#-------------------------------------------------------------------------------

# Name: module1

# Purpose:

#

# Author: Andrew

#

# Created: 22/07/2013

# Copyright: (c) Andrew 2013

# Licence: <your licence>

#-------------------------------------------------------------------------------

def main():

#A few examples of a dictionary

#First we define the dictionary

#it will have nothing in it this time

#ages = {}

ages = {'Andy':59,'Paul':54}

#Add a couple of names to the dictionary

ages['Sue'] = 23

ages['Peter'] = 19

ages['Andrew'] = 78

ages['Karren'] = 45

#It returns TRUE if the dictionary has key-name in it

#but returns FALSE if it doesn't.

#Remember - this is how 'if' statements work -

#they run if something is true

#and they don't when something is false.

if 'Sue' in ages:

print ("Sue is in the dictionary. She is", ages['Sue'], "years old")

else:

print ("Sue is not in the dictionary")

#Use the function keys() -

#This function returns a list

#of all the names of the keys.

#E.g.

print ("The following people are in the dictionary:")

print (ages.keys())

#You could use this function to

#put all the key names in a list:

keys = ages.keys()

print ("Here is the LIST of keys :", keys)

#You can also get a list

#of all the values in a dictionary.

#You use the values() function:

print ("People are aged the following:", ages.values())

#Put it in a list:

values = ages.values()

#You can sort lists, with the sort() function

#It will sort all values in a list

#alphabetically, numerically, etc...

#You can't sort dictionaries -

#they are in no particular order

print (keys)

print ("sorted keys",sorted(ages.keys()))

print (values)

print ("sorted values",sorted (ages.values()))

#You can find the number of entries

#with the len() function:

print ("The dictionary has", len(ages), "entries in it")

if \_\_name\_\_ == '\_\_main\_\_':

main()