目的集合，一张清单;表达形式：[]

shoplist=['apple','mango','carrot','banana']

print("i have ",len(shoplist),'items to buy')

print('these items are:',end=' ')

for item in shoplist:

print(item,end=' ')

shoplist.append('rice')

print(shoplist)

shoplist.sort()

print(shoplist)

olditem=shoplist[0]

del shoplist[0]

print("item to del:",olditem)

print(shoplist)

print(new\_zoo[0])

print(new\_zoo[2][2])

### 元组tuple:将多个对象保存在一起，近似看作列表，类似于字符串，不可改变。表达形式：()

zoo=('python','elephant','penguin')

print(len(zoo))

print(zoo)

new\_zoo='mokey','camel','mule',zoo

print(len(new\_zoo))

print(new\_zoo)

### 字典dictionary:类似地址簿，有keys（唯一）和values（不可改变的对象）联系在一起。表达形式：{key1:value1,key2:value2};不可被排序。

ab={'sw':'sw@126.com','dk':'dk@126.com','ab':'ab@22.com','cd':'cd@11.com'}

print(ab)

print(ab['sw'])

del ab['cd']

print(ab)

for name,address in ab.items():

print("contact {} at {}".format(name,address))

ab['gb']='gb@12.com'

if 'gb' in ab:

print('gb address is:',ab['gb'])

### 集合set

## 函数

### Format()

按照优先顺序替换：print(“{0} is {1}”.format(“jack”,”mark”))

### Print()

打印

### 运算符

+/-/\*//

\*\*：乘方

//：整除

%：取余数

<<:左移 >>:右移

&：按位与 |:按位或

## 控制流

If <condition>:

<Block1>

elif <contition1>:

<block2>

Else:

<block3>

### While

number=23

running=True

while running:

guess=int(input("enter an integer:"))

if guess==number:

print("cong!you got it")

running=False

elif guess<number:

print("a little lower: ")

else:

print("a little higher: ")

else:

print("the while loop is over")

### For:

For x in range(1,5): C++: for (x=0;x<5;x++)

Print(x)

------:1,2,3,4

For x in list(range(5)):

Print(x)

---:0,1,2,3,4

### Break

while True:

s=input("enter sth: ")

if s=='q':

break

print("length of the str is ",len(s))

print("q is input and done")

### Continue: 跳过当前循环块中的剩余语句，并继续该循环的下一次迭 代。

while True:

s=input("enter sth: ")

if s=='q':

break

if len(s)<3:

print("too small")

continue

print("length of the str is ",len(s))

print("q is input and done")

### Def:定义函数

函数定义格式

def function():

print("function1() is printed")

function()

#-----------------

def Max\_ab(a,b):

if a>b:

print('a>b')

elif a==b:

print('a==b')

else:

print("a<b")

Max\_ab(1,3)

函数可使用关键字参数

def function(a,b=5,c=10):

print("a: ",a," b:",b," c:",c)

function(1,3,5)

function(1,c=24)

function(c=30,a=100)

函数可使用可变参数

def total(a=5,\*numbers,\*\*phonebook):

print('a',a)

for single\_item in numbers:

print('single\_item',single\_item)

for first\_part,second\_part in phonebook.items():

print(first\_part,":",second\_part)

print(total(10,1,2,3,4,5,6,7,8,jack=1123,john=2231,inge=1560))

变量的范围

函数体内的变量位局部变量

若需将函数体内的变量声明为全局变量，需在变量前声明:

Global x

DocStrings:文档字符串

def max(x,y):

'''print the maximum of the two numbers.

or other texts'''

if x>y:

return x

elif x==y:

return x

else:

return y

print("max one is ",max(20,3))

print(max.\_\_doc\_\_)

## 模块

编写一个以.py为扩展名的文件，可以重复定义函数

引用已被编译的C语言模块

import sys

print("the commmand line arguments are:")

for i in sys.argv:

print(i)

print("\n\n the pythonpath is:",sys.path,"\n")

### From…import语句

引用一个已被编写保存的.py文档中的函数

Module\_demo.py:

def say\_hi():

print("say hi:this say\_hi() module was referred")

\_\_version\_\_='0.1'

import module\_demo

module\_demo.say\_hi()

print(module\_demo.\_\_version\_\_)

### 包packages

# 应用

## Time

## Pip

Pip镜像地址：

pip install -i https://pypi.tuna.tsinghua.edu.cn/simple module\_name/#例如： Django，wxpython

pip install -i https://pypi.tuna.tsinghua.edu.cn/simple some-package

pip install -i http://pypi.douban.com/simple --trusted-host pypi.douban.com packagename # packagename是要下载的包的名字

pip install -i http://e.pypi.python.org --trusted-host e.pypi.python.org --upgrade pip # 升级pip

### 查看可升级的包

pip list -o

升级脚本:

Pip install somepackage –U # somepakage: package name

安装特定版本：

Pip install somepackage==5.15.4 #安装5.15.4版本的somepackage

## Class

### 创建一个类，默认创建了一个与类同名的方法method,有额外的self变量。

class Person:

def sayhi(self):

print("hello,hi")

p=Person()

p.sayhi()

### \_\_init\_\_方法：在类的对象被实例化instantiated时立即运行。

## Schedule定时方法

import schedule

import time

def morning():

print("good morning")

def noon():

print("good noon")

def evening():

print("good evening")

def task1():

print("task1 was initiated")

def task2():

print("task2")

def task3():

print("task3")

evening()

schedule.every().day.at("23:18").do(task1)

schedule.every().day.at("23:19").do(task2)

schedule.every().day.at("23:20").do(task3)

schedule.every(1).minutes.do(morning)

schedule.every(30).seconds.do(noon)

while True:

schedule.run\_pending()

time.sleep(1)

## 收发邮件

POP3服务器: pop.126.com

SMTP服务器: smtp.126.com

IMAP服务器: imap.126.com

import yagmail

yagmail.register("dake124@126.com","Wdke0101")

yag=yagmail.SMTP(user="dake124@126.com",host="smtp.126.com")

contents=["the 1st test mail"]

#yag.send("dake124@126.com","new 1st test mail",contents)

contents\_01=[

'邮件炸弹搞死你\n'

'炸死你个破邮箱\n'

'<a href="https://www.baidu.com">点击就要命</a>'

'd:\\1.rar'

]

import schedule

import time

def sendmails():

#yag.send("mailbox","subject",content)

yag.send("dake1248@126.com", "mail bombs", contents\_01)

print("send a mail fo fish every 5 seconds")

schedule.every(5).seconds.do(sendmails)

while True:

schedule.run\_pending()

time.sleep(1)

'''

from imbox import Imbox

import keyring

pwd=keyring.get\_password("yagmail", "dake124@126.com")

with Imbox("imap.126.com", "dake124@126.com", pwd, ssl=True) as imbox:

all\_inbox\_messages=imbox.messages()

for uid, message in all\_inbox\_messages:

print(message.subject)

#print(message.body["plain"])

'''

## 分段计税

def tax(income):

if income>10000:

pay=3000\*0.005+2000\*0.01+5000\*0.015+(income-10000)\*0.02

return pay

else:

if income>5000:

pay=3000\*0.005+2000\*0.01+(income-5000)\*0.015

return pay

else:

if income>3000:

pay=3000\*0.005+(income-3000)\*0.01

return pay

else:

pay=income\*0.005

return pay

while True:

inc=float(input("input your income:"))

print("your pay("+str(inc)+") for tax is : "+str(tax(inc)))

## Excel应用

### Openpyxl

### Openpyxl应用

from openpyxl import load\_workbook

workbook=load\_workbook(filename="test.xlsx")

print(workbook.sheetnames)#获取sheet名称

sheet=workbook["Sheet1"]

print(sheet)

print(sheet.dimensions)#获取存在内容的表格的范围:A1:H11

cell1=sheet["A1"]

cell2=sheet["C11"]

print(cell1.value,cell2.value)

"""

获取A1/C11的数据

"""

cell3=sheet.cell(row=1,column=1)

cell4=sheet.cell(row=11,column=3)

print(cell3.value,cell4.value)#打印数值

print(cell3,cell4)#打印cell属性（cell 'sheet1'.Al...)

sheet=workbook.active

cell0=sheet["A1"]

cell1=sheet["C11"]

print(cell0.value,cell0.row,cell0.column,cell0.coordinate)#:Name 1 1 A1

print(cell1.value,cell1.row,cell1.column,cell1.coordinate)#:42 11 3 C11

cell=sheet["A:C"]

print(cell)

for i in cell:

for j in i:

print(j.value)#:打印A:C的所有内容，先打印A1：An,再B1:Bn,再C1：Cn

#:按行打印数据

for i in sheet.iter\_rows(min\_row=2,max\_row=5,min\_col=1,max\_col=2):

for j in i:

print(j.value)

#:按列打印数值

for i in sheet.iter\_cols(min\_row=2,max\_row=5,min\_col=1,max\_col=2):

for j in i:

print(j.value)

#:获取所有行

for i in sheet.rows:

print(i)

#获取所有列

for j in sheet.columns:

print(j)

### Openpyxl应用

from openpyxl import load\_workbook

workbook=load\_workbook(filename="test1.xlsx")

sheet=workbook.active

#写入某个cell:

sheet["H11"]="hello111"

sheet["D1"]="height"

workbook.save(filename="test1.xlsx")#:修改H11内容并将文件另存在test1.xlsx

cell0=sheet["H11"]

print(cell0.value)

print(sheet["D1"].value)

#向表格中插入行数据

"""

sheet=workbook.active

data=[

["唐僧","男","180"],

["孙悟空","猴","199"],

["猪八戒","猪","200"],

["沙僧","男","180”],

]

for row1 in data:

sheet.append(row1)

workbook.save(filename="test2.xlsx")

"""

#sheet=workbook.active

sheet["E1"]="身高"

workbook.save(filename="test1.xlsx")#:修改H11内容并将文件另存在test1.xlsx

"""

for i in range(2,16):

sheet["D{}".format("i")]=2

#sheet["D

"""

#插入空白行/列，位置由idx指定，数量由amount指定

sheet.insert\_cols(idx=2,amount=2)

sheet.insert\_rows(idx=2,amount=4)

workbook.save(filename="test1.xlsx")#:修改H11内容并将文件另存在test1.xlsx

sheet.delete\_cols(idx=1)

sheet.delete\_rows(idx=1)

workbook.save(filename="test1.xlsx")

### Openpyxl应用

from openpyxl import load\_workbook

workbook=load\_workbook(filename="test1.xlsx")

sheet=workbook.active

#移动格子

sheet.move\_range("C1:D4",rows=2,cols=-1)

#建立新sheet

workbook.create\_sheet("new sheet")

workbook.save(filename="test1.xlsx")

#删除sheet

#workbook.remove(sheet)

workbook.save(filename="test1.xlsx")

#修改sheet名称

sheet.title="new sheet name"

workbook.save(filename="test1.xlsx")

### Openpyxl应用

from openpyxl import Workbook

#新建excel文件

workbook=Workbook()

sheet=workbook.active

sheet.title="new sheet"

workbook.save(filename="D:\Python\Python\_programs\new excel")