

[Home \(/\)](#) >> [How to use dictionaries in Python](#)

Oct. 13, 2012

 [Dictionary \(/dictionary/\)](#)

How to use dictionaries in Python

This post will explain how to use dictionaries in Python.

About dictionaries in Python

Use {} curly brackets to construct the dictionary, and [] square brackets to index it. Separate the key and value with colons : and with commas , between each pair. Keys must be quoted. As with lists we can print out the dictionary by printing the reference to it. A dictionary maps a set of objects (keys) to another set of objects (values). A Python dictionary is a mapping of unique keys to values. Dictionaries are mutable, which means they can be changed. The values that the keys point to can be any Python value. Dictionaries are unordered, so the order that the keys are added doesn't necessarily reflect what order they may be reported back.

Create a new dictionary

In order to construct a dictionary you can start with an empty one. >>> mydict={} # This will create a dictionary, which has an initially six key-value pairs, where iphone* is the key and years the values

```
released = {  
    "iphone" : 2007,  
    "iphone 3G" : 2008,  
    "iphone 3GS" : 2009,  
    "iphone 4" : 2010,  
    "iphone 4S" : 2011,  
    "iphone 5" : 2012  
}  
  
print released
```

```
>>>Output  
{'iphone 3G': 2008, 'iphone 4S': 2011, 'iphone': 2007, 'iphone 5': 2012,
```

Add a value to the dictionary

You can assign to an individual dictionary entry to add

```
#the syntax is: mydict[key] = "value"  
released["iphone 5S"] = 2013  
print released
```



[广告 X](#)

Project Manage With Airtable

Finally, a project management tool that teams actually want to use

Airtable

```
>>>Output
{'iphone 5S': 2013, 'iphone 3G': 2008, 'iphone 4S': 2011, 'iphone 3GS': 2009,
 'iphone': 2007, 'iphone 5': 2012, 'iphone 4': 2010}
```

Remove a key and it's value

You can remove key-value pairs with the del operator

```
del released["iphone"]
print released
```

```
>>>output
{'iphone 3G': 2008, 'iphone 4S': 2011, 'iphone 3GS': 2009, 'iphone 5': 2012,
 'iphone 4': 2010}
```

Check the length

The len() function gives the number of pairs in the dictionary.

```
print len(released)
```

Test the dictionary

Check if a key exists in a given dictionary by using the in operator like this:

```
>>> my_dict = {'a' : 'one', 'b' : 'two'}
>>> 'a' in my_dict
True
>>> 'b' in my_dict
True
>>> 'c' in my_dict
False
```

or like this in a for loop

```
for item in released:
    if "iphone 5" in released:
        print "Key found"
        break
    else:
        print "No keys found"
```

```
>>output
Key found
```

Get a value of a specified key

```
print released.get("iphone 3G", "none")
```

Print all keys with a for loop

```
print "-" * 10
print "iphone releases so far: "
print "-" * 10
for release in released:
    print release
```

```
>>output
-----
iphone releases so far:
-----
iphone 3G
iphone 4S
iphone 3GS
iphone
iphone 5
iphone 4
```

Print all key and values

```
for key, val in released.items():
    print key, "=>", val
```

```
>>output
iphone 3G => 2008
iphone 4S => 2011
iphone 3GS => 2009
iphone => 2007
iphone 5 => 2012
iphone 4 => 2010
```

Get only the keys from the dictionary

```
phones = released.keys()
print phones
```

or print them out like this:

```
print "This dictionary contains these keys: ", " ".join(released)
>>iphone 3G iphone 4S iphone 3GS iphone 5 iphone 4
```

or like this:

```
print "This dictionary contains these keys: ", " ", released.keys()
>>['iphone 3G', 'iphone 4S', 'iphone 3GS', 'iphone', 'iphone 5', 'iphone 4']
```

Printing the values

Elements may be referenced via square brackets, using the key: print released["iphone"]

```
print "Values:
",
for year in released:
    releases= released[year]
    print releases
```

```
>>output:
Values:
2008
2011
2009
2007
2012
2010
```

Printing with pprint

```
pprint.pprint(released)
```

Sorting the dictionary

```
for key, value in sorted(released.items()):  
    print key, value
```

```
>>output:  
( 'iphone', 2007)  
( 'iphone 3G', 2008)  
( 'iphone 3GS', 2009)  
( 'iphone 4', 2010)  
( 'iphone 4S', 2011)  
( 'iphone 5', 2012)  
""""
```

```
for phones in sorted(released, key=len):  
    print phones, released[phones]
```

```
>>output:  
iphone 2007  
iphone 5 2012  
iphone 4 2010  
iphone 3G 2008  
iphone 4S 2011  
iphone 3GS 2009
```

Counting

```
count = {}  
for element in released:  
    count[element] = count.get(element, 0) + 1  
print count
```

```
>>output:  
{ 'iphone 3G': 1, 'iphone 4S': 1, 'iphone 3GS': 1, 'iphone': 1,  
  'iphone 5': 1, 'iphone 4': 1}
```

Recommended Python Training – DataCamp (https://www.datacamp.com/?tap_a=5644-dce66f&tap_s=75426-9cf8ad&tm_source=recommended)

For Python training (https://www.datacamp.com/?tap_a=5644-dce66f&tap_s=75426-9cf8ad&tm_source=recommended), our top recommendation is DataCamp.

Datacamp (https://www.datacamp.com/?tap_a=5644-dce66f&tap_s=75426-9cf8ad&tm_source=recommended) provides online interactive courses that combine interactive coding challenges with videos from top instructors in the field.

Datacamp has beginner to advanced Python training that programmers of all levels benefit from.




Tweet

Like 0

Share

Read more about:

 Dictionary (/dictionary/)

Did You Know?

1/15



From its release in 2003 until 2014, the Nokia 1100 has sold over 250 million unit's worldwide, making it the best selling consumer electronic device in history.



0 Comments Pythonforbeginners.com

1 Login ▾

 Recommend 2  Tweet  Share

Sort by Best ▾






Start the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS 

Be the first to comment.

 Subscribe  Add Disqus to your siteAdd DisqusAdd  Disqus' Privacy PolicyPrivacy PolicyPrivacy PolicyPrivacy Policy

Sponsored Links

[A man buys an old plane and turns it into a house. Look inside!](#)

Easyvoyage

[Surrogate Mom Who Gave Birth to 'Twins' Learns One Is Her](#)

PostFun

[The Most Beautiful Royal Women](#)

Hooch

[Just 3 Ingredients Will Help You to Clear Clogged Arteries Without Medication](#)

Food Prevent

[New WiFi Booster Stops Expensive Internet in China](#)

SuperBoost Wifi

[High Blood Pressure: 10 Foods to Avoid](#)

Manuka Feed

Disclosure of Material Connection: Some of the links in the post above are “affiliate links.” This means if you click on the link and purchase the item, I will receive an affiliate commission. Regardless, PythonForBeginners.com only recommend products or services that we try personally and believe will add value to our readers.

SEARCH

Categories

[Basics \(/basics/\)](#)

[Cheatsheet \(/cheatsheet/\)](#)

[Code snippets \(/code-snippets-source-code/\)](#)

[Development \(/development/\)](#)

Dictionary (/dictionary/)

Error Handling (/error-handling/)

Lists (/lists/)

Loops (/loops/)

Modules (/modules-in-python/)

Strings (/python-strings/)

System & OS (/systems-programming/)

Web & Internet (/python-on-the-web/)

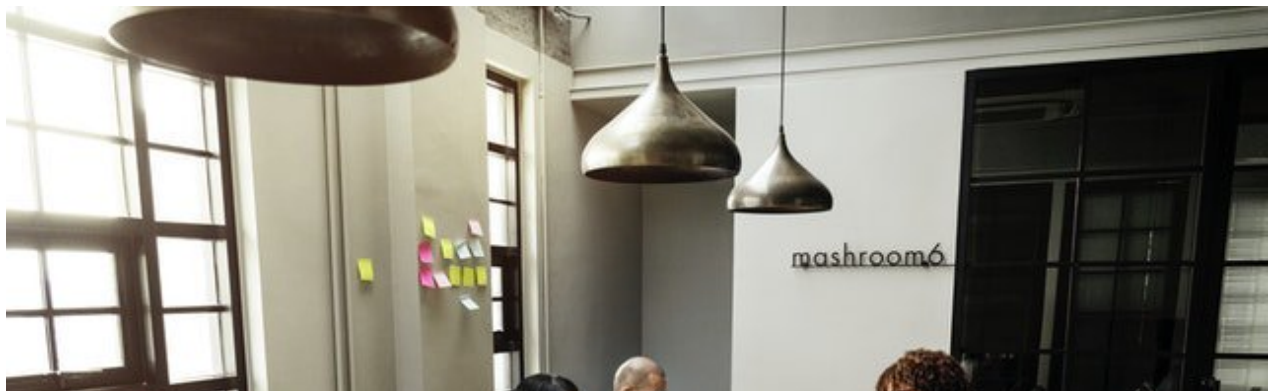
Tweets by [@pythonbeginners](#)



pythonforbeginners

@pythonbeginners

6 Python Programming Projects for Beginners [ow.ly/vs1m30onZU6](https://www.pythonforbeginners.com/dictionary/how-to-use-dictionaries-in-python/) #python #learntocode





Apr 12, 2019

**pythonforbeginners**

@pythonbeginners

SAVE THIS: Python Cheat Sheets ow.ly/uyKG30onZQy#python #learntocode

Apr 11, 2019

**pythonforbeginners**

@pythonbeginners

广告 X

VPN That Works in China

No Logs. Unblock A
Site. 1000s of IP
Locations. 24/7 Supp

© Python For Beginners (<https://www.pythonforbeginners.com>) 2012-2019 | [Privacy Policy \(/privacy-policy/\)](/privacy-policy/) | [Write For Us \(/write/\)](/write/) | [Contact Us \(/contact-us/\)](/contact-us/)