Python Dictionaries

Key/value pairs

https://www.programiz.com/python-programming/dictionary

Definition

- Python dictionary is an unordered collection of items
- Dictionaries are optimized to retrieve values when the key is known
- Keys must be unique
- the values can be of any data type

Creating Python Dictionary

- Create a dictionary by placing items inside curly braces {} separated by commas
- An item has a key and a value expressed as a pair (key: value)

```
# empty dictionary
my_dict = {}

# dictionary with integer keys
my_dict = {1: 'apple', 2: 'ball'}
```

Accessing Elements from Dictionary

- to access values, a dictionary uses keys
- Keys can be used inside square brackets []
- *Trying to access keys which don't exist throws an error*
- my_dict = {'name': 'Jack', 'age': 26}

```
# Output: Jack
print(my_dict['name’])

# Output: 26
print(my_dict.get('age'))
```

Changing and Adding Dictionary elements

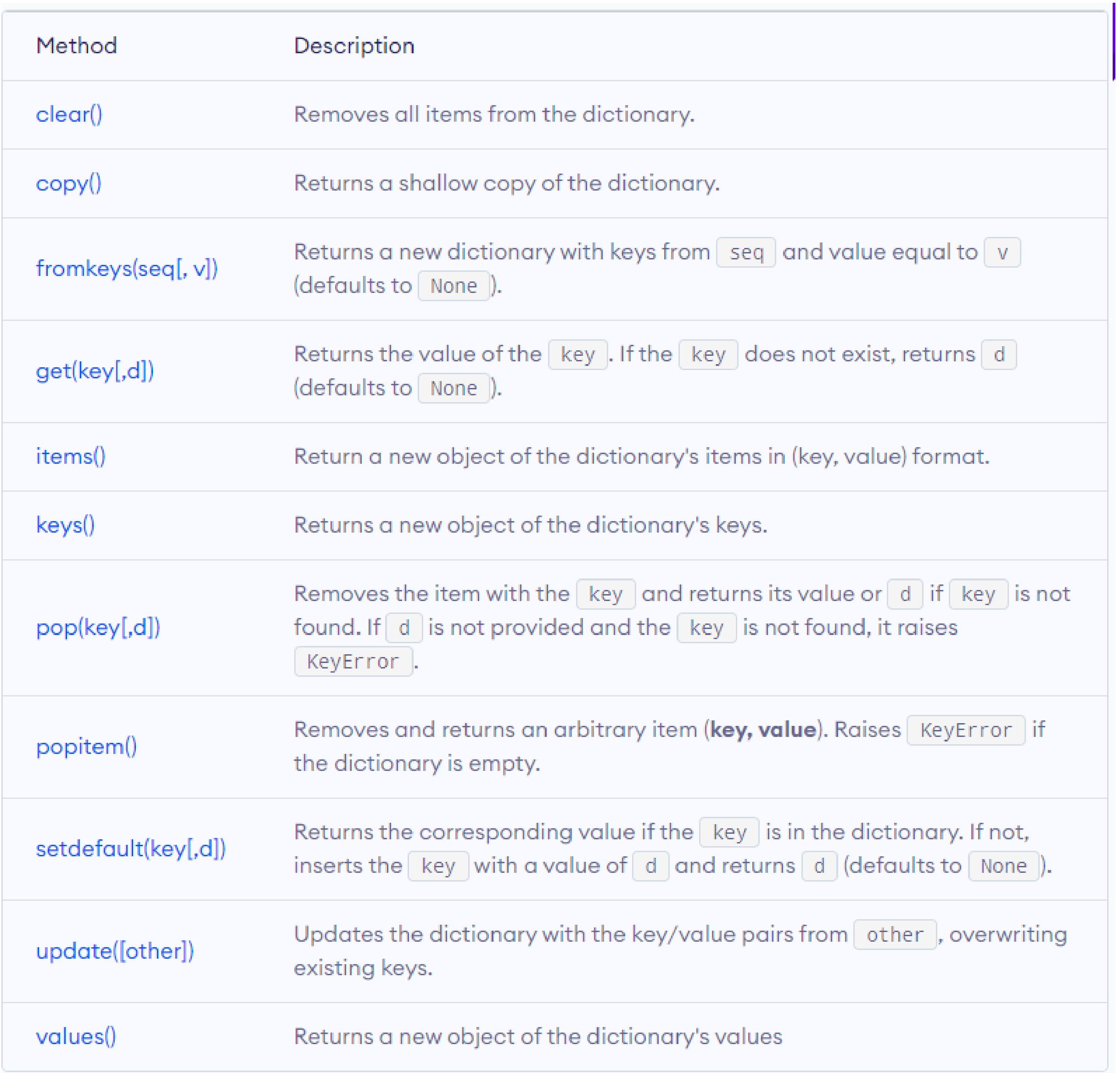
- If the key is already present, the existing value gets updated
- Example: my_dict['age'] = 27
- ---
- If the key is not present, a new (key: value) pair is created
- Add item: my_dict['address'] = 'madison'

Removing elements from Dictionary

- We can remove an item in a dictionary by using the pop() method

```
# remove an item which returns its value
# Output: 27
print(squares.pop(age))
```

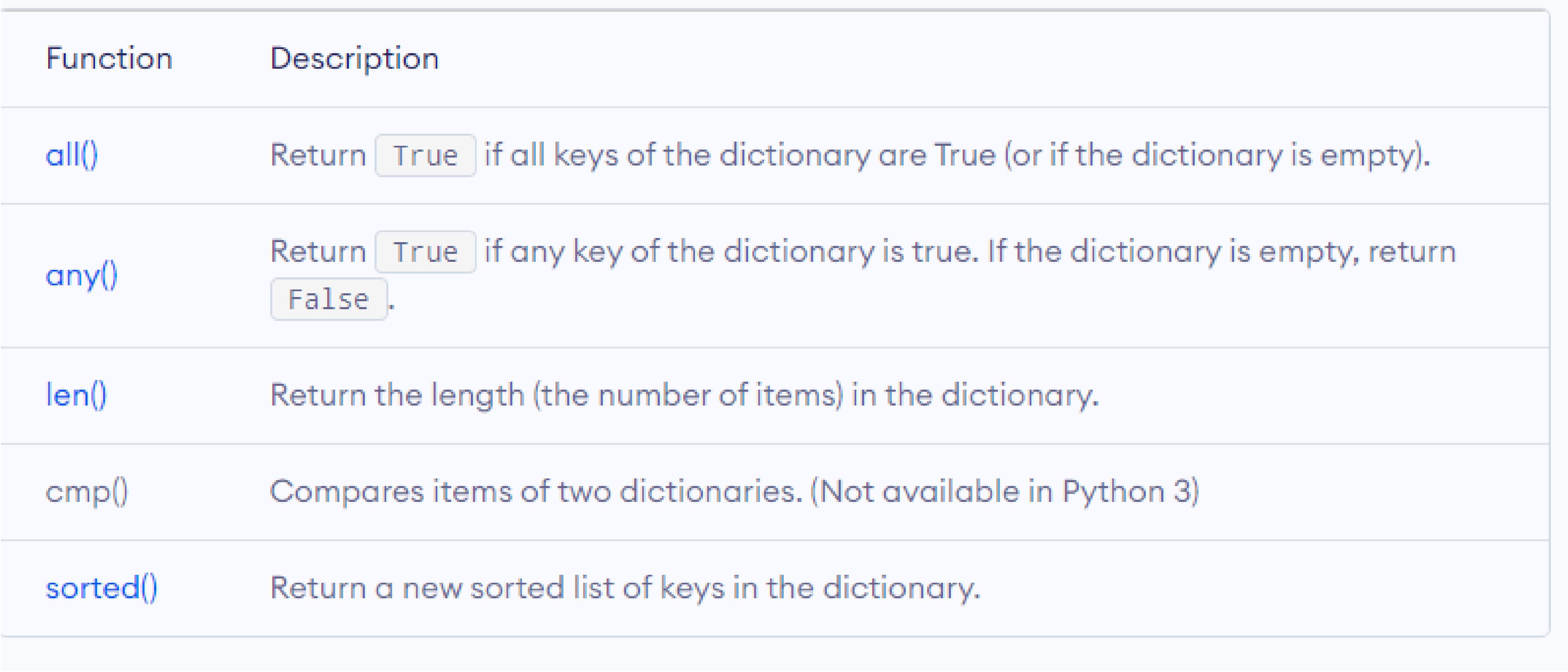
Python Dictionary Methods



Method	Description
clear()	Removes all items from the dictionary.
copy()	Returns a shallow copy of the dictionary.
fromkeys(seq[, v])	Returns a new dictionary with keys from `seq` and value equal to `v` (defaults to `None`).
get(key[,d])	Returns the value of the `key`. If the `key` does not exist, returns `d` (defaults to `None`).
items()	Return a new object of the dictionary's items in (key, value) format.
keys()	Returns a new object of the dictionary's keys.
pop(key[,d])	Removes the item with the `key` and returns its value or `d` if `key` is not found. If `d` is not provided and the `key` is not found, it raises `KeyError`.
popitem()	Removes and returns an arbitrary item (**key, value**). Raises `KeyError` if the dictionary is empty.
setdefault(key[,d])	Returns the corresponding value if the `key` is in the dictionary. If not, inserts the `key` with a value of `d` and returns `d` (defaults to `None`).
update([other])	Updates the dictionary with the key/value pairs from `other`, overwriting existing keys.
values()	Returns a new object of the dictionary's values

Dictionary Built-in Functions

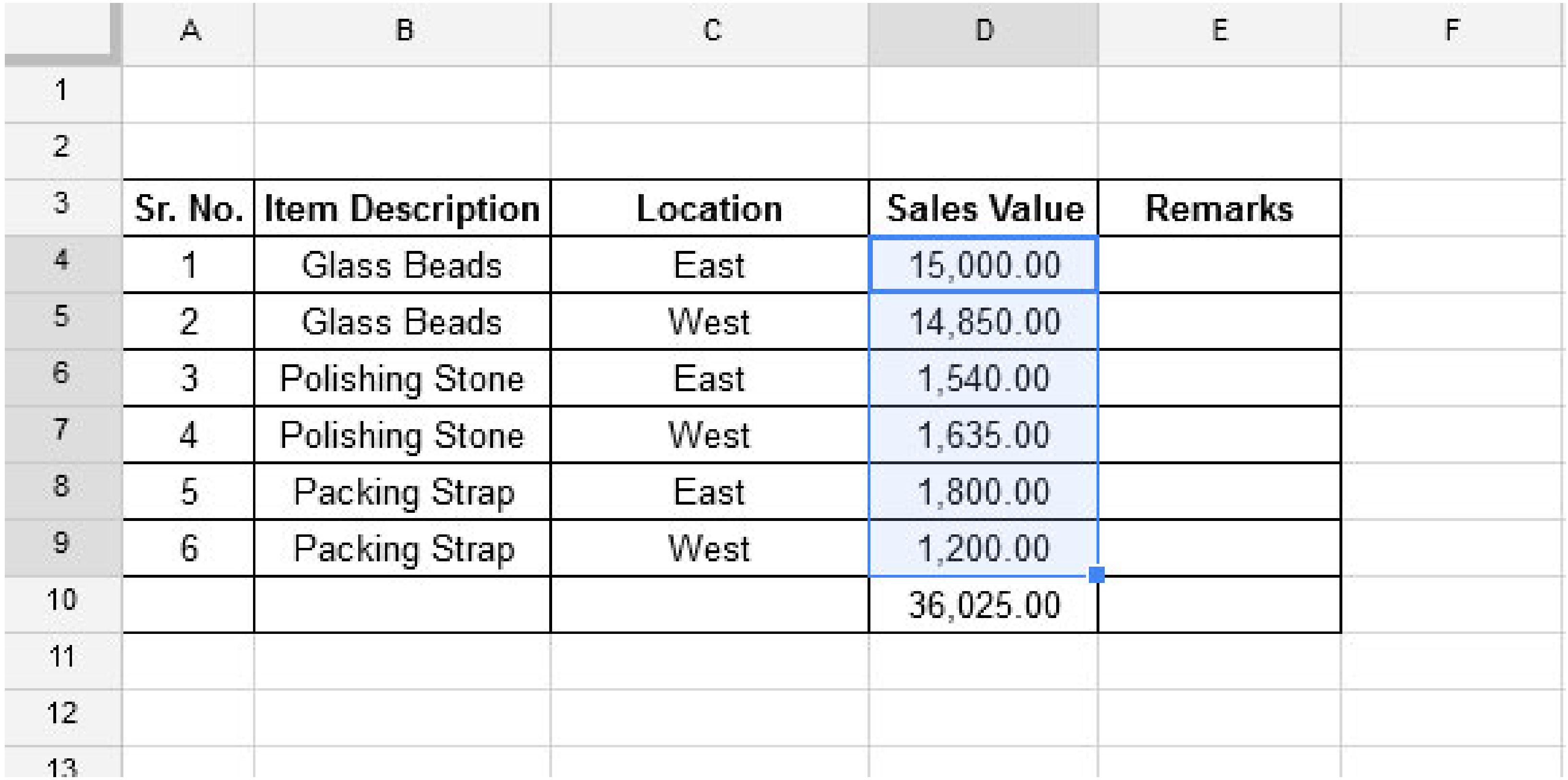
https://www.w3schools.com/python/python_ref_functions.asp



Function	Description
all()	Return `True` if all keys of the dictionary are True (or if the dictionary is empty).
any()	Return `True` if any key of the dictionary is true. If the dictionary is empty, return `False`.
len()	Return the length (the number of items) in the dictionary.
cmp()	Compares items of two dictionaries. (Not available in Python 3)
sorted()	Return a new sorted list of keys in the dictionary.

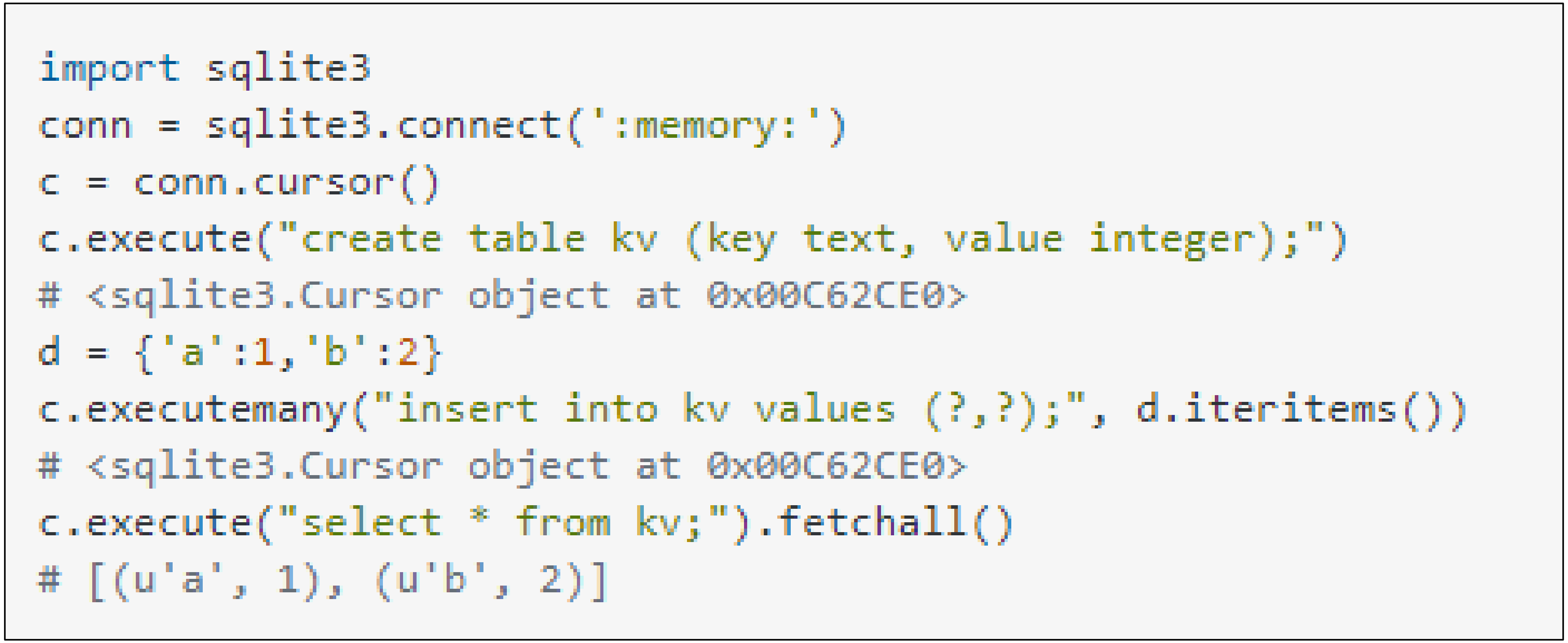
Unique key: Sr. No.

Excel Data



Sr. No.	Item Description	Location	Sales Value	Remarks
1	Glass Beads	East	15,000.00	
2	Glass Beads	West	14,850.00	
3	Polishing Stone	East	1,540.00	
4	Polishing Stone	West	1,635.00	
5	Packing Strap	East	1,800.00	
6	Packing Strap	West	1,200.00	
			36,025.00	

DB key/value Example



```python
import sqlite3
conn = sqlite3.connect(':memory:')
c = conn.cursor()
c.execute("create table kv (key text, value integer);")
# <sqlite3.Cursor object at 0x00C62CE0>
d = {'a':1,'b':2}
c.executemany("insert into kv values (?,?);", d.iteritems())
# <sqlite3.Cursor object at 0x00C62CE0>
c.execute("select * from kv;").fetchall()
# [(u'a', 1), (u'b', 2)]
```