1. 说明列表的浅拷贝和深拷贝的区别？

答：浅拷贝：使用copy()进行浅拷贝，只复制外围数据对象本身，该对象内部

的引用数据类型就只是引用其地址，当改变原始数据中的引用数据时，复制

出来的数据会随之改变。

深拷贝：使用copy模块中的deepcopy()进行深拷贝，既复制外围数据本身，

也复制该对象中的引用数据类型，当改变原始数据中的引用数据时，复制

出来的数据并不会会随之改变。

1. 说明列表和元组的相同点和不同点？

答：相同点：列表和元组都是有顺序可索引，可以存放任何数据类型

不同点： 列表通过[]创建，元组通过()来创建

列表中元素可变，元组中元素不可变

列表可以复制，元组不能复制

列表适合处理数据类型相同的元素，元组适合处理数据类型不

同的元素

1. 请写出字符串支持的所有方法及说明（例如: lower 返回字符串的小写）？

答：replace() 实现字符串中某个元素的替换

1. >>>a = '123456'
2. >>>a.replace('3','x')
3. '12x456'

split() 将字符串分割按某个分隔符分割成若干个子字符串,返回列表

1. >>>a = '1@234@56'
2. >>>a.split('@')
3. ['1','234','56']

partition() 将字符串按照分隔符分割成2段，返回这两段和分隔符的元组

1. >>>a = '1@234@56'
2. >>>a.partition('@')
3. ('a','@','2634@56')

join()返回一个新的字符串，将字符串用join前的字符连接起来

1. >>>a = '1@234@56'
2. >>>'\*'.join(a)
3. '1\*@\*2\*3\*4\*@\*5\*6'

len() 返回字符串的长度

1. >>>a = '1@234@56'
2. >>>len(a)
3. 8

count() 返回指定字符串出现的次数

1. >>>a = '1@234@56'
2. >>>a.count('@')
3. 2

strip() 返回一个新的字符串，移除字符串头尾指定的字符（默认为空格或换行

符）或字符序列。

1. >>>a = '\*123\*123\*'
2. >>>a.strip('\*')
3. '123\*123'

rstrip() 去除字符串右边指定信息

1. >>>a = '\*123\*123\*'
2. >>>a.rstrip('\*')
3. '\*123\*123'

capitalize() 返回新的字符串，首字母大写

1. >>>a = 'bCdEf'
2. >>>a.capitalize()
3. 'Bcdef'

title() 将每个单词首字母大写

1. >>>a = 'hello python!'
2. >>>a.title()
3. 'Hello Python!'

upper() 返回一个新的字符串，将所有字符都转成大写

1. >>>a = 'hello python!'
2. >>>a.upper()
3. 'HELLO PYTHON!'

lower() 返回一个新的字符串，将所有字符都转成小写

1. >>>a = 'HELLO PYTHON!'
2. >>>a.lower()
3. 'hello python!'

swapcase() 返回一个新的字符串，将所有字母大小写互换

1. >>>a = 'HeLlO PyThoN!'
2. >>>a.swapcase()
3. 'hElLo pYtHOn!'

centen() 字符串居中对齐

1. >>> a = 'hello python'
2. >>> a.center(20)
3. '    hello python    '

ljust() 字符串左对齐

1. >>> a = 'hello python'
2. >>> a.center(20)
3. 'hello python        '

rjust() 字符串右对齐

1. >>> a = 'hello python'
2. >>> a.center(20)
3. '        hello python'

isalnum() 检查字符串是否为字母或数字，返回布尔值

1. >>> a = 'hello python'
2. >>> a.isalnum()
3. False

isalpha() 检查字符串是否只由字母组成，返回布尔值

1. >>> a = 'Python3'
2. >>> a.isalpha()
3. False

isdigit() 检查字符串是否只由数字组成，返回布尔值

1. >>> a = '123'
2. >>> a.isdigit()
3. True

isspace() 检查是否为空白符（制表符,换行符,空格），返回布尔值

1. >>> a = '\t\n'
2. >>> a.isspace()
3. True

isupper() 检查字符串是否为大写字母，返回布尔值

1. >>> a = 'Hello'
2. >>> a.isupper()
3. False

islower() 检查字符串是否为小写字母，返回布尔值

1. >>> a = 'Hello'
2. >>> a.islower()
3. False