

```

1 #201021_python阶段复习1
2 #字符串,可以用单引号,双引号,三引号表示
3 # print("It's ok") # 字符串本身含有单引号时,外面用双引号
4 # print('He said:"Are you OK?"') #字符串本身有双引号时,外面用单引号
5 # print('''君问归期未有期
6 # 巴山夜雨涨秋池''') #三引号会记录字符串中换行的效果
7
8 #字符串的拼接
9 # print('1'+ '2') #字符串的拼接
10 # a=6
11 # print('1'+str(a)) #注意str型和int型不能进行拼接,可以先转换为str型
12 # print('1'*6) #注意str型和int型可以用乘号,表示显示若干次
13
14 #转义符\t,\n
15 # print('c:\note1.txt') #因为有转义符,所以显示出来的效果不符合预期
16 # print('c:\\note1.txt') #方案一 \\
17 # print(r'c:\note1.txt') #方案二 字符串前面加r
18 # print('c:/note1.txt') #方案三 /
19
20 #字符串的下标
21 # str1='avdqe'
22 # print(str1[3]) #下标从0开始计算
23 #字符串属于不可变对象,不可以更改某个下标的值
24 # str1[3]='q' #这样写会报错
25
26 #字符串的切片
27 # print(str1[3:6]) #包含起始值,不包含终止值
28 # print(str1[::-1]) #翻转字符串
29 # print(str1[-3:0]) #思考一下,能否取到值?
30
31 #列表
32 #列表是可变对象,可以通过下标对列表中的元素进行修改
33 #列表可以存放任意对象
34 # list1=[[16,90],32,64,128,256,512,1024]
35 # print(list1)
36
37 #列表的切片
38 # list2=list1 #这种是普通的赋值
39 # list1[0]=999
40 # print(list1,id(list1))
41 # print(list2,id(list2))
42 # list2=list1[:] #这种是完整的切片,等价于浅拷贝
43 import copy
44 # list2=copy.copy(list1) #浅拷贝
45
46 # list2=copy.deepcopy(list1) #深拷贝
47 # list1[0]=999
48 # print(list1,id(list1))
49 # print(list2,id(list2))
50 # list1[0][0]=999
51 # print(list1,id(list1[0]))
52 # print(list2,id(list2[0]))
53
54 #添加列表中的元素
55 # list_9=['a']

```

```
56 # list_9.append('b') #在列表的最后添加一个元素
57 # list_9.insert(0,'OPQRST') #在列表的指定位置添加一个元素
58 # print(list_9)
59
60 #删除列表中的元素
61 # list_9.pop() #默认删除最后一个元素,也可以输入一个下标,删除对应位置的值
62 # list_9.remove('a') #根据值删除
63 # print(list_9)
64 # del list_9[0]
65
66 # list10=[1,2,3,4,5]
67 # print(list10[0:2])
68 # print(list10) #切片不会影响列表本身
69
70 #元组是不可变对象,不能修改值,但是可以使用下标或切片
71 # tuple1=(10,20,30)
72 # print(tuple1[0:2])
73
74 #布尔表达式True False
75 # print('a'>='A')
76 # print(1>=2 and 2>1 and 3>2 and 5>4 and 6>5)
77 # print('Aa'=='aA') #字符串比大小,只比较第一位
78 # list1=[[16,90],32,64,128,256,512,1024]
79 # print(16 in list1) #16在子列表中,不在list1中
80
81 def fun1():
82     pass
83 # import inspect #python的自检模块
84 # print(inspect.isfunction(fun1)) #判断fun1是否是函数
85
86 # print(isinstance(123.1,float)) #判断某个实例是否属于某个类
87
88 #分支语句
89 # if True:
90 #     pass
91 # elif True:
92 #     pass
93 # else:
94 #     pass
```