

day002思维导图

布尔值

一个布尔值只有True, False  
空值,是python中一个特殊的值,用None, None不能理解为0, 因为0是有意义的, 而None是一个特殊值, 0或None都被视为False

列表

创建列表:列表名=[列表选项1, 列表选项2,...列表选项n]  
列表方法  
list3=list1+list2, list4=list3\*3  
列表操作  
print(3 in list5) #判断元素是否在列表中  
print(list10[a:b])#列表截取  
print(list6[a][b])#二维列表  
列表是可以改变的

list1.append() #在列表的末尾添加新的元素  
list2.extend() #在末尾一次性追加另一个列表中的多个值  
list.insert() #在下标处添加一个元素, 不覆盖原数据, 原数据向后顺延  
list.pop(x) #默认删除列表最后一位, x为列表下标, 移除列表中指定下标处的元素  
list.remove(x) #移除列表中的某个元素第一个匹配的结果  
list.clear() #清除列表中所有的数据  
list.index(x) #从列表中找出某个值第一个匹配的索引值  
len(list) #列表中元素个数  
max(list) min(list)#获取列表中的最大值, 最小值  
list.count(x) #查看元素在列表中出现的次数  
list.reverse() #倒叙  
list.sort() #升序排序  
list1=list.copy() #在堆区开辟一块空间  
range() #列表生成器, range([start],[end],[step]) 功能:生成数列  
break语句, continue语句

turtle

是一个简单的绘图工具, import turtle

1.运动命令  
forward(d) 向前运动d距离  
backward(d) 向后运动d距离  
right(a) 向右转向多少度  
left(a) 向左  
goto(x,y) 移动坐标到(x,y)  
speed(speed) 笔画绘制的速度0-10  
up() 笔面抬起, 不会绘图  
down() 笔面放下, 绘图  
setheading(a) 改变海龟的朝向  
pensize(b) 改变笔画的宽度  
pencolor("red") 改变笔画的颜色  
reset() 重置turtle状态  
clear() 清空窗口, 不重置turtle状态  
begin\_fill() 开始填充  
fillcolor(color:str) 填充的颜色  
circle(r,steps=e) 绘制一个圆形, 其中r为半径, e为次数  
done() 程序继续执行  
turtle.done()  
undo() 撤销上一次动作  
hideturtle() 隐藏海龟  
showturtle() 显示海龟  
2.笔画控制命令  
end\_fill() 填充结束  
3.其他命令  
screensize(x,y) 修改屏幕大小

元组

元组创建后就不可变

tuple() print(tuple) #创建空的元组  
tuple(1,2,"god") print(tuple) #创建带有元素的元组, 元组中的元素类型可以不同  
#元组元素的访问 格式: 元组名[下标] print(tuple[0]) print(tuple[-1])  
tuple=(1,2,3,[4,5,6]) tuple[0][1]=100 print(tuple) #可以输出, 可以改变元组中的列表  
del tuple #删除元组  
print(t7+t8) #元组可以相加  
print (t10\*10) #元组重复  
print(4 in t11) #判断元素是否在元组中  
t12[3,7]#元组截取 格式: 元组名[开始下标:结束下标]  
t13=((1,2,3),(4,5,6))#二维元组

元组的方法

len() #返回元组中元素的个数  
max() min() #返回元组中最大值, 最小值  
t15=tuple(list1) #将列表转为元组  
for i in (1,2,3,4) #元组的遍历

字典

概述: 使用键值对存储, 注意:字典是无序的  
特性: 字典中的key必须唯一; key必须是不可变对象; 字符串, 整数等都是不可变的, 可以作为key; list可变, 不能作为key  
与list比较: 查找和插入的速度很快, 不会随着key-value的增加而变慢; 需要占用大量的内存, 内存浪费多  
list: 查找和插入的速度随着数据量增多而减慢; 占用空间小, 浪费内存小。

字典的方法

字典名[key] print(dict1["tracy"])  
dict2["age"]=18 #给字典key添加值, 或者添加key-value  
dict2.pop("tracy") #删除字典中的键值对  
for key in dict1: for value in dict1.values() for key,value in dict1.items():  
dict.get(key,default=None) #获取字典中key的value, 若字典中没有返回None

字符串的一些方法

split("str",num) #以str为分隔符截取字符串, 指定num, 则仅截取num个字符串  
splitleines(keepends) #按照(\r,\n,\n,\n)分割, 返回  
join(seq) #以指定的字符串分隔符, 将seq中的所有元素组合成一个字符串  
max() min() #求最大值, 最小值  
replace(oldstr,newstr,count) #用newstr替换oldstr, 默认是全部替换, 如果指定了count, 那么只替换前count个  
#创建一个字符串映射表  
# 要转换的字符串  
t46 = str.maketrans("ac","65")  
str47="tracy is a good man"  
str48 = str47.translate(t46)  
print(str48)  
stratswith(str,strat=0,end=len(str)) #在给定的范围内, 判断是否是以给定的字符串开头, 如果没有指定范围, 默认整个字符串  
endwith(str,strat=0,end=len(str)) #在给定的范围内, 判断是否是以给定的字符串结尾, 如果没有指定范围, 默认整个字符串  
encode(encoding='utf-8',errors='strict') #编码  
decode()#解码  
isalpha() #如果字符串中至少有一个字符且所有的字符都是字母返回True, 否则返回False  
isalnum() #如果字符串中至少有一个字符且所有的字符都是字母或数字返回True, 否则返回False  
isupper() #如果字符串中至少有一个字符且所有的英文字符都是大写的英文字母返回True, 否则返回False  
islower() #如果字符串中至少有一个英文字符且所有的英文字符都是小写的英文字母返回True, 否则返回False  
istitle() #如果字符串是标题化的返回True, 否则返回False  
isdigit() #如果字符串中只包含数字字符返回True, 否则返回False  
isnumeric() #如果字符串中只包含数字字符返回True, 否则返回False  
isdecimal() #字符串只包含十进制字符  
isspace() #如果字符串中只包含空格则返回True, 否则返回False