

```

1  #200925_布尔表达式
2  #浅拷贝,深拷贝
3  #list1=[10,20,30,[40,50],60]
4  # list2=list1  #普通的赋值,两个列表其实指向的是同一个内存地址
5  # list1[0]=98
6  # print(list1,id(list1))
7  # print(list2,id(list2))
8  #如果不想让其中一个列表影响到另一个列表,怎么办?
9  # list2=list1[:]  #切片等价于浅拷贝,两个列表是不同的列表,但是子列表仍然是指向同一个地址
10 # list1[0]=98
11 # list1[3][0]=68  #切片对于子列表而言,仍然是指向的同一个地址
12 # print(list1,id(list1))
13 # print(list2,id(list2))
14 #深拷贝
15 # import copy
16 # list2=copy.copy(list1)  #浅拷贝,等价于切片[:]
17 # list2=copy.deepcopy(list1)  #深拷贝,确保两个列表互相不会影响对方,包括子列表
18 # list1[3][0]=68
19 # print(list1,id(list1))
20 # print(list2,id(list2))
21
22 # print(2>1)  #True,False称之为布尔型,True或者1表示真,False或者0表示假
23 # print('a'>'b')  #字符之间的比较,比的是ASCII码,'a' 97 'b' 98
24 # print('a'>'A')  # 'A' 65
25 # print('aA'>'Aa')  # 'A' 65 字符串之间的比较,只比较第一位,只有第一位相同时,才会向后比较
26 # print(1==1)  #一个等于表示赋值,两个等于表示恒等
27 # print(True*False==False)  #True和False可以进行算术运算,True代表1,False代表0
28
29 #in,not in
30 # print(10 in list1)
31 # print(40 in list1)  #40是属于子列表,而并非属于列表
32 # print(10 not in list1)  #not 表示非,本来是True的,变为False
33 # print(40 not in list1)  #本来是False的,变为True
34
35 #逻辑运算符and,or
36 # print(2>1 and 2>3 and 10>9)  #True and False and True #一假为假,全真为真
37 # print(2>10 or 2>3 or 10>9)  #False or False or True #一真为真,全假为假
38 #优先级,not > and > or
39 # print(1>2 and 3>4 or 2>1 or 5>6 and 9>10)
40 # print(1>2 and (3>4 or 2>1 or 5>6) and 9>10)  #括号可以改变优先级
41
42 #判断某个实例是否属于某个类
43 # print(isinstance(12,int))
44 # print(isinstance(12.1,float))
45 # print(isinstance('12',str))

```