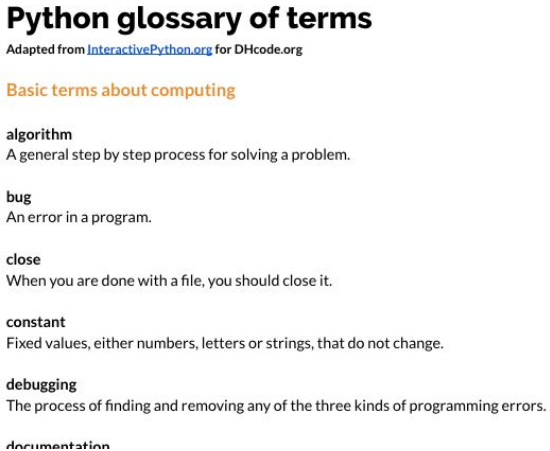
DHcode.org

resources | debugging help

Python + Pandas

Glossary of Terms for Python and Pandas



Python glossary of terms

Adapted from InteractivePython.org for DHcode.org

Basic terms about computing

algorithm
A general step by step process for solving a problem.

bug
An error in a program.

close
When you are done with a file, you should close it.

constant
Fixed values, either numbers, letters or strings, that do not change.

debugging
The process of finding and removing any of the three kinds of programming errors.

documentation

Where to find help

For questions about why to do something:

https://www.digitalocean.com/community/tutorial_series/how-to-code-in-python-3

For questions about how to do something:

https://stackoverflow.com/questions/tagged/python-3
https://www.reddit.com/r/learnpython/wiki/index

For practice (why + how):

https://learnpythonthehardway.org/book/ <<< this is free!
http://introtopython.org/hello_world.html <<< this is free!

Google search tips

- use Python3 , “Python 3” or Python-3 as a keyword
  this helps filter out most (but not all) python 2 advice
- use computational language when you can
- use plain language if you must

computational language

Google

python 3 string len

All Videos Images Shopping News More Settings Tools

About 9,330,000 results (0.40 seconds)

Python 3 String len() Method - Tutorialspoint
https://www.tutorialspoint.com/python3/string_len.htm
Python 3 String len() Method - Learning Python 3 in simple and easy steps : A beginner's tutorial containing complete knowledge of Python Syntax Object ...

Python String len() Method - Tutorialspoint
https://www.tutorialspoint.com/python/string_len.htm
Python String len() Method - Learn **Python** in simple and easy steps starting from basic to advanced concepts with examples including **Python** Syntax Object ...

How to get the size of a string in Python? - Stack Overflow
https://stackoverflow.com/questions/.../how-to-get-the-size-of-a-string-in-python
Feb 11, 2011 - A. The **len**() function in **Python** 2 returns count of bytes allocated to store ... print(**len**('йцы'.decode('utf8'))) **#String** contains Cyrillic symbols **3**.

string length - **Python 3**.x - don't count carriage returns ...	2 answers	Sep 18, 2017
python - Using function **len**() in **python3**	3 answers	Aug 10, 2016
Python 3 variable in **string.len** function	1 answer	Sep 1, 2014
Is there a reason **Python strings** don't have a **string** ...	8 answers	Oct 25, 2008

More results from stackoverflow.com

2. Built-in Functions — Python 3.6.5 documentation
https://docs.python.org/3/library/functions.html
As repr() , return a **string** containing a printable representation of an object, but escape the non-ASCII ..... Return the **length** (the number of items) of an object.

Python string length | len() - GeeksforGeeks
https://www.geeksforgeeks.org/python-string-length-len/
Python string length | **len**(). **len**() function is an inbuilt function in **Python** programming language that returns the **length** of the **string**. Syntax: **len(string)**.

plain language

Google

python 3 how to find the length of a string

All Videos Shopping Images News More Settings Tools

About 9,150,000 results (0.44 seconds)

Python String len() Method - Tutorialspoint
https://www.tutorialspoint.com/python/string_len.htm
Python String len() Method - Learn **Python** in simple and easy steps starting from basic to advanced concepts with **examples** including **Python** Syntax Object ...

Python 3 String len() Method - Tutorialspoint
https://www.tutorialspoint.com/python3/string_len.htm
Python 3 String len() Method - Learning **Python 3** in simple and easy steps : A beginner's tutorial containing complete knowledge of **Python** Syntax Object ...

How to get the size of a string in Python? - Stack Overflow
https://stackoverflow.com/questions/.../how-to-get-the-size-of-a-string-in-python
Feb 11, 2011 - A. The **len**() function in **Python** 2 returns count of bytes allocated to store ... print(**len**('йцы'.decode('utf8'))) **#String** contains Cyrillic symbols **3**.

python - how can I **check the length** of a list?	3 answers	Apr 7, 2017
Python : **Get size** of **string** in bytes	4 answers	Jun 6, 2015
python - How to **get the length** of words in a sentence?	4 answers	Mar 30, 2014
python - **String length** without **len** function	13 answers	Oct 21, 2010

More results from stackoverflow.com

Python string length | len() - GeeksforGeeks
https://www.geeksforgeeks.org/python-string-length-len/
Python string length | **len**(). **len**() function is an inbuilt function in **Python** programming language that returns the **length** of the **string**. Syntax: **len(string)**.

Finding the length of a Python String - YouTube

https://www.youtube.com/watch?v=5VkAlishHc0
Oct 1, 2014 - Uploaded by John Philip Jones
Shows how to **find the length** of a **Python String**. ... **Python 3** Programming Tutorial [18] **Len**() Function and ...

Error messages

Adapted from interactivepython.org

Parse error

Parse errors happen when you make an error in the syntax of your program.

Type error

Type Errors occur when you you try to combine two objects that are not compatible.

Name error

Name errors almost always mean that you have used a variable before it has a value.

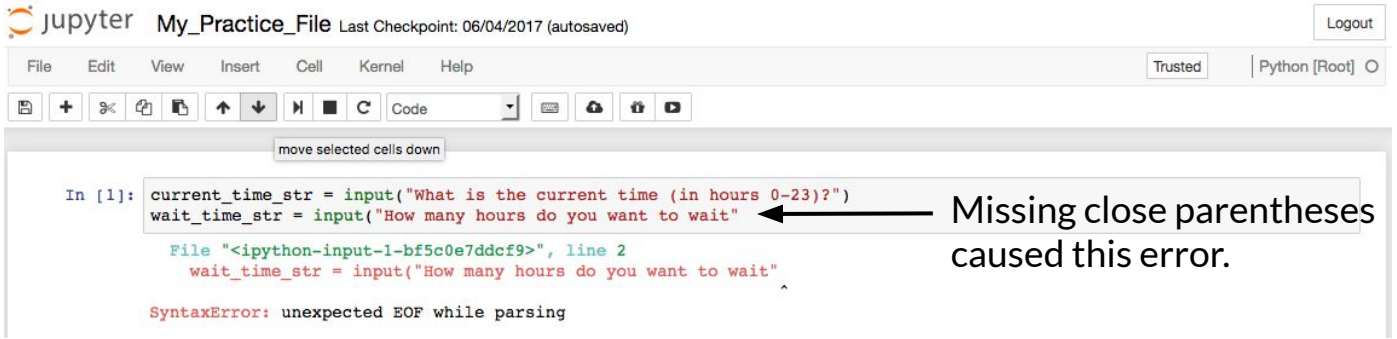
Value error

Value errors occur when the variable expects the value to be a certain data type but it gets something different.

Parse error

Parse errors happen when you make an error in the syntax of your program.

Syntax errors are like making grammatical errors in writing. If you don't use periods and commas in your writing then you are making it hard for other readers to figure out what you are trying to say. Code also has grammatical rules that must be followed or else the program can't figure out what you are trying to say. Usually Parse Errors can be traced back to missing punctuation characters, such as parentheses, quotation marks, or commas.



Type error

Type Errors occur when you you try to combine two objects that are not compatible.

For example you try to add together an integer and a string. Usually type errors can be isolated to lines that are using mathematical operators, and usually the line number given by the error message is an accurate indication of the line.

```
In [19]: number = str(input("What's your favorite number? "))
         What's your favorite number? 7

In [20]: print(number)
         7

In [21]: print(number + 1)
         ---------------------------------------------------------------------------
         TypeError                                 Traceback (most recent call last)
         <ipython-input-21-3ad155d8df06> in <module>()
         ----> 1 print(number + 1)

         TypeError: Can't convert 'int' object to str implicitly
```

Name error

Name errors almost always mean that you have used a variable before it has a value.

Often Name Errors are simply caused by typos in your code. They can be hard to spot if you don't have a good eye for catching spelling mistakes. Other times you may simply mis-remember the name of a variable or even a function you want to call. Or you didn't give the value a variable name yet.

```
In [12]: my_lucky_number = 7
         your_lucky_number = four
         our_lucky_numbers = my_lucky_number + your_lucky_number

---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-12-293e70a8be42> in <module>()
      1 my_lucky_number = 7
----> 2 your_lucky_number = four
      3 our_lucky_numbers = my_lucky_number + your_lucky_number

NameError: name 'four' is not defined
```

The value here is missing the quote marks so it is not written correctly as a string.

But watch out! The next line will also throw an error. The code is setup to add the two numbers. Strings and ints are different data types and can't be added.

Value error

Value errors occur when the variable expects the value to be a certain data type but it gets something different. This tends to happen when you're asking for user input but did not include specific enough directions about how to enter the data.

Be sure to comment your code, printing very explicit instructions and/or being mindful of how you name the variables which will remind you about what types of values to expect.

```
In [25]: current_time_str = input("What is the current time (in hours 0-23)? ")
         What is the current time (in hours 0-23)? 12:34

In [26]: current_time_int = int(current_time_str)
         ---------------------------------------------------------------------------
         ValueError                                Traceback (most recent call last)
         <ipython-input-26-380a22bb36b3> in <module>()
         ----> 1 current_time_int = int(current_time_str)

         ValueError: invalid literal for int() with base 10: '12:34'


In [27]: current_time_str = input("What is the current time (in hours 0-23)? ")
         What is the current time (in hours 0-23)? 10

In [28]: current_time_int = int(current_time_str)
```

This input is correct for how we think about time but does not follow the prompt.

This input does follow the prompt, so it works.