[learnpython.trinket.io/learn-python-part-9-dictionaries](https://learnpython.trinket.io/learn-python-part-9-dictionaries)

**For 35 points, use the above website to fill in the blanks below.**

**Dictionaries**

Dictionaries use \_\_\_\_\_\_\_\_\_\_\_\_\_\_ in order to look up \_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Dictionaries in Python are surrounded by \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ brackets.

Keys and values are separated with a \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

The most commonly used type of keys is \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

The \_\_\_\_\_\_\_\_\_\_ command can be used to list the keys of a dictionary.

The \_\_\_\_\_\_\_\_\_\_ command can be used to list the values of a dictionary.

Give an example of the creation of a dictionary not given on the webpage:

An important difference between dictionaries and lists is \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Think of a dictionary full of keys and values like a crowd of people. You know your friend Maleficent is somewhere in the crowd. You don’t know where, but you’re pretty sure there is only one person with that name in the crowd. You yell your friend’s name and they head right to you.

Other programming languages may call dictionaries: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_,

\_\_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, or \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

**Adding values**

To add a value to a dictionary, just set the dictionary's \_\_\_\_\_\_\_\_\_ equal to \_\_\_\_\_\_\_\_\_\_\_.

The **update(..)** command combines the dictionary in parentheses with the other dictionary.

Create your own dictionary, perhaps mapping your favorite cities to their home countries, with at least 3 entries. Then, add one country:city pair with the update(..) function and another by creating a new key in the dictionary and setting its value. Write the code below:

**Getting values**

To "get back" a value you've stored in a dictionary, ask for it by its \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

If you ask for a key that doesn't exist, Python will return \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Pull out all the names of the cities that start with the letter 'C' from the places dictionary:

>>> places = {'il':'chicago', 'ca':'los angeles', 'co':'boulder', 'oh':'columbus', 'fl':'tampa'}

**Removing values**

Use the **del** keyword to remove items from a dictionary.

If we try to delete something that doesn't exist from the dictionary, we'll get \_\_\_\_\_\_\_\_\_\_

Write code to delete 'lettuce':'green' from the following dictionary:

foods = {'carrot':'orange','radish':'red','lettuce':'green'}