7.1. List of animal

Viết chương trình tìm thú trong vườn thú với yêu cầu sau:  
- Tạo ra một list có các con thú.  
- Nhập vào một con thú cần tìm  
=> Chương trình in ra danh sách các con thú, số lượng các con thú và kết quả tìm kiếm con thú (Con thú cần tìm có/không có trong list).

'''

Created on October 10, 2019

@author: Trung Tâm Tin Học - Trường Đại học Khoa học Tự nhiên TP.HCM

'''

# Tạo ra một list các con thú

list\_animal = ['ant', 'bear', 'cat', 'dog', 'elephant', 'fish', 'goat', 'hippo']

# In danh sách các con thú

print('List of animals:', list\_animal)

# In số lượng các con thú có trong list

# Nhập vào một con thú cần tìm

animal = input('I want to find:\n')

# In kết quả tìm kiếm con thú (Con thú cần tìm có/không có trong list)

7.2. List numbers 1

Viết chương trình xử lý list số theo các yêu cầu sau:  
  
Yêu cầu 1:  
- Tạo list số: Cho phép người dùng lần lượt nhập các phần tử số cho list cho đến khi không muốn nhập nữa.  
- Chương trình sẽ :  
+ In ra list các số vừa nhập.  
+ Tính tổng các phần tử trong list.  
  
Yêu cầu 2:  
- Người dùng nhập vào một số x cần tìm  
- Chương trình sẽ cho biết :  
+ x có xuất hiện trong list hay không? Nếu có thì cho biết x xuất hiện bao nhiêu lần?  
+ x có lớn hơn tất cả các số trong list không?  
+ Nếu không thì x nhỏ hơn những số nào trong list? (In ra tất cả các số lớn hơn x)

7.3. List numbers 2

Viết chương trình thực hiện xử lý list số như sau:  
  
- Tạo list số: Cho phép người dùng lần lượt nhập các phần tử số cho list cho đến khi không muốn nhập nữa.  
- Chương trình sẽ thực hiện những công việc sau:  
+ In ra list các số vừa nhập.  
+ Tìm và in ra tất cả các số nguyên tố có trong list.  
+ Tính trung bình cộng của các phần tử âm/ phần tử dương trong list  
+ Tìm giá trị chẵn lớn nhất/ giá trị lẻ nhỏ nhất trong list  
+ Sắp xếp list theo giá trị tăng dần

7.4. Tuple strings

Viết chương trình thực hiện việc xử lý tuple như sau:  
  
Yêu cầu 1:  
- Tạo 1 tuple có 10 phần tử chuỗi bất kỳ  
- Chương trình sẽ in tuple vừa tạo  
  
Yêu cầu 2:  
- Nhập index dương (0 <= index < 10), index âm (-1 >= index >= -9)  
- Chương trình sẽ in giá trị của phần tử trong tuple có index dương và index âm đã nhập  
  
Yêu cầu 3:  
- Nhập chuỗi cần tìm s\_find  
- Chương trình sẽ tìm và đếm số lần xuất hiện của s\_find trong tuple

'''

Created on October 10, 2019

@author: Trung Tâm Tin Học - Trường Đại học Khoa học Tự nhiên TP.HCM

'''

# Tạo 1 tuple có 10 phần tử chuỗi bất kỳ

tuple\_strings = ('red', 'green', 'yellow', 'blue', 'black', 'white', 'pink', 'orange','red', 'blue')

# In tuple vừa tạo

print('Tuple:', tuple\_strings)

# Nhập index dương (0 <= index < 10), index âm (-1 >= index >= -9)

chuoi\_1 = 'Nhập số từ 0 đến ' + str(len(tuple\_strings) -1) + ': \t'

chuoi\_2 = 'Nhập số từ -1 đến ' + str(-(len(tuple\_strings) -1)) + ':\t'

index\_duong = int(input(chuoi\_1))

index\_am = int(input(chuoi\_2))

# In giá trị của phần tử trong tuple có index dương và index âm đã nhập

# Nhập chuỗi cần tìm s\_find

s\_find = input('Nhập chuỗi cần tìm:\n')

# Tìm và đếm số lần xuất hiện của s\_find trong tuple

7.5. Tuple numbers

Viết chương trình thực hiện việc xử lý trên tuple như sau:  
- Người dùng sẽ tạo:  
+ 1 tuple a chứa 4 số nguyên dương đầu tiên  
+ 1 tuple b chứa 4 số nguyên dương tiếp theo  
- Chương trình sẽ:  
+ Tạo 1 tuple c là sự kết hợp của các phần tử trong tuple a và b  
+ Tạo 1 tuple d từ tuple c với các phần tử được sắp xếp  
+ In phần tử thứ 3 của d  
+ In 3 phần tử cuối cùng của d

'''

Created on October 10, 2019

@author: Trung Tâm Tin Học - Trường Đại học Khoa học Tự nhiên TP.HCM

'''

# Tạo 1 tuple a chứa 4 số nguyên dương đầu tiên

tuple\_a = (1, 2, 3, 4)

print('Tuple a:', tuple\_a)

# Tạo 1 tuple b chứa 4 số nguyên dương tiếp theo

tuple\_b = (5, 7, 6, 8)

print('Tuple b:', tuple\_b)

# Tạo 1 tuple c là sự kết hợp của các phần tử trong tuple a và b

# Tạo 1 tuple d từ tuple c với các phần tử được sắp xếp

# In phần tử thứ 3 của d

# In 3 phần tử cuối cùng của d

7.6. Set numbers

Viết chương trình thực hiện việc xử lý trên set như sau:  
  
- Khai báo và khởi tạo set1, set2  
+ Cho phép người dùng lần lượt nhập các phần tử số cho set1 cho đến khi không muốn nhập nữa  
+ Cho phép người dùng lần lượt nhập các phần tử số cho set2 cho đến khi không muốn nhập nữa  
  
- Chương trình sẽ thực hiện những công việc sau:  
+ In set1, set2  
+ Cho biết mỗi set có bao nhiêu phần tử, tổng giá trị các phần tử của mỗi set  
+ Tìm giá trị lớn nhất, nhỏ nhất của mỗi set  
+ Lấy ra một phần tử ở set1 và in ra phần tử này  
+ Thực hiện set union của set1 và set2 và in kết quả  
+ Thực hiện set intersection của set1 và set2 và in kết quả  
+ Thực hiện set difference của set1 với set2 và in kết quả  
+ Thực hiện set symmetric difference của set1 với set2 và in kết quả  
+ Sắp xếp set1 tăng dần và set2 giảm dần

'''

Created on October 10, 2019

@author: Trung Tâm Tin Học - Trường Đại học Khoa học Tự nhiên TP.HCM

'''

# Khai báo và khởi tạo set1, set2

# Cho phép người dùng lần lượt nhập các phần tử số cho set1 cho đến khi không muốn nhập nữa

set1=set()

gt="1"

while gt!="":

gt=input('Nhap gia tri cho set 1 (enter ket thuc): ')

if gt!="":

set1.add(int(gt))

# Cho phép người dùng lần lượt nhập các phần tử số cho set2 cho đến khi không muốn nhập nữa

set2=set()

gt="1"

while gt!="":

gt=input('Nhap gia tri cho set 2 (enter ket thuc): ')

if gt!="":

set2.add(int(gt))

# In set1, set2

print("Set 1: ", set1)

print("Set 2: ", set2)

# Cho biết mỗi set có bao nhiêu phần tử

# Tổng giá trị các phần tử của mỗi set

# Tìm giá trị lớn nhất, nhỏ nhất của set1

# Tìm giá trị lớn nhất, nhỏ nhất của set2

# Lấy ra một phần tử ở set1 và in ra phần tử này

# Thực hiện set union của set1 và set2 và in kết quả

# Thực hiện set intersection của set1 và set2 và in kết quả

# Thực hiện set difference của set1 với set2 và in kết quả

# Thực hiện set symmetric difference của set1 với set2 và in kết quả

# Sắp xếp set1 tăng dần và set2 giảm dần

7.7. Loại bỏ các từ trùng lặp trong chuỗi và sắp xếp theo thứ tự bảng chữ cái

Viết chương trình xử lý chuỗi với yêu cầu sau:  
- Người dùng nhập vào một chuỗi các từ tách biệt bởi khoảng trắng (input)  
- Chương trình xử lý chuỗi: loại bỏ các từ trùng lặp, sắp xếp theo thứ tự bảng chữ cái,  
và in kết quả (output)  
  
Ví dụ:  
- input: hello world and practice makes perfect and hello world again  
- output: again and hello makes perfect practice world

'''

Created on October 10, 2019

@author: Trung Tâm Tin Học - Trường Đại học Khoa học Tự nhiên TP.HCM

'''

# Nhập vào một chuỗi các từ tách biệt bởi khoảng trắng

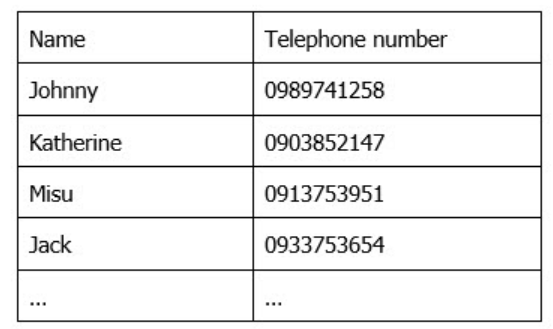
s = input("Nhập chuỗi của bạn: ")

words = [word for word in s.split(" ")]

# Loại bỏ các từ trùng lặp, sắp xếp theo thứ tự bảng chữ cái, và in kết quả

7.8. Danh bạ điện thoại

Viết chương trình thực hiện việc xử lý danh bạ điện thoại như sau:  
  
- Tạo một danh bạ kiểu dictionary để lưu trữ danh bạ điện thoại với các cặp key-value (ví dụ như hình 7.8)  
- Nhập tên cần tìm search\_name  
- Nhập tên, số điện thoại  
  
=> Chương trình sẽ thực hiện những công việc sau:  
- Tìm search\_name trong danh bạ. Nếu không tìm thấy thì in thông tin tên – số điện thoại. Nếu không tìm thấy thì thông báo là không tìm thấy.  
- Thêm một liên hệ mới với thông tin: tên – số điện thoại đã nhập  
- In danh bạ



'''

Created on October 10, 2019

@author: Trung Tâm Tin Học - Trường Đại học Khoa học Tự nhiên TP.HCM

'''

# Tạo một danh bạ kiểu dictionary để lưu trữ danh bạ điện thoại với các cặp key-value

danh\_ba = {'Johnny': '0989741258', 'Katherine':'0903852147', 'Misu': '0913753951', 'Jack' : '0933753654'}

i = 1

while i == 1:

cv = int(input('Bạn muốn làm gì? 1: Xem danh bạ; 2: Tìm kiếm, 3: Thêm mới\t'))

# In danh bạ điện thoại

if cv == 1:

print('Danh bạ điện thoại:')

print('Tên \t Số điện thoại')

# Tìm search\_name trong danh bạ.

# Nếu không tìm thấy thì in thông tin tên – số điện thoại. Nếu không tìm thấy thì thông báo là không tìm thấy.

elif cv == 2:

name\_search = input('Nhập tên cần tìm:\n')

# Thêm một liên hệ mới với thông tin: tên – số điện thoại đã nhập

elif cv == 3:

ten = input('Nhập tên:\n')

fone = input('Nhập số điện thoại:\n')

print('Danh bạ điện thoại:')

print('Tên \t Số điện thoại')

i = int(input('Tiếp tục lựa chọn? 1: Có; 0: Không\t'))

7.9. Từ điển

Viết chương trình thực hiện việc xử lý từ điển Anh – Việt như sau:  
  
- Tạo một từ điển (key: từ tiếng Anh, value: list nhiều nghĩa Tiếng Việt)  
ví dụ: check: tờ séc, hóa đơn, kiểm tra, chiếu cờ tướng  
  
=> Chương trình sẽ thực hiện những công việc sau:  
- Hiển thị từ điển, cho biết trong từ điển hiện tại có bao nhiêu từ  
- Thêm từ vào từ điển, và hiển thị từ điển sau khi thêm  
- Tìm kiếm từ tiếng Anh => nếu tìm thấy thì hiển thị key và value. Nếu không tìm thấy thì thông báo không tìm thấy  
- Xóa một từ trong từ điển, dựa trên key cung cấp , và hiển thị từ điển sau khi xóa

'''

Created on October 10, 2019

@author: Trung Tâm Tin Học - Trường Đại học Khoa học Tự nhiên TP.HCM

'''

# Tạo một từ điển ((key: từ tiếng Anh, value: list nghĩa Tiếng Việt))

dictionary = {'work': ['công việc', 'việc làm', 'tác phẩm'], 'bark': ['vỏ cây', 'tiếng sủa', 'thuyền ba cột buồm'], 'bat':['con dơi', 'gậy', 'đánh bóng'], 'board': ['bảng', 'ban quản lý', 'lên tàu'], 'bowl' : ['cái bát', 'khán đài', 'quả bóng quần'], 'stamp' : ['con tem', 'phiếu mua hàng', 'con dấu'], 'club' : ['câu lạc bộ', 'gậy đánh golf', 'dùi cui']}

# Chương trình thực hiện các công việc:

i = 1

while i == 1:

cv = int(input('Bạn muốn làm gì? 1: Hiển thị từ điển; 2: Tra từ, 3: Thêm từ, 4: Xóa từ\t'))

# Hiển thị từ điển, và cho biết trong từ điển hiện tại có bao nhiêu từ?

if cv == 1:

print('Dictionary:')

print('Từ Anh \t Nghĩa Việt')

# Tìm kiếm từ tiếng Anh => nếu tìm thấy thì hiển thị key và value. Nếu không tìm thấy thì thông báo không tìm thấy

elif cv == 2:

name\_search = input('Nhập từ cần tra:\t')

# Thêm từ vào từ điển, Hiển thị từ điển sau khi thêm

elif cv == 3:

word = input('Nhập từ Anh:\t')

meaning = input('Nhập nghĩa Việt:\t')

# Xóa một từ trong từ điển, dựa trên key cung cấp , và Hiển thị từ điển sau khi xóa

elif cv == 4:

word\_delete = input('Nhập từ cần xóa:\t')

x = int(input('Bạn có thật sự muốn xóa hay không? 1: Xóa, 0: Không\t'))

i = int(input('Tiếp tục lựa chọn? 1: Có; 0: Không\t'))

7.10. Sự khác biệt giữa List, Tuple, Set, Dictionary

Trong ngôn ngữ Python, kiểu dữ liệu List là một trong các kiểu cấu trúc dữ liệu tập hợp bao gồm: List, Tuple, Set, Dictionary. Tuy nhiên, việc hiểu ý nghĩa và cách sử dụng của từng kiểu đôi khi chúng ta lại thấy hơi khó hiểu, do đó, chúng ta cùng ôn lại 4 kiểu cấu trúc dữ liệu này và sự khác biệt giữa chúng như thế nào nhé!

<https://drive.google.com/file/d/1ZFAW4fjKjEVs6RBc8diQCUqib4reh14L/view?usp=sharing>

Ôn tập bài 7: Trong bài 7, chúng ta được học và sử dụng các phương thức dành cho List, Tuple, Dictionary vào việc giải các bài toán.  
  
Để ôn bài, chúng ta cùng làm bài trắc nghiệm sau:

<https://docs.google.com/forms/d/e/1FAIpQLScDQbFDA72QSiGsRXUJ5cRGLeZ6cnVy5OnYeopYjh0BWf-VQQ/viewform?hr_submission=ChkIvZrx2r8BEhAI9-TUwPoEEgcImPPvvvoEEAE>