Python教程

# Python语言的学习（上）

**1、基础知识**

数据结构：常量、变量、元组（列表）、字典

数据类型：int float str

算法：分支和循环

面向对象：类、属性、方法

封装、继承、多态

**2、Python语言的特点：**

例子：定义三个不同类型的变量，赋值，输出

无头文件、无主函数、无大括号、无分号，简洁，靠缩进来区分语句，且变量的定义不需要书写类型，直接赋值，Python自动识别数据类型，字符串赋值时可用双引号或单引号，两者区别：双引号可用于打印对象中含有单引号时

注释：单行注释---#，多行注释---’’’(一对缩进相同)

注意：不要用一些关键字命名：selenium

**3、输入输出函数：**

输入函数：input

例子：a=input(“提示信息：”)、

区别：input函数自动识别数据类型，

多个数据同时输入：a,b = input('请输入：').split()

输入：happy hello

注：input函数输入的所有数据都为str类型

数据类型查看函数：type(变量名)

例子：print type(a)

输出函数：print('结果为：',a,b)

例子：print ("%d,%f,%s")%(a,b,c) –字符串采用%s

**4、元组/列表/序列**

**定义：**

s=(1,2.5,'much')----tuple

s=[1,2.5,'much']----list

#tuple类型的元组内的值不可改变，而list类型的元组可以改变

与数组不同的地方：1.括号；2.元素类型

**引用：**

#引用方式：

#1整体应用---元组名

例子：print(s)

#2下标引用---下标从0开始

例子：print(s[2])

#3下限、上限和步长

例子：print(s[1:7:2])

---下限表示从谁s[1]打印，上限表示打印到谁s[7]但是该数默认不打印，步长是下一元素的增量，步长为1可省略，

#4 特殊引用

例子：print(s[-2])、print(s[1::2])

练习：定义一个1-9的元组，1、输出倒数第3个元素；2、输出值468

思考：如何定义1-999的元组？

range(n):产生一个0到n-1的列表

range(m,n)：产生一个m到n-1的列表

range(m,n,i)：产生一个m到n-1，步长为i的列表

输出：list(range(1,1000))

**操作方法：**

* + - 1. 逆序存放：s.reverse()---改变原来元组的值
      2. 排序存放：s.sort()------改变原来元组的值
      3. 排序：sorted(s)---------不改变原来元组的值，只返回一个排序结果
      4. 插入：s.insert(n,m)-----在某一位置(s[n]前面)插入该值m
      5. 追加：s.append(n)-------在该元组末尾追加n
      6. 连接两个列表：m.extend(n) ---将m和n两个列表连接
      7. 删除指定元素：m.pop(n) ------删除m[n]并返回该值
      8. 删除指定元素第一次出现的值：m.remove(n) –将第一次出现的元素n删除
      9. 返回该值在列表中出现的次数：m.count(n) –返回元素n在列表中出现的次数
      10. 返回列表中元素最大值：max(s)
      11. 返回列表中元素最小值：min(s)
      12. 长度：len(s) ---返回列表元素的个数
      13. 删除：del s[n] ----删除s[n]，n为下标
      14. 获得元素的下标：s.index(n) -------得到元素n在列表中的下标
      15. 清空列表：s.clear() --------清空列表
      16. in 和 not in ---用来检查是否在列表中

练习：列表[11,13,5,7,0,56,23,34,72]，

求该列表中的最大值，最小值及列表中一共有几个元素

获取56元素在列表的位置

追加元素7

删除元素0

排序列表（由大到小）

将列表[66,67,68]与原列表组合

集合

定义：s = {3,1,2}，集合是无序的，且不重复

引用：不能通过下标引用

操作：

添加-s.add(n)---向集合s添加元素n

删除-s.remove(n) ----删除集合中的元素n，如果元素不存在，报Keyerror异常

删除-s.discard(n) ----删除集合中的元素n

清空-s.clear() ------清空集合

in 和 not in ---用来检查是否在集合中

**5、字典**

**定义：**#大括号，键值对形式存在{‘key’:value}

s={'a':10,'b':20,'c':15}

#字典内的元素没有顺序，不能够通过下标引用

**引用：**只能通过键来引用或整体引用

print(s['c'])；print(s)

**赋值：**当key不存在时即可为字典插入，但是如果存在，即为修改

s['d']=11

**操作方法：**

1、取出所有的键：s.keys()

2、取出所有的值：s.values()

3、取出所有的键值对，作为一个元组内的元素：s.items()

4、清空字典：s.clear()

5、删除：del s[‘key’]

**6、分支**

|  |  |  |  |
| --- | --- | --- | --- |
| 分类 | If-分支 | If-else分支 | If-elif嵌套 |
| 语法 | If 条件：  语句体 | If 条件**：**  语句体1  else**：**  语句体2 | If 条件1：  语句体1  elif 条件2：  语句体2  elif 条件3：  语句体3  else：  语句体 |
| 例子 | a=0  if a>0:  a+=1  print a | a=input()  if a>0:  a+=1  else:  a-=1  print a | a=input()  if a>0:  a+=1  elif a==0:  a-=1  elif a<0:  a+=2  print a |

例子：

输入一个数，判断该数的等级：90-100：等价为A……60以下：等级为E

注：Python不支持&&、||，需要and、or代替，同时也支持多个条件连接：0<a<10

预习：

面向对象的类、属性、方法、实例化、

作业：输出100以内的所有质数（只能被1和其本身整除的数）

1. **Python语言学习下**

**1、循环**

循环四要素：

1. 定义循环变量；2、循环条件；3、循环体；4、循环变量发生变化

n = 0; n<5; print n; n+=1

把程序做成循环的步骤：

A、确定循环体，全部选中用Tab键内推，用循环结构包裹

B、从上到下依次编写：1/2/4

**a、while循环**

**语法：**

循环变量

while 条件**：**

循环体

循环变量发生变化

**例子：**

i=0

while i<5:

print i

i=i+1

练习：计算1+2+3+4……+100的和

**b、for循环**

**语法：**

for 变量 in 序列:

循环体

**例子：**

for i in [1,2,1,4]:

print i,'ok'

range（n）：产生一个0到n-1的序列

range(m,n)：产生一个m到n-1的序列

range(m,n,i)产生一个m到n-1且步长为i的序列

**c、continue和break**

continue：停止当前这次循环，继续下次循环

break：跳出所有循环

**例子：**

for i in range(1,5):

if i==3:

break/continue

print i

else:

print i

break：跳出最近这层循环

continue：停止当前这次循环，继续判断下次循环是否成立

练习：

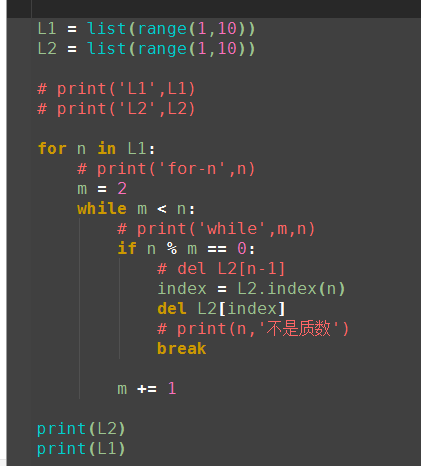
求10阶乘

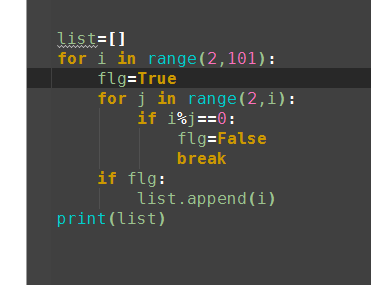
求100以内能被3整除的数，并将作为列表输出

列表[1,2,3,4,3,4,2,5,5,8,9,7],将此列表去重，得到一个唯一元素的列表

求斐波那契数列 1 2 3 5 8 13 ……

求10000以内的质数





**2、函数**

**a、定义：**

def 函数名(参数1，参数2）：----（参数可有可无）

语句体

return 变量----（return可有可无）

**b、调用:**

函数名

变量=函数名(参数1，参数2)----（变量，参数可有可无）

1、调用；2、传参；3、接收返回值

**c、例子：**

def add(a,b):

print a+b

a,b=input('请输入两个数，并用逗号隔开：')

add(a,b)

**练习：**

设计一个计算器，输入两个数，自动实现加减乘除（进阶：根据用户输入的计算符号计算结果）

**3、面向对象的类、属性、方法**

1. **基础：**

类：具有相同的特性，且能够完成某些动作的事物组成的一个集合

属性：类中的事物所具有的特性，趋于静态

方法：类中的事物所能够完成的动作或功能，趋于动态

实例/对象：类中某一个具体的对象或实例

实例化：定义类中某一个对象的过程

1. **类的定义：简单类的创建1🡪2🡪3**

用法：

class 类名（object）：

属性

方法

注意：属性和方法可以为空，里面直接写pass即可，表示是一个空类

例子：

class Person(object):

sex = ‘’

height = ‘’

age = ‘’

def takefood(self,food):

print food

print '小明吃',food

注：dir(类名)，可查看类中的方法

1. **类的实例化：**

变量=类名('xx') ---是否需要传参就看\_\_init\_\_方法中除了self是否还有其他参数

1. **类中方法的调用：**

实例名.方法名('xx') ---是否需要传参就看该方法中除了self是否还有其他参数

**练习：**

a、定义一个学生类：Student、内部含有一个方法：study，实现打印：小明学习xx课程

b、定义一个类名：Student—学生、类内部含有一个属性：sno—学号，一个方法：study—学习，实现打印：学号为xx的学生，学习xx课程

1. **继承**

语法：

例子：

练习：

定义一个Teacher类，继承Person类，拥有自身的属性gh，自身的方法：teach教课（课程）；

1、实现gh为xx的老师，教xx课

2、实现gh为xx老师，在xx上班，一月工资xx

学习Python网站：廖雪峰、 3wschool、 博客（网易、新浪、CSDN）