

The 8 Best Free Python Cheat Sheets for Beginners and Experts

September 19, 2018 By sak1m 4 Comments

Last Updated on July 31, 2019



Download a PDF version of this Post

Python Cheat Sheet can be really helpful when you're working on a project or trying a set of exercises related to a specific topic.

“ If you are just getting started with Data Science or Machine Learning, i've got you covered in this blog post about **Learning how to learn Data Science** (Python, Maths and Statistics).

in the form of cheat sheet.

Content Insight 

7+ Best Python Cheat Sheets ...

Here's a curated a list of Python Cheat Sheets and most commonly used Python Libraries. You'll be able to download them with ease and grasp the fundamentals for long-term benefits




1. Python for Data Science Cheat Sheet

This Python Cheat Sheet presents the Python basics that you need to do data science and will guide you through variables and data types, Strings, Lists, to eventually land at the fundamental package for scientific computing with Python, Numpy.



Also, This cheat sheet is free additional material that complements DataCamp's [Intro to Python for Data Science](#) course, where you learn by doing.

Variable Assignment <pre>>>> x=5 >>> x 5</pre>	<pre>>>> my_list[1] >>> my_list[-3] Slice >>> my_list[1:3] >>> my_list[1:] >>> my_list[:3] >>> my_list[:] Subset Lists of Lists >>> my_list2[1][0] >>> my_list2[1][:2] my_list[list][itemOfList]</pre>	Select item at index 1 Select 3rd last item Select items at index 1 and 2 Select items after index 0 Select items before index 3 Copy my_list
Calculations With Variables <pre>>>> x+2 7 >>> x-2 3 >>> x*2 10 >>> x**2 25 >>> x%2 1 >>> x/float(2) 2.5</pre>	Sum of two variables Subtraction of two variables Multiplication of two variables Exponentiation of a variable Remainder of a variable Division of a variable	
Types and Type Conversion <pre>str() int() float() bool()</pre>	'5', '3.45', 'True' 5, 3, 1 5.0, 1.0 True, True, True	Variables to strings Variables to integers Variables to floats Variables to booleans
Asking For Help <pre>>>> help(str)</pre>	List Operations <pre>>>> my_list + my_list ['my', 'list', 'is', 'nice', 'my', 'list', 'is', 'nice'] >>> my_list * 2 ['my', 'list', 'is', 'nice', 'my', 'list', 'is', 'nice'] True</pre>	Get the index of an item Count an item Append an item at a time Remove an item Remove an item Reverse the list Append an item Remove an item Insert an item Sort the list
Strings <pre>>>> my_string = 'thisStringIsAwesome' >>> my_string 'thisStringIsAwesome'</pre>	String Operations <pre>>>> my_string * 2 'thisStringIsAwesomethisStringIsAwesome' >>> my_string + 'Innit' 'thisStringIsAwesomeInnit' >>> 'm' in my_string True</pre>	String Methods <pre>>>> my_string.upper() String to uppercase >>> my_string.lower() String to lowercase >>> my_string.count('w') Count String elements >>> my_string.replace('e', 'i') Replace String elements >>> my_string.strip() Strip whitespaces</pre>

Leading open data science platform powered by Python

Free IDE that is included with Anaconda

Create and share documents with live code, visualizations, text, ...

Numpy Arrays <pre>>>> my_list = [1, 2, 3, 4] >>> my_array = np.array(my_list) >>> my_2darray = np.array([[1,2,3],[4,5,6]])</pre>	Also see Lists
Selecting Numpy Array Elements <pre>>>> my_array[1] 2 >>> my_array[0:2] array([1, 2]) >>> my_2darray[:,0] array([1, 4])</pre>	Index starts at 0 Select item at index 1 Select items at index 0 and 1 my_2darray[rows, columns]
Numpy Array Operations <pre>>>> my_array > 3 array([False, False, False, True], dtype=bool) >>> my_array * 2 array([2, 4, 6, 8]) >>> my_array + np.array([5, 6, 7, 8]) array([6, 8, 10, 12])</pre>	
Numpy Array Functions <pre>>>> my_array.shape >>> np.append(other_array) >>> np.insert(my_array, 1, 5) >>> np.delete(my_array, [1]) >>> np.mean(my_array) >>> np.median(my_array) >>> my_array.corrcoeff() >>> np.std(my_array)</pre>	Get the dimensions of the array Append items to an array Insert items in an array Delete items in an array Mean of the array Median of the array Correlation coefficient Standard deviation

DataCamp
Learn Python for Data Science Interactively

You can [Download Pdf here](#)

2. Python Cheat Sheet for Data Science: Basics

This cheat sheet is the companion to free [Python Programming Beginner Course](#) course offered by Dataquest which can start you on your data science journey.

You'll be able to practice reading files, Strings, Numeric Types, and Mathematical Operations, Lists and Dictionaries, Modules and Functions, Boolean Comparisons,



DATAQUEST

Data Science Cheat Sheet

Python Basics

BASICS, PRINTING AND GETTING HELP

`x = 3` - Assign 3 to the variable `x` `help(x)` - Show documentation for the `str` data type
`print(x)` - Print the value of `x` `help(print)` - Show documentation for the `print()` function
`type(x)` - Return the type of the variable `x` (in this case, `int` for integer)

READING FILES

```
f = open("my_file.txt", "r")
file_as_string = f.read()
- Open the file my_file.txt and assign its
  contents to s

import csv
f = open("my_dataset.csv", "r")
csvreader = csv.reader(f)
csv_as_list = list(csvreader)
- Open the CSV file my_dataset.csv and assign its
  data to the list of lists csv_as_list
```

STRINGS

```
s = "hello" - Assign the string "hello" to the
             variable s

s = """She said,
there's a good idea."""
- Assign a multi-line string to the variable s. Also
  used to create strings that contain both " and '
  characters

len(s) - Return the number of characters in s

s.startswith("hel") - Test whether s starts with
                    the substring "hel"

s.endswith("lo") - Test whether s ends with the
                 substring "lo"

"{ } plus { } is { }".format(3,1,4) - Return the
string with the values 3, 1, and 4 inserted

s.replace("e", "z") - Return a new string based
on s with all occurrences of "e" replaced with "z"

s.split(" ") - Split the string s into a list of
strings, separating on the character " " and
return that list
```

NUMERIC TYPES AND MATHEMATICAL OPERATIONS

```
i = int("5") - Convert the string "5" to the
              integer 5 and assign the result to i

f = float("2.5") - Convert the string "2.5" to
                 the float value 2.5 and assign the result to f

5 + 5 - Addition
5 - 5 - Subtraction
10 / 2 - Division
5 * 2 - Multiplication
```

```
3 ** 2 - Raise 3 to the power of 2 (or 32)
27 ** (1/3) - The 3rd root of 27 (or 3√27)
x += 1 - Assign the value of x + 1 to x
x -= 1 - Assign the value of x - 1 to x
```

LISTS

```
l = [100, 21, 88, 3] - Assign a list containing the
                     integers 100, 21, 88, and 3 to the variable l

l = list() - Create an empty list and assign the
           result to l

l[0] - Return the first value in the list l
l[-1] - Return the last value in the list l
l[1:3] - Return a slice (list) containing the second
        and third values of l

len(l) - Return the number of elements in l
sum(l) - Return the sum of the values of l
min(l) - Return the minimum value from l
max(l) - Return the maximum value from l

l.append(16) - Append the value 16 to the end of l
l.sort() - Sort the items in l in ascending order
"".join(["A", "B", "C", "D"]) - Converts the list
["A", "B", "C", "D"] into the string "A B C D"
```

DICTIONARIES

```
d = {"CA": "Canada", "GB": "Great Britain",
     "IN": "India"} - Create a dictionary with keys of
"CA", "GB", and "IN" and corresponding values
of "Canada", "Great Britain", and "India"

d["GB"] - Return the value from the dictionary d
         that has the key "GB"

d.get("AU", "Sorry") - Return the value from the
                      dictionary d that has the key "AU", or the string
                      "Sorry" if the key "AU" is not found in d

d.keys() - Return a list of the keys from d
d.values() - Return a list of the values from d
d.items() - Return a list of (key, value) pairs
           from d
```

MODULES AND FUNCTIONS

The body of a function is defined through indentation.

```
import random - Import the module random
from math import sqrt - Import the function
sqrt from the module math
```

```
def calculate(addition_one, addition_two,
              exponent=1, factor=1):
    result = (value_one + value_two) ** exponent * factor
    return result
```

- Define a new function `calculate` with two required and two optional named arguments which calculates and returns a result.

`addition(3, 5, factor=10)` - Run the `addition` function with the values 3 and 5 and the named argument `10`

BOOLEAN COMPARISONS

```
x == 5 - Test whether x is equal to 5
x != 5 - Test whether x is not equal to 5
x > 5 - Test whether x is greater than 5
x < 5 - Test whether x is less than 5
x >= 5 - Test whether x is greater than or equal to 5
x <= 5 - Test whether x is less than or equal to 5
x == 5 or name == "alfred" - Test whether x is
                             equal to 5 or name is equal to "alfred"
x == 5 and name == "alfred" - Test whether x is
                             equal to 5 and name is equal to "alfred"
5 in l - Checks whether the value 5 exists in the list l
"GB" in d - Checks whether the value "GB" exists in
           the keys for d
```

IF STATEMENTS AND LOOPS

The body of if statements and loops are defined through indentation.

```
if x > 5:
    print("{} is greater than five".format(x))
elif x < 0:
    print("{} is negative".format(x))
else:
    print("{} is between zero and five".format(x))
- Test the value of the variable x and run the code
  body based on the value
```

```
for value in l:
    print(value)
- Iterate over each value in l, running the code in
  the body of the loop with each iteration
```

```
while x < 10:
    x += 1
- Run the code in the body of the loop until the
  value of x is no longer less than 10
```

3. Beginner's Python Cheat Sheet

This Python Cheat is from the Book [Python Crash Course](#) which aims to remind you of syntax rules and grasp all the important concepts in Python programming as a beginner.

You will also create a game with Pygame, Create Data Visualization with PyGal and build Web Apps with Django.

4. Python Cheat Sheet for Data Science: Intermediate

This cheat sheet assumes you are familiar with the content of the Python Basic Cheat Sheet from Dataquest.

This Python cheat sheet provides in-depth focus on Lists, Strings, Range, Dictionaries, Sets, Regular Expressions, List Comprehension, Functions for Looping, DateTime, Random, Counter and Try Except.

You can **Download** [here](#)

5. Importing Data in Python Cheat Sheet

This Python Cheat Sheet from Datacamp provides everything that you need to kickstart your data science learning with Python.

Moreover, you'll have a handy reference guide to importing your data, from flat files to files native to other software, and relational databases.

You'll also learn how you can get data from files native to other software such as Excel spreadsheets, Stata, SAS and MATLAB files and relational databases.

6. Python NumPy Cheat Sheet

This cheat sheet assumes you are familiar with NumPy. If you're interested in learning NumPy, you can start learning about NumPy in [Python Data Science Course](#) from Dataquest.

This Python Numpy Cheat Sheet will make you familiar with NumPy Array and how you can Import and Export Data for analysis. You'll also index data and retrieve results, using NumPy with Scalar Math, Vector Math, and Statistics will hold no secrets for you any longer.

You can **Download here**

7. Python Data Visualization: Bokeh Cheat Sheet

This Python Cheat Sheet will guide you to interactive plotting and statistical charts with Bokeh. Python Bokeh Cheat Sheet is a free additional material for [Interactive Data Visualization with Bokeh](#) Course and is a handy one-page reference for those who need an extra push to get started with Bokeh.

This cheat sheet will walk you through making beautiful plots and also introduce you to the basics of statistical charts.

8. Python for Data Science: Pandas Cheat Sheet

Pandas is a data-centric Python package. It's common when first learning pandas to have trouble remembering all the functions and methods that you need, and it's nice to have a handy reference.

We hope this cheat sheet will help you out! If you are interested in learning, you can signup for free and start learning Pandas for [Data Science Course](#) offered by Dataquest.

You can **Download** [here](#)

Thanks for making it to the end 😊


Share & Learn! Do add your favourite Python Cheat Sheet in the comments below.

You may also be interested in checking the list of [Machine Learning Cheat Sheets](#) in Python and Maths or listening to a [Python Podcast](#) to bootstrap your knowledge in Python.


If you liked this article, i've got few practical reads for you.

- [Learning how to learn Python, Statistics and Maths for Data Science](#)
- [7 Best \(and Affordable !!!\) Data Science Specializations](#)
- [5 Must have Skills in Python for Every Data Scientist](#)

“ I've also got this [Data Science Newsletter](#) that you might be into. I send a tiny email once or twice every quarter with some useful resource I've found.

Don't worry, I hate spam as much as you. Feel free to subscribe. 

Banner Image Credits: <https://uxdesign.cc/ui-cheat-sheet-text-fields-2152112615f8>

[Download a PDF version of this post](#) Filed Under: [Python](#) Tagged With: [Bokeh Cheat Sheet](#), [Data Visualization Cheat Sheet](#), [Data Visualization Cheat Sheet Python](#), [Numpy Cheat Sheet](#), [Pandas Cheat Sheet](#), [Python Basics Cheat Sheet](#), [Python Bokeh Cheat Sheet](#), [Python Cheat Sheet](#), [Python Cheat Sheet Advanced](#), [Python Cheat Sheet Basics](#), [Python Cheat Sheet Beginner](#), [Python Cheat Sheet Expert](#), [Python Cheat Sheets](#), [Python Data Science Cheat Sheet](#), [Python Data Visualization Cheat Sheet](#), [Python Import Data Cheat Sheet](#), [Python Numpy Cheat Sheet](#)

Stay In The Know!

[GO](#)

join our mailing list to get data science updates once every
fortnight!

Spot a mistake or want to help improve this page? **Please let us
know.**

LOAD COMMENTS

Copyright | © 2019 sinxLoud. All rights reserved.

[TERMS](#) — [PRIVACY](#) — [DMCA](#) — [AFFILIATE DISCLOSURE](#) — [COPYRIGHT](#)