BINF2111 – Introduction to Bioinformatics Computing

Python – you have the power!

Richard Allen White III, PhD
RAW Lab
Lecture 16 – Thursday Oct 28th, 2021

Learning Objectives

- Writing Scripts

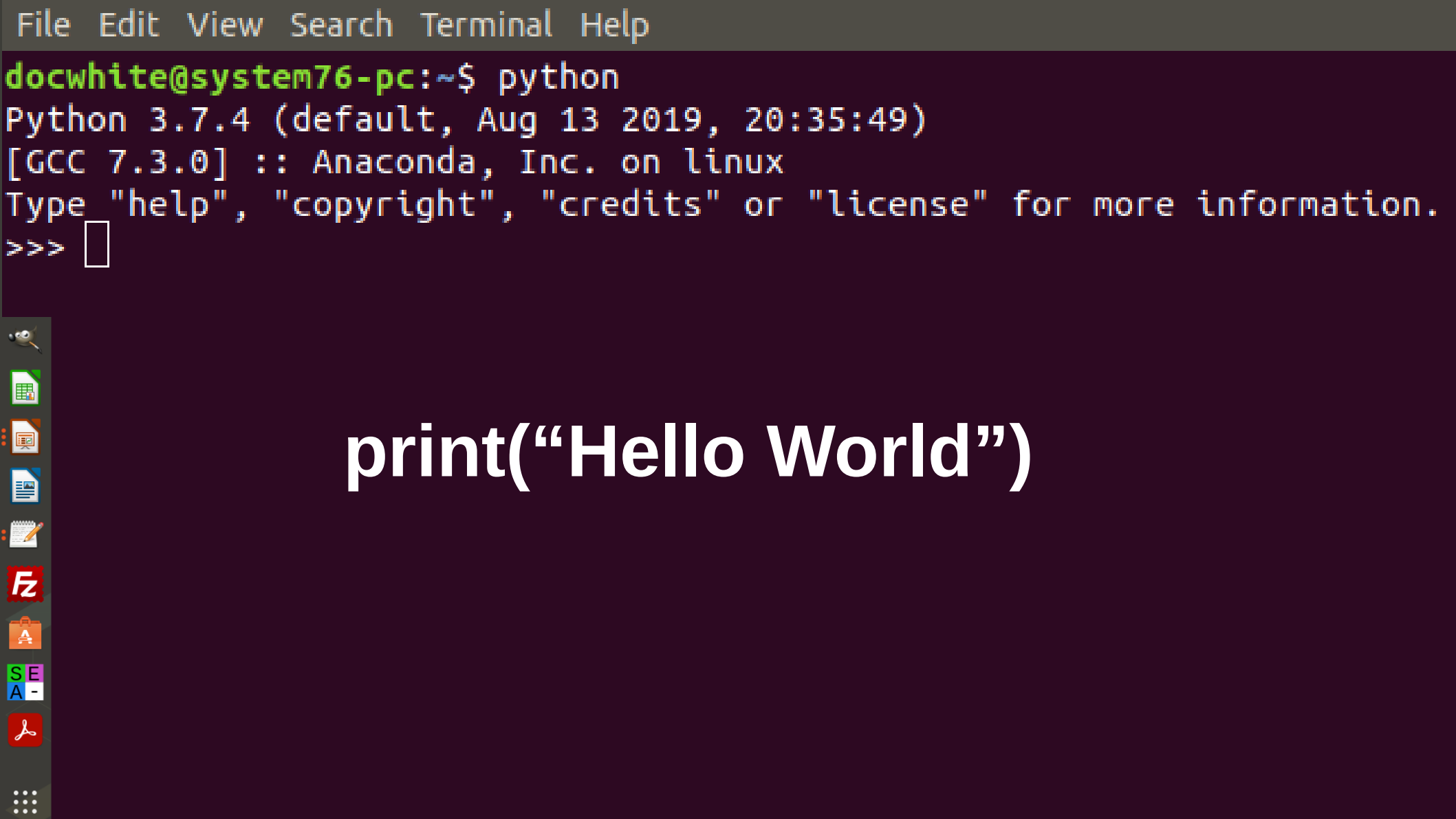
- Sting operations

- Replace/slicing

- Lists

- Conda installing?

- Quiz 15


File Edit View Search Terminal Help
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
print("Hello World")

Writing scripts

All scripts start with a #! (Shebang line)

#!/usr/bin/env python (what is it for bash or shell)

This tells the script where python is located. If you don’t do this, python has to be invoked manually

python <script>

<script.py>

Reserved words in Python

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield

Python Variables

Variables – variables allow you to store values within data types

Data types:

Numbers

Strings

List (Grocery List, can change)

Tuple (Fixed List Like Months, Jan… Feb… etc.)

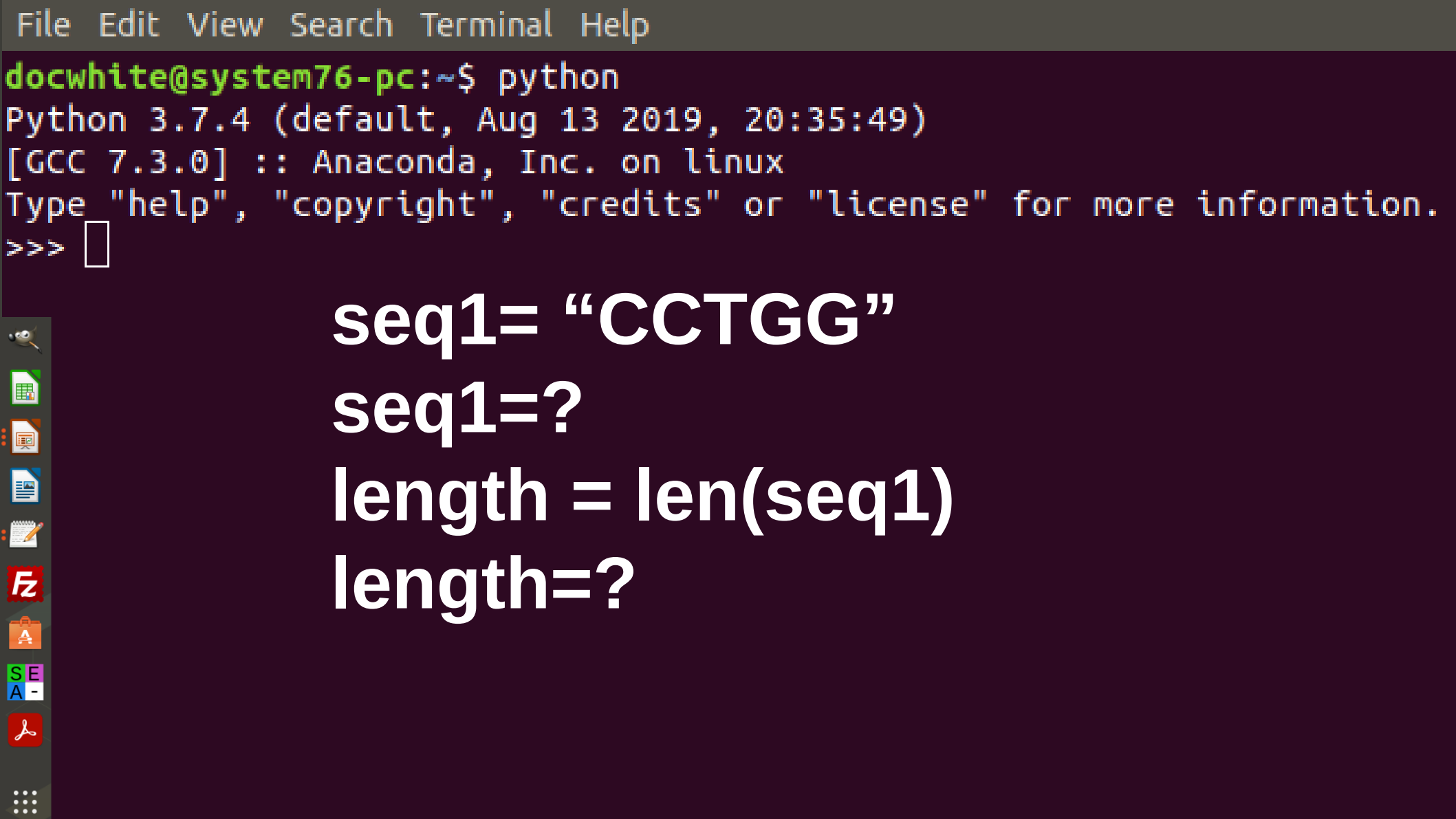
Dictionary (Indexed List with Key>Value)

Python Functions vs. Methods

Functions – global, takes an argument, e.g. len(“string”)

Methods – specifically bound to an object an can also take an argument e.g. seq.count(“T”)

Both methods and functions can be limited in the kind of variables they handle - i.e. both examples above apply only to strings.

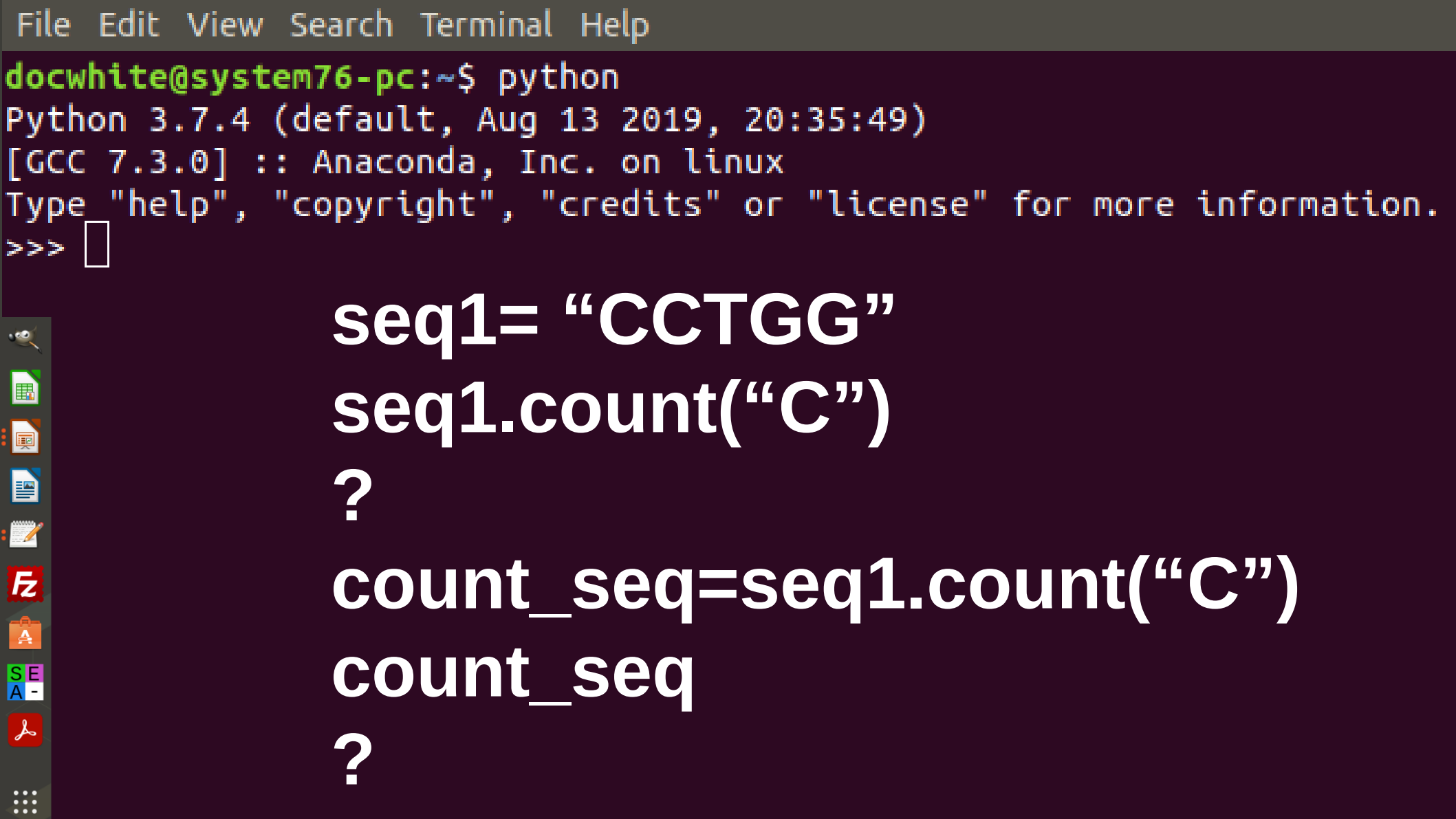

File Edit View Search Terminal Help
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
seq1= “CCTGG”
seq1=?
length = len(seq1)
length=?

Python String operators

String Operators – String operators to manipulate variables.

count – counts instances of a substring in a string

Variable.count(“pattern”)


File Edit View Search Terminal Help
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
seq1= “CCTGG”
seq1.count(“C”)
?
count_seq=seq1.count(“C”)
count_seq
?

Python String - Replace

String Methods – String methods to manipulate variables.

replace – replaces one substring with another. CASE SENSITIVE

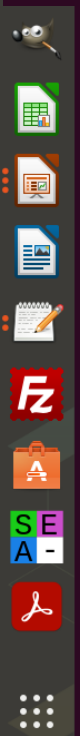
Variable.replace('pattern','pattern2')

Replace is a method

To use the result of Variable.replace(), assign its value to a variable

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```



```
seq1.replace('T','U')
?
re_seq=seq1.replace('T','U')
re_seq
?
```

Python String – Case change

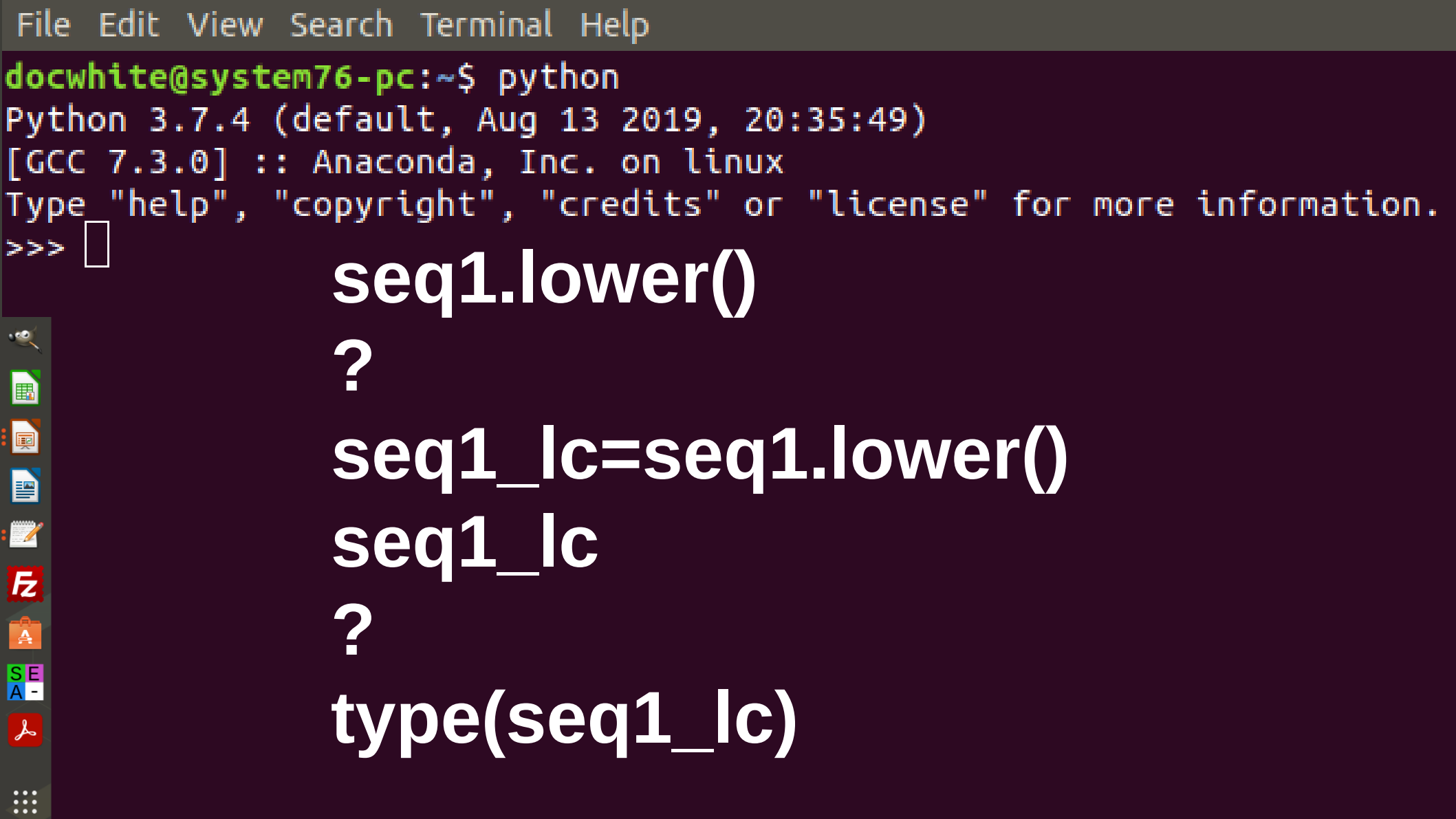
String Methods – String methods to manipulate variables.

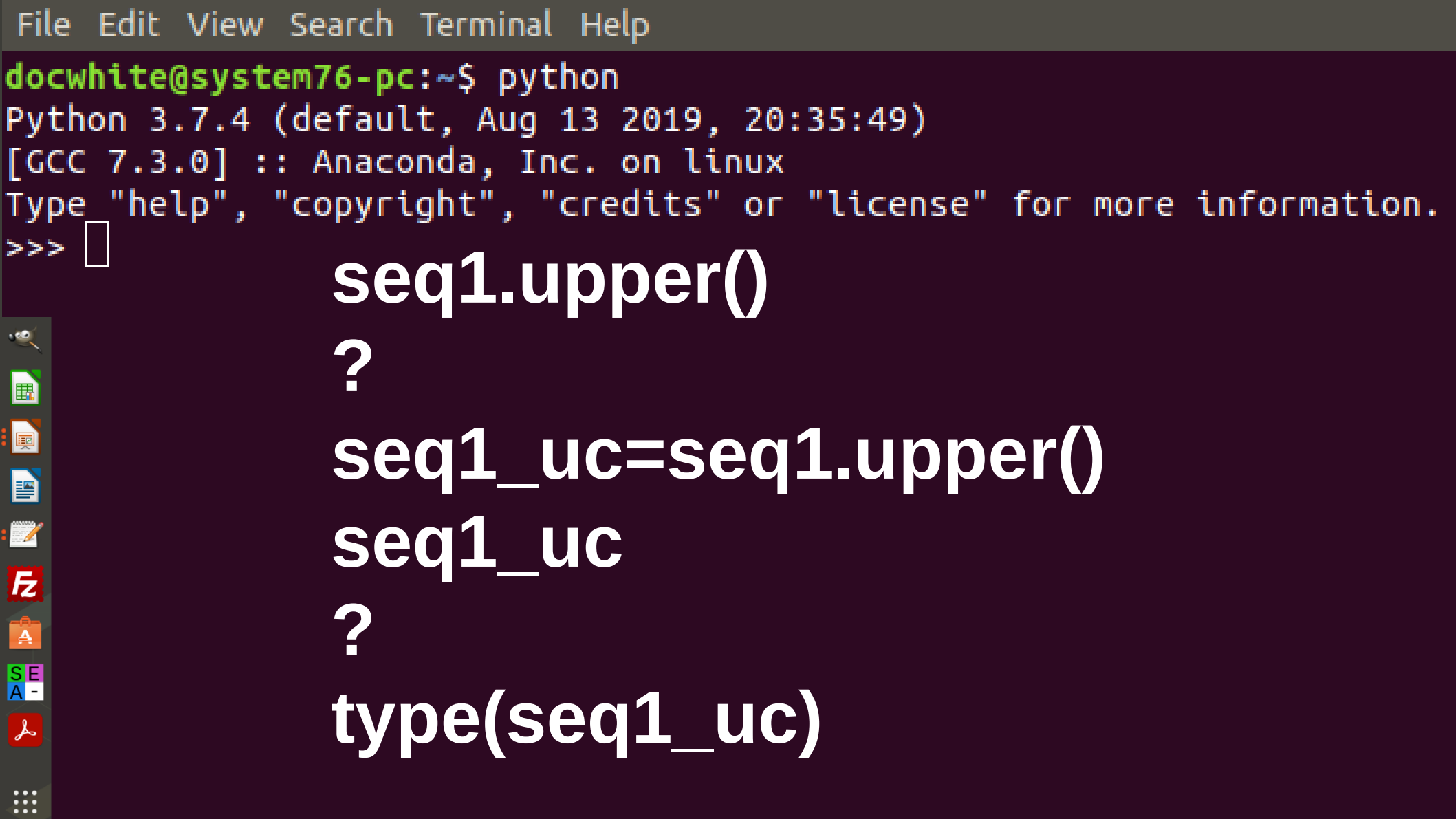
UPPER, lower – converts strings to upper or lowercase

Variable.lower() or Variable.upper()

lower/upper is a method

To use the result of Variable.lower(), assign its value to a variable


File Edit View Search Terminal Help
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
seq1.lower()
?
seq1_lc=seq1.lower()
seq1_lc
?
type(seq1_lc)


File Edit View Search Terminal Help
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
seq1.upper()
?
seq1_uc=seq1.upper()
seq1_uc
?
type(seq1_uc)

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
print(seq1_uc[2:4])
?
print(seq1_uc[1:3])
?
type(seq1_uc)
?
```

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
name = 'Jose Figueroa'
name.startswith('J')
?
type(name.startswith('J'))
?
```

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
name = 'Jose Figueroa'
print(name.split())
?
type(name.split())
?
```

Python String – Split

string.split(“pattern”)[number]

```
>>> name = "Jose Figueroa"
>>> print(name.split())
['Jose', 'Figueroa']
   0       1
```

What is this doing? Splits a string into individual elements at a defined separator (space in this case). This creates a list with elements 0 and 1.

Python String – Split

string.split(“pattern”)[number]

What is this doing? The line is split by the delimiter space using the **method** split. The **function** print addresses an individual element of the list created when the line is split. This is a slice of a list.

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
name = 'Jose Figueroa'
print(name.split(' ')[0])
?
print(name.split(' ')[1])
?
```

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
name = 'Jose Figueroa'
FirstFourChar = name[0:4]
FirstFourChar
?
print(FirstFourChar)
?
```

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

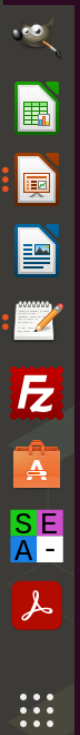
protein = ‘vlspadktnv’

#Convert to upper case

#Print the 3 to 5

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```



```
protein = 'vlspadktnv'

#Convert to upper case
print(protein.upper())

#Print the 3 to 5
print(protein[3:5]
```

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
protein = 'vlspadktnv'

#Print first residue


#Count the number of valine
```

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
protein = 'vlspadktnv'

#Print first residue
print(protein[0])

#Count the number of valine
print(protein.count('v'))
```

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

protein = ‘vlspadktnvw’

#Print valine count as strings

#Print tryptophan count as strings

File Edit View Search Terminal Help

```
docwhite@system76-pc:~$ python
Python 3.7.4 (default, Aug 13 2019, 20:35:49)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

```
#Print valine count as strings
valine_count = protein.count('v')
print("valine: " + str(valine_count))

#Print tryptophan count as strings
typ_count = protein.count('w')
print("tryptophan: " + str(typ_count))
```

Python list

What is a list?

Instead of assigning a different variable name to every value (ex. 10 different amino acids each to a different variable).

We can create a list of values. This list can expand to hold as many values as you need.

Data types:

Numbers

Strings

List

Tuple

Dictionary

Python list

What is a list?

A list is an object type that consists of comma separated objects such as numerical values and strings. (You can also make lists of lists).

We can split a line into a list of fields and access the elements we want.

We can also “loop” or iterate through a list in order to look through the list.

Aren't lists and arrays the same thing?

Sort of. Same but different. In python, just use lists.

Python list

What is a list?

A list is an object type that consists of comma separated objects such as numerical values and strings. (You can also make lists of lists).

We can split a line into a list of fields and access the elements we want.

We can also “loop” or iterate through a list in order to look through the list.

Aren't lists and arrays the same thing?

Sort of. Same but different. In python, just use lists.

Python list - create

How to create a list:

Lists can be created by setting the list name equal to a collection of comma separated strings or values

snplist = [“rs12913832”, “rs916977”, “rs1667394”]

numlist = [1, 2, 3, 5, 6000]

Python list - slice

How to print a slice of a list:

First, make some lists:

>>snplist = [“rs12913832”, “rs916977”, “rs1667394”]

>>numlist = [1, 2, 3, 5, 6000]

Then, print selected values – slice notation

>>print snplist[1]

>>print numlist[3:5]

Python list - slice

How to print a slice of a list:

First, make some lists:

>>snplist = [“rs12913832”, “rs916977”, “rs1667394”]

>>numlist = [1, 2, 3, 5, 6000]

Then, print selected values – slice notation

>>print snplist[1] #prints the 1st element (2nd item in list)

>>print numlist[3:5] #prints element 3,4 (4th and 5th item in list)

Remember a slice prints the elements [inclusive:exclusive]

[3:4] = element 3

[3:5] = element 3,4.

The indexing starts at 0 always.

Python list - Initialize

Initialize (or clear) a list:

```
listname = []
```

This will create a placeholder list for you to add to.

Methods like list.append() so you can put things into the list.

It will also empty out a list. This is nice when you have a very large list and you want to clear up memory.

Python list - Sort

Methods to reverse and sort a list:

Reversing and sorting changes the list permanently without reassigning to the list.

```
>>> ranks = ["kingdom","phylum","class","order","family"]
>>> print ranks
['kingdom', 'phylum', 'class', 'order', 'family']
>>> ranks.reverse()
>>> print ranks
['family', 'order', 'class', 'phylum', 'kingdom']
>>> ranks.sort()
>>> print ranks
['class', 'family', 'kingdom', 'order', 'phylum']
```

Quiz 15

- On canvas now