第八周Python实验报告

**学号**：117060400110 **姓名**：蒙柳双

**班级**：17应用统计学1班 **指导老师**：林卫中老师

**实验目的：**

1. 了解什么是列表（list）、集合以及它们的一些特点
2. 了解列表一些操作、会运用于一些列表类型特有的函数或方法，运用有关列表知识去解决随机密码的生成，
3. 会运用set(x)函数生成集合，然后运用一些集合知识与random库去解决重复元素判定问题
4. 会熟练的去定义并且调用函数，对循环结构的使用
5. 运用有关字典的知识去解文本字符分析问题

**试验要求：**

1.上课认真听老师解答问题的过程

2.上课不玩手机

3.理解实验课的每一道题

实验内容步骤：

程序练习6-1

随机密码生成。编写程序，在26个字母大小写和9个数字组成的列表中随机生成10个8位数密码。

**核心代码：**

**一．**

S=['q','w','e','r','t','y','u','i','o','p','a','s','d','f','g','h','j','k','l','z','x','c','v','b','n','m','Q','W','E','R','T','Y','U','I','O','P','A','S','D','F','G','H','J','K','L','Z','X','C','V','B','N'',M',1,2,3,4,5,6,7,8,9]

import random

def generatePSW():

for i in range(8):

print(random.choice(s),end = '')

for i in range(10):

PSW = generatePSW()

print()

**二．**

**含有简化列表的代码**

**（1）**

ch = []

c = 'A'

while c <= 'Z':

ch.append((c))

c = chr(ord(c) + 1)

c = 'a'

while c <= 'z':

ch.append((c))

c = chr(ord(c) + 1)

c = '1'

while c <= '9':

ch.append((c))

c = chr(ord(c) + 1)

import random

def generatePSW():

for i in range(8):

print(random.choice(ch),end = '')

for i in range(10):

PSW = generatePSW()

print()

**实验结果：**



**（2）**

ch = []

c = 'A'

while c <= 'Z':

ch.append((c))

c = chr(ord(c) + 1)

c = 'a'

while c <= 'z':

ch.append((c))

c = chr(ord(c) + 1)

c = '1'

while c <= '9':

ch.append((c))

c = chr(ord(c) + 1)

import random

def generatePSW(chlist,n=8):

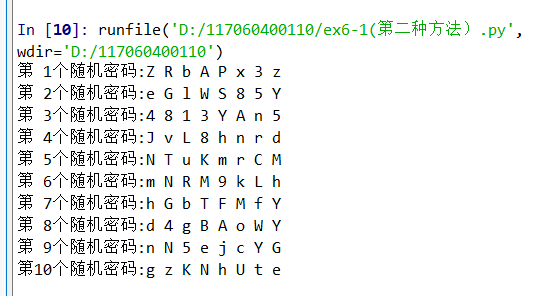
PSW = random.sample(chlist,n)

return " ".join(PSW)

for i in range(10):

print("第{0:2}个随机密码:{1}".format(i+1,generatePSW(ch)))

**实验结果：**



**（3）**

ch = []

c = 'A'

while c <= 'Z':

ch.append((c))

c = chr(ord(c) + 1)

c = 'a'

while c <= 'z':

ch.append((c))

c = chr(ord(c) + 1)

c = '1'

while c <= '9':

ch.append((c))

c = chr(ord(c) + 1)

import random

def generatePSW(chlist,n=8):

psw = []

size = len(chlist)

for i in range(8):

k = random.randint(0,size-1)

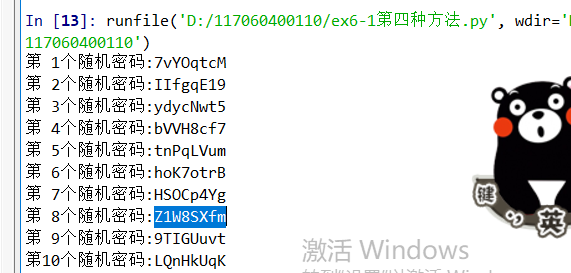
psw.append(chlist[k])

return "".join(psw)

for i in range(10):

print("第{0:2}个随机密码:{1}".format(i+1,generatePSW(ch)))

**实验结果：**



1. **这是未能完成题目要求的代码**

ch = []

c = 'A'

while c <= 'Z':

ch.append((c))

c = chr(ord(c) + 1)

c = 'a'

while c <= 'z':

ch.append((c))

c = chr(ord(c) + 1)

c = '1'

while c <= '9':

ch.append((c))

c = chr(ord(c) + 1)

import random

def generatePSW(chist,n = 8):

psw = []

for i in range(8):

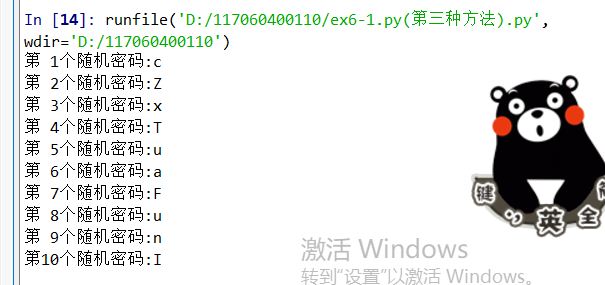
psw.append(random.choice(ch))

return "".join(psw)

for i in range(10):

print("第{0:2}个随机密码:{1}".format(i+1,generatePSW(ch)))

**实验结果：**



**注：最后一个代码还未能完成题目的要求，说实话，目前我还没有找出问题在哪，待我再去研究研究**

**程序练习6-2**

重复元素判定。编写一个函数，接受列表作为参数，如果一个元素在列表中出现不止一次，则返回True，但不要改变原来列表的值。同时编写调用这个函数和测试结果的函数。

**核心代码：**

def isRepetitive(ls):

for i in ls:

if ls.count(i) > 1:

return True

return False

ls1 = [1,2,3,1,4,5]

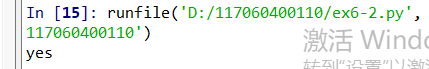
if isRepetitive(ls1):

print("yes")

else:

print("no")

**实验结果：**



**错误代码示范：**

def isRepetitive(ls):

for i in ls:

if ls.count(i) > 1:

return True

return False #这是当第一个数字出现的次数为1，则就开始return False,就不会往下执行了

ls1 = [0,2,2,3,1,4,5]

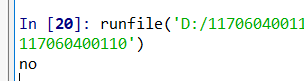
if isRepetitive(ls1):

print("yes")

else:

print("no")

**实验结果：**



**注：这是最初在写代码的时候没有注意到要求是如果有某个元素出现的次数>1，则return True;所有的元素出现的次数都等于1，则return False**

**程序练习6-3**

重复元素判定续。利用集合的无重复性改变程序练习题6-2的程序，获得一个更快更简洁的版本。

**核心代码：**

def isRepetitive(ls):

if len(ls) > len(set(ls)):

return True

else:

return False

ls1 = [0,2,2,3,1,4,5]

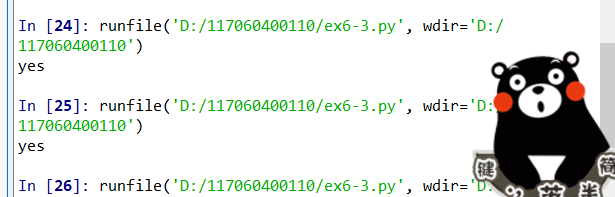
if isRepetitive(ls1):

print("yes")

else:

print("no")

**实验结果：**



**程序练习6-4**

文本字符分析。编写程序接受字符串，按字符出现频率的降序打印字母。分别尝试录入一些中英文文章片段，比较不同语言之间字符频率的差别。

**核心代码：**

**（这是复制老师的，复制精髓我还没能掌握，其实连题目都还没能好好的理解到底要怎么解决，题目还没能好好理解，待我日后再去仔细研究研究）**

**from operator import itemgetter**

**def analyseText(text):**

**D = {}**

**tlist = list(text)**

**for e in tlist:**

**if e in D:**

**D[e] = D[e] + 1**

**else:**

**D[e] = 1**

**return D**

**D = analyseText(text)**

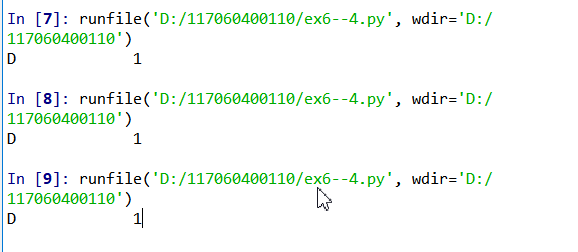
**D = sorted(D.items(),key=itemgetter(1), reverse=True)**

**for d in D:**

**ch, count = d[0],d[1]**

**print("{0:<10}{1:>5}".format(ch,count))**

**实验结果：**



（虽然我知道结果不应该是这样子，但是我还没.......能好好理解）

**实验总结：**

1. **在第一题中，最开始不会将61个元素放在一个集合当中，简化代码，最后通过老师的帮助，将61个元素的代码简化了，目前有三种方法的将随机密码生产输出，但第四种方法还存在错误，只能随机生成10个1位密码（日后再研究改进）**

**2.重复元素判定。首先要会引用到count(x)函数计算一个函数出现的次数，再者就是定义一个函数，然后调用函数**

**3.重复元素判定续。利用集合的无重复性和列表的长度跟集合长度大小比较**

**4.文本字符分析。最后一题还是不太能理解题目以及不能理解代码代表什么意思，等我理解弄明白之后再上传自己的理解（凡事总有从不会到理解的）。**