```python
'''
    dictionaries.py -- sample uses of dictionaries

    Jeff Ondich, 24 October 2006
    Tianna Avery, November 2018, updated for Python 3

    Python's "dictionaries" are associative arrays (like Perl's
"hashes").
    You can think of them as arrays whose indices can be (almost) any
object, not
    just integers starting at 0.  Dictionaries are also referred to as
"mapping
    types" in some of the Python documentation.
'''


# Initialize a dictionary.
dictionary = {'goat' : 'domestic', 'moose' : 'wild', 'kudu' : 'wild',
'emu' : 'weird'}

# It's cool that you can print(a dictionary so easily.
print(dictionary)

# Iterating over the (key, value) pairs in a dictionary is a snap.
for key in dictionary:
    print('{0} => {1}'.format(key, dictionary[key]))
print()

# Iterate over the keys in alphabetical order.
print(dictionary.keys())
sorted_keys = dictionary.keys()
sorted_keys = sorted(sorted_keys)
for key in sorted_keys:
    print('{0} => {1}'.format(key, dictionary[key]))
print()

# Test for the presence of a given key using "in".
animal = input('Animal, please: ')
if animal in dictionary:
    print('The {0} is {1}'.format(animal, dictionary[animal]))
else:
    print('The {0} is not in the dictionary.'.format(animal))

# Delete a (key, value) pair via the key.
del dictionary['kudu']
print(dictionary)

# Go to https://docs.python.org/3/library/stdtypes.html#typesmapping
# and experiment with some of the operations described there.
```