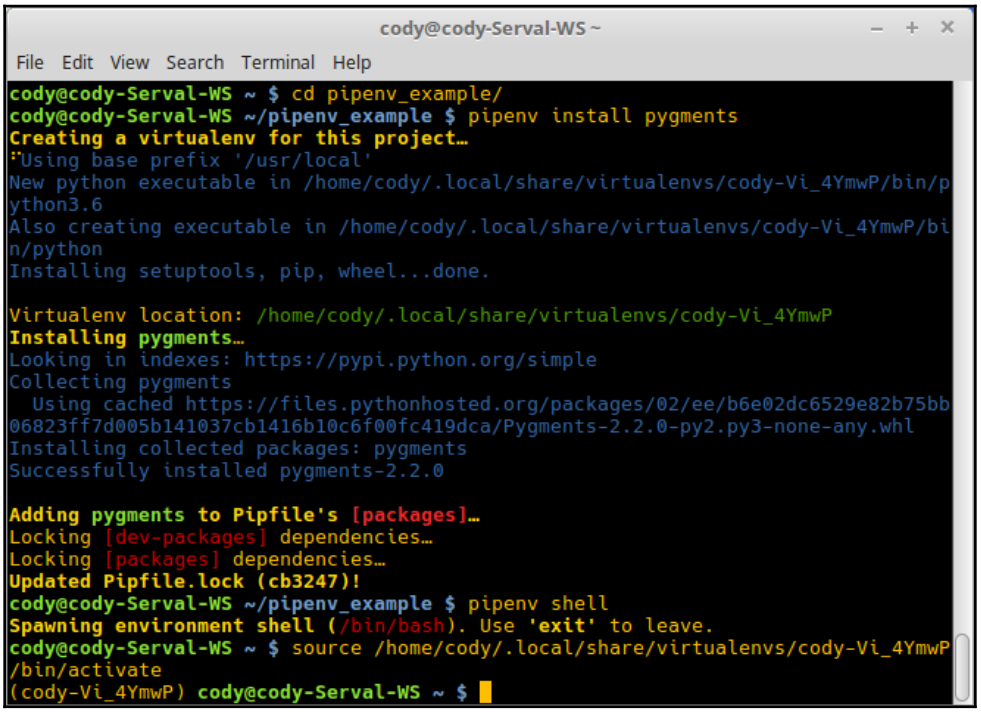
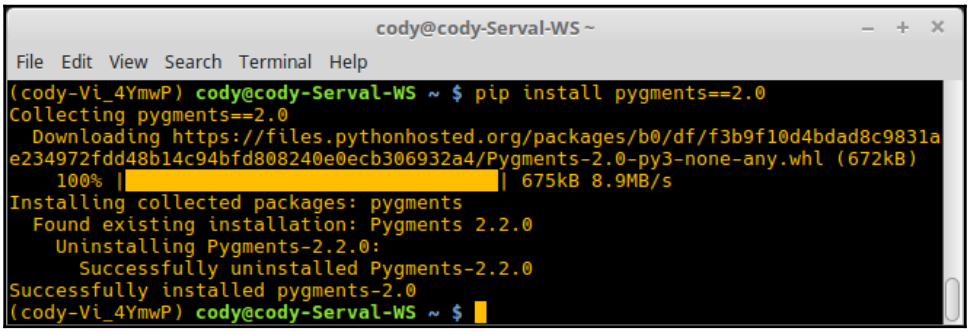
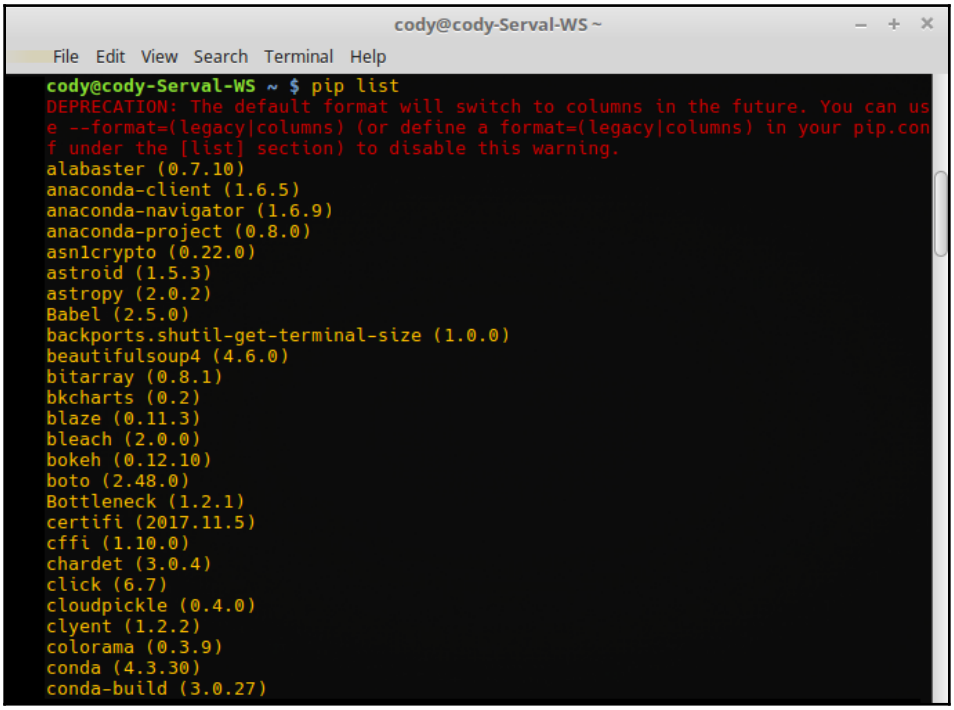
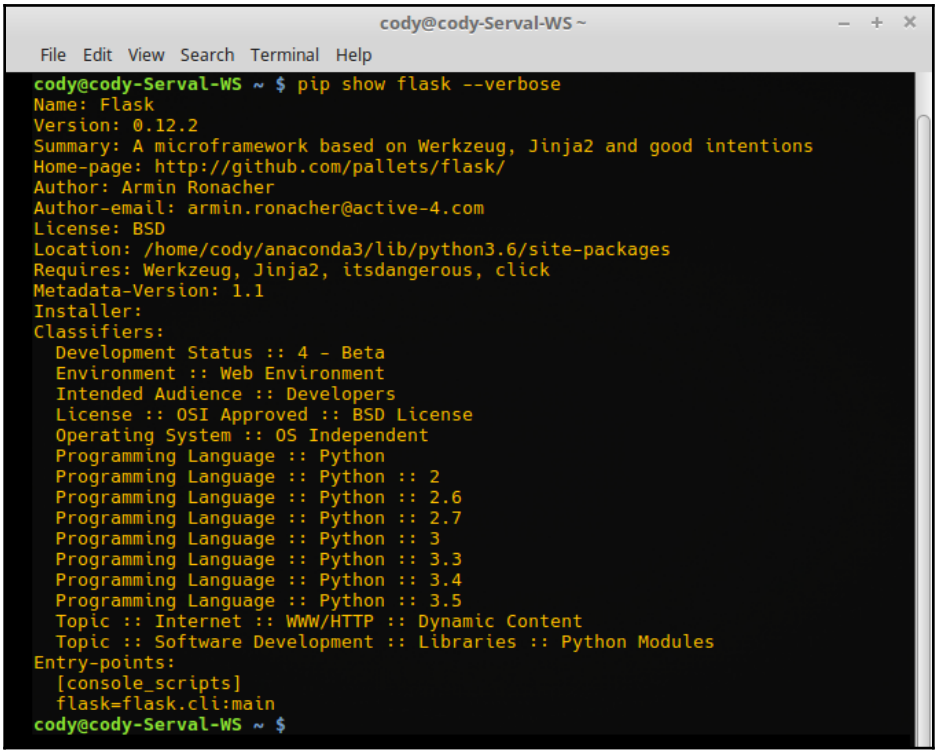
Chapter 1: Working with Python Modules





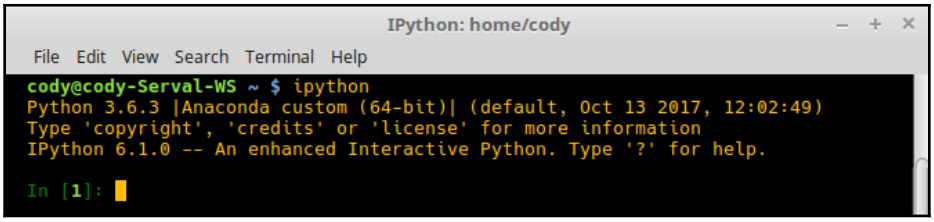
```c
#include <Python.h>

int
main(int argc, char *argv[])
{
    wchar_t *program = Py_DecodeLocale(argv[0], NULL);
    if (program == NULL) {
        fprintf(stderr, "Fatal error: cannot decode argv[0]\n");
        exit(1);
    }
    Py_SetProgramName(program);  /* optional but recommended */
    Py_Initialize();
    PyRun_SimpleString("from time import time,ctime\n"
                       "print('Today is', ctime(time()))\n");
    if (Py_FinalizeEx() < 0) {
        exit(120);
    }
    PyMem_RawFree(program);
    return 0;
}
```

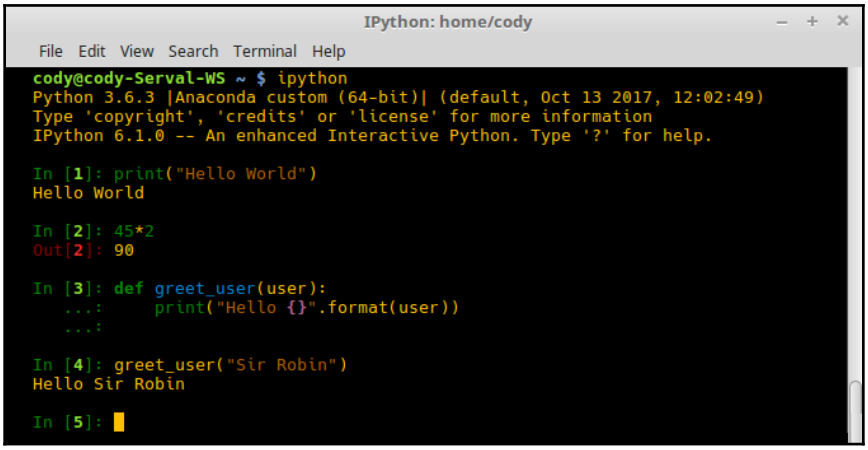


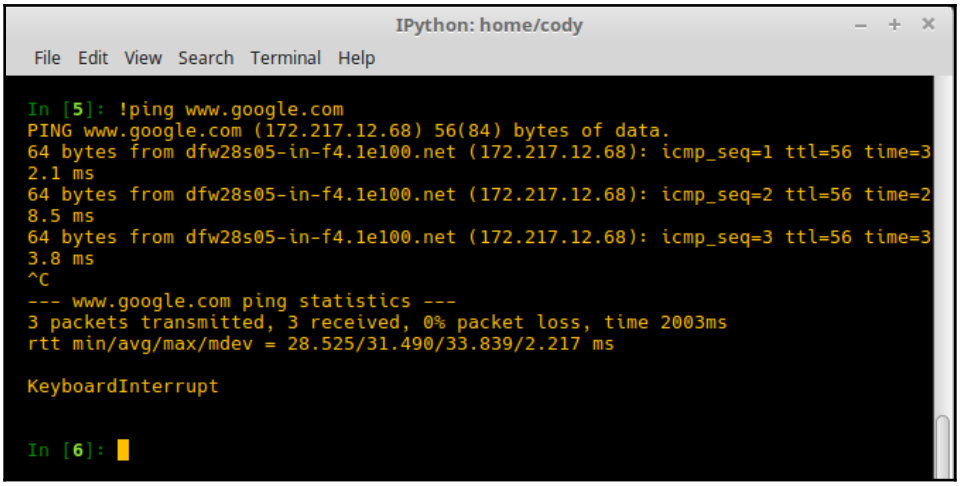

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ pip show flask --verbose
Name: Flask
Version: 0.12.2
Summary: A microframework based on Werkzeug, Jinja2 and good intentions
Home-page: http://github.com/pallets/flask/
Author: Armin Ronacher
Author-email: armin.ronacher@active-4.com
License: BSD
Location: /home/cody/anaconda3/lib/python3.6/site-packages
Requires: Werkzeug, Jinja2, itsdangerous, click
Metadata-Version: 1.1
Installer:
Classifiers:
Development Status :: 4 - Beta
Environment :: Web Environment
Intended Audience :: Developers
License :: OSI Approved :: BSD License
Operating System :: OS Independent
Programming Language :: Python
Programming Language :: Python :: 2
Programming Language :: Python :: 2.6
Programming Language :: Python :: 2.7
Programming Language :: Python :: 3
Programming Language :: Python :: 3.3
Programming Language :: Python :: 3.4
Programming Language :: Python :: 3.5
Topic :: Internet :: WWW/HTTP :: Dynamic Content
Topic :: Software Development :: Libraries :: Python Modules
Entry-points:
[console_scripts]
flask=flask.cli:main
cody@cody-Serval-WS ~ $

Chapter 2: Utilizing the Python Interpreter

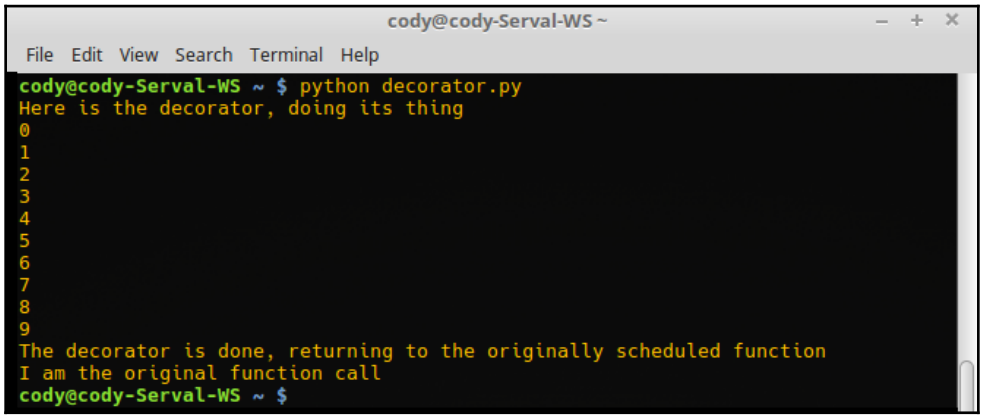
```c
#include <Python.h>

int
main(int argc, char *argv[])
{
    wchar_t *program = Py_DecodeLocale(argv[0], NULL);
    if (program == NULL) {
        fprintf(stderr, "Fatal error: cannot decode argv[0]\n");
        exit(1);
    }
    Py_SetProgramName(program);  /* optional but recommended */
    Py_Initialize();
    PyRun_SimpleString("from time import time,ctime\n"
                       "print('Today is', ctime(time()))\n");
    if (Py_FinalizeEx() < 0) {
        exit(120);
    }
    PyMem_RawFree(program);
    return 0;
}
```

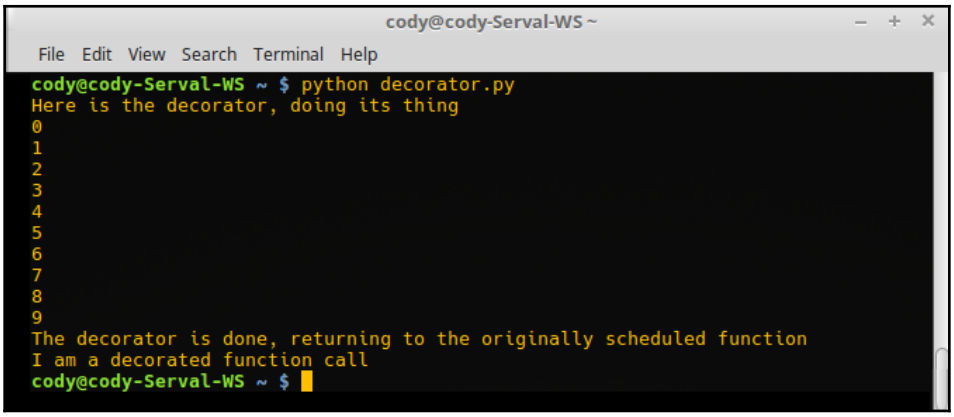


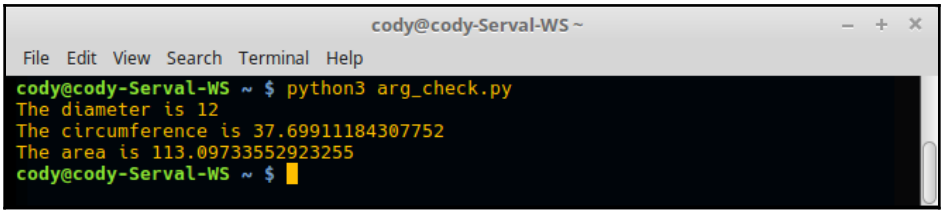


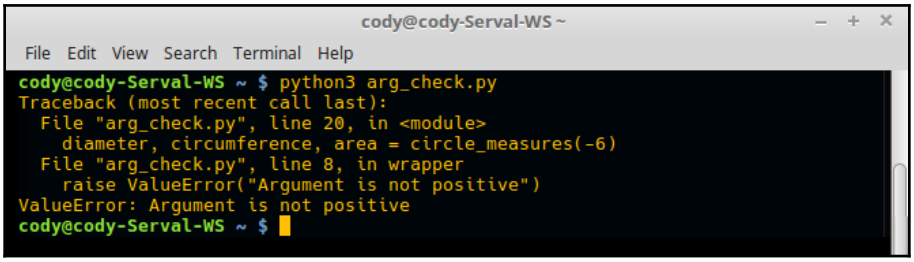

IPython: home/cody
File Edit View Search Terminal Help
In [5]: !ping www.google.com
PING www.google.com (172.217.12.68) 56(84) bytes of data.
64 bytes from dfw28s05-in-f4.1e100.net (172.217.12.68): icmp_seq=1 ttl=56 time=3
2.1 ms
64 bytes from dfw28s05-in-f4.1e100.net (172.217.12.68): icmp_seq=2 ttl=56 time=2
8.5 ms
64 bytes from dfw28s05-in-f4.1e100.net (172.217.12.68): icmp_seq=3 ttl=56 time=3
3.8 ms
^C
--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 28.525/31.490/33.839/2.217 ms
KeyboardInterrupt
In [6]:

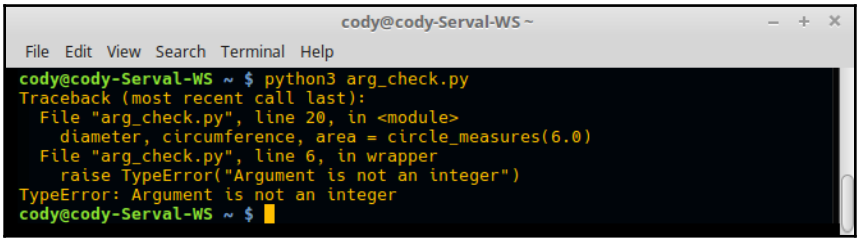
Chapter 3: Working with Decorators

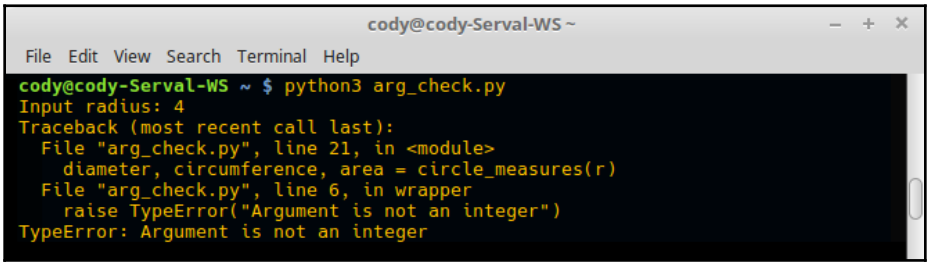


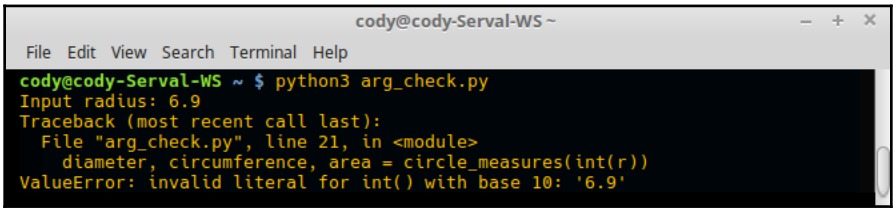




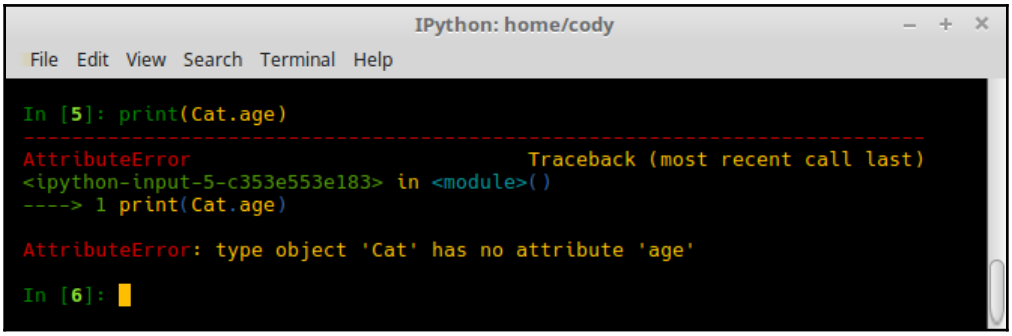

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3 arg_check.py
Traceback (most recent call last):
File "arg_check.py", line 20, in <module>
diameter, circumference, area = circle_measures(-6)
File "arg_check.py", line 8, in wrapper
raise ValueError("Argument is not positive")
ValueError: Argument is not positive
cody@cody-Serval-WS ~ $

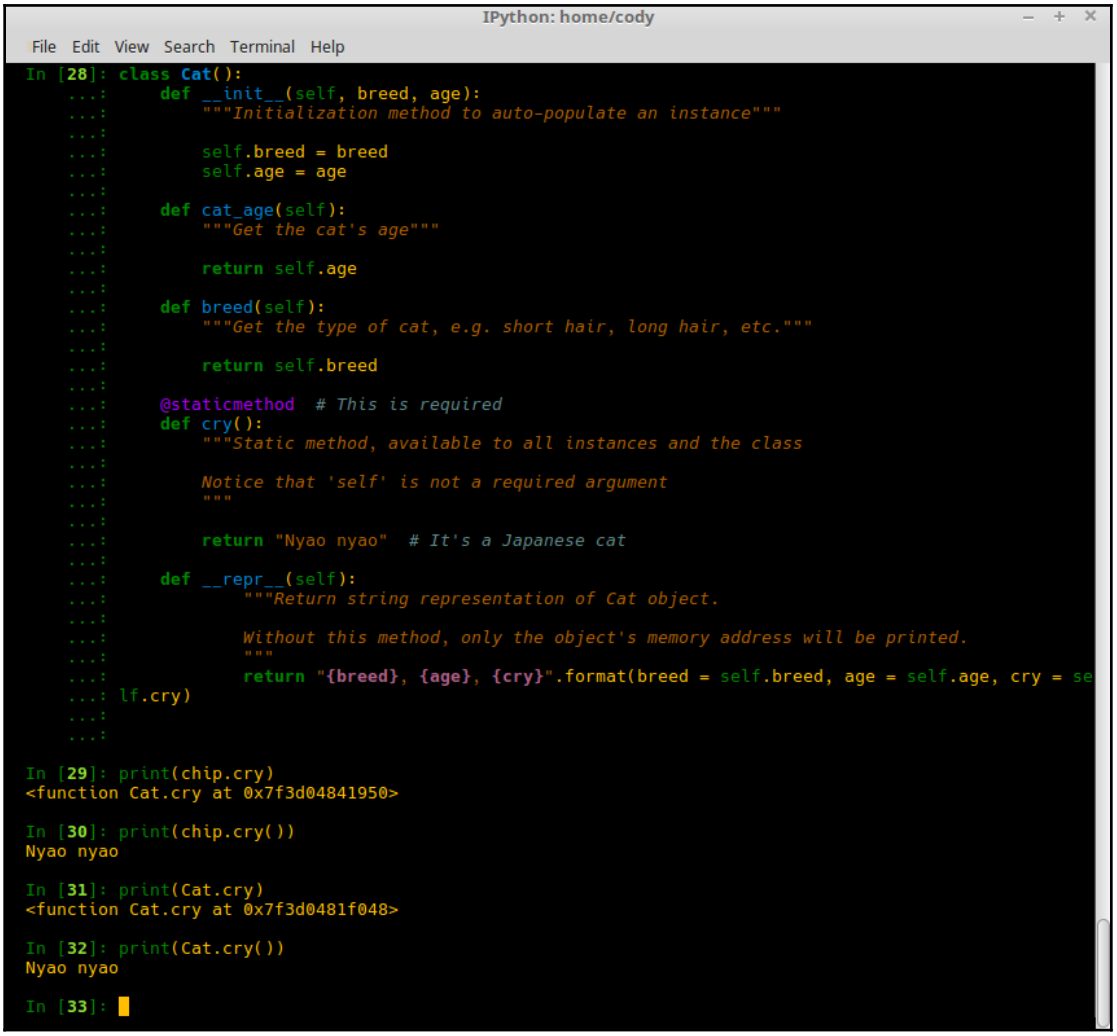

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3 arg_check.py
Traceback (most recent call last):
File "arg_check.py", line 20, in <module>
diameter, circumference, area = circle_measures(6.0)
File "arg_check.py", line 6, in wrapper
raise TypeError("Argument is not an integer")
TypeError: Argument is not an integer
cody@cody-Serval-WS ~ $


cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3 arg_check.py
Input radius: 4
Traceback (most recent call last):
File "arg_check.py", line 21, in <module>
diameter, circumference, area = circle_measures(r)
File "arg_check.py", line 6, in wrapper
raise TypeError("Argument is not an integer")
TypeError: Argument is not an integer

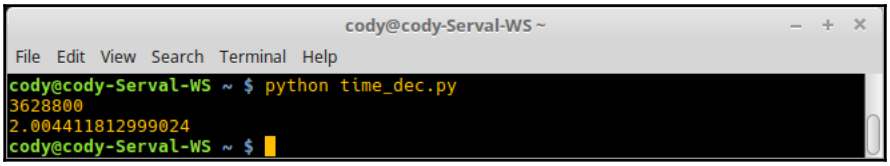

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3 arg_check.py
Input radius: 6.9
Traceback (most recent call last):
File "arg_check.py", line 21, in <module>
diameter, circumference, area = circle_measures(int(r))
ValueError: invalid literal for int() with base 10: '6.9'


In [3]: print(chip.age)
4
In [4]: print(chip.breed)
domestic shorthair
In [5]:

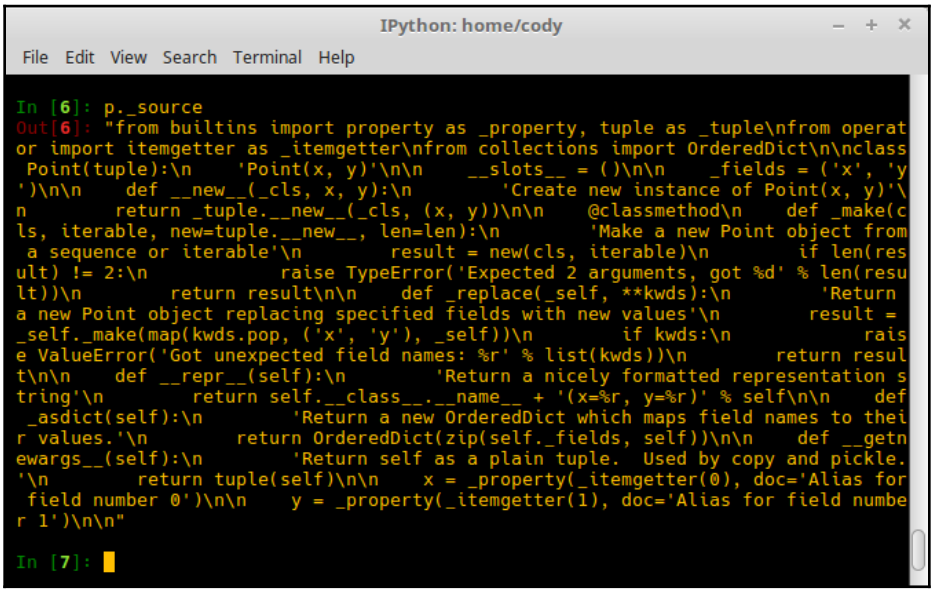

IPython: home/cody
File Edit View Search Terminal Help
In [5]: print(Cat.age)
AttributeError Traceback (most recent call last)
<ipython-input-5-c353e553e183> in <module>()
----> 1 print(Cat.age)
AttributeError: type object 'Cat' has no attribute 'age'
In [6]:

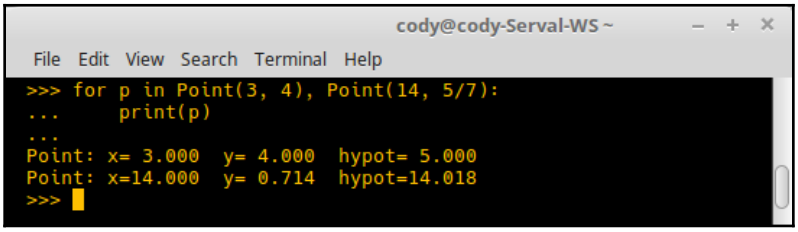

IPython: home/cody
File Edit View Search Terminal Help
In [28]: class Cat():
...: def __init__(self, breed, age):
...: """Initialization method to auto-populate an instance"""
...:
...: self.breed = breed
...: self.age = age
...:
...: def cat_age(self):
...: """Get the cat's age"""
...:
...: return self.age
...:
...: def breed(self):
...: """Get the type of cat, e.g. short hair, long hair, etc."""
...:
...: return self.breed
...:
...: @staticmethod # This is required
...: def cry():
...: """Static method, available to all instances and the class
...:
...: Notice that 'self' is not a required argument
...: """
...:
...: return "Nyao nyao" # It's a Japanese cat
...:
...: def __repr__(self):
...: """Return string representation of Cat object.
...:
...: Without this method, only the object's memory address will be printed.
...: """
...: return "{breed}, {age}, {cry}".format(breed = self.breed, age = self.age, cry = se
...: lf.cry)
...:
...:
In [29]: print(chip.cry)
<function Cat.cry at 0x7f3d04841950>
In [30]: print(chip.cry())
Nyao nyao
In [31]: print(Cat.cry)
<function Cat.cry at 0x7f3d0481f048>
In [32]: print(Cat.cry())
Nyao nyao
In [33]:

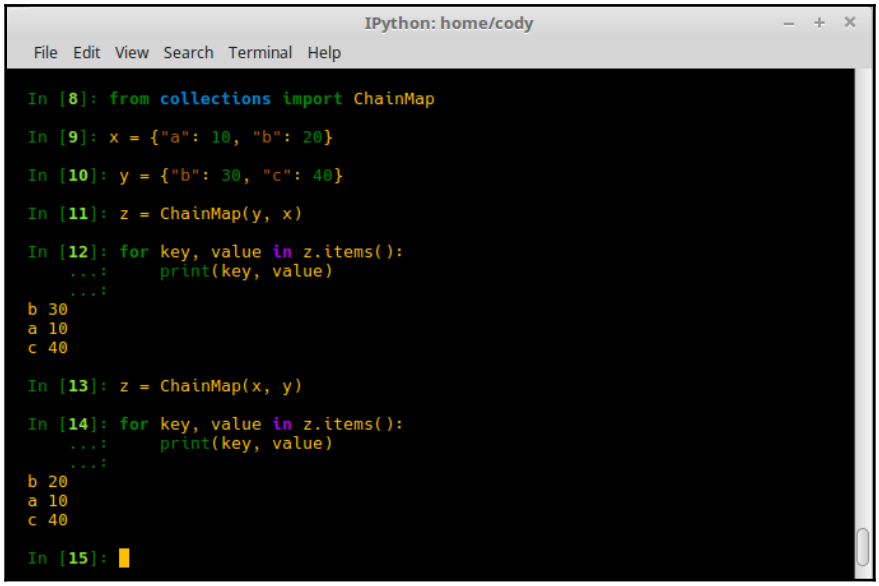

IPython: home/cody
File Edit View Search Terminal Help
In [43]: captain = Cat("grey shorthair", 2)
In [44]: print(captain.type())
Some sort of domestic cat.
In [45]: class Japanese_Bobtail(Cat):
...: pass
...:
In [46]: rascal = Japanese_Bobtail("shorthair", 5)
In [47]: print(rascal.type())
Japanese_Bobtail
In [48]:

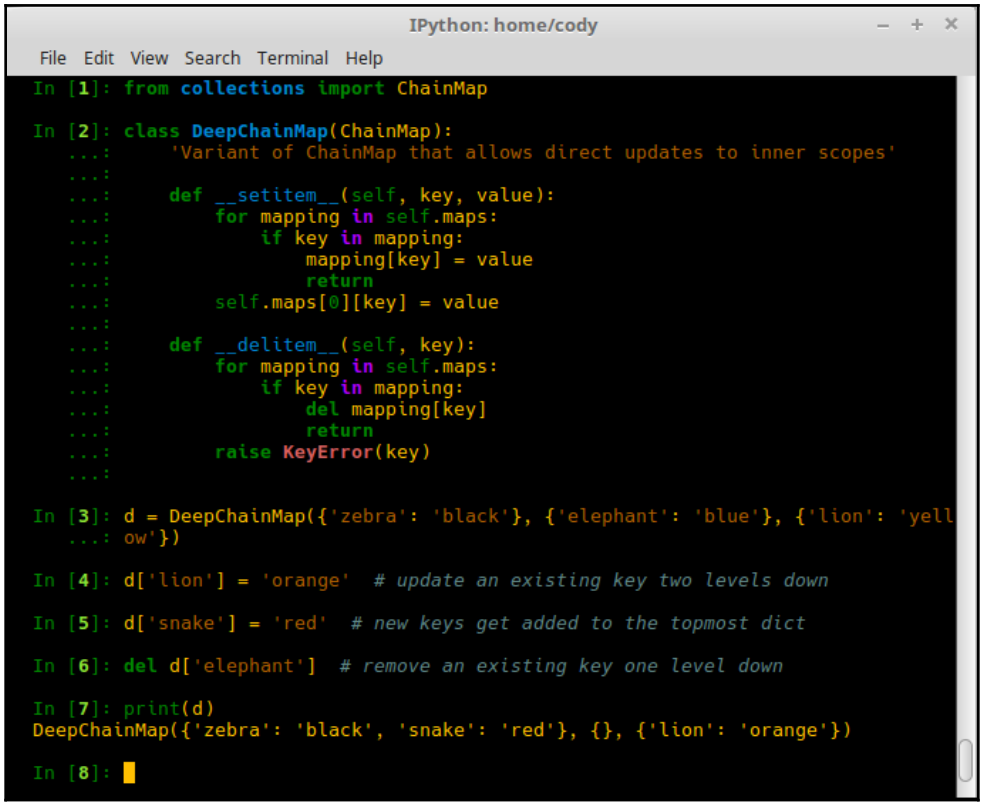

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python time_dec.py
3628800
2.004411812999024
cody@cody-Serval-WS ~ $

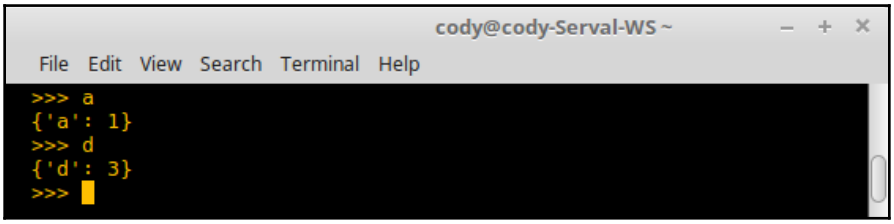
Chapter 4: Using Python Collections

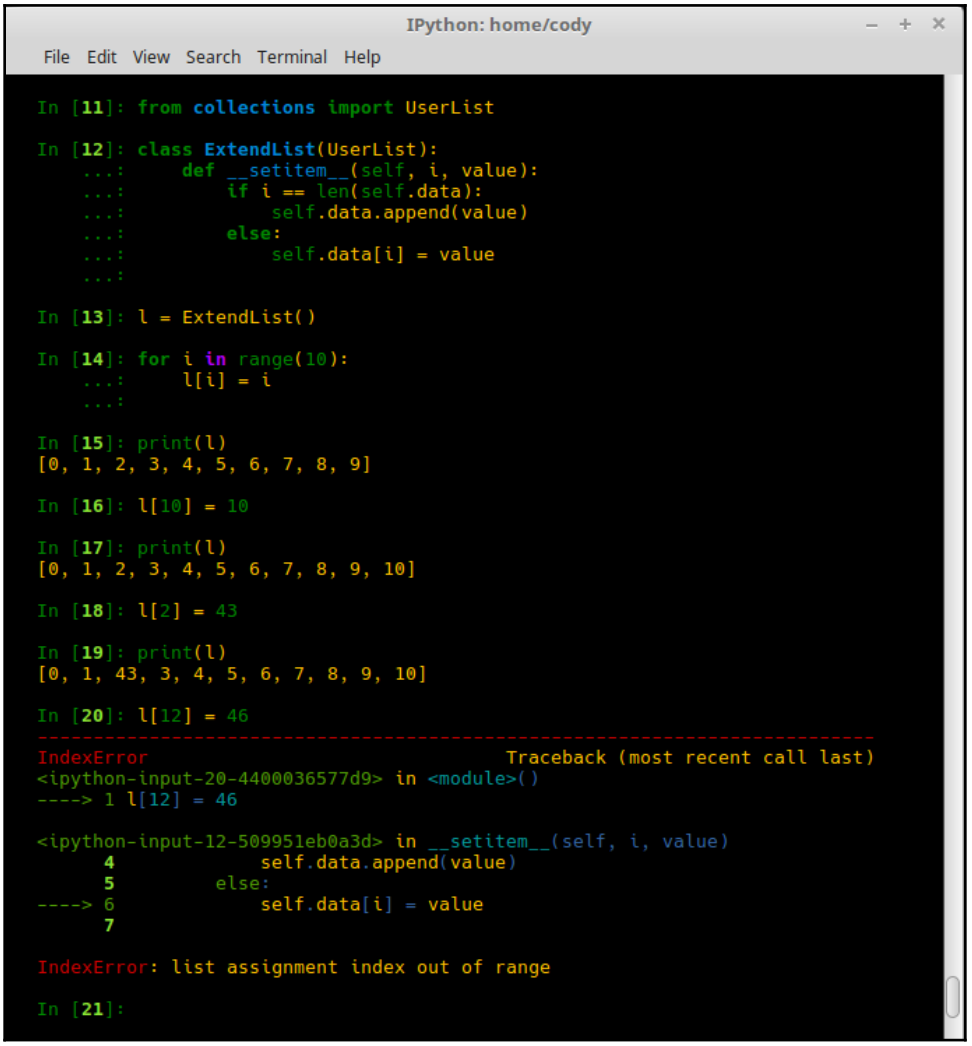


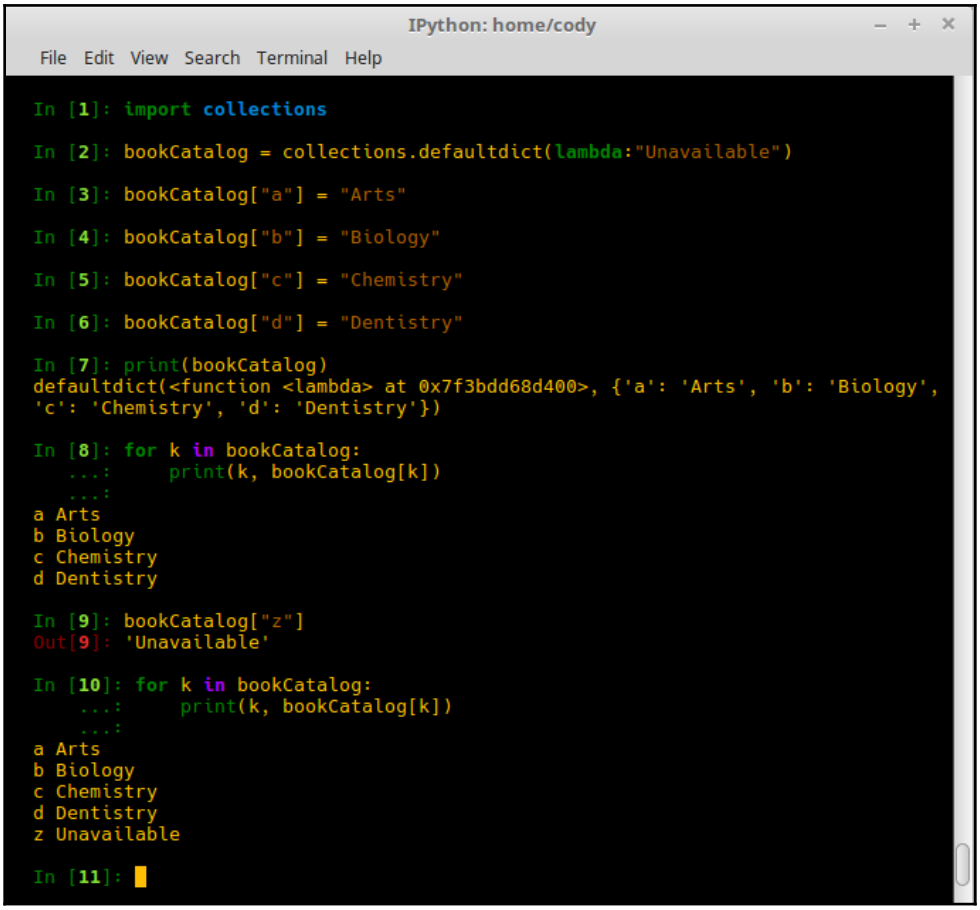


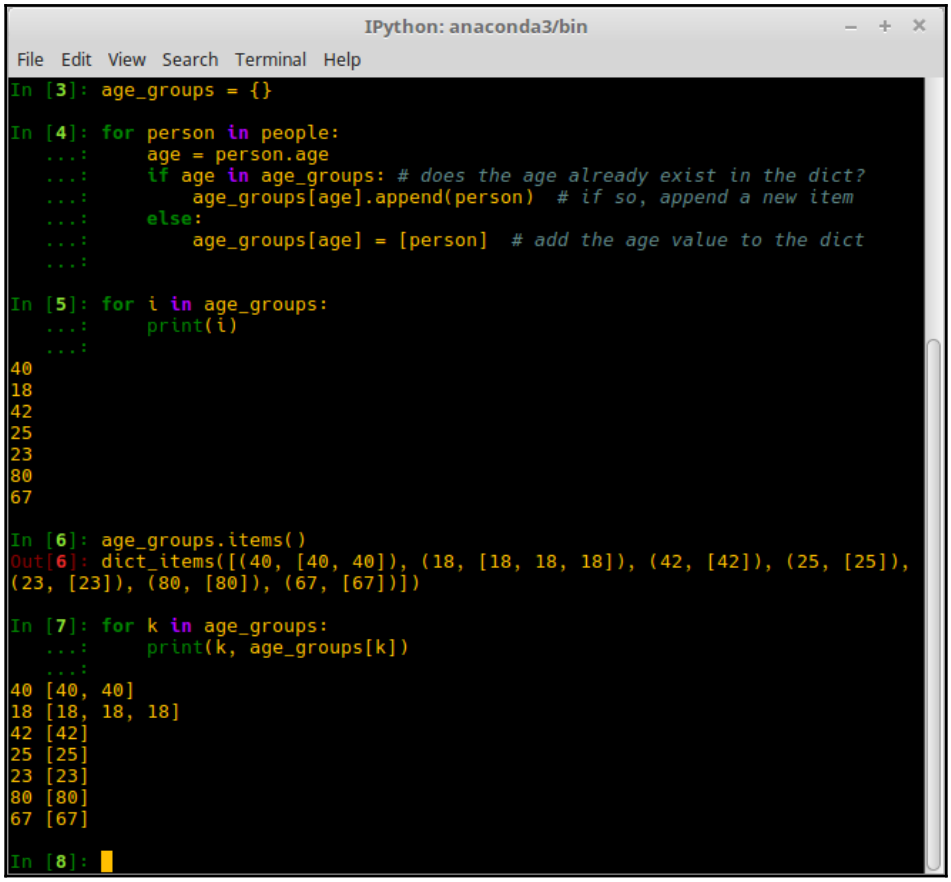

IPython: home/cody
File Edit View Search Terminal Help
In [8]: from collections import ChainMap
In [9]: x = {"a": 10, "b": 20}
In [10]: y = {"b": 30, "c": 40}
In [11]: z = ChainMap(y, x)
In [12]: for key, value in z.items():
...: print(key, value)
...:
b 30
a 10
c 40
In [13]: z = ChainMap(x, y)
In [14]: for key, value in z.items():
...: print(key, value)
...:
b 20
a 10
c 40
In [15]:

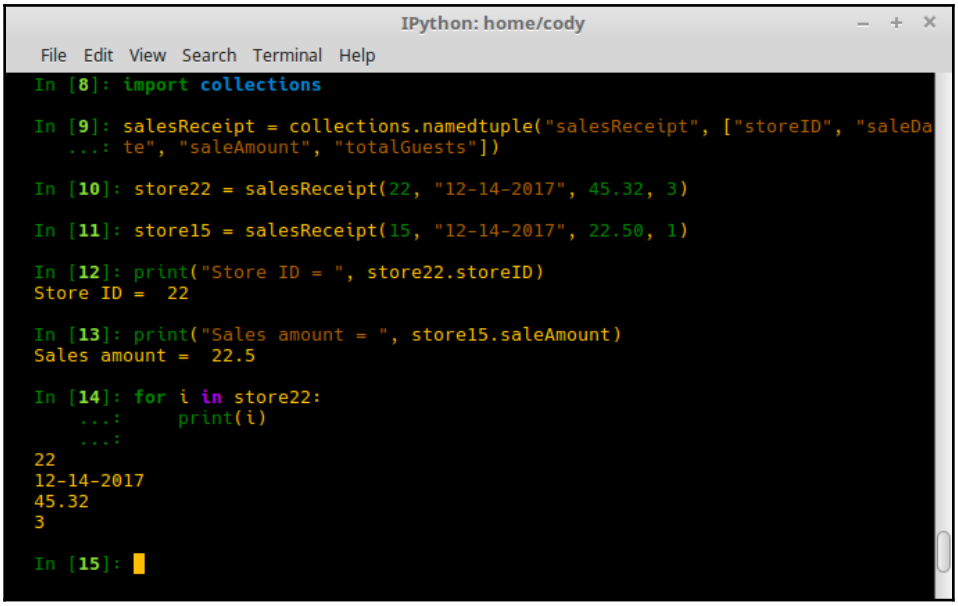

IPython: home/cody
File Edit View Search Terminal Help
In [1]: from collections import ChainMap
In [2]: class DeepChainMap(ChainMap):
...: 'Variant of ChainMap that allows direct updates to inner scopes'
...:
...: def __setitem__(self, key, value):
...: for mapping in self.maps:
...: if key in mapping:
...: mapping[key] = value
...: return
...: self.maps[0][key] = value
...:
...: def __delitem__(self, key):
...: for mapping in self.maps:
...: if key in mapping:
...: del mapping[key]
...: return
...: raise KeyError(key)
...:
In [3]: d = DeepChainMap({'zebra': 'black'}, {'elephant': 'blue'}, {'lion': 'yell
...: ow'})
In [4]: d['lion'] = 'orange' # update an existing key two levels down
In [5]: d['snake'] = 'red' # new keys get added to the topmost dict
In [6]: del d['elephant'] # remove an existing key one level down
In [7]: print(d)
DeepChainMap({'zebra': 'black', 'snake': 'red'}, {}, {'lion': 'orange'})
In [8]:


cody@cody-Serval-WS ~
File Edit View Search Terminal Help
>>> a
{'a': 1}
>>> d
{'d': 3}
>>>

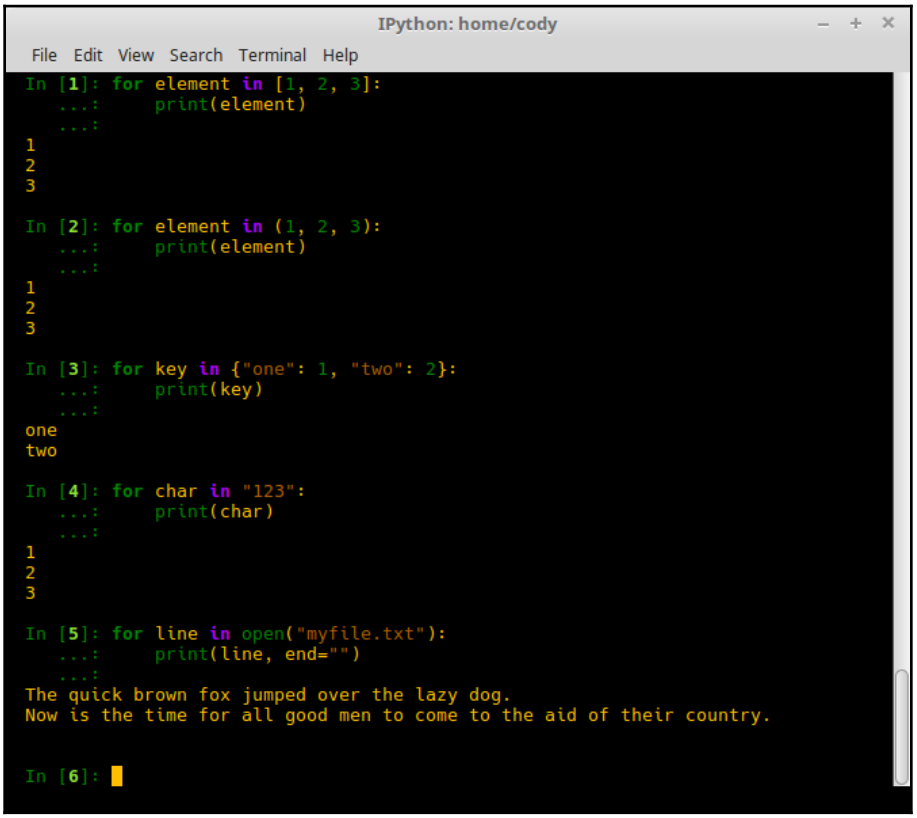

IPython: home/cody
File Edit View Search Terminal Help
In [11]: from collections import UserList
In [12]: class ExtendList(UserList):
...: def __setitem__(self, i, value):
...: if i == len(self.data):
...: self.data.append(value)
...: else:
...: self.data[i] = value
...:
In [13]: l = ExtendList()
In [14]: for i in range(10):
...: l[i] = i
...:
In [15]: print(l)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
In [16]: l[10] = 10
In [17]: print(l)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
In [18]: l[2] = 43
In [19]: print(l)
[0, 1, 43, 3, 4, 5, 6, 7, 8, 9, 10]
In [20]: l[12] = 46
IndexError Traceback (most recent call last)
<ipython-input-20-4400036577d9> in <module>()
----> 1 l[12] = 46
<ipython-input-12-509951eb0a3d> in __setitem__(self, i, value)
4 self.data.append(value)
5 else:
----> 6 self.data[i] = value
7
IndexError: list assignment index out of range
In [21]:

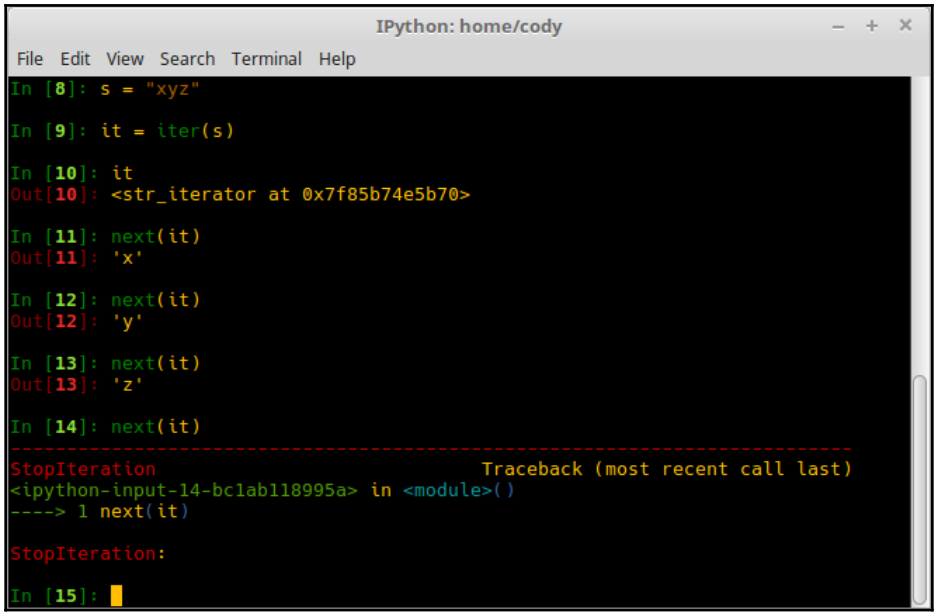

IPython: home/cody
File Edit View Search Terminal Help
In [1]: import collections
In [2]: bookCatalog = collections.defaultdict(lambda:"Unavailable")
In [3]: bookCatalog["a"] = "Arts"
In [4]: bookCatalog["b"] = "Biology"
In [5]: bookCatalog["c"] = "Chemistry"
In [6]: bookCatalog["d"] = "Dentistry"
In [7]: print(bookCatalog)
defaultdict(<function <lambda> at 0x7f3bdd68d400>, {'a': 'Arts', 'b': 'Biology',
'c': 'Chemistry', 'd': 'Dentistry'})
In [8]: for k in bookCatalog:
...: print(k, bookCatalog[k])
...:
a Arts
b Biology
c Chemistry
d Dentistry
In [9]: bookCatalog["z"]
Out[9]: 'Unavailable'
In [10]: for k in bookCatalog:
...: print(k, bookCatalog[k])
...:
a Arts
b Biology
c Chemistry
d Dentistry
z Unavailable
In [11]:

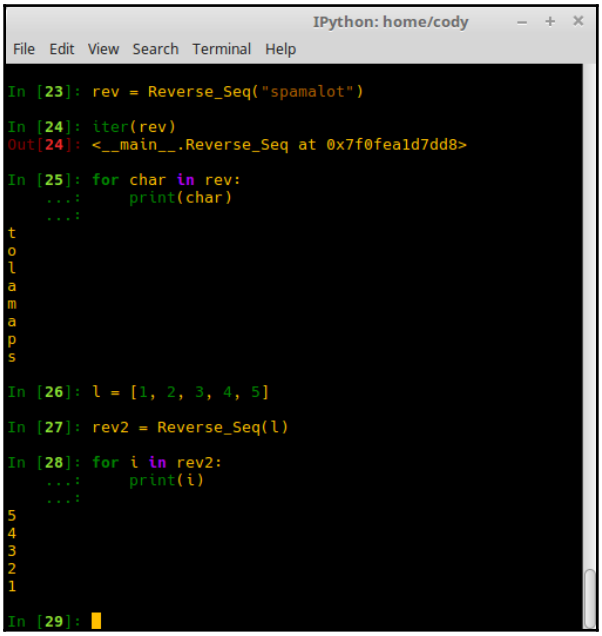

IPython: anaconda3/bin
File Edit View Search Terminal Help
In [3]: age_groups = {}
In [4]: for person in people:
...: age = person.age
...: if age in age_groups: # does the age already exist in the dict?
...: age_groups[age].append(person) # if so, append a new item
...: else:
...: age_groups[age] = [person] # add the age value to the dict
...:
In [5]: for i in age_groups:
...: print(i)
...:
40
18
42
25
23
80
67
In [6]: age_groups.items()
Out[6]: dict_items([(40, [40, 40]), (18, [18, 18, 18]), (42, [42]), (25, [25]), (23, [23]), (80, [80]), (67, [67])])
In [7]: for k in age_groups:
...: print(k, age_groups[k])
...:
40 [40, 40]
18 [18, 18, 18]
42 [42]
25 [25]
23 [23]
80 [80]
67 [67]
In [8]:

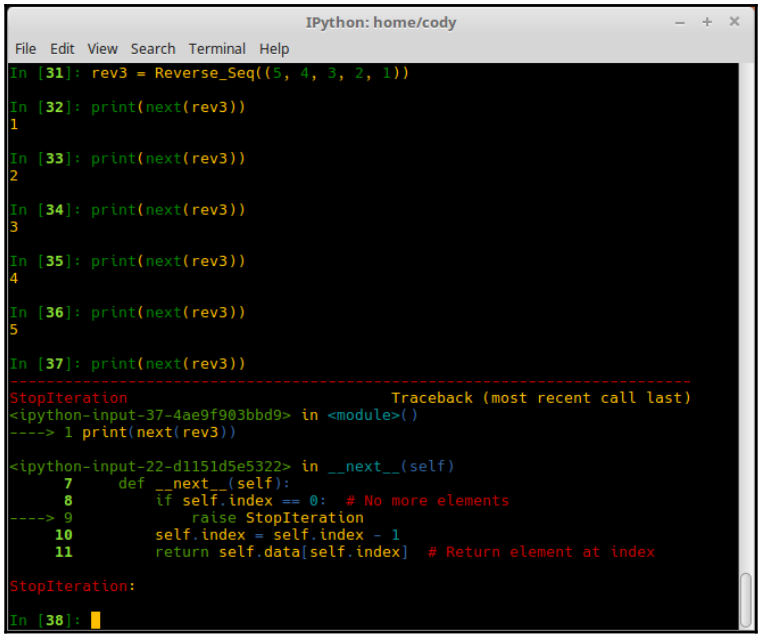

IPython: home/cody
File Edit View Search Terminal Help
In [8]: import collections
In [9]: salesReceipt = collections.namedtuple("salesReceipt", ["storeID", "saleDa
...: te", "saleAmount", "totalGuests"])
In [10]: store22 = salesReceipt(22, "12-14-2017", 45.32, 3)
In [11]: store15 = salesReceipt(15, "12-14-2017", 22.50, 1)
In [12]: print("Store ID = ", store22.storeID)
Store ID = 22
In [13]: print("Sales amount = ", store15.saleAmount)
Sales amount = 22.5
In [14]: for i in store22:
...: print(i)
...:
22
12-14-2017
45.32
3
In [15]:

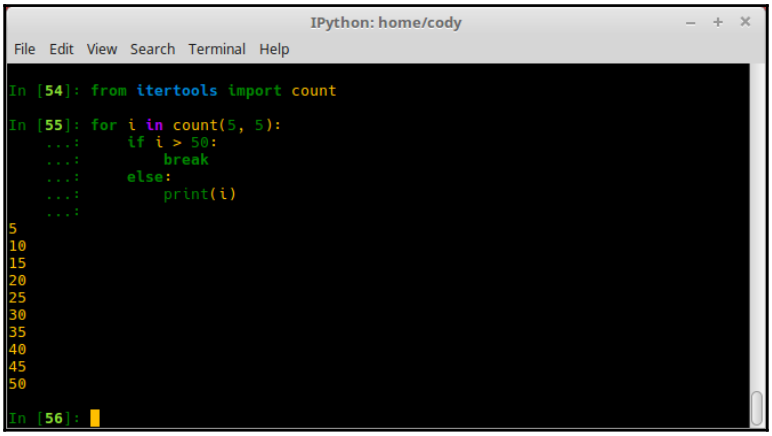
Chapter 5: Generators, Coroutines, and Parallel Processing

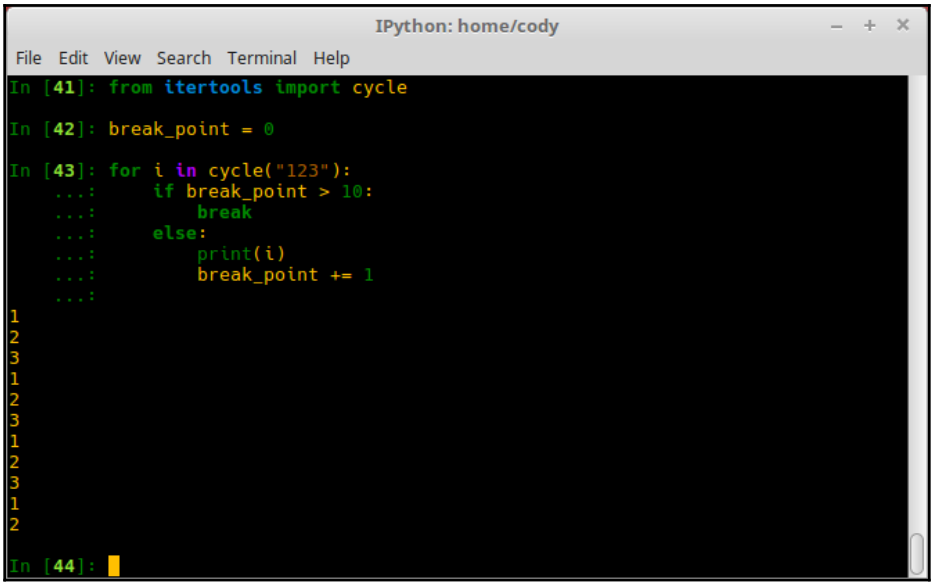


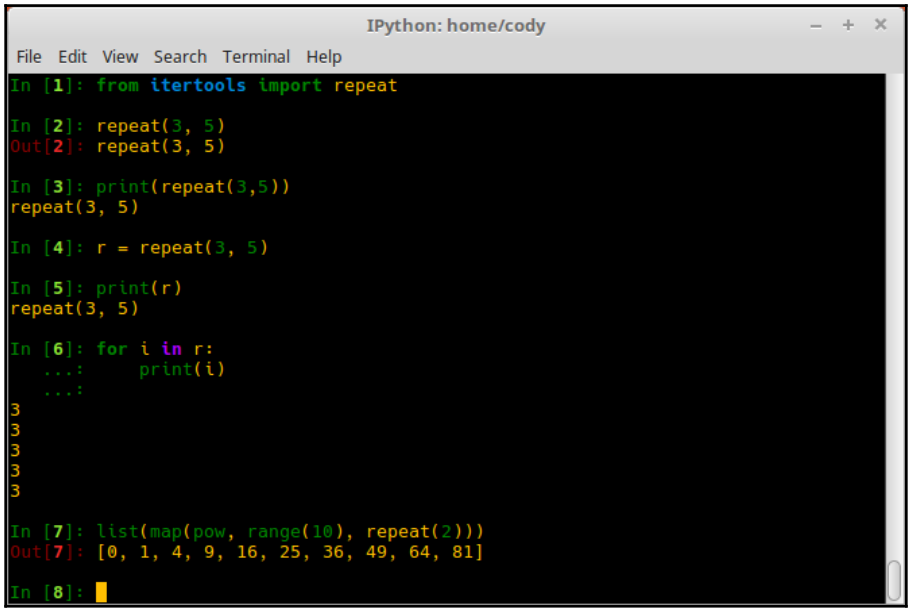

IPython: home/cody
File Edit View Search Terminal Help
In [8]: s = "xyz"
In [9]: it = iter(s)
In [10]: it
Out[10]: <str_iterator at 0x7f85b74e5b70>
In [11]: next(it)
Out[11]: 'x'
In [12]: next(it)
Out[12]: 'y'
In [13]: next(it)
Out[13]: 'z'
In [14]: next(it)
StopIteration Traceback (most recent call last)
<ipython-input-14-bc1ab118995a> in <module>()
----> 1 next(it)
StopIteration:
In [15]:

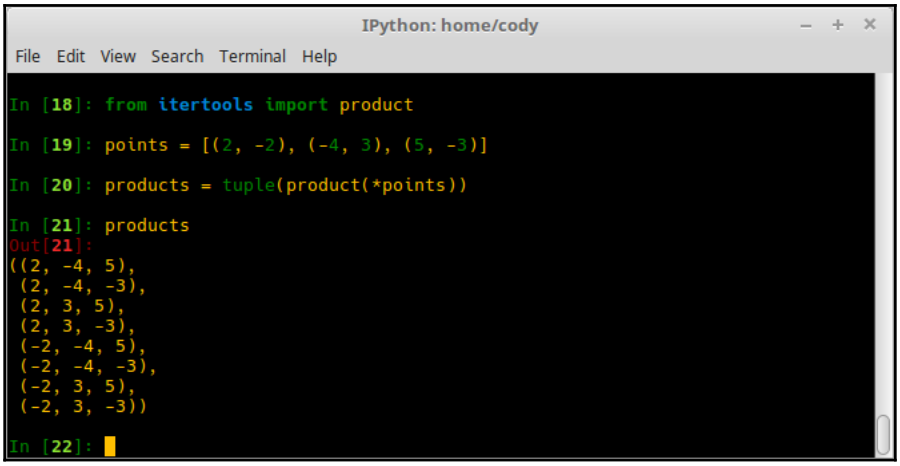

IPython: home/cody
File Edit View Search Terminal Help
In [23]: rev = Reverse_Seq("spamalot")
In [24]: iter(rev)
Out[24]: <__main__.Reverse_Seq at 0x7f0fea1d7dd8>
In [25]: for char in rev:
...: print(char)
...:
t
o
l
a
m
a
p
s
In [26]: l = [1, 2, 3, 4, 5]
In [27]: rev2 = Reverse_Seq(l)
In [28]: for i in rev2:
...: print(i)
...:
5
4
3
2
1
In [29]:

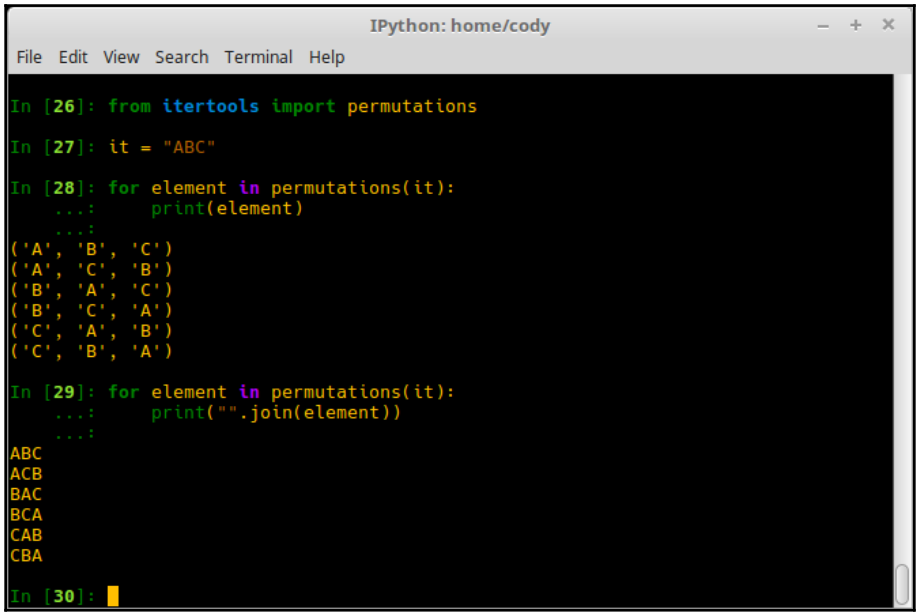

IPython: home/cody
File Edit View Search Terminal Help
In [31]: rev3 = Reverse_Seq((5, 4, 3, 2, 1))
In [32]: print(next(rev3))
1
In [33]: print(next(rev3))
2
In [34]: print(next(rev3))
3
In [35]: print(next(rev3))
4
In [36]: print(next(rev3))
5
In [37]: print(next(rev3))
StopIteration Traceback (most recent call last)
<ipython-input-37-4ae9f903bbd9> in <module>()
----> 1 print(next(rev3))
<ipython-input-22-d1151d5e5322> in __next__(self)
7 def __next__(self):
8 if self.index == 0: # No more elements
----> 9 raise StopIteration
10 self.index = self.index - 1
11 return self.data[self.index] # Return element at index
StopIteration:
In [38]:

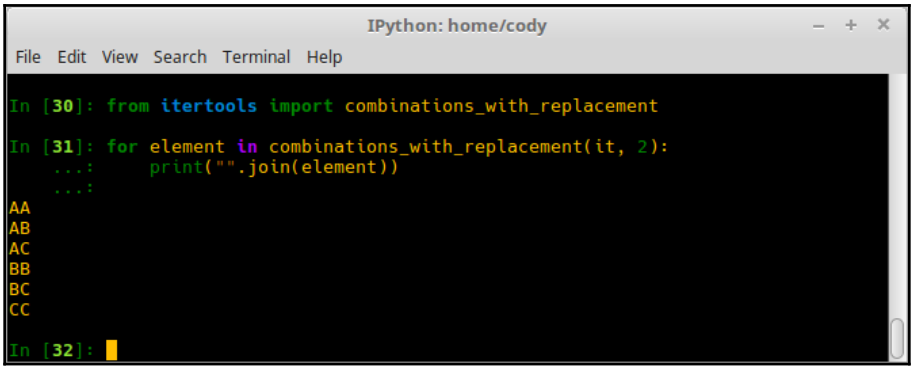

IPython: home/cody
File Edit View Search Terminal Help
In [54]: from itertools import count
In [55]: for i in count(5, 5):
...: if i > 50:
...: break
...: else:
...: print(i)
...:
5
10
15
20
25
30
35
40
45
50
In [56]:

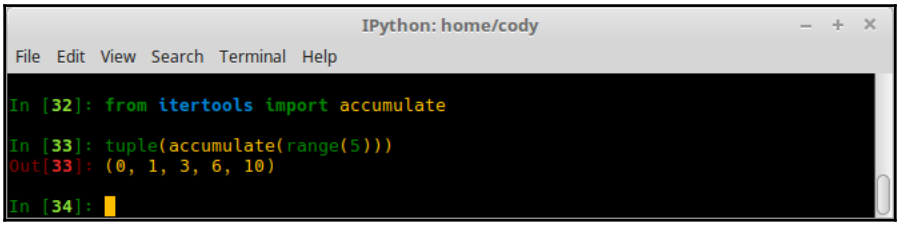

IPython: home/cody
File Edit View Search Terminal Help
In [41]: from itertools import cycle
In [42]: break_point = 0
In [43]: for i in cycle("123"):
...: if break_point > 10:
...: break
...: else:
...: print(i)
...: break_point += 1
...:
1
2
3
1
2
3
1
2
3
1
2
In [44]:

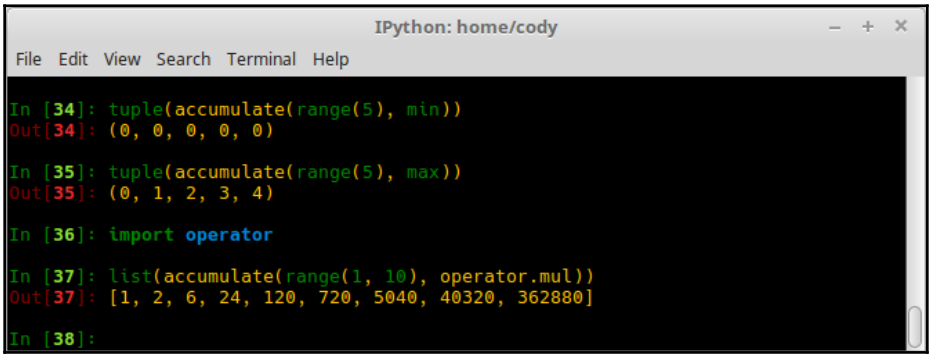

IPython: home/cody
File Edit View Search Terminal Help
In [1]: from itertools import repeat
In [2]: repeat(3, 5)
Out[2]: repeat(3, 5)
In [3]: print(repeat(3,5))
repeat(3, 5)
In [4]: r = repeat(3, 5)
In [5]: print(r)
repeat(3, 5)
In [6]: for i in r:
...: print(i)
...:
3
3
3
3
3
In [7]: list(map(pow, range(10), repeat(2)))
Out[7]: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
In [8]:

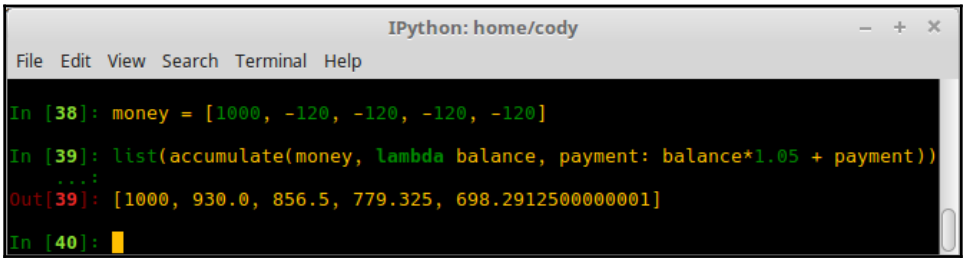

IPython: home/cody
File Edit View Search Terminal Help
In [18]: from itertools import product
In [19]: points = [(2, -2), (-4, 3), (5, -3)]
In [20]: products = tuple(product(*points))
In [21]: products
Out[21]:
((2, -4, 5),
(2, -4, -3),
(2, 3, 5),
(2, 3, -3),
(-2, -4, 5),
(-2, -4, -3),
(-2, 3, 5),
(-2, 3, -3))
In [22]:

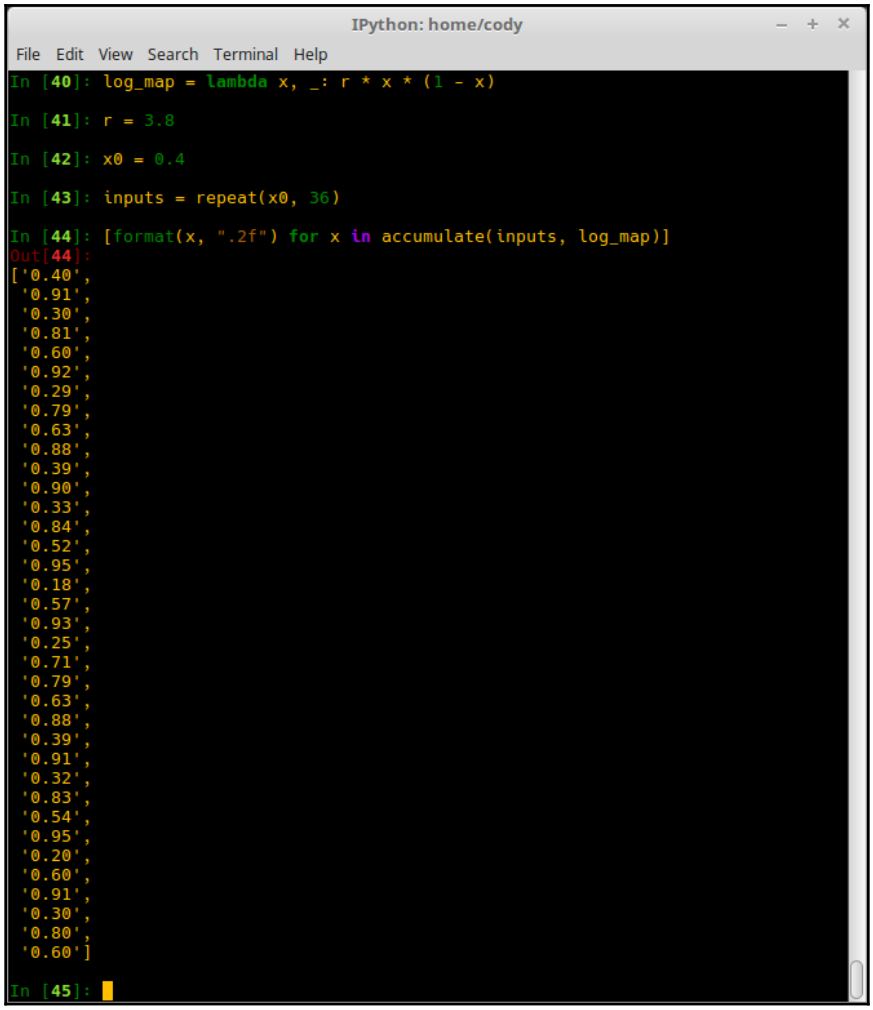

IPython: home/cody
File Edit View Search Terminal Help
In [26]: from itertools import permutations
In [27]: it = "ABC"
In [28]: for element in permutations(it):
...: print(element)
...:
('A', 'B', 'C')
('A', 'C', 'B')
('B', 'A', 'C')
('B', 'C', 'A')
('C', 'A', 'B')
('C', 'B', 'A')
In [29]: for element in permutations(it):
...: print("".join(element))
...:
ABC
ACB
BAC
BCA
CAB
CBA
In [30]:


IPython: home/cody
File Edit View Search Terminal Help
In [30]: from itertools import combinations_with_replacement
In [31]: for element in combinations_with_replacement(it, 2):
...: print("".join(element))
...:
AA
AB
AC
BB
BC
CC
In [32]:


IPython: home/cody
File Edit View Search Terminal Help
In [32]: from itertools import accumulate
In [33]: tuple(accumulate(range(5)))
Out[33]: (0, 1, 3, 6, 10)
In [34]:

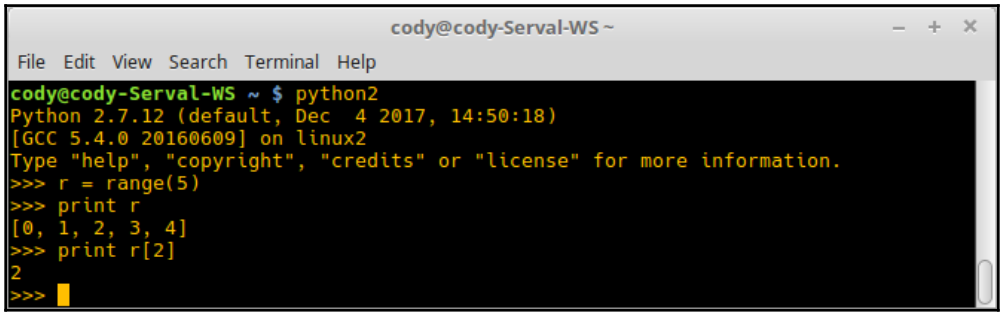

IPython: home/cody
File Edit View Search Terminal Help
In [34]: tuple(accumulate(range(5), min))
Out[34]: (0, 0, 0, 0, 0)
In [35]: tuple(accumulate(range(5), max))
Out[35]: (0, 1, 2, 3, 4)
In [36]: import operator
In [37]: list(accumulate(range(1, 10), operator.mul))
Out[37]: [1, 2, 6, 24, 120, 720, 5040, 40320, 362880]
In [38]:

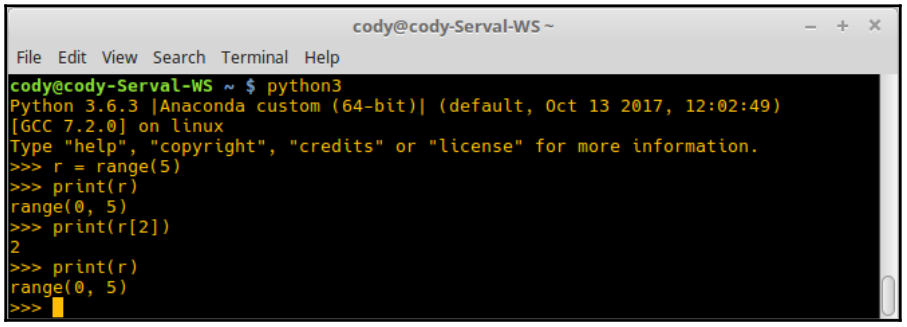

IPython: home/cody
File Edit View Search Terminal Help
In [38]: money = [1000, -120, -120, -120, -120]
In [39]: list(accumulate(money, lambda balance, payment: balance*1.05 + payment))
...:
Out[39]: [1000, 930.0, 856.5, 779.325, 698.2912500000001]
In [40]:

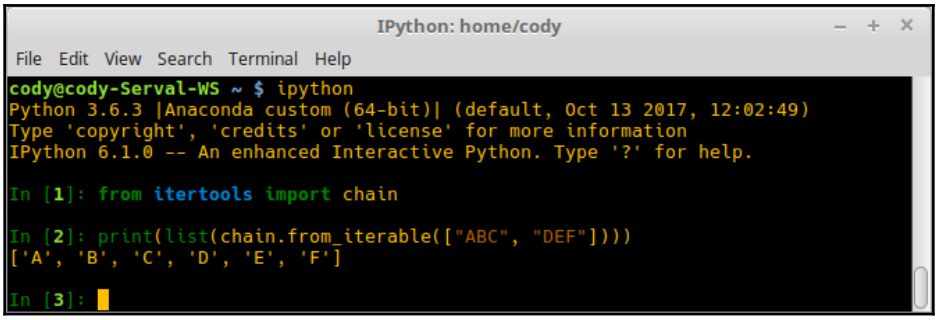

IPython: home/cody
File Edit View Search Terminal Help
In [40]: log_map = lambda x, _: r * x * (1 - x)
In [41]: r = 3.8
In [42]: x0 = 0.4
In [43]: inputs = repeat(x0, 36)
In [44]: [format(x, ".2f") for x in accumulate(inputs, log_map)]
Out[44]:
['0.40',
'0.91',
'0.30',
'0.81',
'0.60',
'0.92',
'0.29',
'0.79',
'0.63',
'0.88',
'0.39',
'0.90',
'0.33',
'0.84',
'0.52',
'0.95',
'0.18',
'0.57',
'0.93',
'0.25',
'0.71',
'0.79',
'0.63',
'0.88',
'0.39',
'0.91',
'0.32',
'0.83',
'0.54',
'0.95',
'0.20',
'0.60',
'0.91',
'0.30',
'0.80',
'0.60']
In [45]:

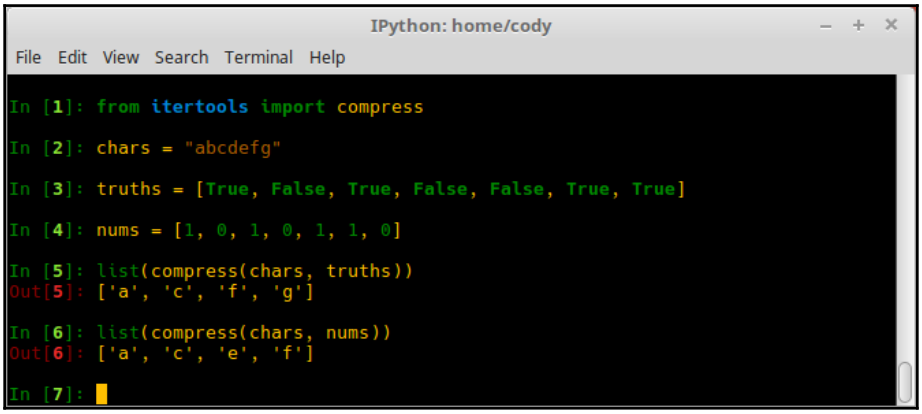

IPython: home/cody
File Edit View Search Terminal Help
In [45]: from itertools import chain
In [46]: this_list = list(chain(["spam", "more spam"], (1, 2, 3), ["here's", "mor
...: e", "values"]))
In [47]: print(this_list)
['spam', 'more spam', 1, 2, 3, "here's", 'more', 'values']
In [48]:

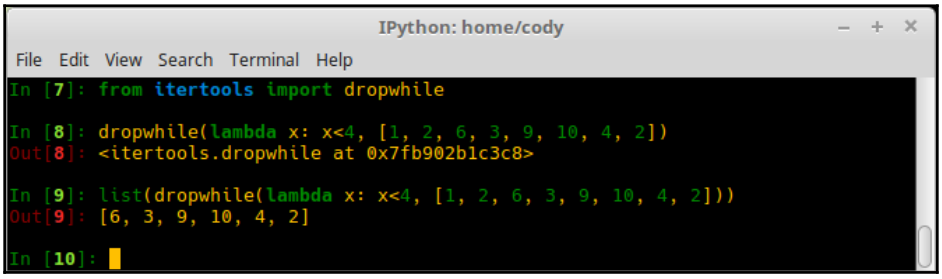

IPython: home/cody
File Edit View Search Terminal Help
In [48]: this_list = ["spam", "more spam"]
In [49]: numbers = [1, 2, 3]
In [50]: more_list = ["here's", "more", "values"]
In [51]: this_list += numbers + more_list
In [52]: this_list
Out[52]: ['spam', 'more spam', 1, 2, 3, "here's", 'more', 'values']
In [53]:

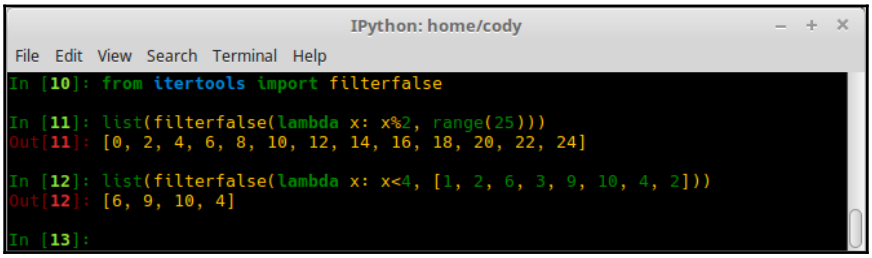

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python2
Python 2.7.12 (default, Dec 4 2017, 14:50:18)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> r = range(5)
>>> print r
[0, 1, 2, 3, 4]
>>> print r[2]
2
>>>

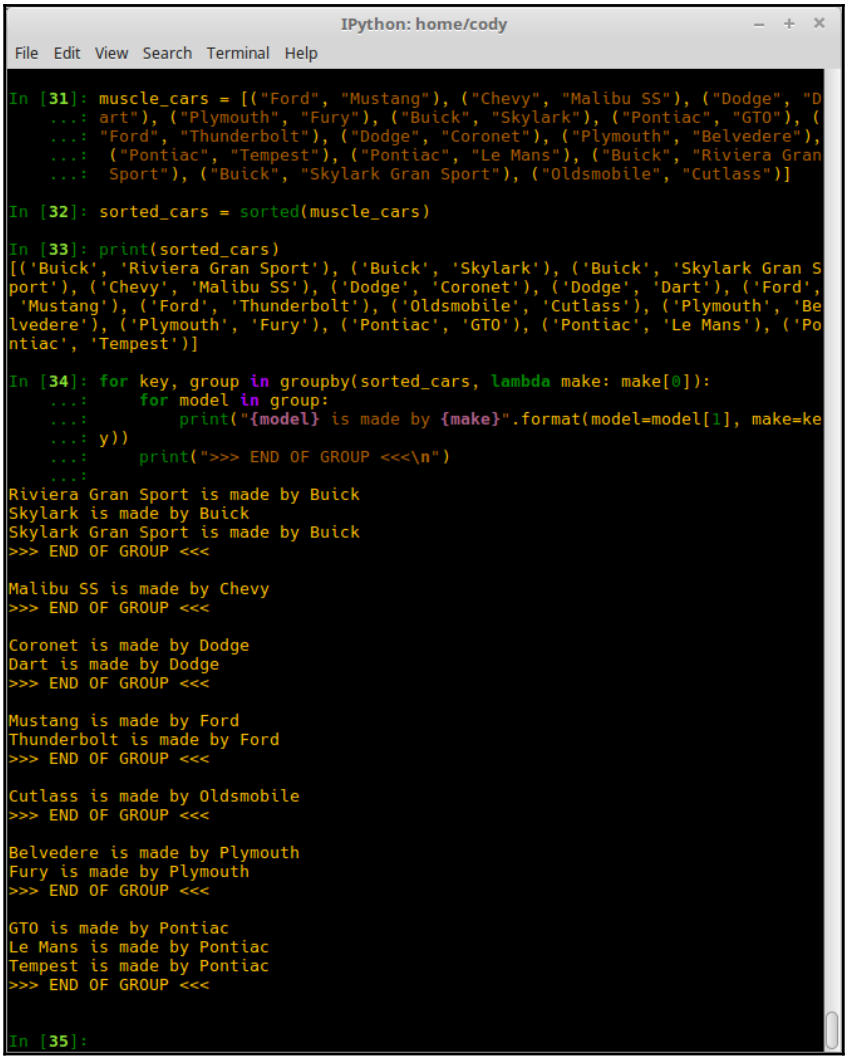

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3
Python 3.6.3 |Anaconda custom (64-bit)| (default, Oct 13 2017, 12:02:49)
[GCC 7.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> r = range(5)
>>> print(r)
range(0, 5)
>>> print(r[2])
2
>>> print(r)
range(0, 5)
>>>

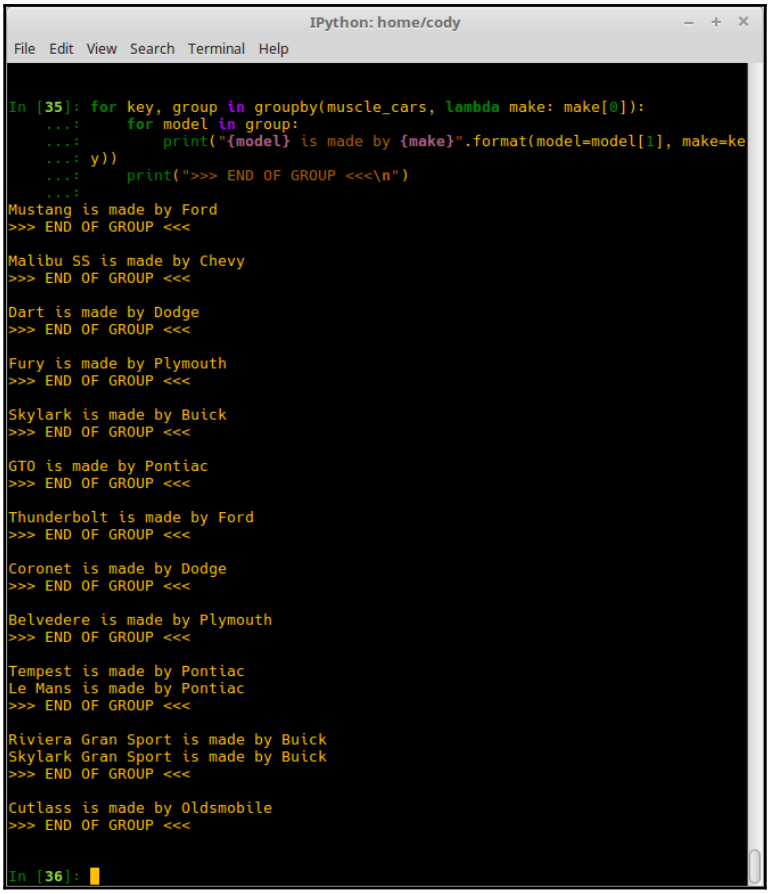

IPython: home/cody
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ ipython
Python 3.6.3 |Anaconda custom (64-bit)| (default, Oct 13 2017, 12:02:49)
Type 'copyright', 'credits' or 'license' for more information
IPython 6.1.0 -- An enhanced Interactive Python. Type '?' for help.
In [1]: from itertools import chain
In [2]: print(list(chain.from_iterable(["ABC", "DEF"])))
['A', 'B', 'C', 'D', 'E', 'F']
In [3]:

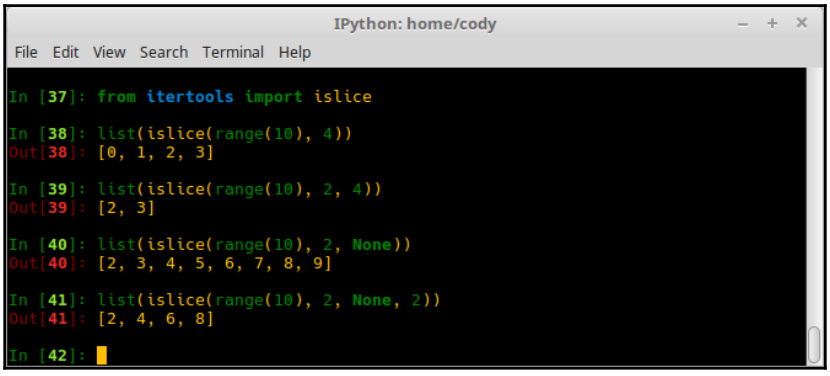

IPython: home/cody
File Edit View Search Terminal Help
In [1]: from itertools import compress
In [2]: chars = "abcdefg"
In [3]: truths = [True, False, True, False, False, True, True]
In [4]: nums = [1, 0, 1, 0, 1, 1, 0]
In [5]: list(compress(chars, truths))
Out[5]: ['a', 'c', 'f', 'g']
In [6]: list(compress(chars, nums))
Out[6]: ['a', 'c', 'e', 'f']
In [7]:

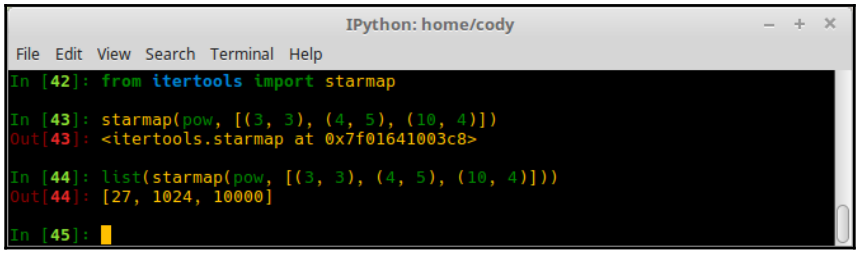

IPython: home/cody
File Edit View Search Terminal Help
In [7]: from itertools import dropwhile
In [8]: dropwhile(lambda x: x<4, [1, 2, 6, 3, 9, 10, 4, 2])
Out[8]: <itertools.dropwhile at 0x7fb902b1c3c8>
In [9]: list(dropwhile(lambda x: x<4, [1, 2, 6, 3, 9, 10, 4, 2]))
Out[9]: [6, 3, 9, 10, 4, 2]
In [10]:

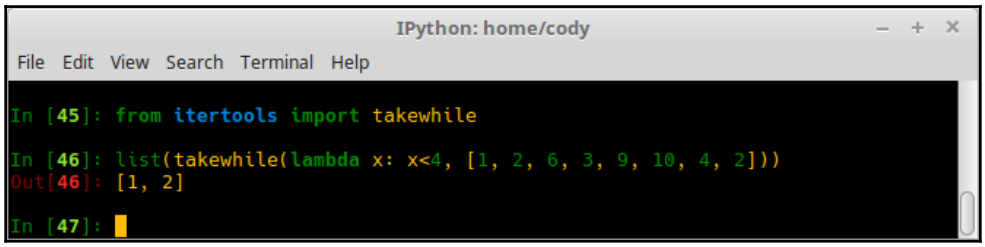

IPython: home/cody
File Edit View Search Terminal Help
In [10]: from itertools import filterfalse
In [11]: list(filterfalse(lambda x: x%2, range(25)))
Out[11]: [0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24]
In [12]: list(filterfalse(lambda x: x<4, [1, 2, 6, 3, 9, 10, 4, 2]))
Out[12]: [6, 9, 10, 4]
In [13]:

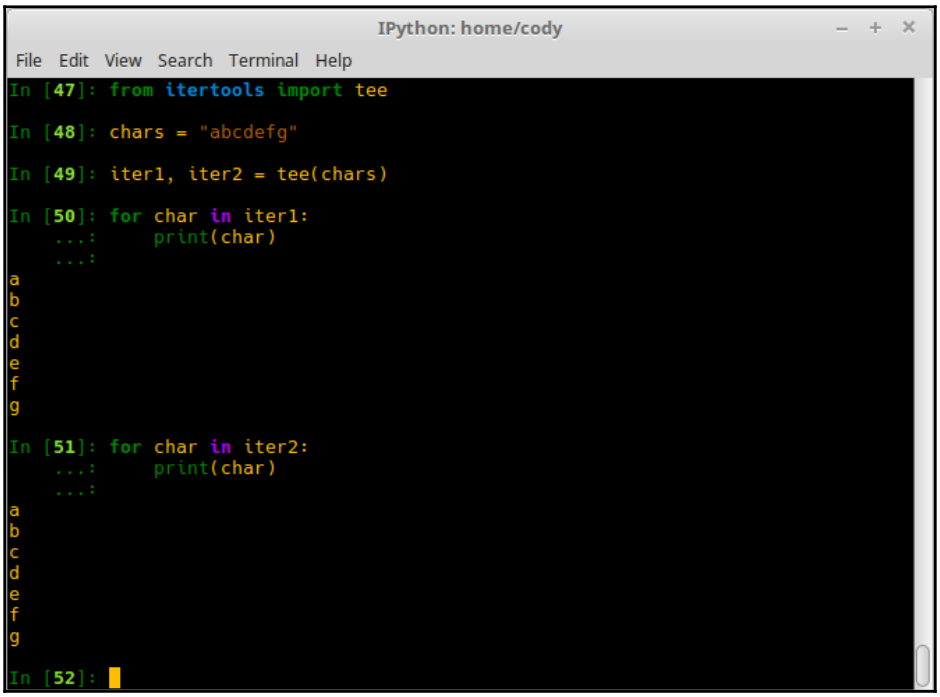

IPython: home/cody
File Edit View Search Terminal Help
In [31]: muscle_cars = [("Ford", "Mustang"), ("Chevy", "Malibu SS"), ("Dodge", "D
...: art"), ("Plymouth", "Fury"), ("Buick", "Skylark"), ("Pontiac", "GTO"), (
...: "Ford", "Thunderbolt"), ("Dodge", "Coronet"), ("Plymouth", "Belvedere"),
...: ("Pontiac", "Tempest"), ("Pontiac", "Le Mans"), ("Buick", "Riviera Gran
...: Sport"), ("Buick", "Skylark Gran Sport"), ("Oldsmobile", "Cutlass")]
In [32]: sorted_cars = sorted(muscle_cars)
In [33]: print(sorted_cars)
[('Buick', 'Riviera Gran Sport'), ('Buick', 'Skylark'), ('Buick', 'Skylark Gran S
port'), ('Chevy', 'Malibu SS'), ('Dodge', 'Coronet'), ('Dodge', 'Dart'), ('Ford',
'Mustang'), ('Ford', 'Thunderbolt'), ('Oldsmobile', 'Cutlass'), ('Plymouth', 'Be
lvedere'), ('Plymouth', 'Fury'), ('Pontiac', 'GTO'), ('Pontiac', 'Le Mans'), ('Po
ntiac', 'Tempest')]
In [34]: for key, group in groupby(sorted_cars, lambda make: make[0]):
...: for model in group:
...: print("{model} is made by {make}".format(model=model[1], make=ke
...: y))
...: print(">>> END OF GROUP <<<\n")
...:
Riviera Gran Sport is made by Buick
Skylark is made by Buick
Skylark Gran Sport is made by Buick
>>> END OF GROUP <<<
Malibu SS is made by Chevy
>>> END OF GROUP <<<
Coronet is made by Dodge
Dart is made by Dodge
>>> END OF GROUP <<<
Mustang is made by Ford
Thunderbolt is made by Ford
>>> END OF GROUP <<<
Cutlass is made by Oldsmobile
>>> END OF GROUP <<<
Belvedere is made by Plymouth
Fury is made by Plymouth
>>> END OF GROUP <<<
GTO is made by Pontiac
Le Mans is made by Pontiac
Tempest is made by Pontiac
>>> END OF GROUP <<<
In [35]:


IPython: home/cody
File Edit View Search Terminal Help
In [35]: for key, group in groupby(muscle_cars, lambda make: make[0]):
...: for model in group:
...: print("{model} is made by {make}".format(model=model[1], make=ke
...: y))
...: print(">>> END OF GROUP <<<\n")
...:
Mustang is made by Ford
>>> END OF GROUP <<<
Malibu SS is made by Chevy
>>> END OF GROUP <<<
Dart is made by Dodge
>>> END OF GROUP <<<
Fury is made by Plymouth
>>> END OF GROUP <<<
Skylark is made by Buick
>>> END OF GROUP <<<
GTO is made by Pontiac
>>> END OF GROUP <<<
Thunderbolt is made by Ford
>>> END OF GROUP <<<
Coronet is made by Dodge
>>> END OF GROUP <<<
Belvedere is made by Plymouth
>>> END OF GROUP <<<
Tempest is made by Pontiac
Le Mans is made by Pontiac
>>> END OF GROUP <<<
Riviera Gran Sport is made by Buick
Skylark Gran Sport is made by Buick
>>> END OF GROUP <<<
Cutlass is made by Oldsmobile
>>> END OF GROUP <<<
In [36]:

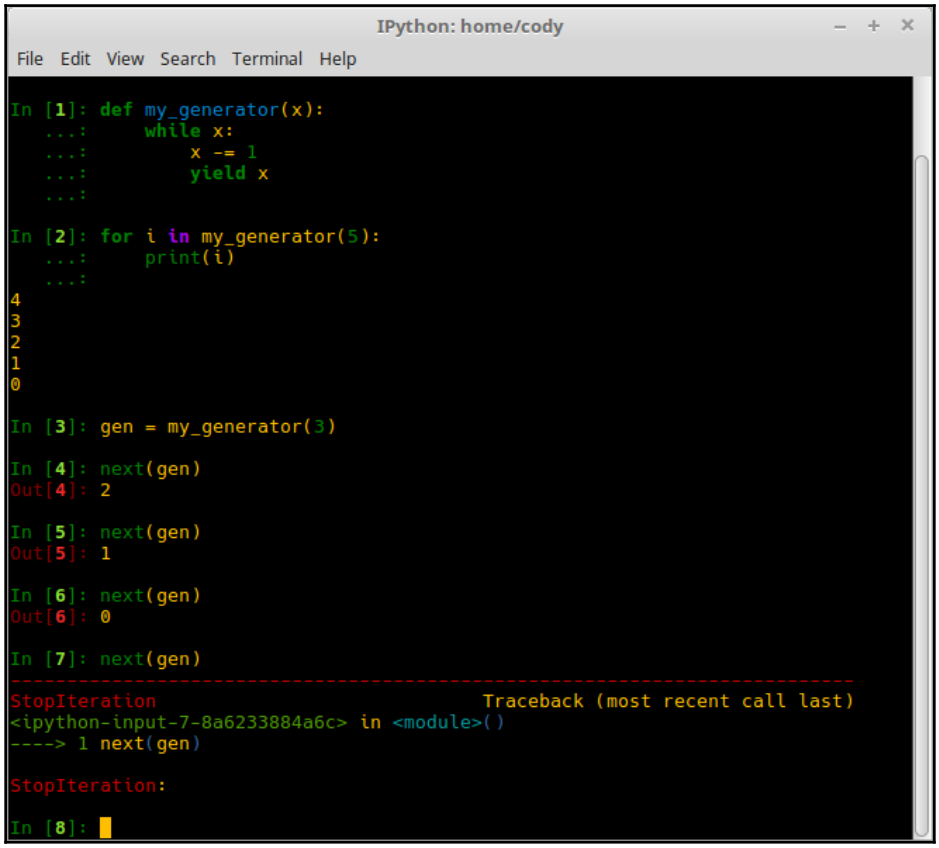

IPython: home/cody
File Edit View Search Terminal Help
In [37]: from itertools import islice
In [38]: list(islice(range(10), 4))
Out[38]: [0, 1, 2, 3]
In [39]: list(islice(range(10), 2, 4))
Out[39]: [2, 3]
In [40]: list(islice(range(10), 2, None))
Out[40]: [2, 3, 4, 5, 6, 7, 8, 9]
In [41]: list(islice(range(10), 2, None, 2))
Out[41]: [2, 4, 6, 8]
In [42]:

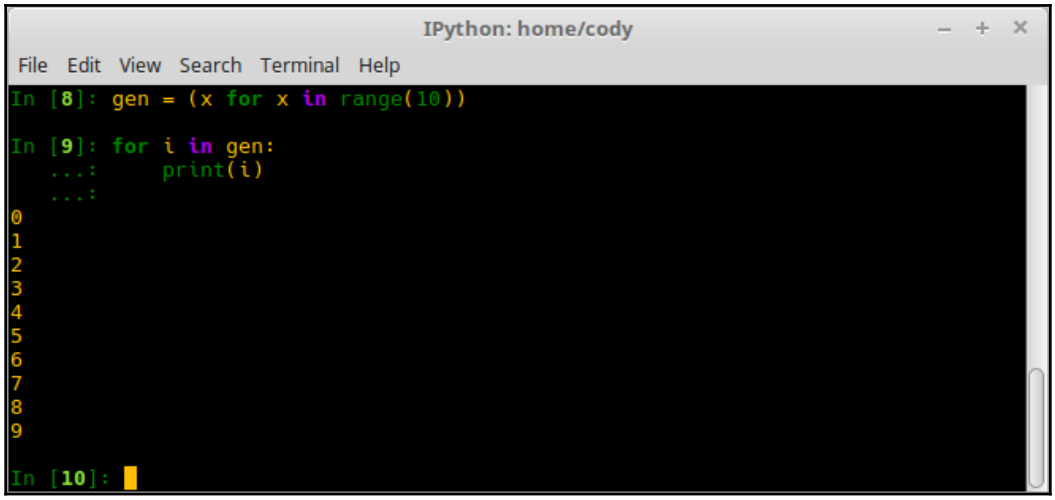

IPython: home/cody
File Edit View Search Terminal Help
In [42]: from itertools import starmap
In [43]: starmap(pow, [(3, 3), (4, 5), (10, 4)])
Out[43]: <itertools.starmap at 0x7f01641003c8>
In [44]: list(starmap(pow, [(3, 3), (4, 5), (10, 4)]))
Out[44]: [27, 1024, 10000]
In [45]:


IPython: home/cody
File Edit View Search Terminal Help
In [45]: from itertools import takewhile
In [46]: list(takewhile(lambda x: x<4, [1, 2, 6, 3, 9, 10, 4, 2]))
Out[46]: [1, 2]
In [47]:

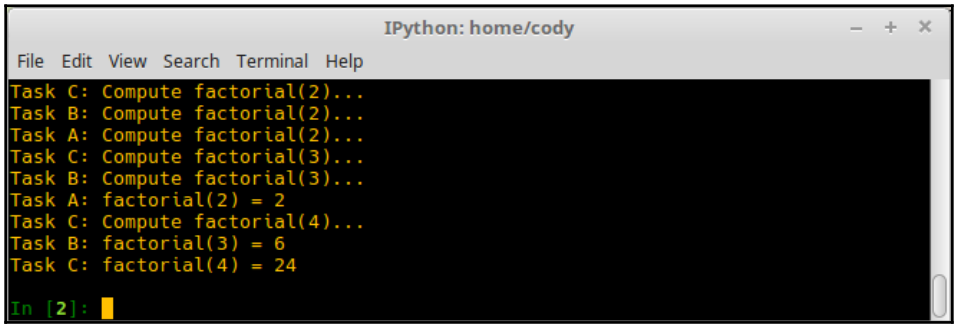

IPython: home/cody
File Edit View Search Terminal Help
In [47]: from itertools import tee
In [48]: chars = "abcdefg"
In [49]: iter1, iter2 = tee(chars)
In [50]: for char in iter1:
...: print(char)
...:
a
b
c
d
e
f
g
In [51]: for char in iter2:
...: print(char)
...:
a
b
c
d
e
f
g
In [52]:

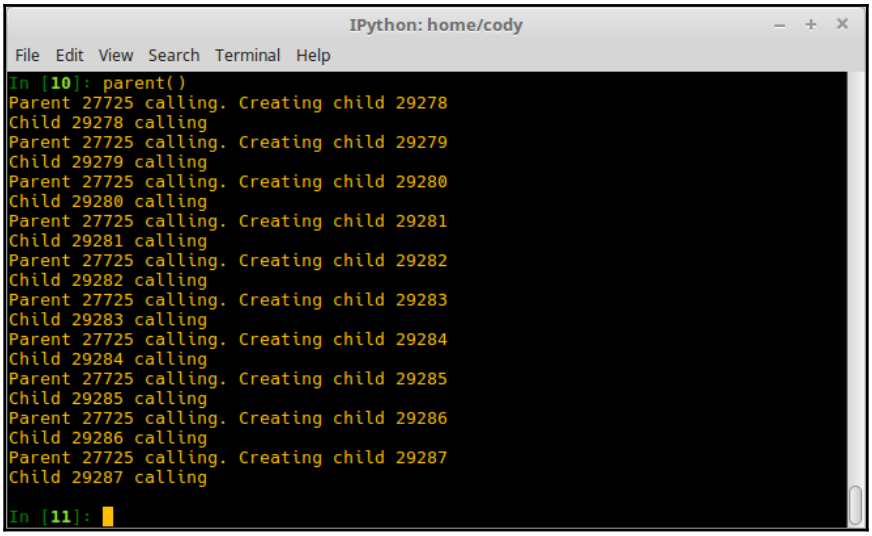

IPython: home/cody
File Edit View Search Terminal Help
In [52]: from itertools import zip_longest
In [53]: for iterable in zip_longest("abcdefg", "zyx", fillvalue="spam"):
...:   print(iterable)
...:
('a', 'z')
('b', 'y')
('c', 'x')
('d', 'spam')
('e', 'spam')
('f', 'spam')
('g', 'spam')
In [54]:

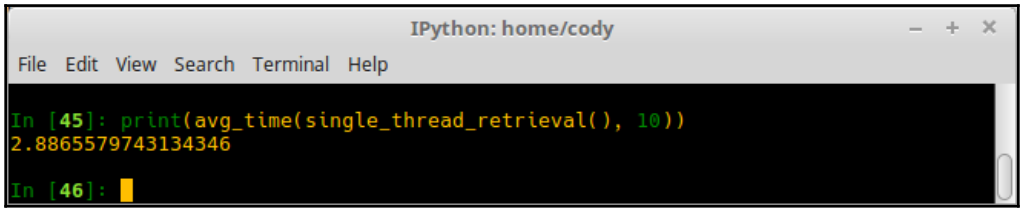

IPython: home/cody
File Edit View Search Terminal Help
In [1]: def my_generator(x):
...:   while x:
...:   x -= 1
...:   yield x
...:
In [2]: for i in my_generator(5):
...:   print(i)
...:
4
3
2
1
0
In [3]: gen = my_generator(3)
In [4]: next(gen)
Out[4]: 2
In [5]: next(gen)
Out[5]: 1
In [6]: next(gen)
Out[6]: 0
In [7]: next(gen)
StopIteration Traceback (most recent call last)
<ipython-input-7-8a6233884a6c> in <module>()
----> 1 next(gen)
StopIteration:
In [8]:

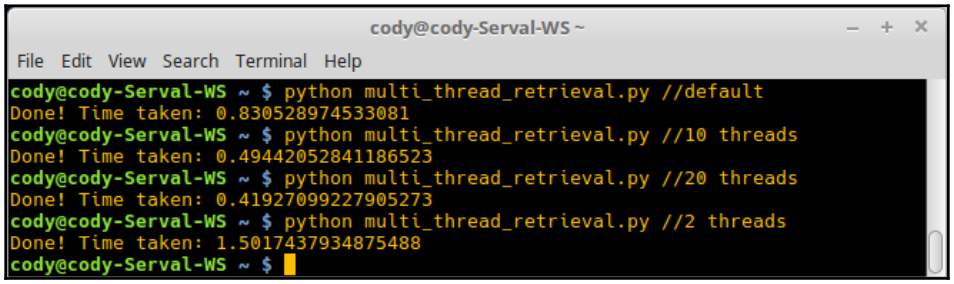

IPython: home/cody
File Edit View Search Terminal Help
In [8]: gen = (x for x in range(10))
In [9]: for i in gen:
...: print(i)
...:
0
1
2
3
4
5
6
7
8
9
In [10]:

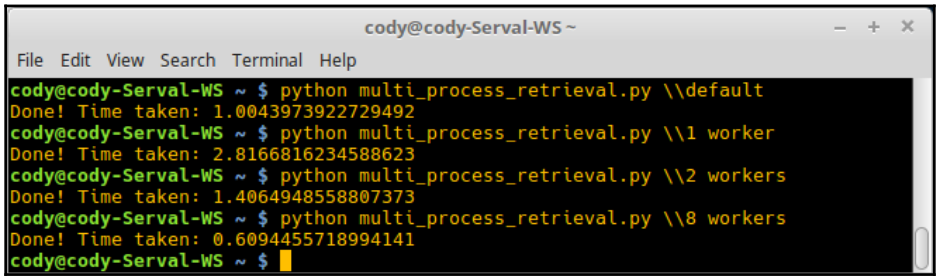

IPython: home/cody
File Edit View Search Terminal Help
In [10]: def cor():
...: hi = yield "Hello"
...: yield hi
...:
In [11]: cor = cor()
In [12]: print(next(cor))
Hello
In [13]: print(cor.send("World"))
World
In [14]:

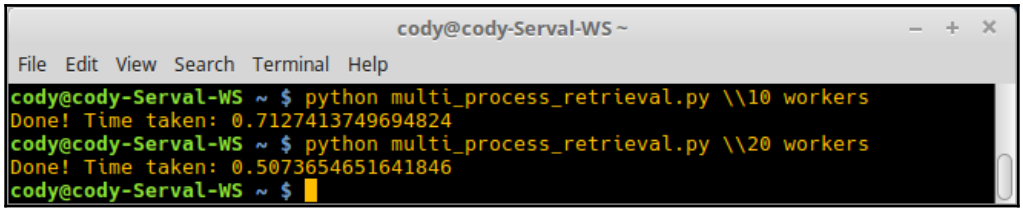

IPython: home/cody
File Edit View Search Terminal Help
Task C: Compute factorial(2)...
Task B: Compute factorial(2)...
Task A: Compute factorial(2)...
Task C: Compute factorial(3)...
Task B: Compute factorial(3)...
Task A: factorial(2) = 2
Task C: Compute factorial(4)...
Task B: factorial(3) = 6
Task C: factorial(4) = 24
In [2]:


IPython: home/cody
File Edit View Search Terminal Help
In [10]: parent()
Parent 27725 calling. Creating child 29278
Child 29278 calling
Parent 27725 calling. Creating child 29279
Child 29279 calling
Parent 27725 calling. Creating child 29280
Child 29280 calling
Parent 27725 calling. Creating child 29281
Child 29281 calling
Parent 27725 calling. Creating child 29282
Child 29282 calling
Parent 27725 calling. Creating child 29283
Child 29283 calling
Parent 27725 calling. Creating child 29284
Child 29284 calling
Parent 27725 calling. Creating child 29285
Child 29285 calling
Parent 27725 calling. Creating child 29286
Child 29286 calling
Parent 27725 calling. Creating child 29287
Child 29287 calling
In [11]:

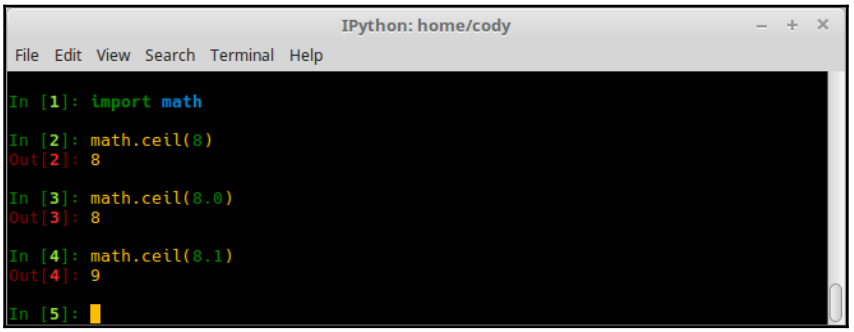

IPython: home/cody
File Edit View Search Terminal Help
In [45]: print(avg_time(single_thread_retrieval(), 10))
2.8865579743134346
In [46]:

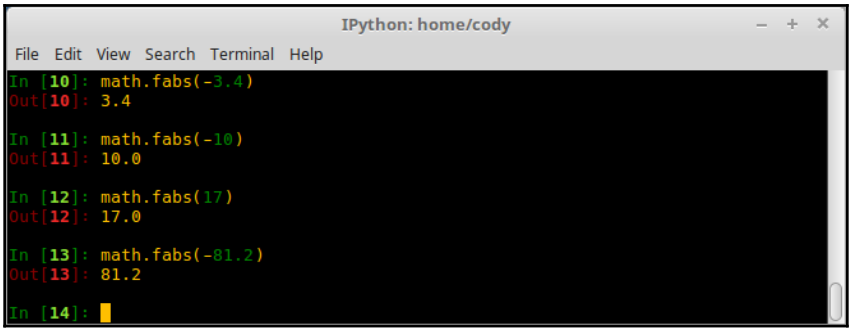

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python multi_thread_retrieval.py //default
Done! Time taken: 0.830528974533081
cody@cody-Serval-WS ~ $ python multi_thread_retrieval.py //10 threads
Done! Time taken: 0.49442052841186523
cody@cody-Serval-WS ~ $ python multi_thread_retrieval.py //20 threads
Done! Time taken: 0.41927099227905273
cody@cody-Serval-WS ~ $ python multi_thread_retrieval.py //2 threads
Done! Time taken: 1.5017437934875488
cody@cody-Serval-WS ~ $

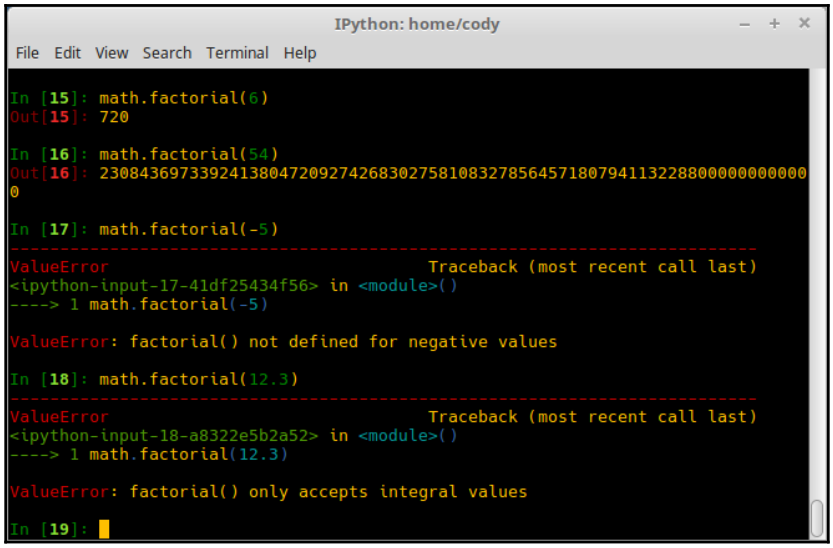

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python multi_process_retrieval.py \\default
Done! Time taken: 1.0043973922729492
cody@cody-Serval-WS ~ $ python multi_process_retrieval.py \\1 worker
Done! Time taken: 2.8166816234588623
cody@cody-Serval-WS ~ $ python multi_process_retrieval.py \\2 workers
Done! Time taken: 1.4064948558807373
cody@cody-Serval-WS ~ $ python multi_process_retrieval.py \\8 workers
Done! Time taken: 0.6094455718994141
cody@cody-Serval-WS ~ $


cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python multi_process_retrieval.py \\10 workers
Done! Time taken: 0.7127413749694824
cody@cody-Serval-WS ~ $ python multi_process_retrieval.py \\20 workers
Done! Time taken: 0.5073654651641846
cody@cody-Serval-WS ~ $

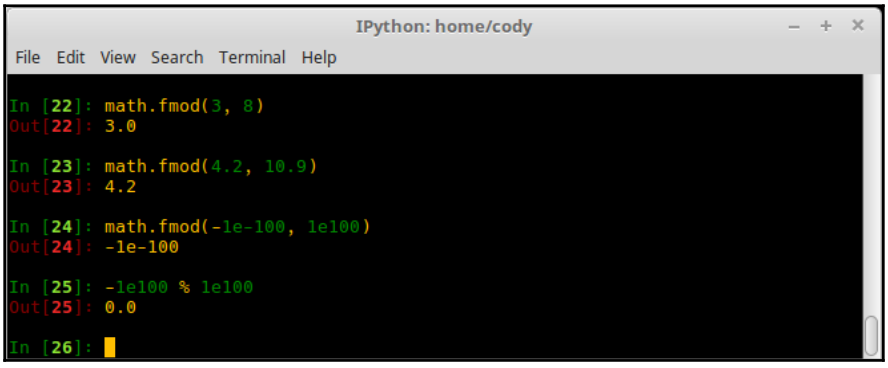
Chapter 6: Working with Python's Math Module

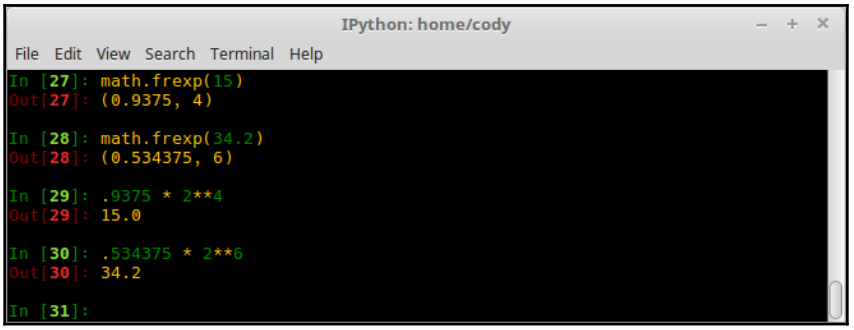


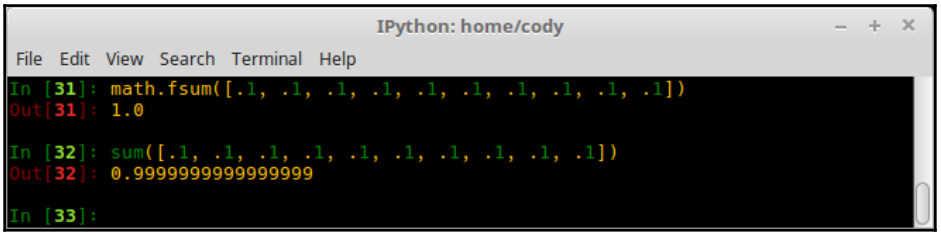


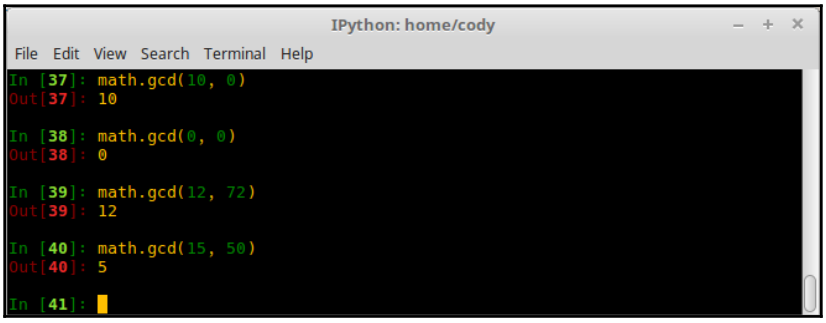

IPython: home/cody
File Edit View Search Terminal Help
In [15]: math.factorial(6)
Out[15]: 720
In [16]: math.factorial(54)
Out[16]: 230843697339241380472092742683027581083278564571807941132288000000000000
In [17]: math.factorial(-5)
ValueError Traceback (most recent call last)
<ipython-input-17-41df25434f56> in <module>()
----> 1 math.factorial(-5)
ValueError: factorial() not defined for negative values
In [18]: math.factorial(12.3)
ValueError Traceback (most recent call last)
<ipython-input-18-a8322e5b2a52> in <module>()
----> 1 math.factorial(12.3)
ValueError: factorial() only accepts integral values
In [19]:

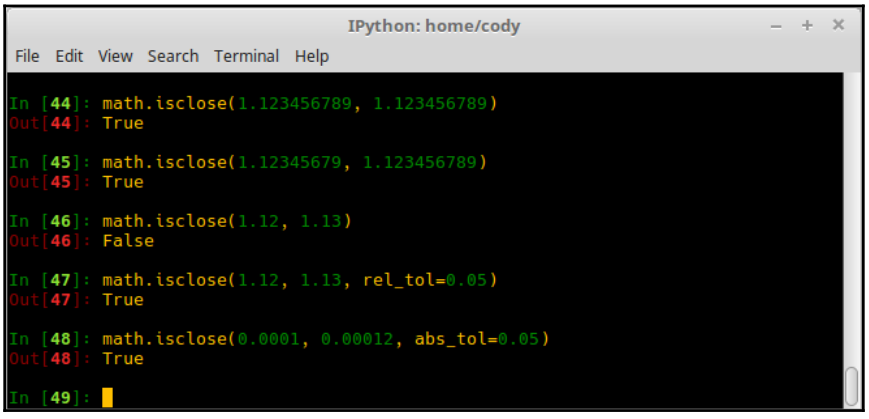

IPython: home/cody
File Edit View Search Terminal Help
In [19]: math.floor(14)
Out[19]: 14
In [20]: math.floor(3.0)
Out[20]: 3
In [21]: math.floor(5.9)
Out[21]: 5
In [22]:

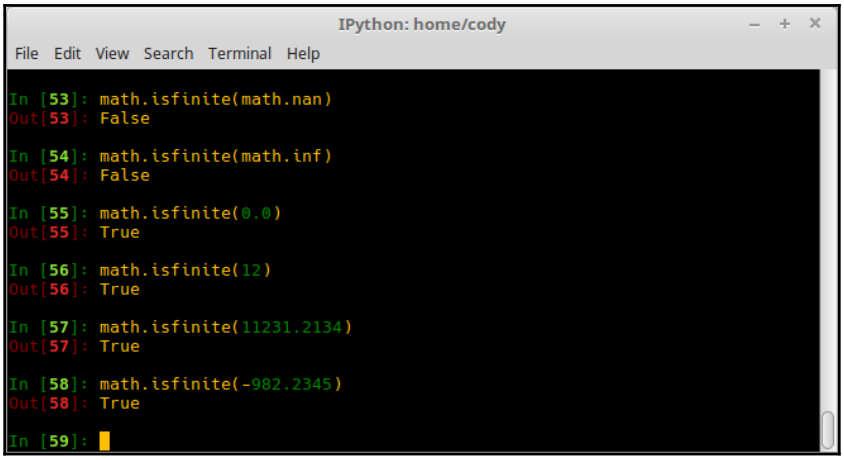

IPython: home/cody
File Edit View Search Terminal Help
In [22]: math.fmod(3, 8)
Out[22]: 3.0
In [23]: math.fmod(4.2, 10.9)
Out[23]: 4.2
In [24]: math.fmod(-1e-100, 1e100)
Out[24]: -1e-100
In [25]: -1e100 % 1e100
Out[25]: 0.0
In [26]:

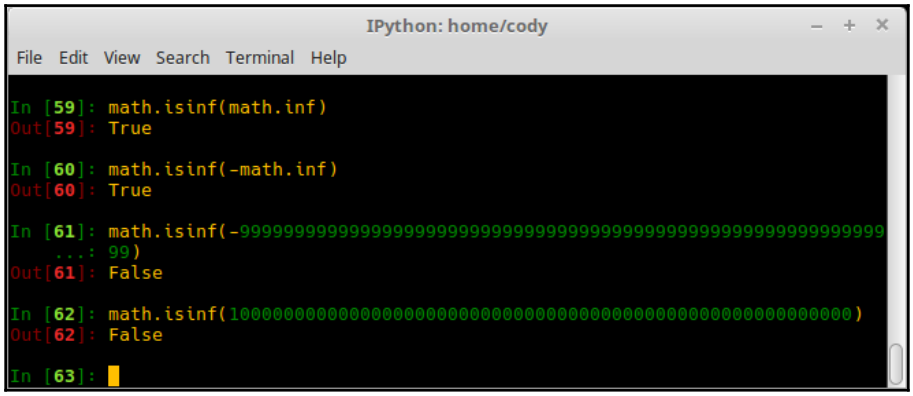

IPython: home/cody
File Edit View Search Terminal Help
In [27]: math.frexp(15)
Out[27]: (0.9375, 4)
In [28]: math.frexp(34.2)
Out[28]: (0.534375, 6)
In [29]: .9375 * 2**4
Out[29]: 15.0
In [30]: .534375 * 2**6
Out[30]: 34.2
In [31]:

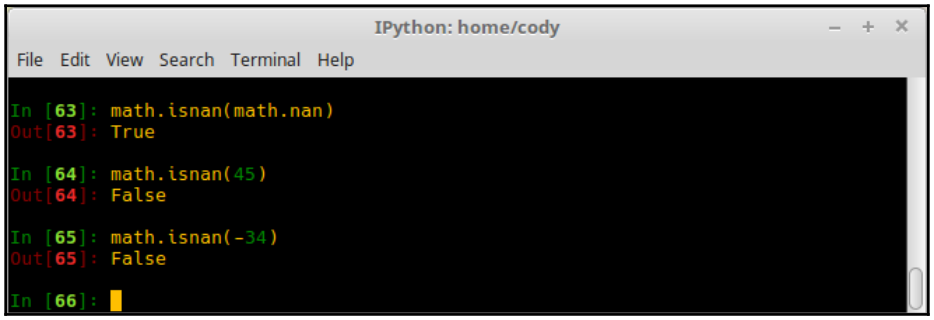

IPython: home/cody
File Edit View Search Terminal Help
In [31]: math.fsum([.1, .1, .1, .1, .1, .1, .1, .1, .1, .1])
Out[31]: 1.0
In [32]: sum([.1, .1, .1, .1, .1, .1, .1, .1, .1, .1])
Out[32]: 0.9999999999999999
In [33]:

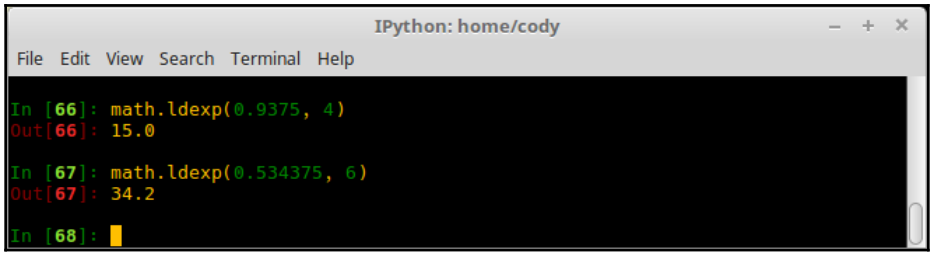

IPython: home/cody
File Edit View Search Terminal Help
In [37]: math.gcd(10, 0)
Out[37]: 10
In [38]: math.gcd(0, 0)
Out[38]: 0
In [39]: math.gcd(12, 72)
Out[39]: 12
In [40]: math.gcd(15, 50)
Out[40]: 5
In [41]:

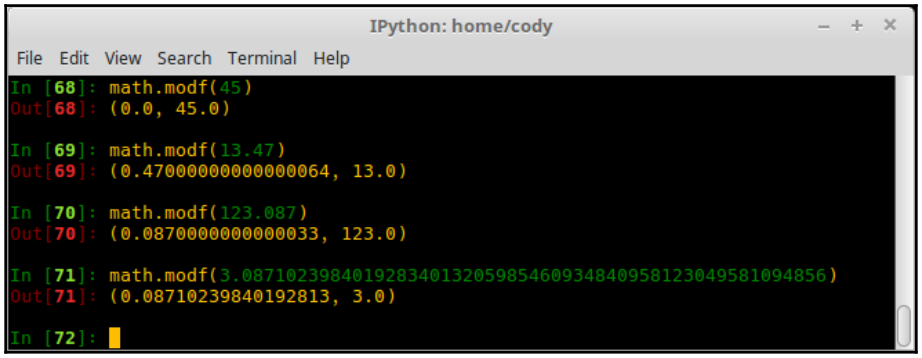

IPython: home/cody
File Edit View Search Terminal Help
In [44]: math.isclose(1.123456789, 1.123456789)
Out[44]: True
In [45]: math.isclose(1.12345679, 1.123456789)
Out[45]: True
In [46]: math.isclose(1.12, 1.13)
Out[46]: False
In [47]: math.isclose(1.12, 1.13, rel_tol=0.05)
Out[47]: True
In [48]: math.isclose(0.0001, 0.00012, abs_tol=0.05)
Out[48]: True
In [49]:

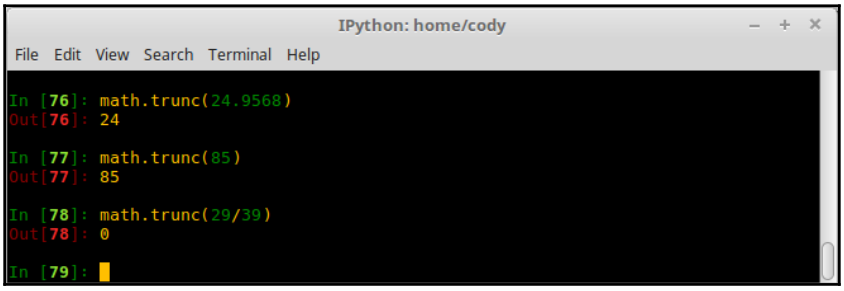

IPython: home/cody
File Edit View Search Terminal Help
In [53]: math.isfinite(math.nan)
Out[53]: False
In [54]: math.isfinite(math.inf)
Out[54]: False
In [55]: math.isfinite(0.0)
Out[55]: True
In [56]: math.isfinite(12)
Out[56]: True
In [57]: math.isfinite(11231.2134)
Out[57]: True
In [58]: math.isfinite(-982.2345)
Out[58]: True
In [59]:

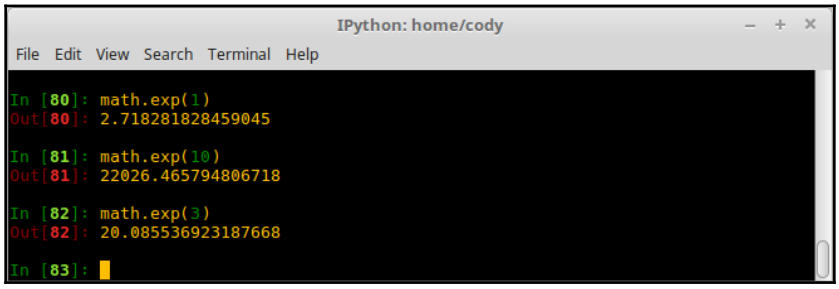

IPython: home/cody
File Edit View Search Terminal Help
In [59]: math.isinf(math.inf)
Out[59]: True
In [60]: math.isinf(-math.inf)
Out[60]: True
In [61]: math.isinf(-999999999999999999999999999999999999999999999999999999999999999999999999
...: 99)
Out[61]: False
In [62]: math.isinf(10000000000000000000000000000000000000000000000000000000000000000)
Out[62]: False
In [63]:


IPython: home/cody
File Edit View Search Terminal Help
In [63]: math.isnan(math.nan)
Out[63]: True
In [64]: math.isnan(45)
Out[64]: False
In [65]: math.isnan(-34)
Out[65]: False
In [66]:


IPython: home/cody
File Edit View Search Terminal Help
In [66]: math.ldexp(0.9375, 4)
Out[66]: 15.0
In [67]: math.ldexp(0.534375, 6)
Out[67]: 34.2
In [68]:


IPython: home/cody
File Edit View Search Terminal Help
In [68]: math.modf(45)
Out[68]: (0.0, 45.0)
In [69]: math.modf(13.47)
Out[69]: (0.47000000000000064, 13.0)
In [70]: math.modf(123.087)
Out[70]: (0.0870000000000033, 123.0)
In [71]: math.modf(3.0871023984019283401320598546093484095812304958109485 6)
Out[71]: (0.08710239840192813, 3.0)
In [72]:

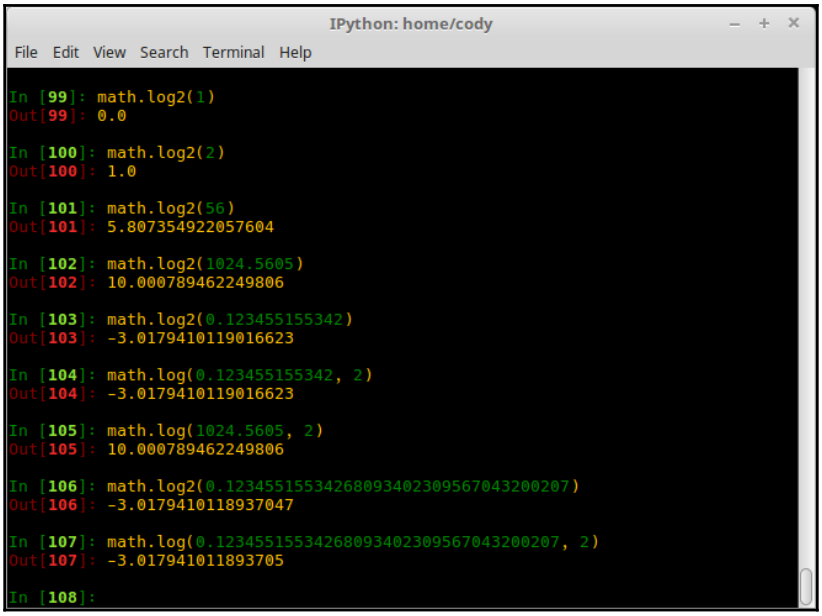

IPython: home/cody
File Edit View Search Terminal Help
In [76]: math.trunc(24.9568)
Out[76]: 24
In [77]: math.trunc(85)
Out[77]: 85
In [78]: math.trunc(29/39)
Out[78]: 0
In [79]:

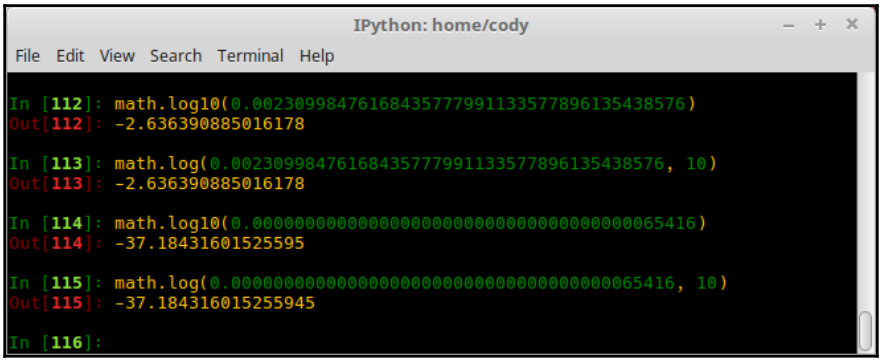

IPython: home/cody
File Edit View Search Terminal Help
In [80]: math.exp(1)
Out[80]: 2.718281828459045
In [81]: math.exp(10)
Out[81]: 22026.465794806718
In [82]: math.exp(3)
Out[82]: 20.085536923187668
In [83]:

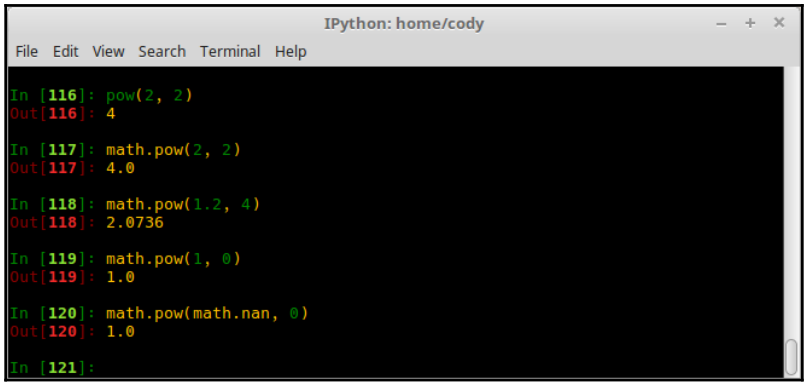

IPython: home/cody
File Edit View Search Terminal Help
In [88]: math.exp(1e-9) - 1
Out[88]: 1.000000082740371e-09
In [89]: math.expm1(1e-9)
Out[89]: 1.0000000005000001e-09
In [90]:

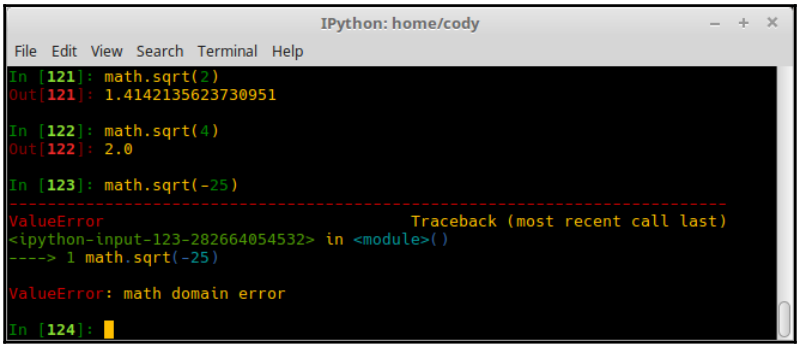

IPython: home/cody
File Edit View Search Terminal Help
In [93]: math.log(2)
Out[93]: 0.6931471805599453
In [94]: math.log(2, 10)
Out[94]: 0.30102999566398114
In [95]: math.log(2, 2)
Out[95]: 1.0
In [96]:

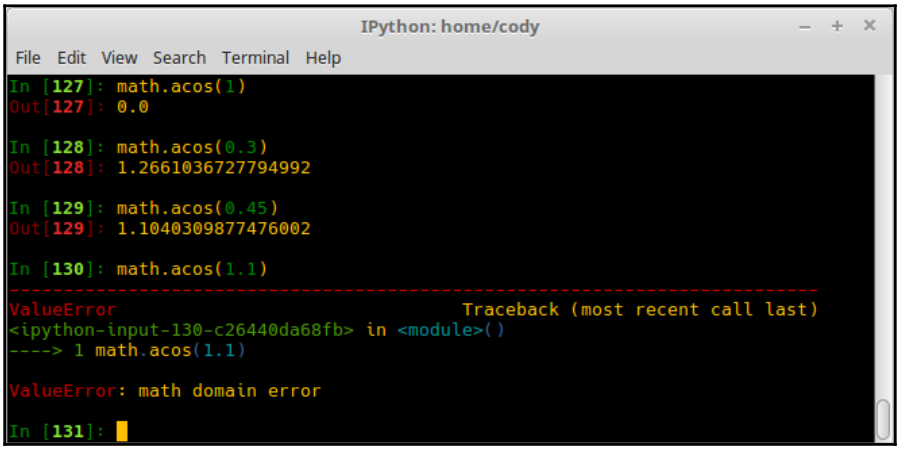

IPython: home/cody
File Edit View Search Terminal Help
In [96]: math.log1p(1)
Out[96]: 0.6931471805599453
In [97]: math.log1p(10)
Out[97]: 2.3978952727983707
In [98]: math.log1p(0.0000000000003)
Out[98]: 2.99999999999955e-13
In [99]:

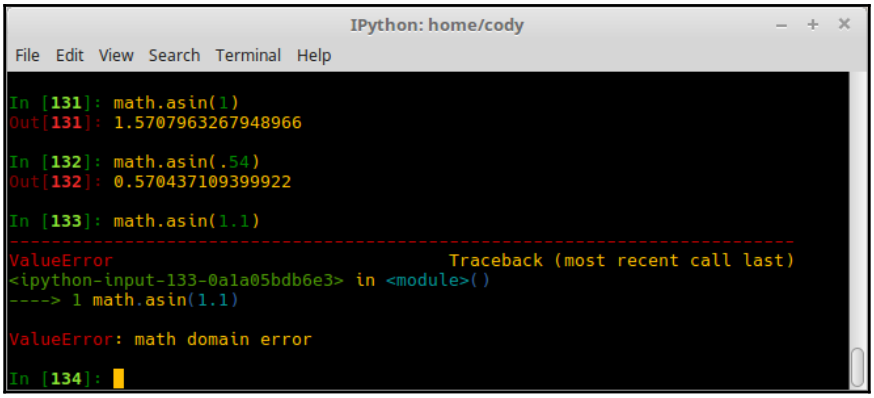

IPython: home/cody
File Edit View Search Terminal Help
In [99]: math.log2(1)
Out[99]: 0.0
In [100]: math.log2(2)
Out[100]: 1.0
In [101]: math.log2(56)
Out[101]: 5.807354922057604
In [102]: math.log2(1024.5605)
Out[102]: 10.000789462249806
In [103]: math.log2(0.123455155342)
Out[103]: -3.0179410119016623
In [104]: math.log(0.123455155342, 2)
Out[104]: -3.0179410119016623
In [105]: math.log(1024.5605, 2)
Out[105]: 10.000789462249806
In [106]: math.log2(0.12345515534268093402309567043200207)
Out[106]: -3.0179410118937047
In [107]: math.log(0.12345515534268093402309567043200207, 2)
Out[107]: -3.017941011893705
In [108]:

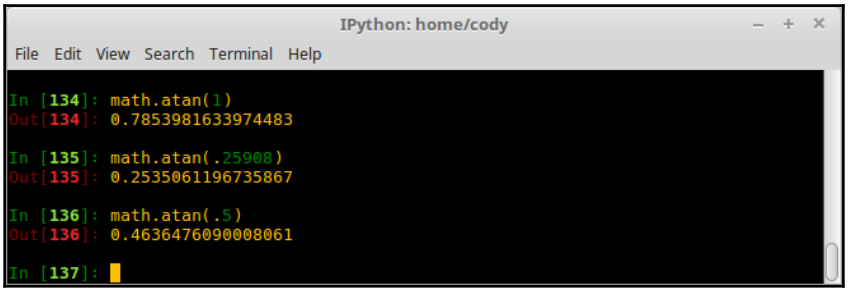

IPython: home/cody
File Edit View Search Terminal Help
In [112]: math.log10(0.0023099847616843577799113357789613543857 6)
Out[112]: -2.636390885016178
In [113]: math.log(0.0023099847616843577799113357789613543857 6, 10)
Out[113]: -2.636390885016178
In [114]: math.log10(0.0000000000000000000000000000000000000065416)
Out[114]: -37.18431601525595
In [115]: math.log(0.0000000000000000000000000000000000000065416, 10)
Out[115]: -37.184316015255945
In [116]:

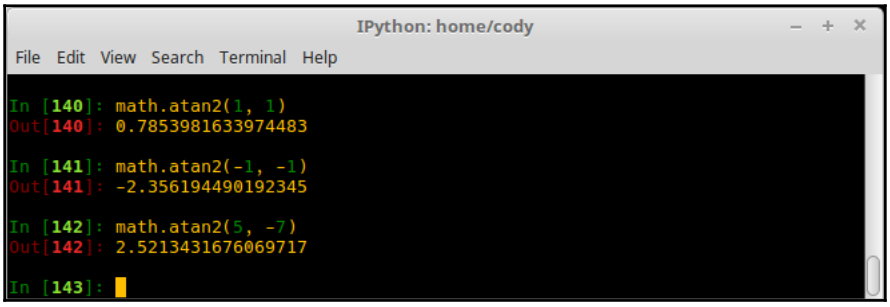

IPython: home/cody
File Edit View Search Terminal Help
In [116]: pow(2, 2)
Out[116]: 4
In [117]: math.pow(2, 2)
Out[117]: 4.0
In [118]: math.pow(1.2, 4)
Out[118]: 2.0736
In [119]: math.pow(1, 0)
Out[119]: 1.0
In [120]: math.pow(math.nan, 0)
Out[120]: 1.0
In [121]:

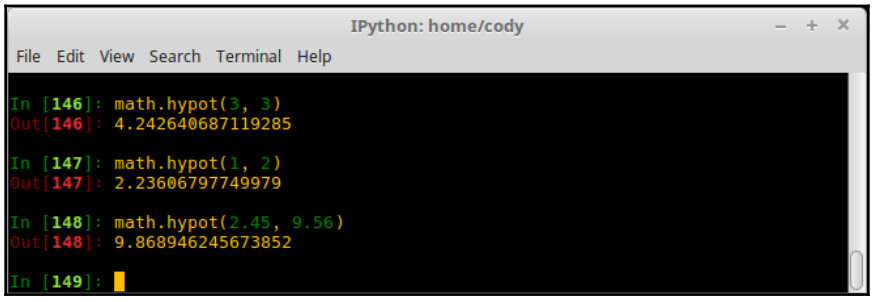

IPython: home/cody
File Edit View Search Terminal Help
In [121]: math.sqrt(2)
Out[121]: 1.4142135623730951
In [122]: math.sqrt(4)
Out[122]: 2.0
In [123]: math.sqrt(-25)
ValueError Traceback (most recent call last)
<ipython-input-123-282664054532> in <module>()
----> 1 math.sqrt(-25)
ValueError: math domain error
In [124]:

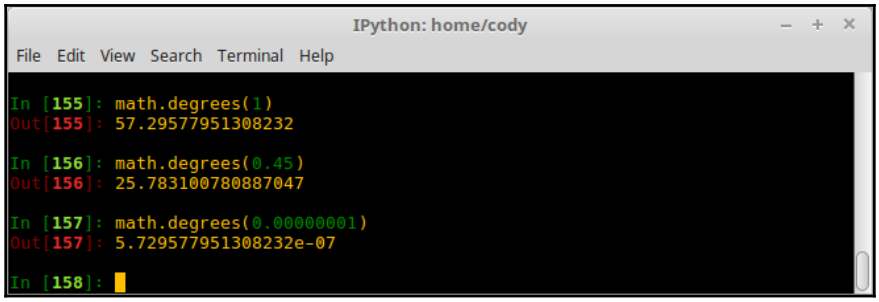

IPython: home/cody
File Edit View Search Terminal Help
In [127]: math.acos(1)
Out[127]: 0.0
In [128]: math.acos(0.3)
Out[128]: 1.2661036727794992
In [129]: math.acos(0.45)
Out[129]: 1.1040309877476002
In [130]: math.acos(1.1)
ValueError Traceback (most recent call last)
<ipython-input-130-c26440da68fb> in <module>()
----> 1 math.acos(1.1)
ValueError: math domain error
In [131]:

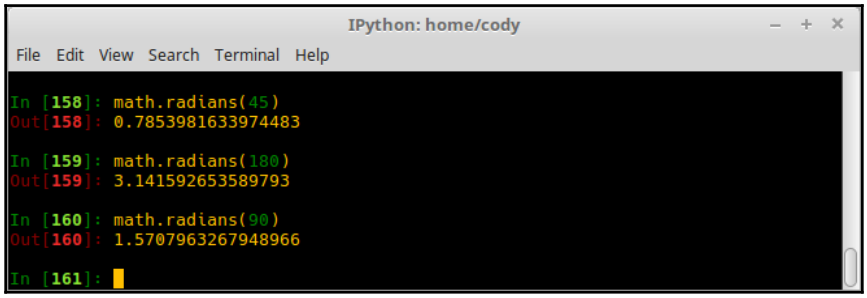

IPython: home/cody
File Edit View Search Terminal Help
In [131]: math.asin(1)
Out[131]: 1.5707963267948966
In [132]: math.asin(.54)
Out[132]: 0.570437109399922
In [133]: math.asin(1.1)
ValueError Traceback (most recent call last)
<ipython-input-133-0a1a05bdb6e3> in <module>()
----> 1 math.asin(1.1)
ValueError: math domain error
In [134]:

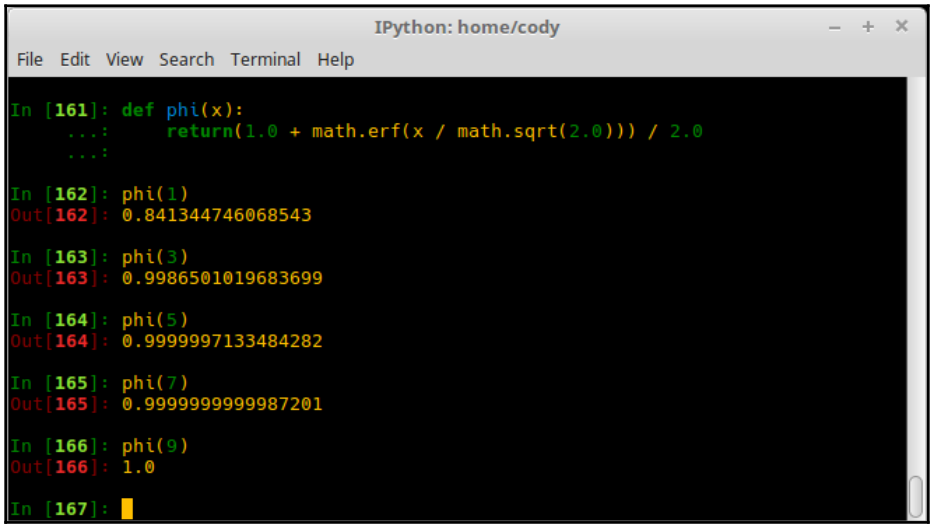

IPython: home/cody
File Edit View Search Terminal Help
In [134]: math.atan(1)
Out[134]: 0.7853981633974483
In [135]: math.atan(.25908)
Out[135]: 0.2535061196735867
In [136]: math.atan(.5)
Out[136]: 0.4636476090008061
In [137]:

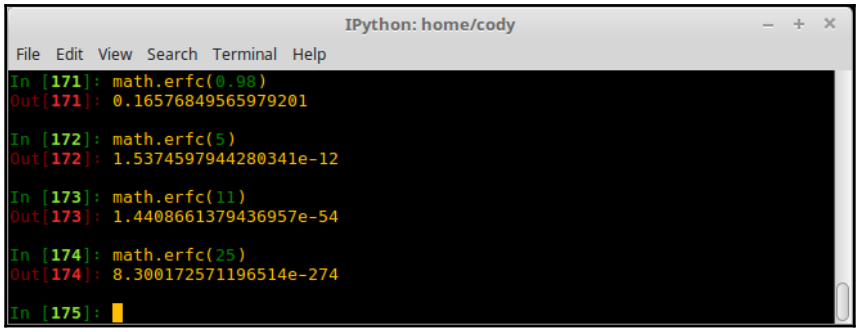

IPython: home/cody
File Edit View Search Terminal Help
In [140]: math.atan2(1, 1)
Out[140]: 0.7853981633974483
In [141]: math.atan2(-1, -1)
Out[141]: -2.356194490192345
In [142]: math.atan2(5, -7)
Out[142]: 2.5213431676069717
In [143]:

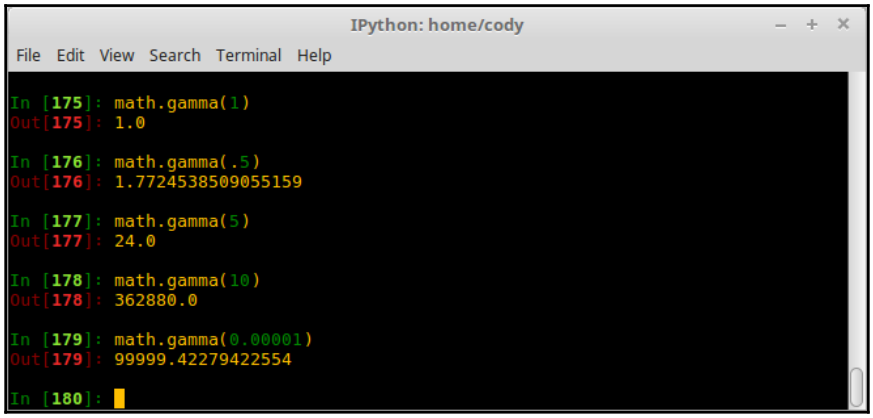

IPython: home/cody
File Edit View Search Terminal Help
In [146]: math.hypot(3, 3)
Out[146]: 4.242640687119285
In [147]: math.hypot(1, 2)
Out[147]: 2.23606797749979
In [148]: math.hypot(2.45, 9.56)
Out[148]: 9.868946245673852
In [149]:

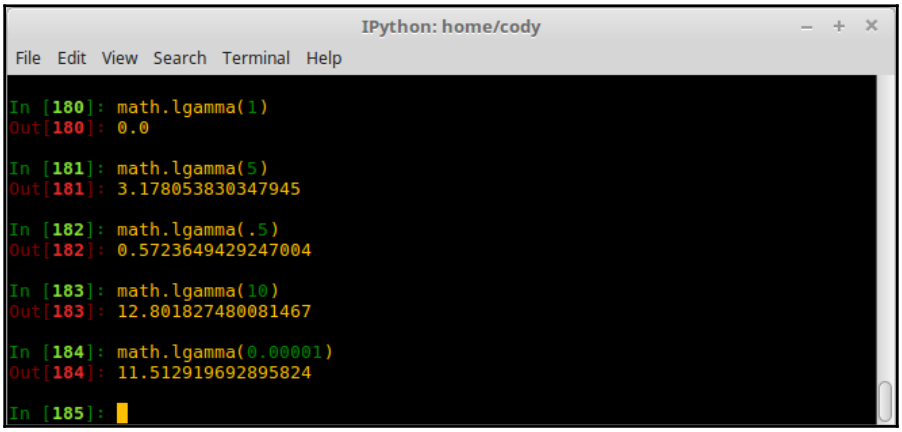

IPython: home/cody
File Edit View Search Terminal Help
In [155]: math.degrees(1)
Out[155]: 57.29577951308232
In [156]: math.degrees(0.45)
Out[156]: 25.783100780887047
In [157]: math.degrees(0.00000001)
Out[157]: 5.729577951308232e-07
In [158]:

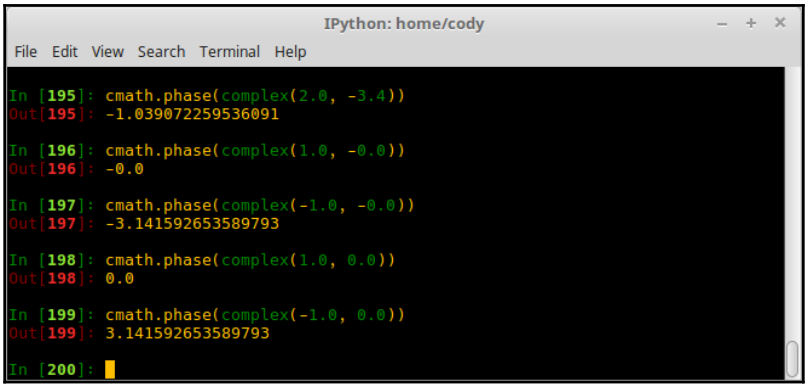

IPython: home/cody
File Edit View Search Terminal Help
In [158]: math.radians(45)
Out[158]: 0.7853981633974483
In [159]: math.radians(180)
Out[159]: 3.141592653589793
In [160]: math.radians(90)
Out[160]: 1.5707963267948966
In [161]:

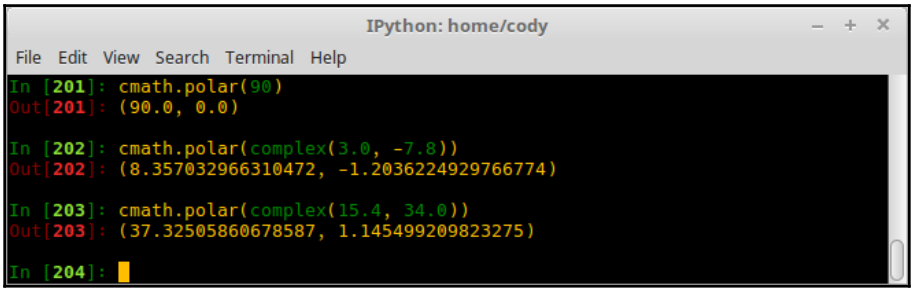

IPython: home/cody
File Edit View Search Terminal Help
In [161]: def phi(x):
...: return(1.0 + math.erf(x / math.sqrt(2.0))) / 2.0
...:
In [162]: phi(1)
Out[162]: 0.841344746068543
In [163]: phi(3)
Out[163]: 0.9986501019683699
In [164]: phi(5)
Out[164]: 0.9999997133484282
In [165]: phi(7)
Out[165]: 0.9999999999987201
In [166]: phi(9)
Out[166]: 1.0
In [167]:

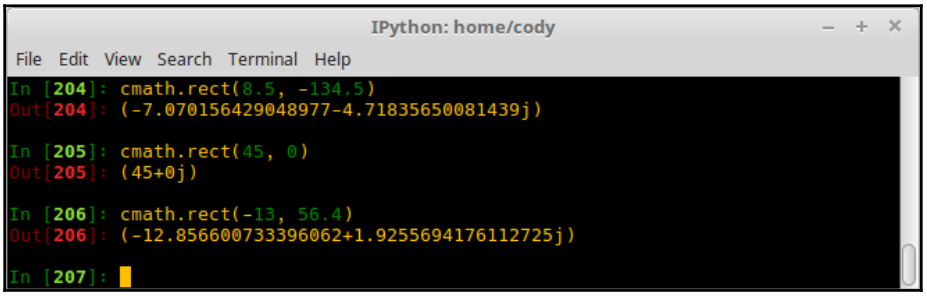

IPython: home/cody
File Edit View Search Terminal Help
In [171]: math.erfc(0.98)
Out[171]: 0.16576849565979201
In [172]: math.erfc(5)
Out[172]: 1.5374597944280341e-12
In [173]: math.erfc(11)
Out[173]: 1.4408661379436957e-54
In [174]: math.erfc(25)
Out[174]: 8.300172571196514e-274
In [175]:


IPython: home/cody
File Edit View Search Terminal Help
In [175]: math.gamma(1)
Out[175]: 1.0
In [176]: math.gamma(.5)
Out[176]: 1.7724538509055159
In [177]: math.gamma(5)
Out[177]: 24.0
In [178]: math.gamma(10)
Out[178]: 362880.0
In [179]: math.gamma(0.00001)
Out[179]: 99999.42279422554
In [180]:

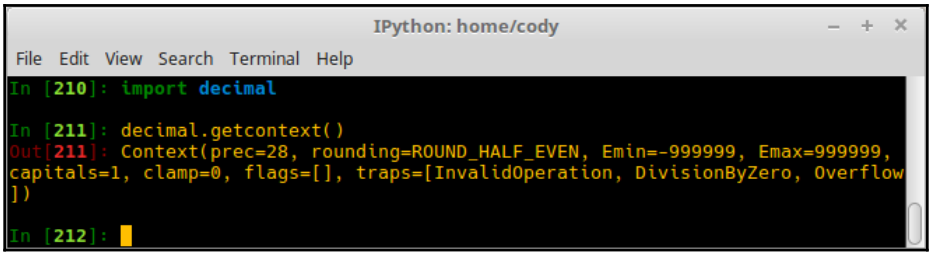

IPython: home/cody
File Edit View Search Terminal Help
In [180]: math.lgamma(1)
Out[180]: 0.0
In [181]: math.lgamma(5)
Out[181]: 3.178053830347945
In [182]: math.lgamma(.5)
Out[182]: 0.5723649429247004
In [183]: math.lgamma(10)
Out[183]: 12.801827480081467
In [184]: math.lgamma(0.00001)
Out[184]: 11.512919692895824
In [185]:

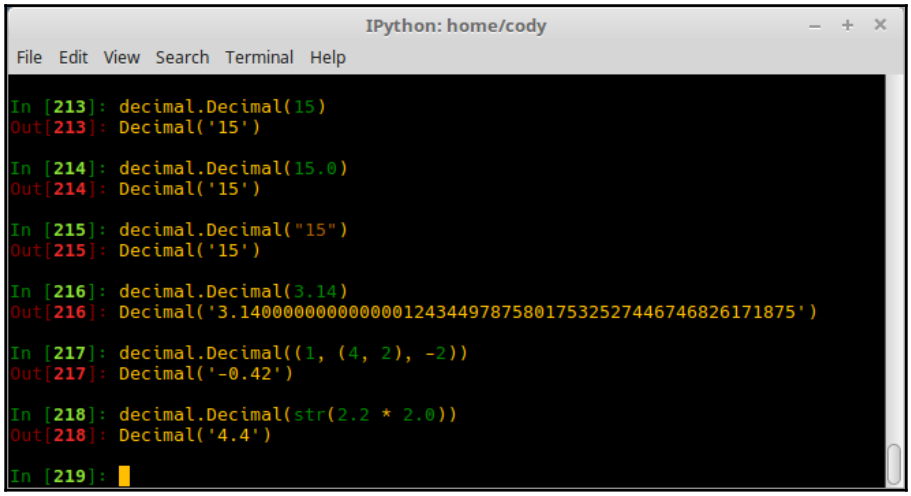

IPython: home/cody
File Edit View Search Terminal Help
In [195]: cmath.phase(complex(2.0, -3.4))
Out[195]: -1.039072259536091
In [196]: cmath.phase(complex(1.0, -0.0))
Out[196]: -0.0
In [197]: cmath.phase(complex(-1.0, -0.0))
Out[197]: -3.141592653589793
In [198]: cmath.phase(complex(1.0, 0.0))
Out[198]: 0.0
In [199]: cmath.phase(complex(-1.0, 0.0))
Out[199]: 3.141592653589793
In [200]:

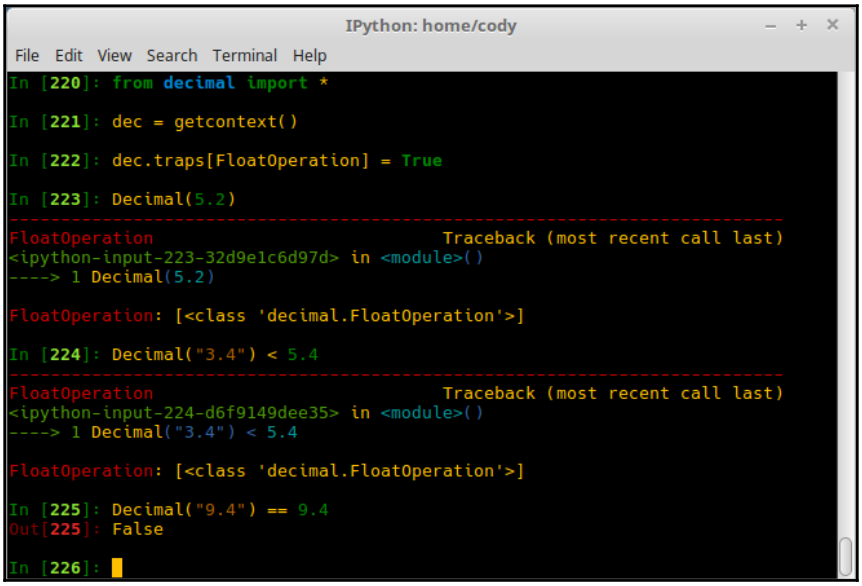

IPython: home/cody
File Edit View Search Terminal Help
In [201]: cmath.polar(90)
Out[201]: (90.0, 0.0)
In [202]: cmath.polar(complex(3.0, -7.8))
Out[202]: (8.357032966310472, -1.2036224929766774)
In [203]: cmath.polar(complex(15.4, 34.0))
Out[203]: (37.32505860678587, 1.145499209823275)
In [204]:

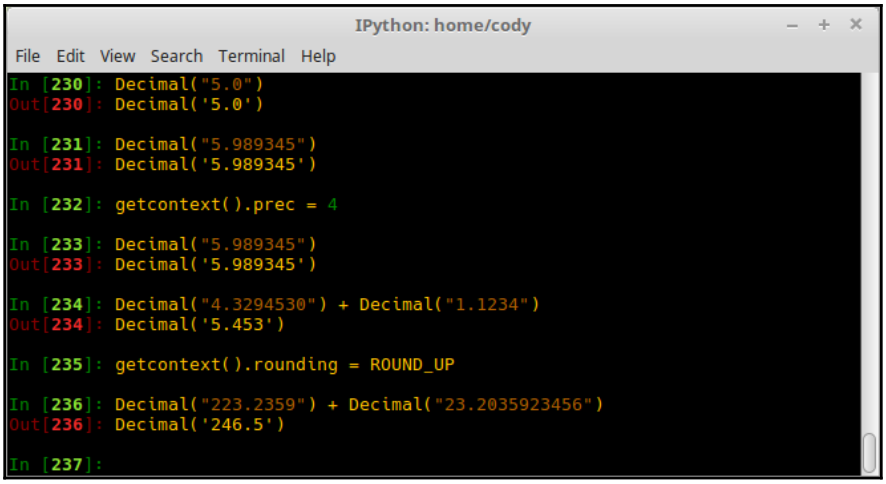

IPython: home/cody
File Edit View Search Terminal Help
In [204]: cmath.rect(8.5, -134.5)
Out[204]: (-7.070156429048977-4.71835650081439j)
In [205]: cmath.rect(45, 0)
Out[205]: (45+0j)
In [206]: cmath.rect(-13, 56.4)
Out[206]: (-12.856600733396062+1.9255694176112725j)
In [207]:

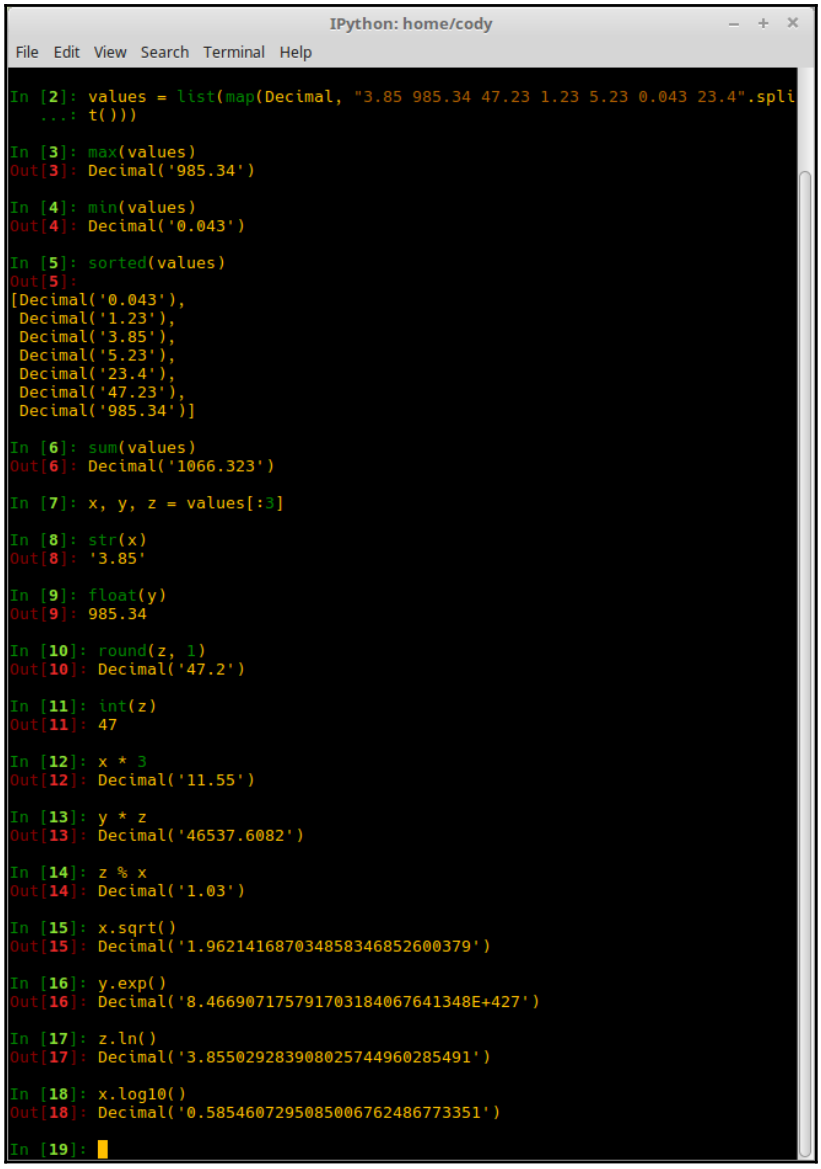

IPython: home/cody
File Edit View Search Terminal Help
In [207]: 1.1 + 2.2
Out[207]: 3.3000000000000003
In [208]:

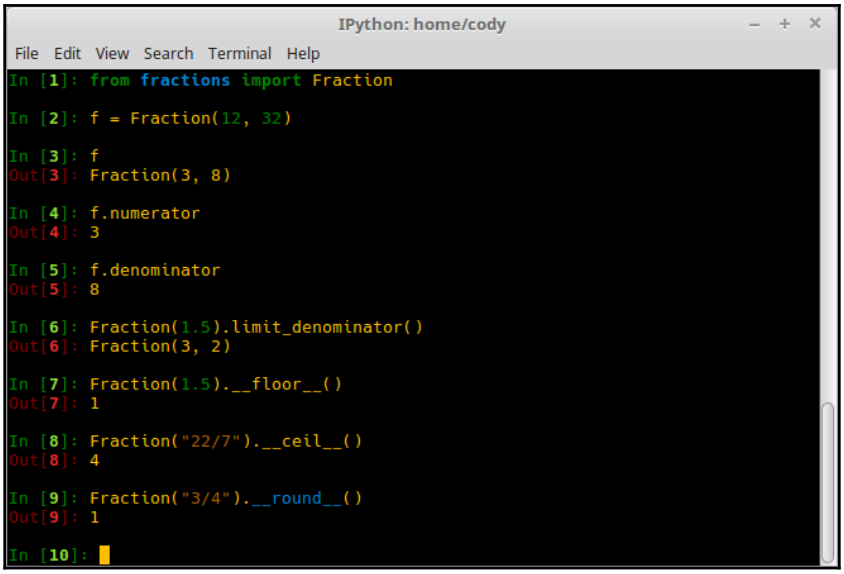

IPython: home/cody
File Edit View Search Terminal Help
In [210]: import decimal
In [211]: decimal.getcontext()
Out[211]: Context(prec=28, rounding=ROUND_HALF_EVEN, Emin=-999999, Emax=999999,
capitals=1, clamp=0, flags=[], traps=[InvalidOperation, DivisionByZero, Overflow
])
In [212]:

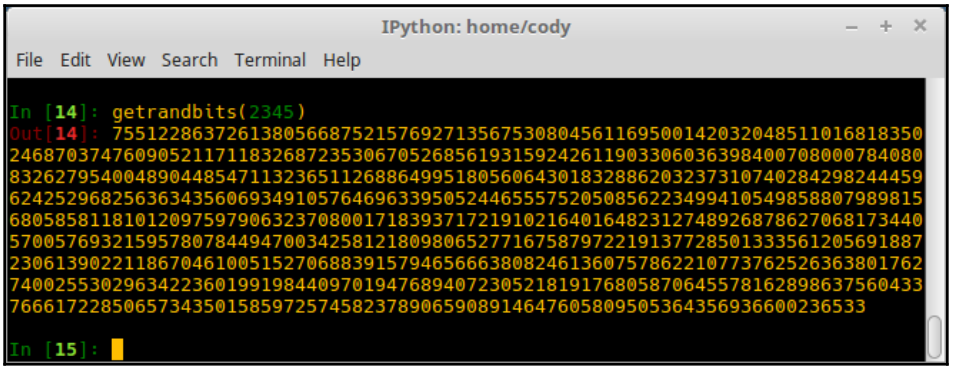

IPython: home/cody
File Edit View Search Terminal Help
In [213]: decimal.Decimal(15)
Out[213]: Decimal('15')
In [214]: decimal.Decimal(15.0)
Out[214]: Decimal('15')
In [215]: decimal.Decimal("15")
Out[215]: Decimal('15')
In [216]: decimal.Decimal(3.14)
Out[216]: Decimal('3.140000000000001243449787580175325274467468261718 75')
In [217]: decimal.Decimal((1, (4, 2), -2))
Out[217]: Decimal('-0.42')
In [218]: decimal.Decimal(str(2.2 * 2.0))
Out[218]: Decimal('4.4')
In [219]:

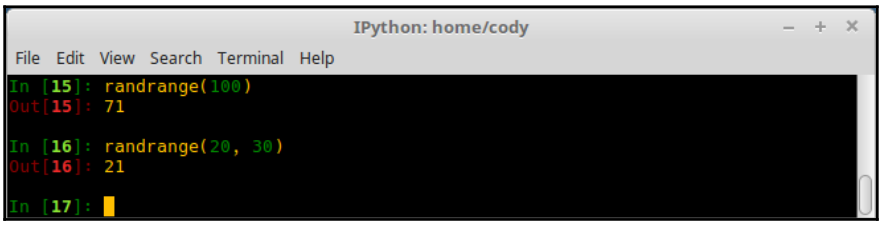

IPython: home/cody
File Edit View Search Terminal Help
In [220]: from decimal import *
In [221]: dec = getcontext()
In [222]: dec.traps[FloatOperation] = True
In [223]: Decimal(5.2)
FloatOperation Traceback (most recent call last)
<ipython-input-223-32d9e1c6d97d> in <module>()
----> 1 Decimal(5.2)
FloatOperation: [<class 'decimal.FloatOperation'>]
In [224]: Decimal("3.4") < 5.4
FloatOperation Traceback (most recent call last)
<ipython-input-224-d6f9149dee35> in <module>()
----> 1 Decimal("3.4") < 5.4
FloatOperation: [<class 'decimal.FloatOperation'>]
In [225]: Decimal("9.4") == 9.4
Out[225]: False
In [226]:

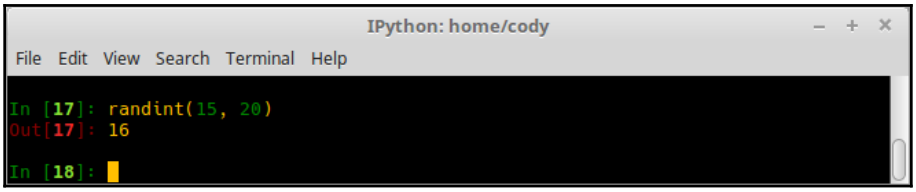

IPython: home/cody
File Edit View Search Terminal Help
In [230]: Decimal("5.0")
Out[230]: Decimal('5.0')
In [231]: Decimal("5.989345")
Out[231]: Decimal('5.989345')
In [232]: getcontext().prec = 4
In [233]: Decimal("5.989345")
Out[233]: Decimal('5.989345')
In [234]: Decimal("4.3294530") + Decimal("1.1234")
Out[234]: Decimal('5.453')
In [235]: getcontext().rounding = ROUND_UP
In [236]: Decimal("223.2359") + Decimal("23.2035923456")
Out[236]: Decimal('246.5')
In [237]:

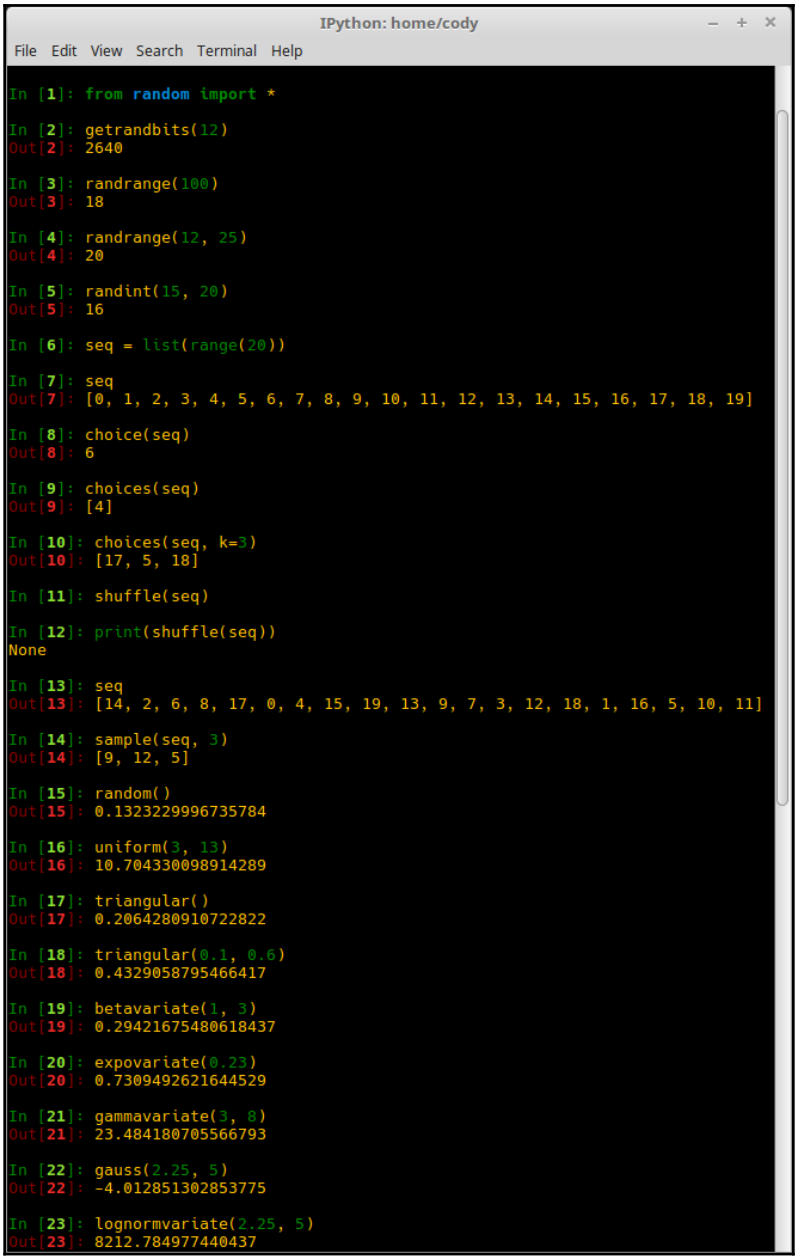

IPython: home/cody
File Edit View Search Terminal Help
In [2]: values = list(map(Decimal, "3.85 985.34 47.23 1.23 5.23 0.043 23.4".spli
...: t()))
In [3]: max(values)
Out[3]: Decimal('985.34')
In [4]: min(values)
Out[4]: Decimal('0.043')
In [5]: sorted(values)
Out[5]:
[Decimal('0.043'),
Decimal('1.23'),
Decimal('3.85'),
Decimal('5.23'),
Decimal('23.4'),
Decimal('47.23'),
Decimal('985.34')]
In [6]: sum(values)
Out[6]: Decimal('1066.323')
In [7]: x, y, z = values[:3]
In [8]: str(x)
Out[8]: '3.85'
In [9]: float(y)
Out[9]: 985.34
In [10]: round(z, 1)
Out[10]: Decimal('47.2')
In [11]: int(z)
Out[11]: 47
In [12]: x * 3
Out[12]: Decimal('11.55')
In [13]: y * z
Out[13]: Decimal('46537.6082')
In [14]: z % x
Out[14]: Decimal('1.03')
In [15]: x.sqrt()
Out[15]: Decimal('1.962141687034858346852600379')
In [16]: y.exp()
Out[16]: Decimal('8.466907175791703184067641348E+427')
In [17]: z.ln()
Out[17]: Decimal('3.855029283908025744960285491')
In [18]: x.log10()
Out[18]: Decimal('0.585460729508500676248677335 1')
In [19]:

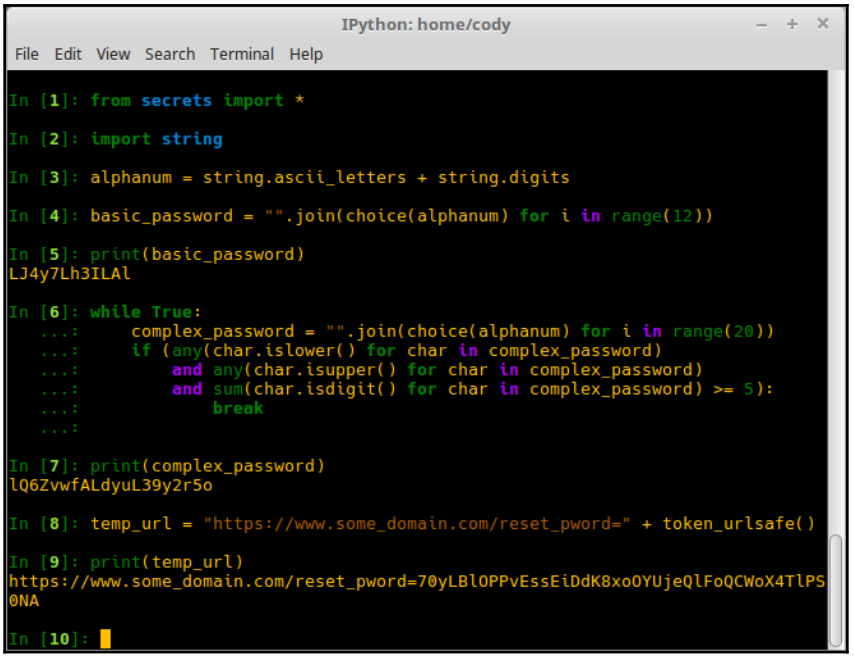

IPython: home/cody
File Edit View Search Terminal Help
In [1]: from fractions import Fraction
In [2]: f = Fraction(12, 32)
In [3]: f
Out[3]: Fraction(3, 8)
In [4]: f.numerator
Out[4]: 3
In [5]: f.denominator
Out[5]: 8
In [6]: Fraction(1.5).limit_denominator()
Out[6]: Fraction(3, 2)
In [7]: Fraction(1.5).__floor__()
Out[7]: 1
In [8]: Fraction("22/7").__ceil__()
Out[8]: 4
In [9]: Fraction("3/4").__round__()
Out[9]: 1
In [10]:

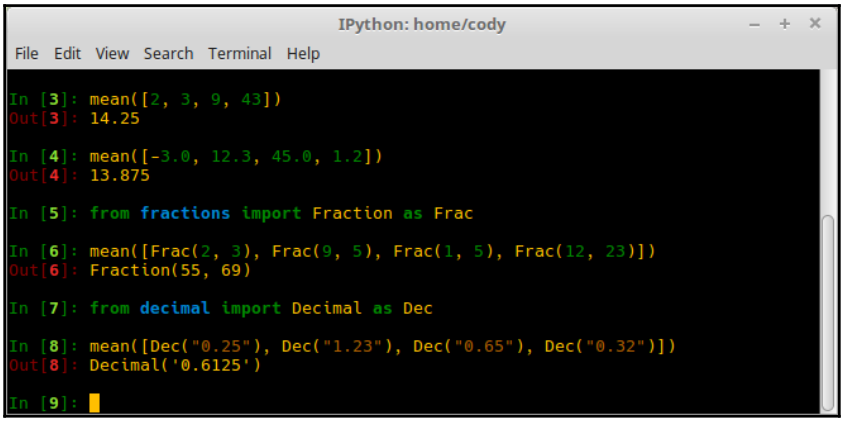

IPython: home/cody
File Edit View Search Terminal Help
In [14]: getrandbits(2345)
Out[14]: 7551228637261380566875215769271356753080456116950014203204851101681835024687037476090521171183268723530670526856193159242611903306036398400708000784080832627954004890448547113236511268864995180560643018328862032373107402842982444596242529682563634356069349105764696339505244655575205085622349941054985880798981568058581181012097597906323708001718393717219102164016482312748926878627068173440570057693215957807844947003425812180980652771675879722191377285013335612056918872306139022118670461005152706883915794656663808246136075786221077376252636380176274002553029634223601991984409701947689407230521819176805870645578162898637560433766617228506573435015859725745823789065908914647605809505364356936600236533
In [15]:

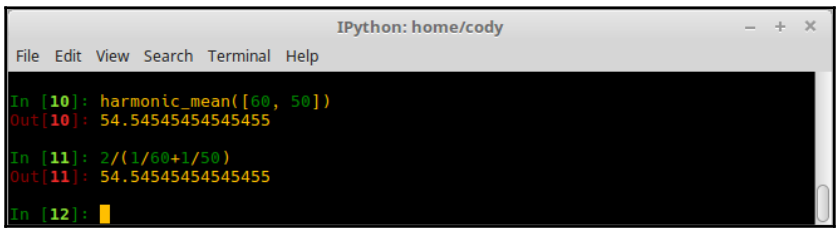

IPython: home/cody
File Edit View Search Terminal Help
In [15]: randrange(100)
Out[15]: 71
In [16]: randrange(20, 30)
Out[16]: 21
In [17]:

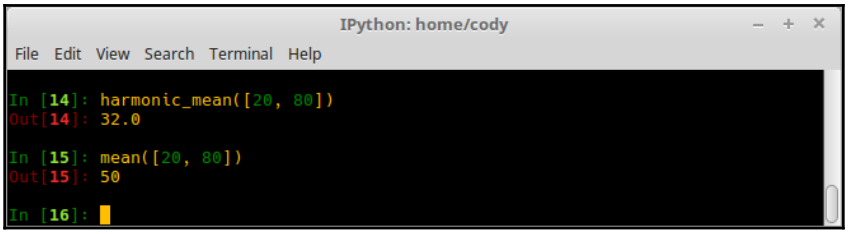

IPython: home/cody
File Edit View Search Terminal Help
In [17]: randint(15, 20)
Out[17]: 16
In [18]:

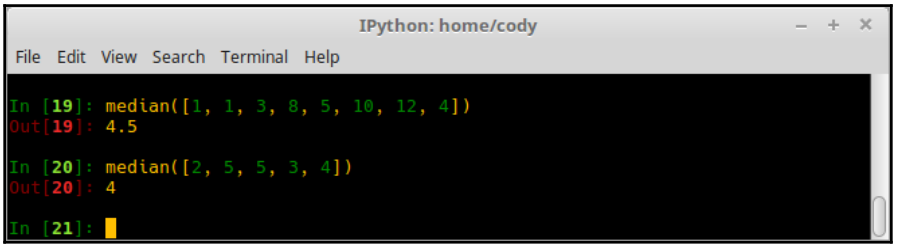

IPython: home/cody
File Edit View Search Terminal Help
In [1]: from random import *
In [2]: getrandbits(12)
Out[2]: 2640
In [3]: randrange(100)
Out[3]: 18
In [4]: randrange(12, 25)
Out[4]: 20
In [5]: randint(15, 20)
Out[5]: 16
In [6]: seq = list(range(20))
In [7]: seq
Out[7]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
In [8]: choice(seq)
Out[8]: 6
In [9]: choices(seq)
Out[9]: [4]
In [10]: choices(seq, k=3)
Out[10]: [17, 5, 18]
In [11]: shuffle(seq)
In [12]: print(shuffle(seq))
None
In [13]: seq
Out[13]: [14, 2, 6, 8, 17, 0, 4, 15, 19, 13, 9, 7, 3, 12, 18, 1, 16, 5, 10, 11]
In [14]: sample(seq, 3)
Out[14]: [9, 12, 5]
In [15]: random()
Out[15]: 0.1323229996735784
In [16]: uniform(3, 13)
Out[16]: 10.704330098914289
In [17]: triangular()
Out[17]: 0.2064280910722822
In [18]: triangular(0.1, 0.6)
Out[18]: 0.4329058795466417
In [19]: betavariate(1, 3)
Out[19]: 0.29421675480618437
In [20]: expovariate(0.23)
Out[20]: 0.7309492621644529
In [21]: gammavariate(3, 8)
Out[21]: 23.484180705566793
In [22]: gauss(2.25, 5)
Out[22]: -4.012851302853775
In [23]: lognormvariate(2.25, 5)
Out[23]: 8212.784977440437

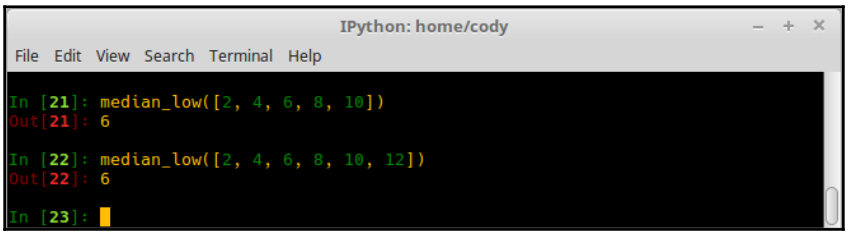

IPython: home/cody
File Edit View Search Terminal Help
In [1]: from secrets import *
In [2]: import string
In [3]: alphanum = string.ascii_letters + string.digits
In [4]: basic_password = "".join(choice(alphanum) for i in range(12))
In [5]: print(basic_password)
LJ4y7Lh3ILAl
In [6]: while True:
...: complex_password = "".join(choice(alphanum) for i in range(20))
...: if (any(char.islower() for char in complex_password)
...: and any(char.isupper() for char in complex_password)
...: and sum(char.isdigit() for char in complex_password) >= 5):
...: break
...:
In [7]: print(complex_password)
lQ6ZvwfALdyuL39y2r5o
In [8]: temp_url = "https://www.some_domain.com/reset_pword=" + token_urlsafe()
In [9]: print(temp_url)
https://www.some_domain.com/reset_pword=70yLBlOPPvEssEiDdK8xoOYUjeQlFoQCWoX4TlPS
0NA
In [10]:

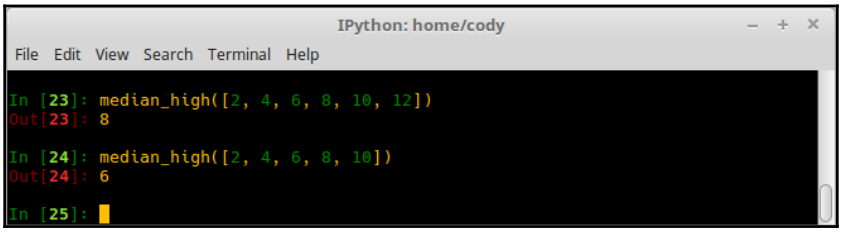

IPython: home/cody
File Edit View Search Terminal Help
In [3]: mean([2, 3, 9, 43])
Out[3]: 14.25
In [4]: mean([-3.0, 12.3, 45.0, 1.2])
Out[4]: 13.875
In [5]: from fractions import Fraction as Frac
In [6]: mean([Frac(2, 3), Frac(9, 5), Frac(1, 5), Frac(12, 23)])
Out[6]: Fraction(55, 69)
In [7]: from decimal import Decimal as Dec
In [8]: mean([Dec("0.25"), Dec("1.23"), Dec("0.65"), Dec("0.32")])
Out[8]: Decimal('0.6125')
In [9]:

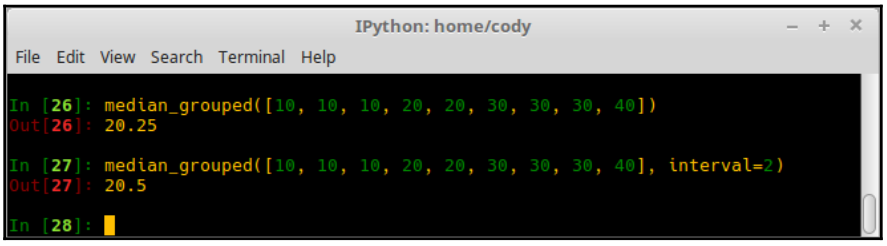

IPython: home/cody
File Edit View Search Terminal Help
In [10]: harmonic_mean([60, 50])
Out[10]: 54.54545454545455
In [11]: 2/(1/60+1/50)
Out[11]: 54.54545454545455
In [12]:

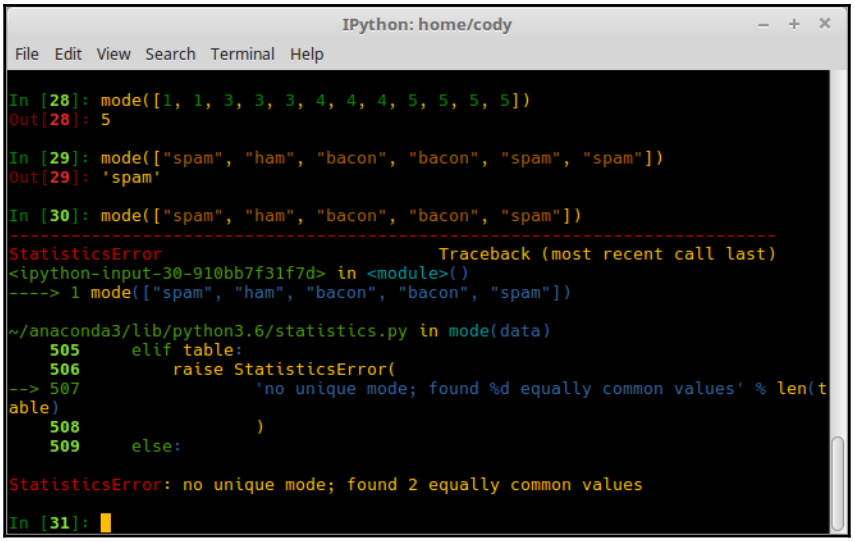

IPython: home/cody
File Edit View Search Terminal Help
In [14]: harmonic_mean([20, 80])
Out[14]: 32.0
In [15]: mean([20, 80])
Out[15]: 50
In [16]:

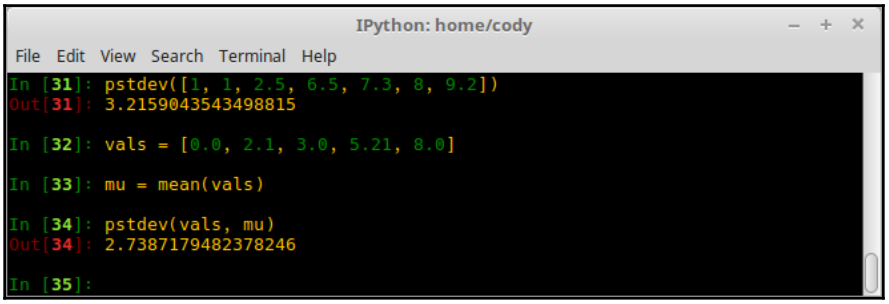

IPython: home/cody
File Edit View Search Terminal Help
In [19]: median([1, 1, 3, 8, 5, 10, 12, 4])
Out[19]: 4.5
In [20]: median([2, 5, 5, 3, 4])
Out[20]: 4
In [21]:


IPython: home/cody
File Edit View Search Terminal Help
In [21]: median_low([2, 4, 6, 8, 10])
Out[21]: 6
In [22]: median_low([2, 4, 6, 8, 10, 12])
Out[22]: 6
In [23]:

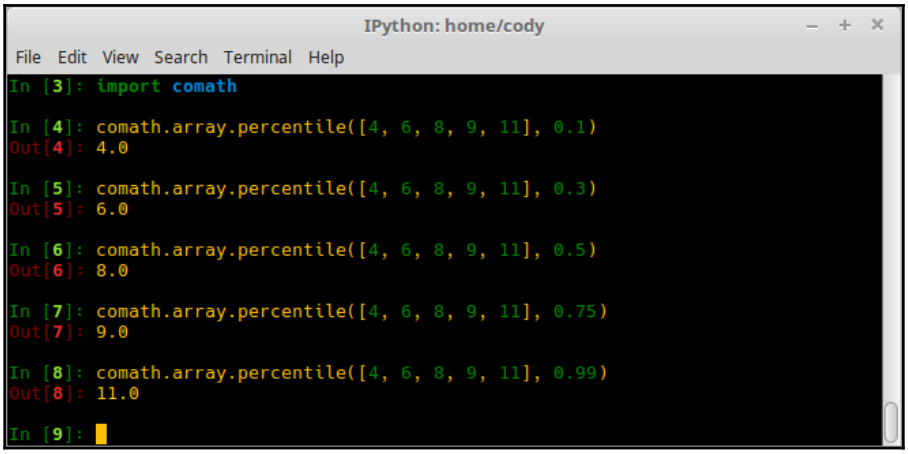

IPython: home/cody
File Edit View Search Terminal Help
In [23]: median_high([2, 4, 6, 8, 10, 12])
Out[23]: 8
In [24]: median_high([2, 4, 6, 8, 10])
Out[24]: 6
In [25]:

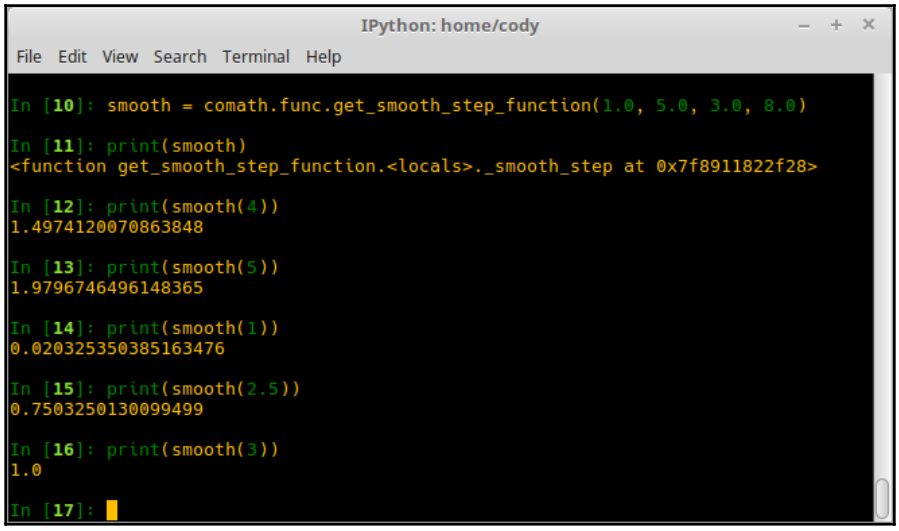

IPython: home/cody
File Edit View Search Terminal Help
In [26]: median_grouped([10, 10, 10, 20, 20, 30, 30, 30, 40])
Out[26]: 20.25
In [27]: median_grouped([10, 10, 10, 20, 20, 30, 30, 30, 40], interval=2)
Out[27]: 20.5
In [28]:

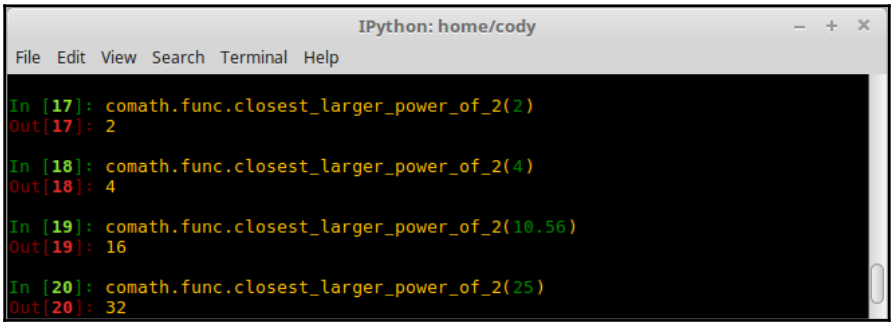

IPython: home/cody
File Edit View Search Terminal Help
In [28]: mode([1, 1, 3, 3, 3, 4, 4, 4, 5, 5, 5, 5])
Out[28]: 5
In [29]: mode(["spam", "ham", "bacon", "bacon", "spam", "spam"])
Out[29]: 'spam'
In [30]: mode(["spam", "ham", "bacon", "bacon", "spam"])
StatisticsError Traceback (most recent call last)
<ipython-input-30-910bb7f31f7d> in <module>()
----> 1 mode(["spam", "ham", "bacon", "bacon", "spam"])
~/anaconda3/lib/python3.6/statistics.py in mode(data)
505 elif table:
506 raise StatisticsError(
--> 507 'no unique mode; found %d equally common values' % len(table)
508)
509 else:
StatisticsError: no unique mode; found 2 equally common values
In [31]:

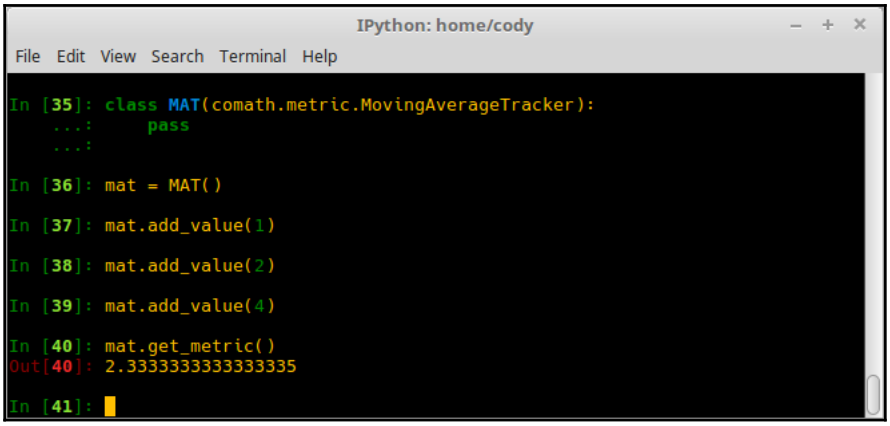

IPython: home/cody
File Edit View Search Terminal Help
In [31]: pstdev([1, 1, 2.5, 6.5, 7.3, 8, 9.2])
Out[31]: 3.2159043543498815
In [32]: vals = [0.0, 2.1, 3.0, 5.21, 8.0]
In [33]: mu = mean(vals)
In [34]: pstdev(vals, mu)
Out[34]: 2.7387179482378246
In [35]:


IPython: home/cody
File Edit View Search Terminal Help
In [35]: pvariance(vals)
Out[35]: 7.500576
In [36]: pvariance(vals, 5.0)
Out[36]: 7.500576
In [37]: pvariance(vals, 0.5)
Out[37]: 7.500576000000001
In [38]: pvariance(vals, 50)
Out[38]: 7.5005760000000095
In [39]:


IPython: home/cody
File Edit View Search Terminal Help
In [3]: import comath
In [4]: comath.array.percentile([4, 6, 8, 9, 11], 0.1)
Out[4]: 4.0
In [5]: comath.array.percentile([4, 6, 8, 9, 11], 0.3)
Out[5]: 6.0
In [6]: comath.array.percentile([4, 6, 8, 9, 11], 0.5)
Out[6]: 8.0
In [7]: comath.array.percentile([4, 6, 8, 9, 11], 0.75)
Out[7]: 9.0
In [8]: comath.array.percentile([4, 6, 8, 9, 11], 0.99)
Out[8]: 11.0
In [9]:

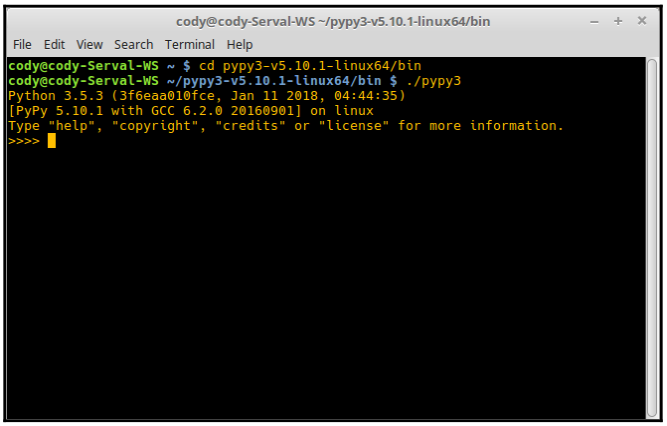

IPython: home/cody
File Edit View Search Terminal Help
In [10]: smooth = comath.func.get_smooth_step_function(1.0, 5.0, 3.0, 8.0)
In [11]: print(smooth)
<function get_smooth_step_function.<locals>._smooth_step at 0x7f8911822f28>
In [12]: print(smooth(4))
1.4974120070863848
In [13]: print(smooth(5))
1.9796746496148365
In [14]: print(smooth(1))
0.020325350385163476
In [15]: print(smooth(2.5))
0.7503250130099499
In [16]: print(smooth(3))
1.0
In [17]:

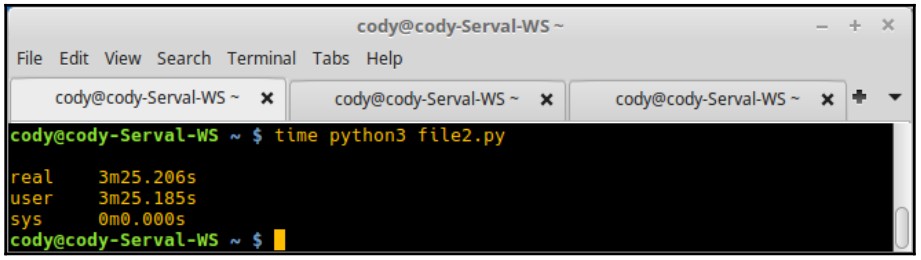

IPython: home/cody
File Edit View Search Terminal Help
In [17]: comath.func.closest_larger_power_of_2(2)
Out[17]: 2
In [18]: comath.func.closest_larger_power_of_2(4)
Out[18]: 4
In [19]: comath.func.closest_larger_power_of_2(10.56)
Out[19]: 16
In [20]: comath.func.closest_larger_power_of_2(25)
Out[20]: 32

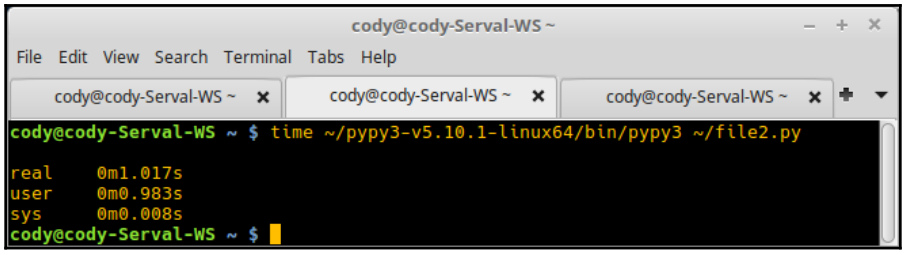

IPython: home/cody
File Edit View Search Terminal Help
In [35]: class MAT(comath.metric.MovingAverageTracker):
...: pass
...:
In [36]: mat = MAT()
In [37]: mat.add_value(1)
In [38]: mat.add_value(2)
In [39]: mat.add_value(4)
In [40]: mat.get_metric()
Out[40]: 2.3333333333333335
In [41]:

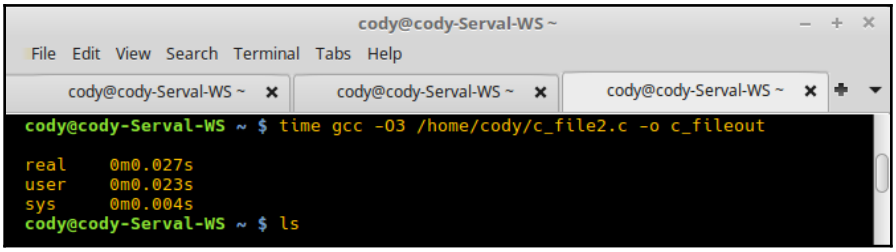

IPython: home/cody
File Edit View Search Terminal Help
In [50]: seg = comath.segment.LineSegment(2, 5)
In [51]: seg.contains(3)
Out[51]: True
In [52]: seg.contains(7)
Out[52]: False
In [53]: seg.intersection([2, 3])
Out[53]: {2, 3}
In [54]: seg.intersection([2, 3, 5, 6])
Out[54]: {2, 3, 5}
In [55]:

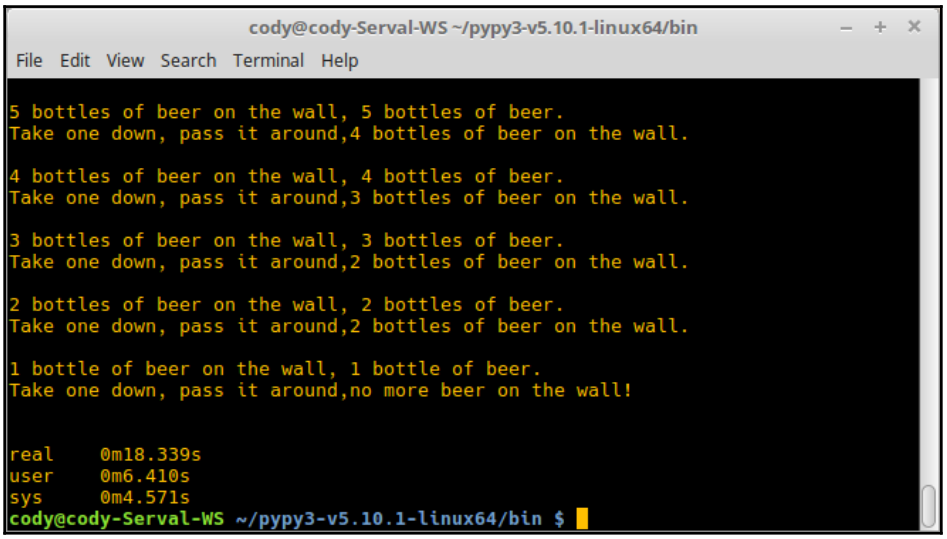
Chapter 7: Improving Python Performance with PyPy

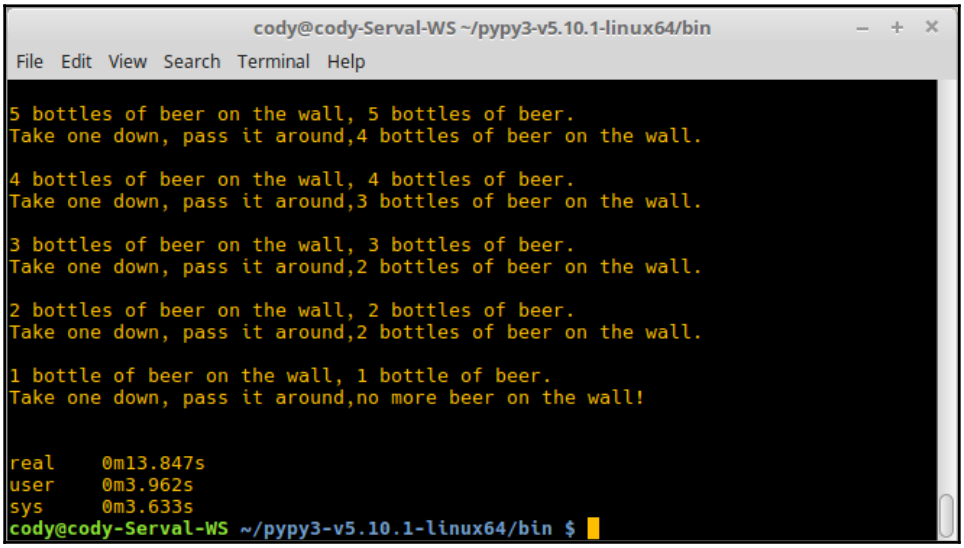


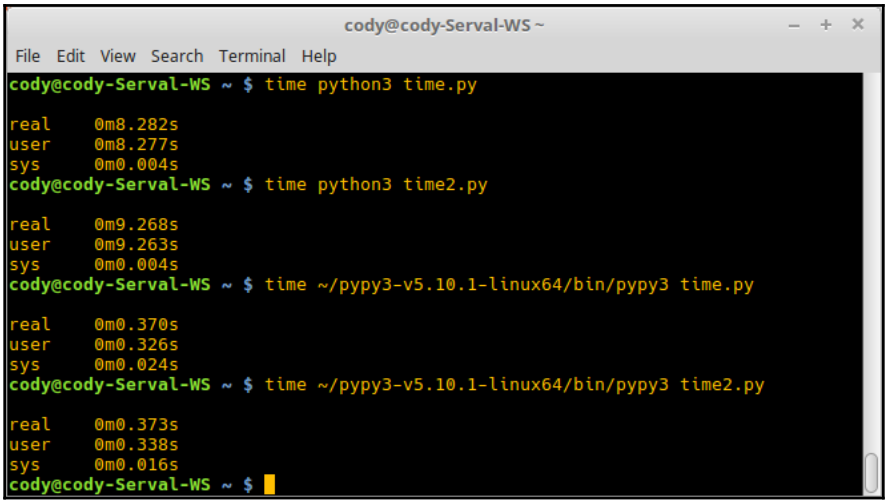


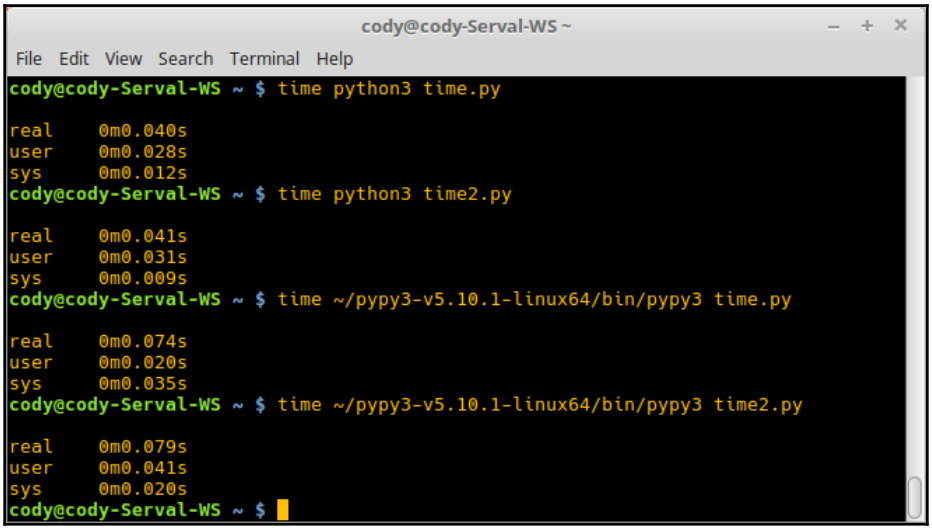


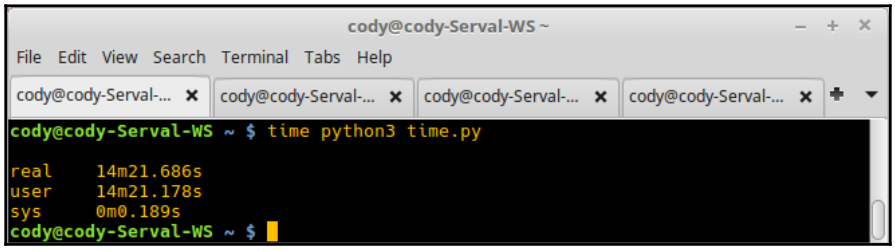

cody@cody-Serval-WS ~
File Edit View Search Terminal Tabs Help
cody@cody-Serval-WS ~
cody@cody-Serval-WS ~
cody@cody-Serval-WS ~
cody@cody-Serval-WS ~ $ time gcc -O3 /home/cody/c_file2.c -o c_fileout
real 0m0.027s
user 0m0.023s
sys 0m0.004s
cody@cody-Serval-WS ~ $ ls

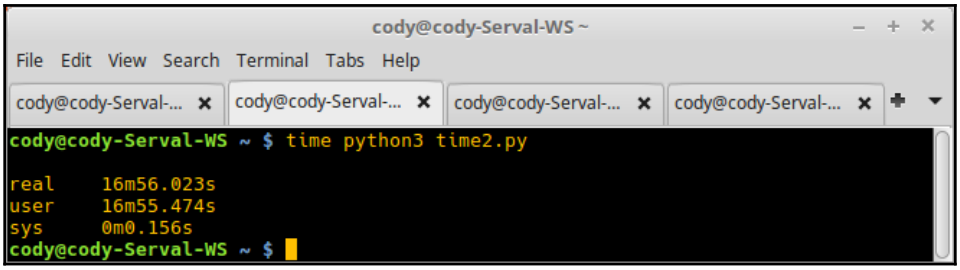

cody@cody-Serval-WS ~/pypy3-v5.10.1-linux64/bin
File Edit View Search Terminal Help
5 bottles of beer on the wall, 5 bottles of beer.
Take one down, pass it around,4 bottles of beer on the wall.
4 bottles of beer on the wall, 4 bottles of beer.
Take one down, pass it around,3 bottles of beer on the wall.
3 bottles of beer on the wall, 3 bottles of beer.
Take one down, pass it around,2 bottles of beer on the wall.
2 bottles of beer on the wall, 2 bottles of beer.
Take one down, pass it around,2 bottles of beer on the wall.
1 bottle of beer on the wall, 1 bottle of beer.
Take one down, pass it around,no more beer on the wall!
real 0m18.339s
user 0m6.410s
sys 0m4.571s
cody@cody-Serval-WS ~/pypy3-v5.10.1-linux64/bin $

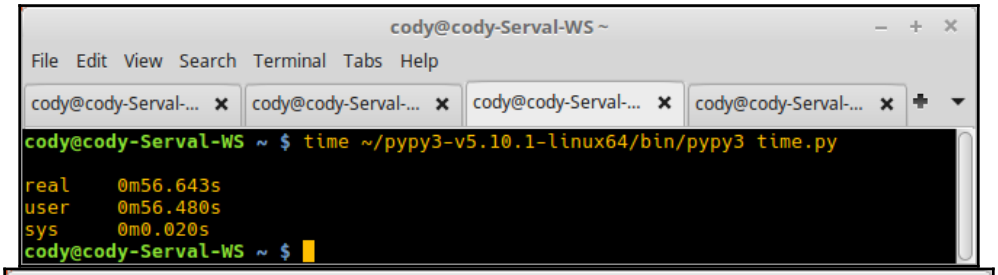

cody@cody-Serval-WS ~/pypy3-v5.10.1-linux64/bin
File Edit View Search Terminal Help
5 bottles of beer on the wall, 5 bottles of beer.
Take one down, pass it around,4 bottles of beer on the wall.
4 bottles of beer on the wall, 4 bottles of beer.
Take one down, pass it around,3 bottles of beer on the wall.
3 bottles of beer on the wall, 3 bottles of beer.
Take one down, pass it around,2 bottles of beer on the wall.
2 bottles of beer on the wall, 2 bottles of beer.
Take one down, pass it around,2 bottles of beer on the wall.
1 bottle of beer on the wall, 1 bottle of beer.
Take one down, pass it around,no more beer on the wall!
real 0m13.847s
user 0m3.962s
sys 0m3.633s
cody@cody-Serval-WS ~/pypy3-v5.10.1-linux64/bin $

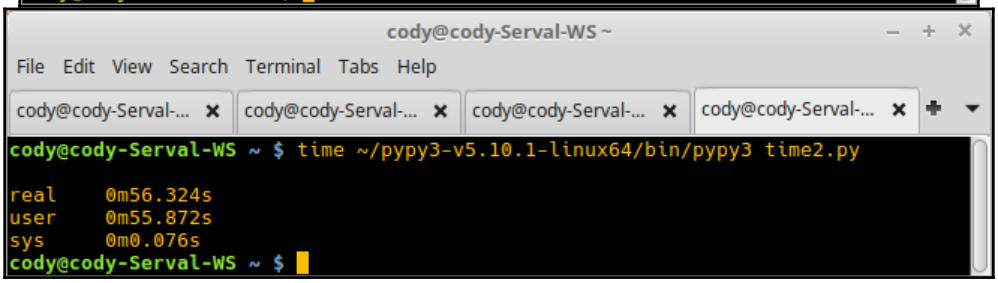

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ time python3 time.py
real 0m8.282s
user 0m8.277s
sys 0m0.004s
cody@cody-Serval-WS ~ $ time python3 time2.py
real 0m9.268s
user 0m9.263s
sys 0m0.004s
cody@cody-Serval-WS ~ $ time ~/pypy3-v5.10.1-linux64/bin/pypy3 time.py
real 0m0.370s
user 0m0.326s
sys 0m0.024s
cody@cody-Serval-WS ~ $ time ~/pypy3-v5.10.1-linux64/bin/pypy3 time2.py
real 0m0.373s
user 0m0.338s
sys 0m0.016s
cody@cody-Serval-WS ~ $

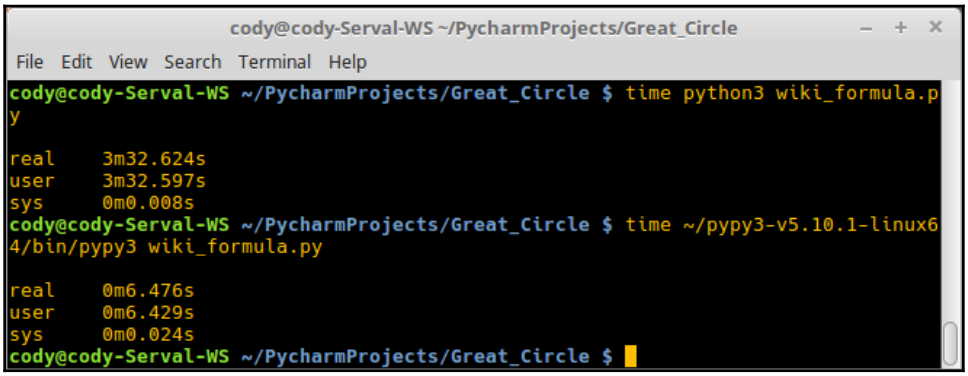

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ time python3 time.py
real 0m0.040s
user 0m0.028s
sys 0m0.012s
cody@cody-Serval-WS ~ $ time python3 time2.py
real 0m0.041s
user 0m0.031s
sys 0m0.009s
cody@cody-Serval-WS ~ $ time ~/pypy3-v5.10.1-linux64/bin/pypy3 time.py
real 0m0.074s
user 0m0.020s
sys 0m0.035s
cody@cody-Serval-WS ~ $ time ~/pypy3-v5.10.1-linux64/bin/pypy3 time2.py
real 0m0.079s
user 0m0.041s
sys 0m0.020s
cody@cody-Serval-WS ~ $

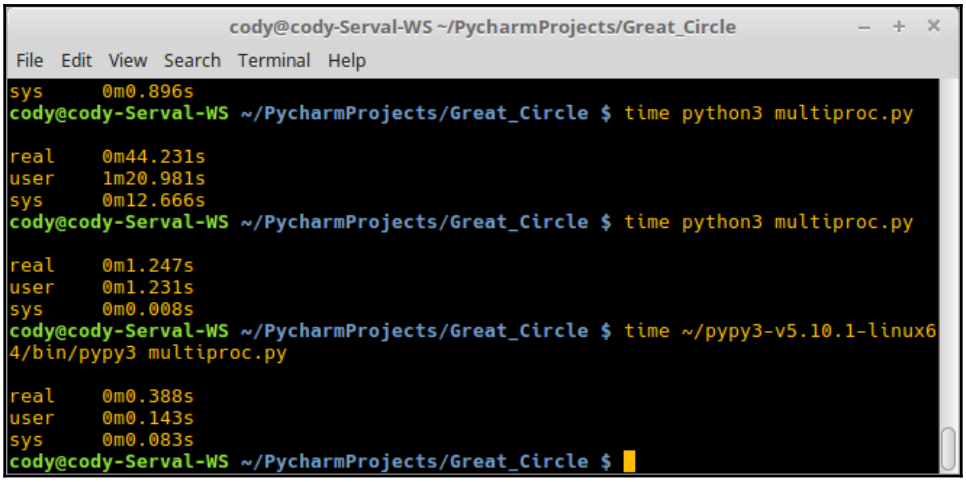

cody@cody-Serval-WS ~
File Edit View Search Terminal Tabs Help
cody@cody-Serval-...
cody@cody-Serval-...
cody@cody-Serval-...
cody@cody-Serval-...
cody@cody-Serval-WS ~ $ time python3 time.py
real 14m21.686s
user 14m21.178s
sys 0m0.189s
cody@cody-Serval-WS ~ $


cody@cody-Serval-WS ~
File Edit View Search Terminal Tabs Help
cody@cody-Serval-... cody@cody-Serval-... cody@cody-Serval-... cody@cody-Serval-...
cody@cody-Serval-WS ~ $ time python3 time2.py
real 16m56.023s
user 16m55.474s
sys 0m0.156s
cody@cody-Serval-WS ~ $

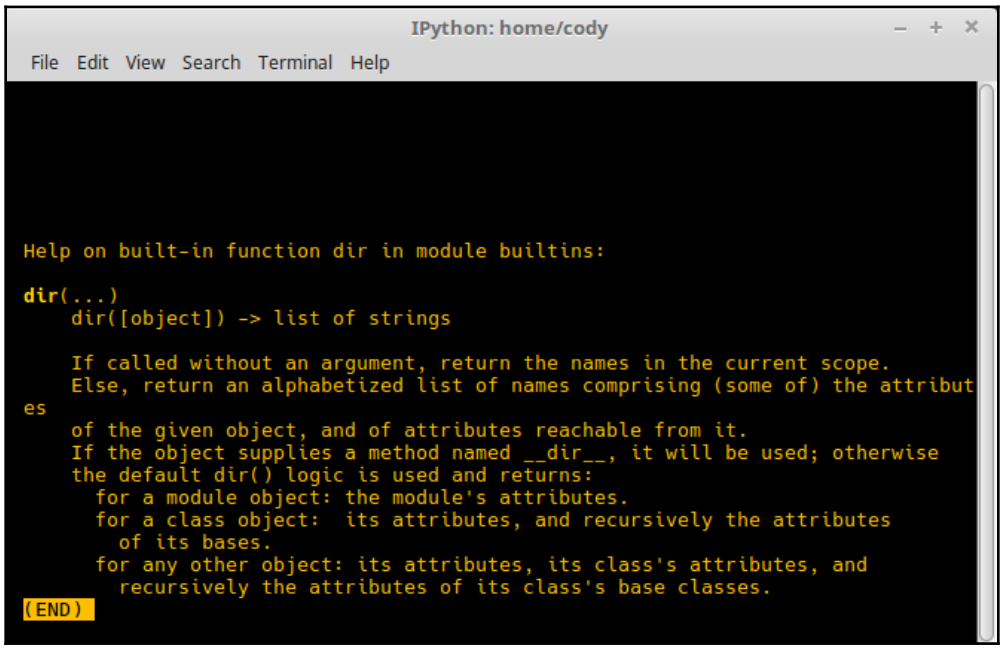

cody@cody-Serval-WS ~
File Edit View Search Terminal Tabs Help
cody@cody-Serval-... cody@cody-Serval-... cody@cody-Serval-... cody@cody-Serval-...
cody@cody-Serval-WS ~ $ time ~/pypy3-v5.10.1-linux64/bin/pypy3 time.py
real 0m56.643s
user 0m56.480s
sys 0m0.020s
cody@cody-Serval-WS ~ $

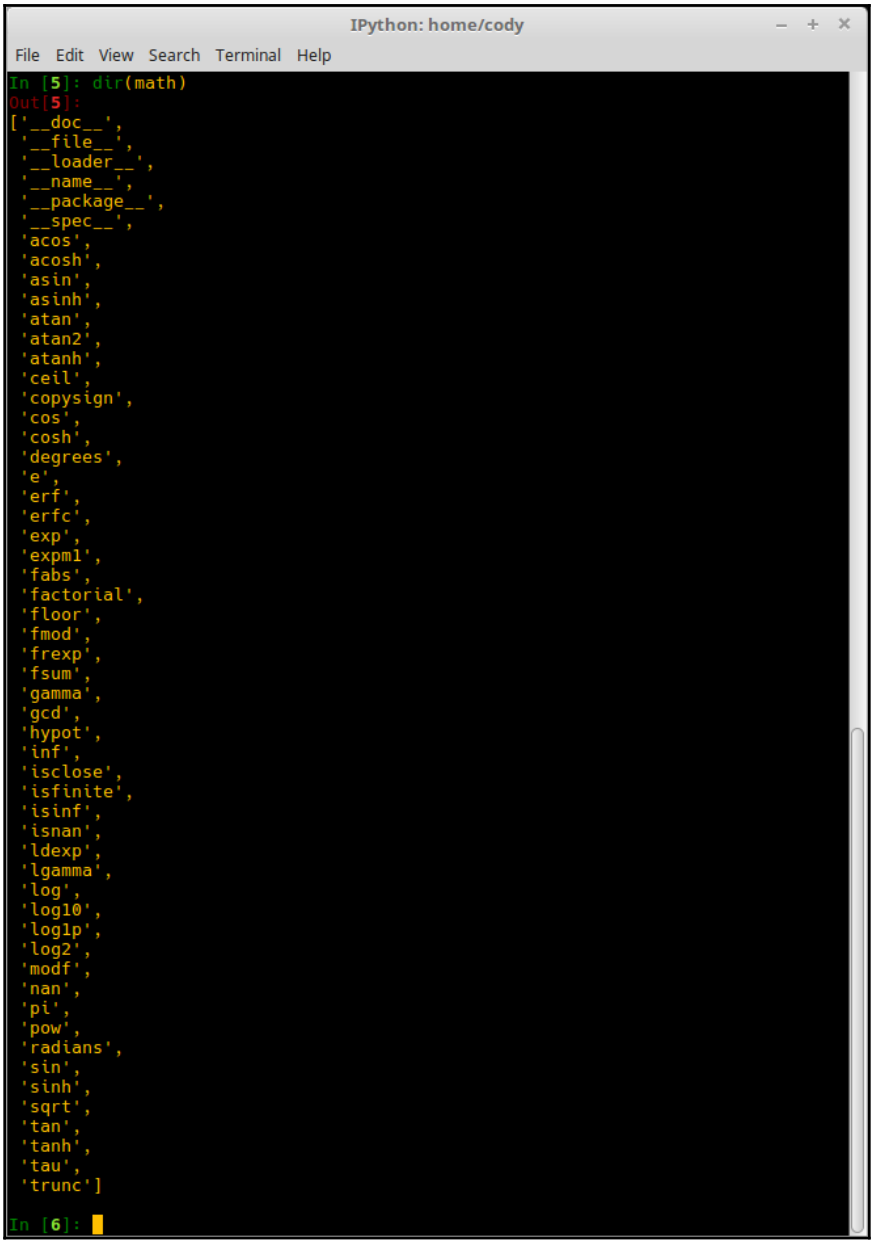

cody@cody-Serval-WS ~
File Edit View Search Terminal Tabs Help
cody@cody-Serval-... cody@cody-Serval-... cody@cody-Serval-... cody@cody-Serval-...
cody@cody-Serval-WS ~ $ time ~/pypy3-v5.10.1-linux64/bin/pypy3 time2.py
real 0m56.324s
user 0m55.872s
sys 0m0.076s
cody@cody-Serval-WS ~ $

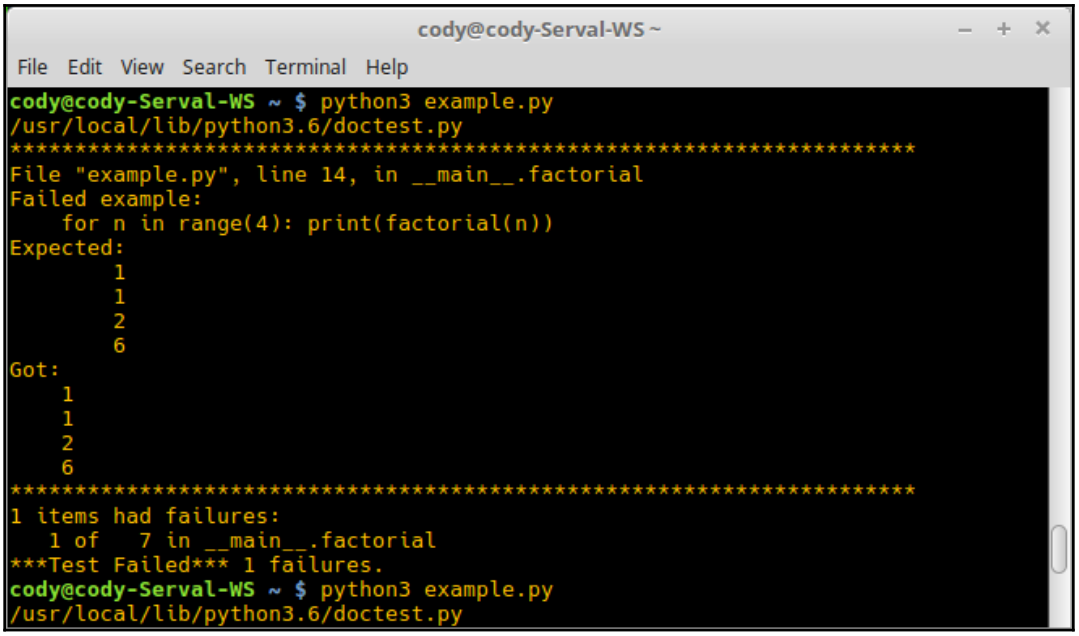

cody@cody-Serval-WS ~/PycharmProjects/Great_Circle
File Edit View Search Terminal Help
cody@cody-Serval-WS ~/PycharmProjects/Great_Circle $ time python3 wiki_formula.p
y
real 3m32.624s
user 3m32.597s
sys 0m0.008s
cody@cody-Serval-WS ~/PycharmProjects/Great_Circle $ time ~/pypy3-v5.10.1-linux6
4/bin/pypy3 wiki_formula.py
real 0m6.476s
user 0m6.429s
sys 0m0.024s
cody@cody-Serval-WS ~/PycharmProjects/Great_Circle $


cody@cody-Serval-WS ~/PycharmProjects/Great_Circle
File Edit View Search Terminal Help
sys 0m0.896s
cody@cody-Serval-WS ~/PycharmProjects/Great_Circle $ time python3 multiproc.py
real 0m44.231s
user 1m20.981s
sys 0m12.666s
cody@cody-Serval-WS ~/PycharmProjects/Great_Circle $ time python3 multiproc.py
real 0m1.247s
user 0m1.231s
sys 0m0.008s
cody@cody-Serval-WS ~/PycharmProjects/Great_Circle $ time ~/pypy3-v5.10.1-linux64/bin/pypy3 multiproc.py
real 0m0.388s
user 0m0.143s
sys 0m0.083s
cody@cody-Serval-WS ~/PycharmProjects/Great_Circle $

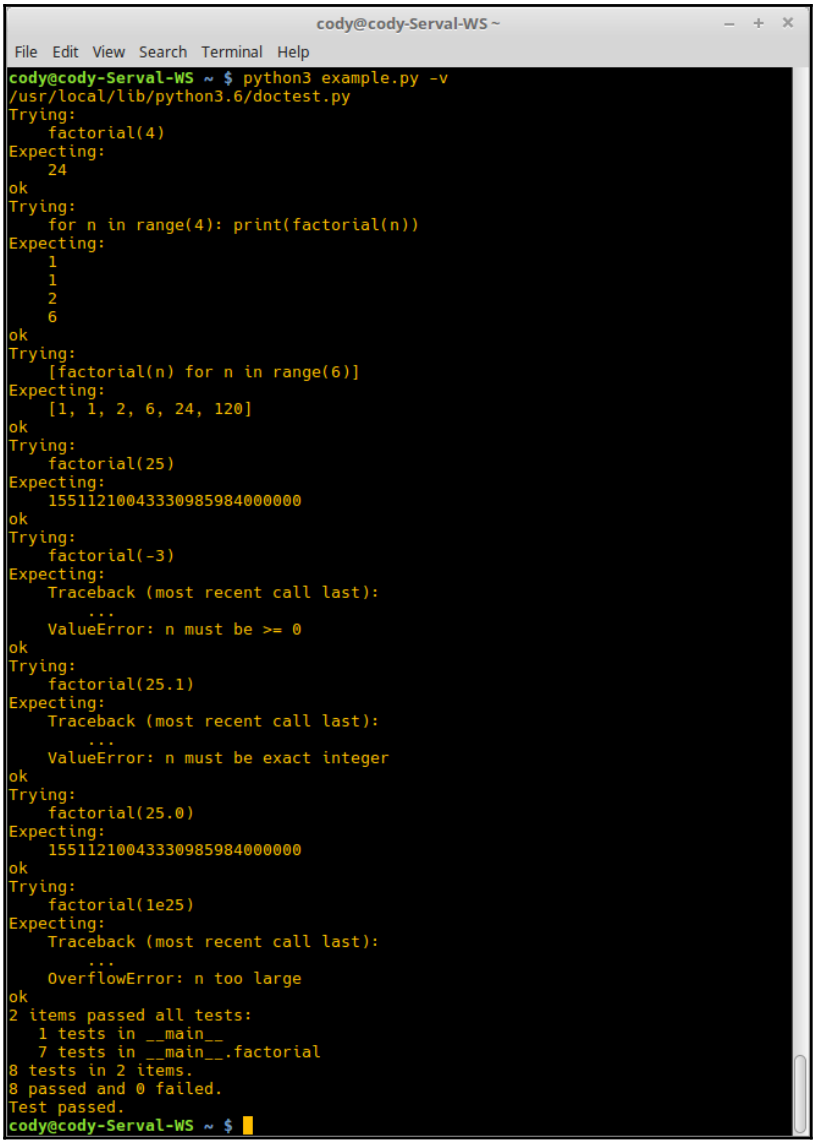
Chapter 9: Documenting with LyX

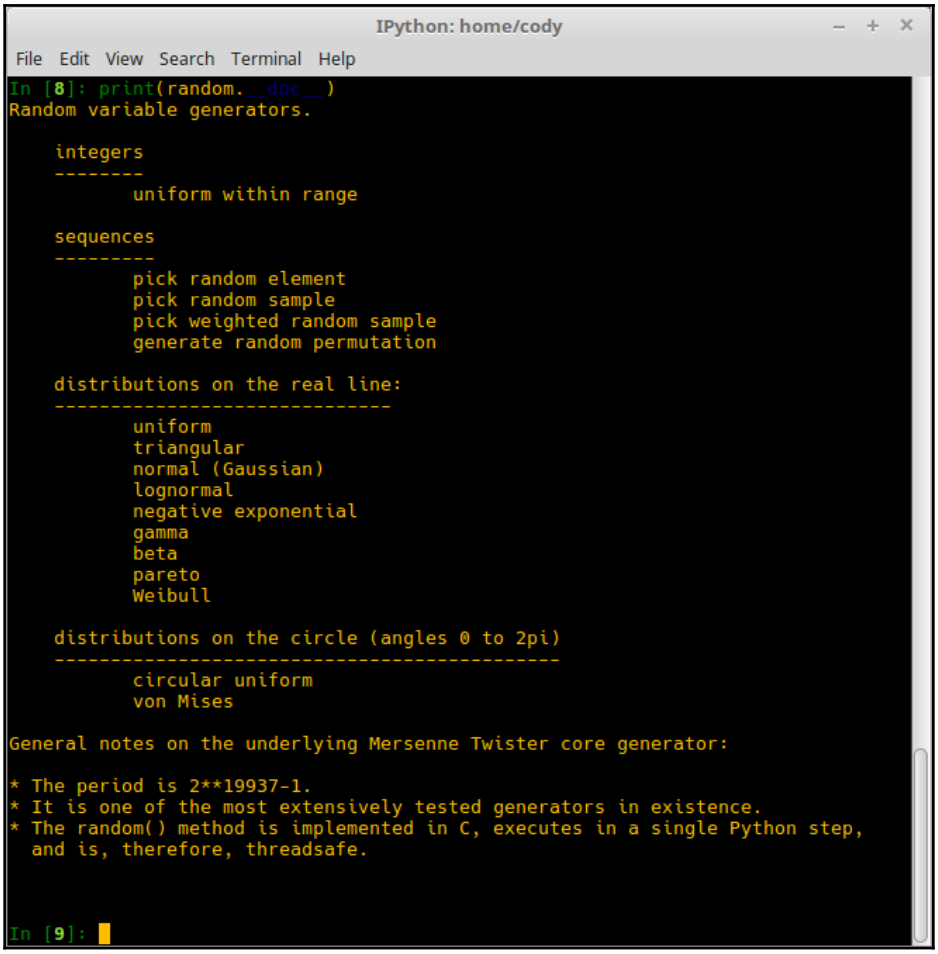


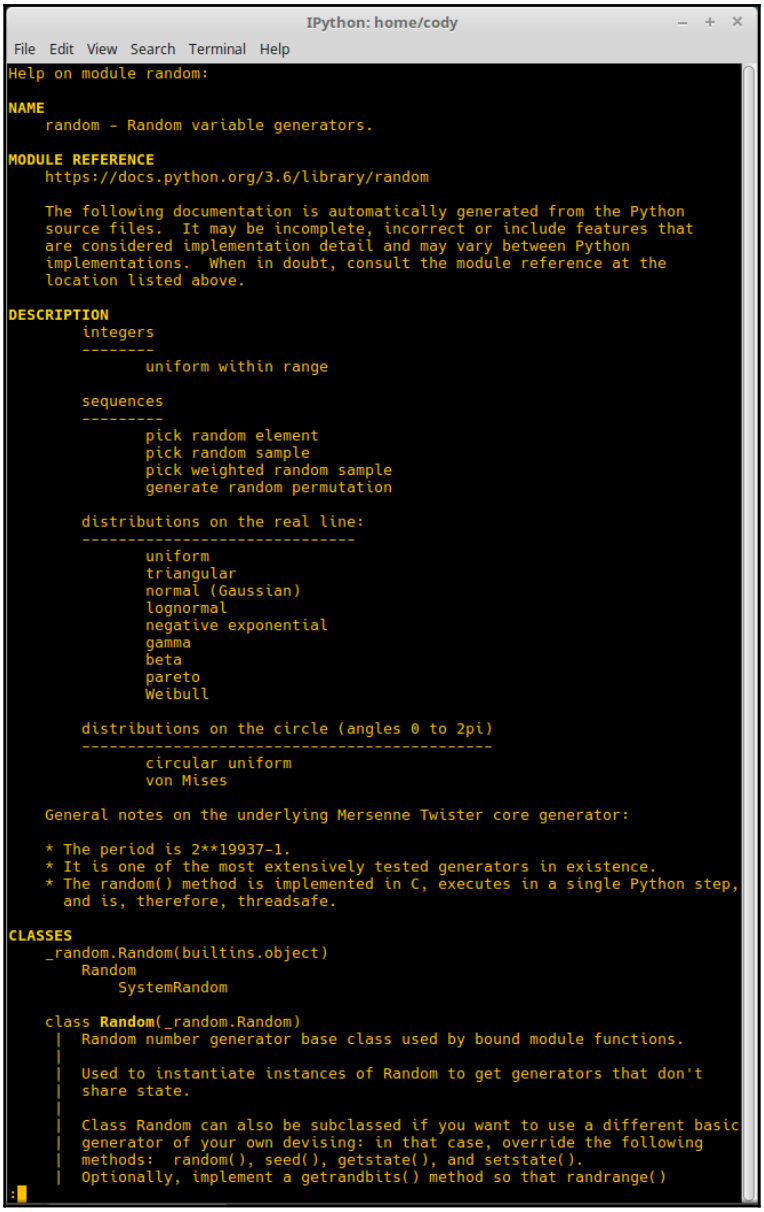

IPython: home/cody
File Edit View Search Terminal Help
In [5]: dir(math)
Out[5]:
['__doc__',
'__file__',
'__loader__',
'__name__',
'__package__',
'__spec__',
'acos',
'acosh',
'asin',
'asinh',
'atan',
'atan2',
'atanh',
'ceil',
'copysign',
'cos',
'cosh',
'degrees',
'e',
'erf',
'erfc',
'exp',
'expm1',
'fabs',
'factorial',
'floor',
'fmod',
'frexp',
'fsum',
'gamma',
'gcd',
'hypot',
'inf',
'isclose',
'isfinite',
'isinf',
'isnan',
'ldexp',
'lgamma',
'log',
'log10',
'log1p',
'log2',
'modf',
'nan',
'pi',
'pow',
'radians',
'sin',
'sinh',
'sqrt',
'tan',
'tanh',
'tau',
'trunc']
In [6]:

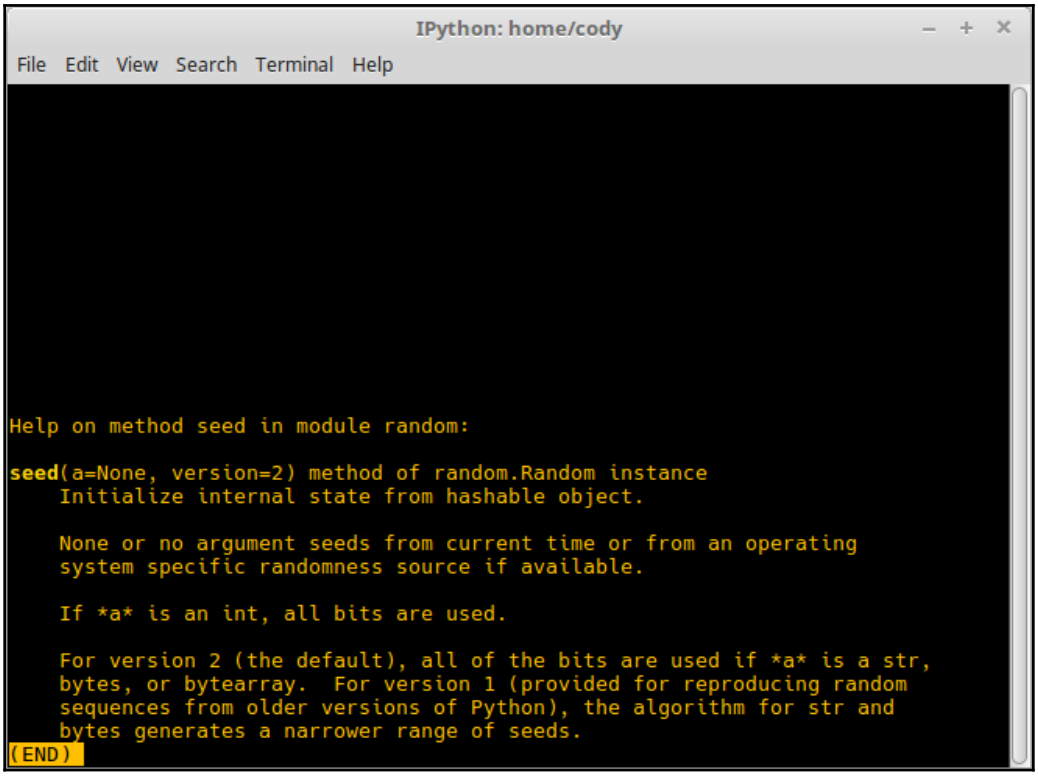

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3 example.py
/usr/local/lib/python3.6/doctest.py
****
File "example.py", line 14, in __main__.factorial
Failed example:
for n in range(4): print(factorial(n))
Expected:
1
1
2
6
Got:
1
1
2
6
****
1 items had failures:
1 of 7 in __main__.factorial
Test Failed 1 failures.
cody@cody-Serval-WS ~ $ python3 example.py
/usr/local/lib/python3.6/doctest.py

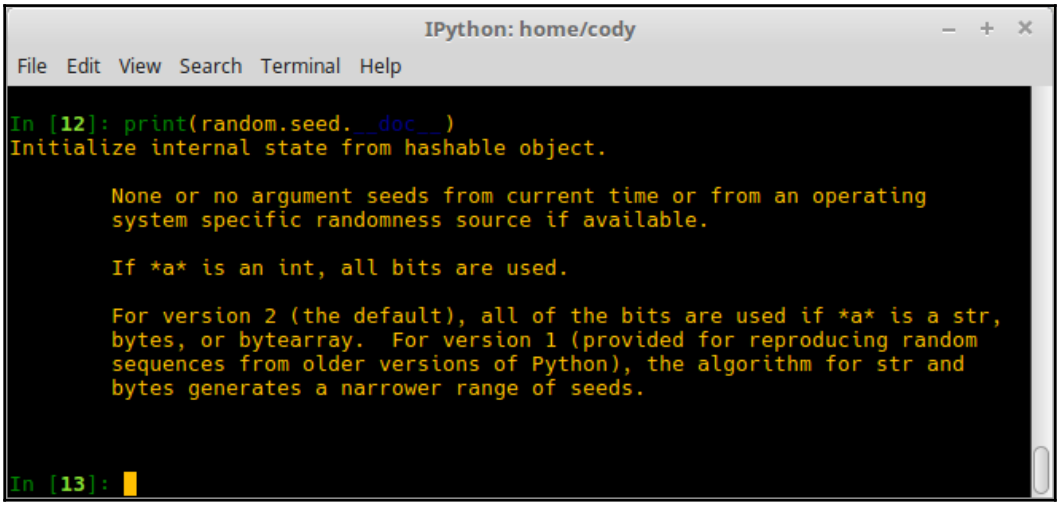

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3 example.py
/usr/local/lib/python3.6/doctest.py
cody@cody-Serval-WS ~ $

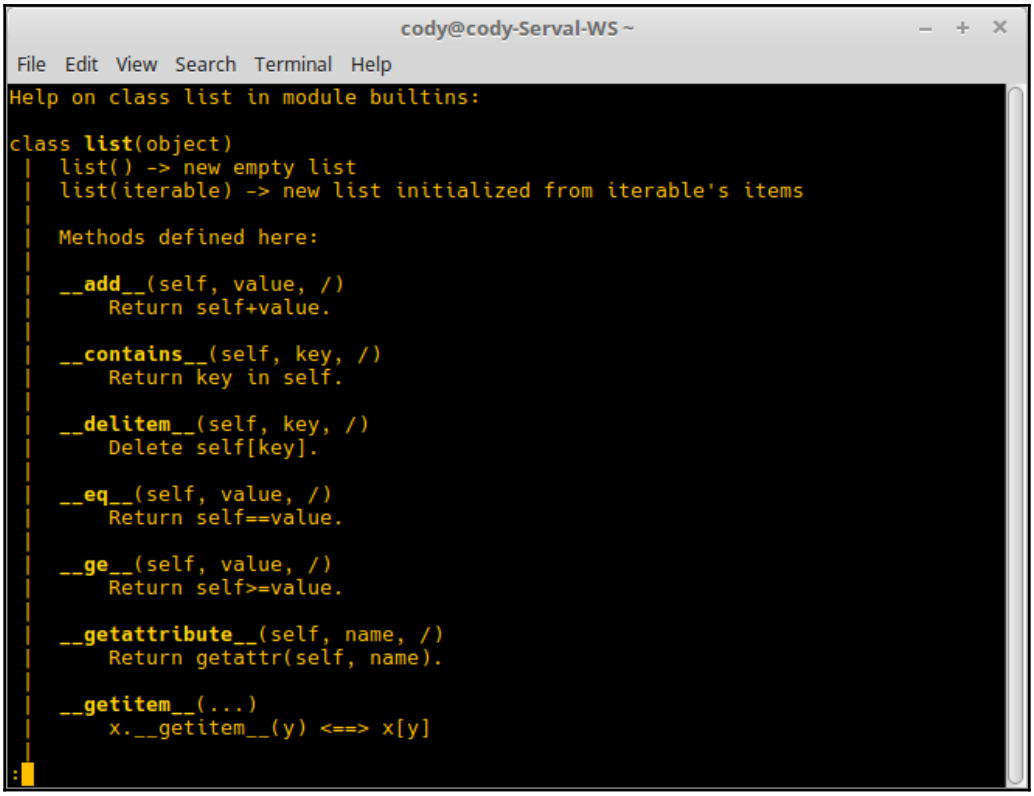

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
cody@cody-Serval-WS ~ $ python3 example.py -v
/usr/local/lib/python3.6/doctest.py
Trying:
factorial(4)
Expecting:
24
ok
Trying:
for n in range(4): print(factorial(n))
Expecting:
1
1
2
6
ok
Trying:
[factorial(n) for n in range(6)]
Expecting:
[1, 1, 2, 6, 24, 120]
ok
Trying:
factorial(25)
Expecting:
15511210043330985984000000
ok
Trying:
factorial(-3)
Expecting:
Traceback (most recent call last):
...
ValueError: n must be >= 0
ok
Trying:
factorial(25.1)
Expecting:
Traceback (most recent call last):
...
ValueError: n must be exact integer
ok
Trying:
factorial(25.0)
Expecting:
15511210043330985984000000
ok
Trying:
factorial(1e25)
Expecting:
Traceback (most recent call last):
...
OverflowError: n too large
ok
2 items passed all tests:
1 tests in __main__
7 tests in __main__.factorial
8 tests in 2 items.
8 passed and 0 failed.
Test passed.
cody@cody-Serval-WS ~ $

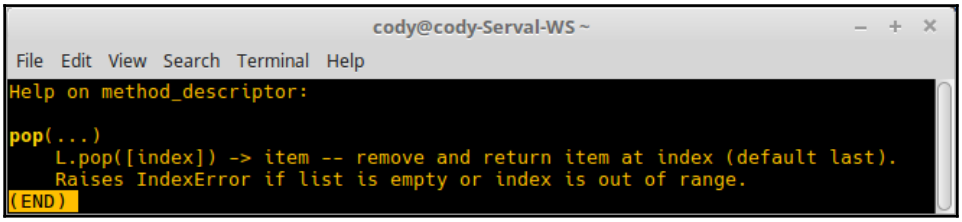

IPython: home/cody
File Edit View Search Terminal Help
In [8]: print(random.__doc__)
Random variable generators.
integers
----
uniform within range
sequences
----
pick random element
pick random sample
pick weighted random sample
generate random permutation
distributions on the real line:
----
uniform
triangular
normal (Gaussian)
lognormal
negative exponential
gamma
beta
pareto
Weibull
distributions on the circle (angles 0 to 2pi)
----
circular uniform
von Mises
General notes on the underlying Mersenne Twister core generator:
* The period is 2**19937-1.
* It is one of the most extensively tested generators in existence.
* The random() method is implemented in C, executes in a single Python step,
and is, therefore, threadsafe.
In [9]:

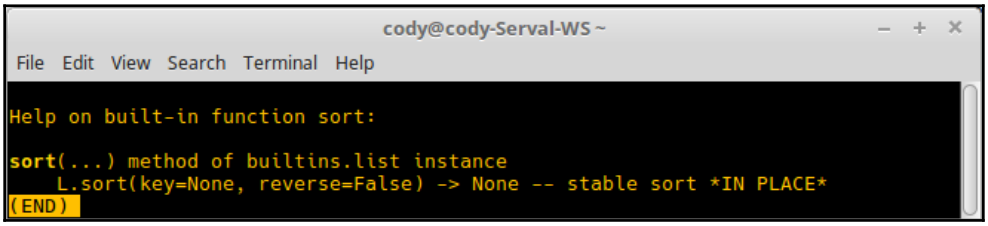

IPython: home/cody
File Edit View Search Terminal Help
Help on module random:
NAME
random - Random variable generators.
MODULE REFERENCE
https://docs.python.org/3.6/library/random
The following documentation is automatically generated from the Python
source files. It may be incomplete, incorrect or include features that
are considered implementation detail and may vary between Python
implementations. When in doubt, consult the module reference at the
location listed above.
DESCRIPTION
integers
uniform within range
sequences
pick random element
pick random sample
pick weighted random sample
generate random permutation
distributions on the real line:
uniform
triangular
normal (Gaussian)
lognormal
negative exponential
gamma
beta
pareto
Weibull
distributions on the circle (angles 0 to 2pi)
circular uniform
von Mises
General notes on the underlying Mersenne Twister core generator:
* The period is 2**19937-1.
* It is one of the most extensively tested generators in existence.
* The random() method is implemented in C, executes in a single Python step,
and is, therefore, threadsafe.
CLASSES
_random.Random(builtins.object)
Random
SystemRandom
class Random(_random.Random)
| Random number generator base class used by bound module functions.
|
| Used to instantiate instances of Random to get generators that don't
| share state.
|
| Class Random can also be subclassed if you want to use a different basic
| generator of your own devising: in that case, override the following
| methods: random(), seed(), getstate(), and setstate().
| Optionally, implement a getrandbits() method so that randrange()
:


IPython: home/cody
File Edit View Search Terminal Help
Help on method seed in module random:
seed(a=None, version=2) method of random.Random instance
Initialize internal state from hashable object.
None or no argument seeds from current time or from an operating
system specific randomness source if available.
If *a* is an int, all bits are used.
For version 2 (the default), all of the bits are used if *a* is a str,
bytes, or bytearray. For version 1 (provided for reproducing random
sequences from older versions of Python), the algorithm for str and
bytes generates a narrower range of seeds.
(END)

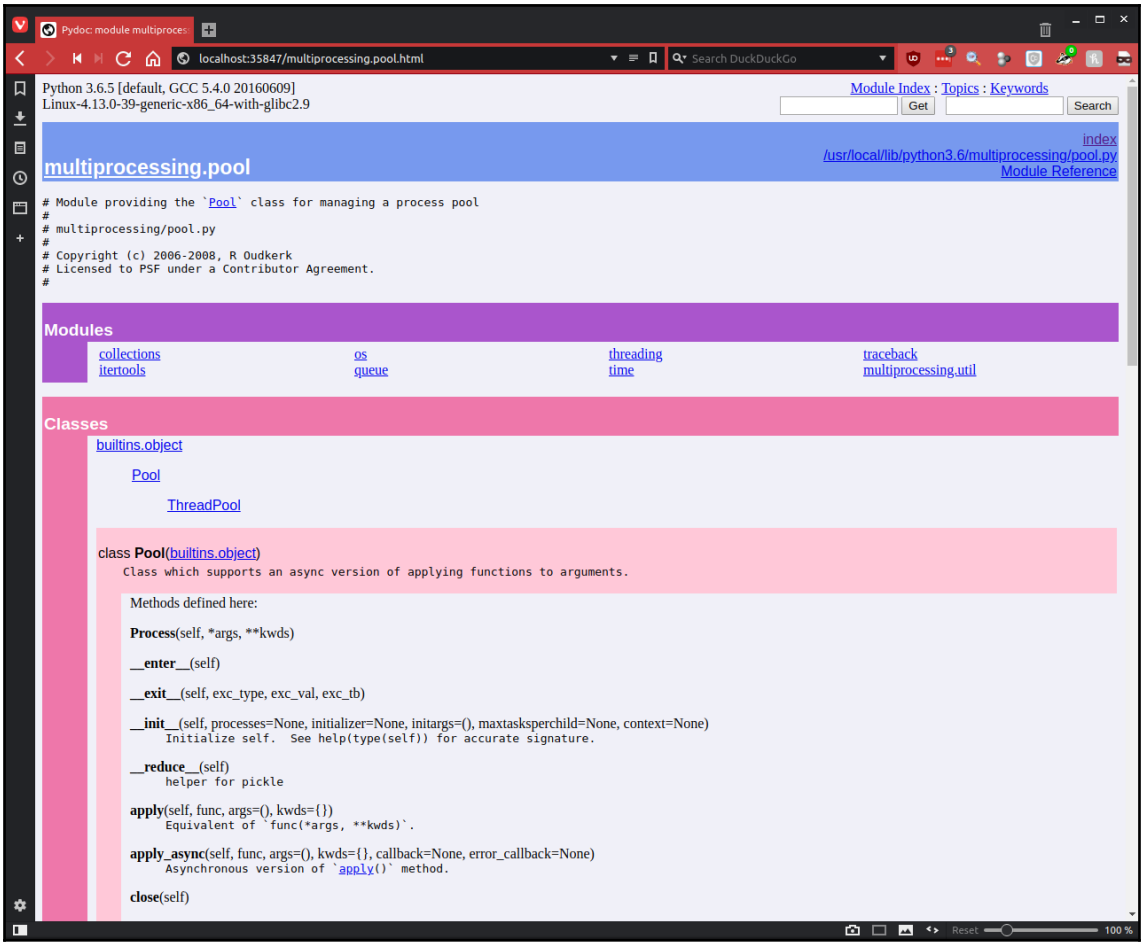

IPython: home/cody
File Edit View Search Terminal Help
In [12]: print(random.seed.__doc__)
Initialize internal state from hashable object.
None or no argument seeds from current time or from an operating
system specific randomness source if available.
If *a* is an int, all bits are used.
For version 2 (the default), all of the bits are used if *a* is a str,
bytes, or bytearray. For version 1 (provided for reproducing random
sequences from older versions of Python), the algorithm for str and
bytes generates a narrower range of seeds.
In [13]:


cody@cody-Serval-WS ~
File Edit View Search Terminal Help
Help on class list in module builtins:
class list(object)
| list() -> new empty list
| list(iterable) -> new list initialized from iterable's items
| Methods defined here:
| __add__(self, value, /)
| Return self+value.
| __contains__(self, key, /)
| Return key in self.
| __delitem__(self, key, /)
| Delete self[key].
| __eq__(self, value, /)
| Return self==value.
| __ge__(self, value, /)
| Return self>=value.
| __getattribute__(self, name, /)
| Return getattr(self, name).
| __getitem__(...)
| x.__getitem__(y) <==> x[y]
:

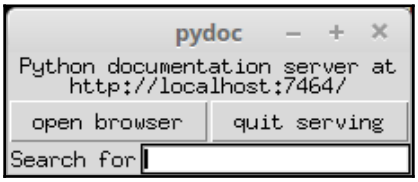

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
Help on method_descriptor:
pop(...)
L.pop([index]) -> item -- remove and return item at index (default last).
Raises IndexError if list is empty or index is out of range.
(END)

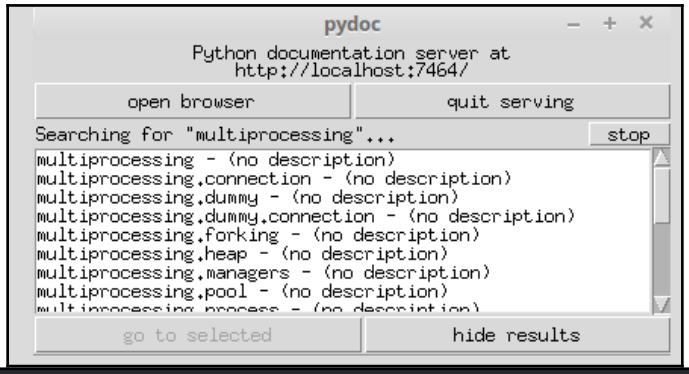

cody@cody-Serval-WS ~
File Edit View Search Terminal Help
Help on built-in function sort:
sort(...) method of builtins.list instance
L.sort(key=None, reverse=False) -> None -- stable sort *IN PLACE*
(END)

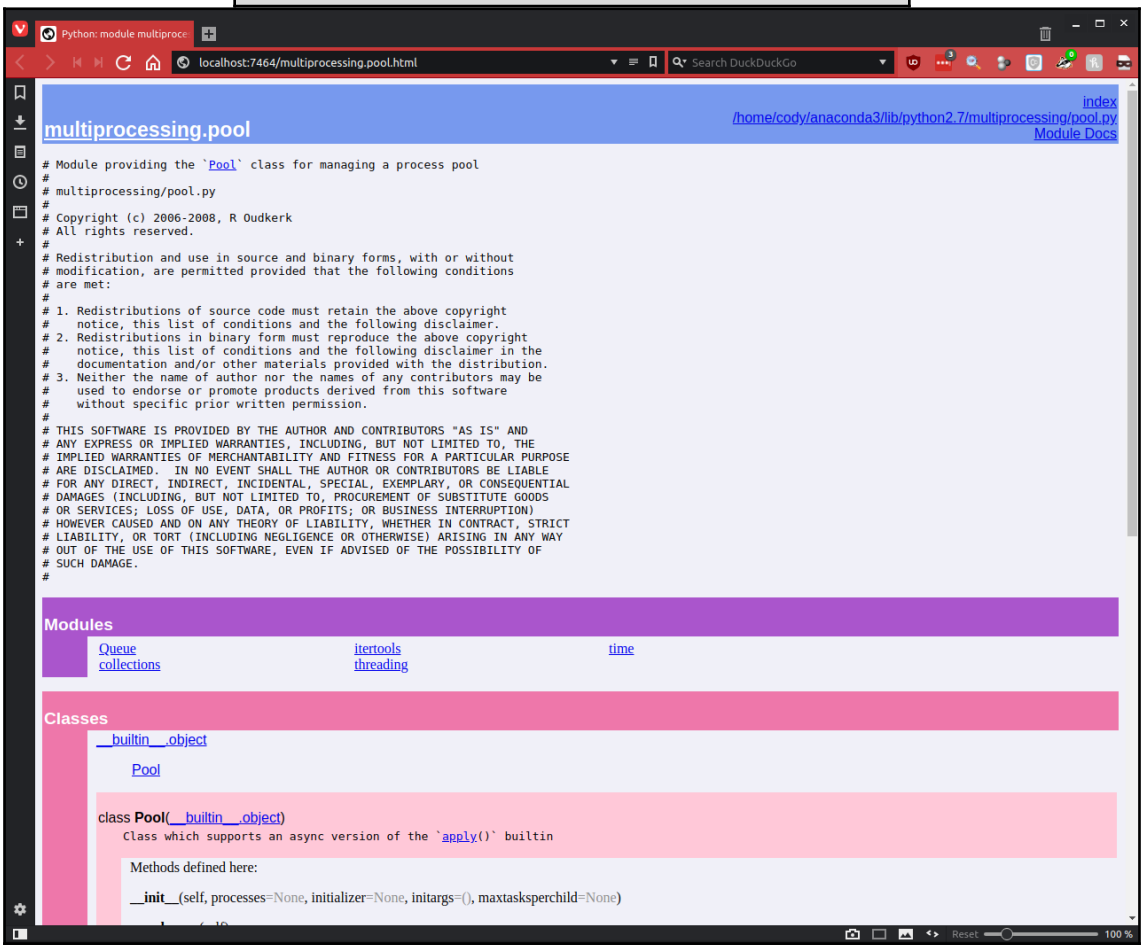

Pydoc: Index of Modules
localhost:40567
Search DuckDuckGo
Python 3.6.5 [default, GCC 5.4.0 20160609]
Linux-4.13.0-39-generic-x86_64-with-glibc2.9
Module Index : Topics : Keywords
Get
Search
Index of Modules
Built-in Modules
_ast
_codecs
_collections
_functools
_imp
_io
_locale
_operator
_signal
_socket
_sre
_ssl
_stat
_string
_symtable
_thread
_tracemalloc
_warnings
_weakref
atexit
builtins
errno
faulthandler
gc
itertools
marshal
posix
pwd
sys
time
xxsubtype
zipimport
example
/usr/local/lib/python36.zip
/usr/local/lib/python3.6
__future__
_bootlocale
_collections_abc
_compat_pickle
_compression
_dummy_thread
_markupbase
_osx_support
_pydecimal
_pyio
_sitebuiltins
_strptime
_sysconfigdata_m_linux_x86_64-linux-gnu
_threading_local
_weakrefset
abc
aifc
antigravity
argparse
ast
asynchat
asyncio (package)
dbm (package)
decimal
difflib
dis
distutils (package)
doctest
dummy_threading
email (package)
encodings (package)
ensurepip (package)
enum
filecmp
fileinput
fnmatch
formatter
fractions
ftplib
functools
genericpath
getopt
getpass
gettext
glob
multiprocessing (package)
netrc
nntplib
ntpath
nturl2path
numbers
opcode
operator
optparse
os
pathlib
pdb
pickle
pickletools
pipes
pkgutil
platform
plistlib
poplib
posixpath
pprint
profile
pstats
sre_constants
sre_parse
ssl
stat
statistics
string
stringprep
struct
subprocess
sunau
symbol
symtable
sysconfig
tabnanny
tarfile
telnetlib
tempfile
test (package)
textwrap
this
threading
timeit
tkinter (package)
Reset
100 %

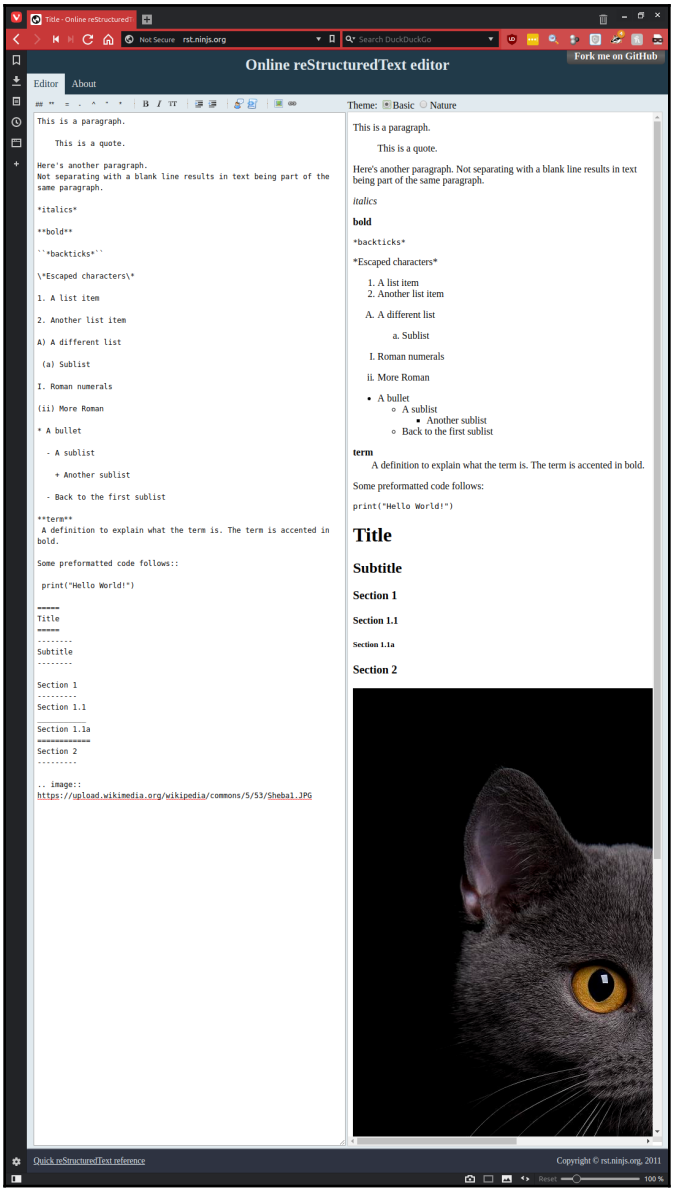

Pydoc: module multiproces
localhost:35847/multiprocessing.pool.html
Search DuckDuckGo
Python 3.6.5 [default, GCC 5.4.0 20160609]
Linux-4.13.0-39-generic-x86_64-with-glibc2.9
Module Index : Topics : Keywords
Get
Search
multiprocessing.pool
index
/usr/local/lib/python3.6/multiprocessing/pool.py
Module Reference
Module providing the `Pool` class for managing a process pool
#
multiprocessing/pool.py
#
Copyright (c) 2006-2008, R Oudkerk
Licensed to PSF under a Contributor Agreement.
#
Modules
collections
itertools
os
queue
threading
time
traceback
multiprocessing.util
Classes
builtins.object
Pool
ThreadPool
class Pool(builtins.object)
Class which supports an async version of applying functions to arguments.
Methods defined here:
Process(self, *args, **kwds)
__enter__(self)
__exit__(self, exc_type, exc_val, exc_tb)
__init__(self, processes=None, initializer=None, initargs=(), maxtasksperchild=None, context=None)
Initialize self. See help(type(self)) for accurate signature.
__reduce__(self)
helper for pickle
apply(self, func, args=(), kwds={})
Equivalent of `func(*args, **kwds)`.
apply_async(self, func, args=(), kwds={}, callback=None, error_callback=None)
Asynchronous version of `apply()` method.
close(self)
Reset
100 %

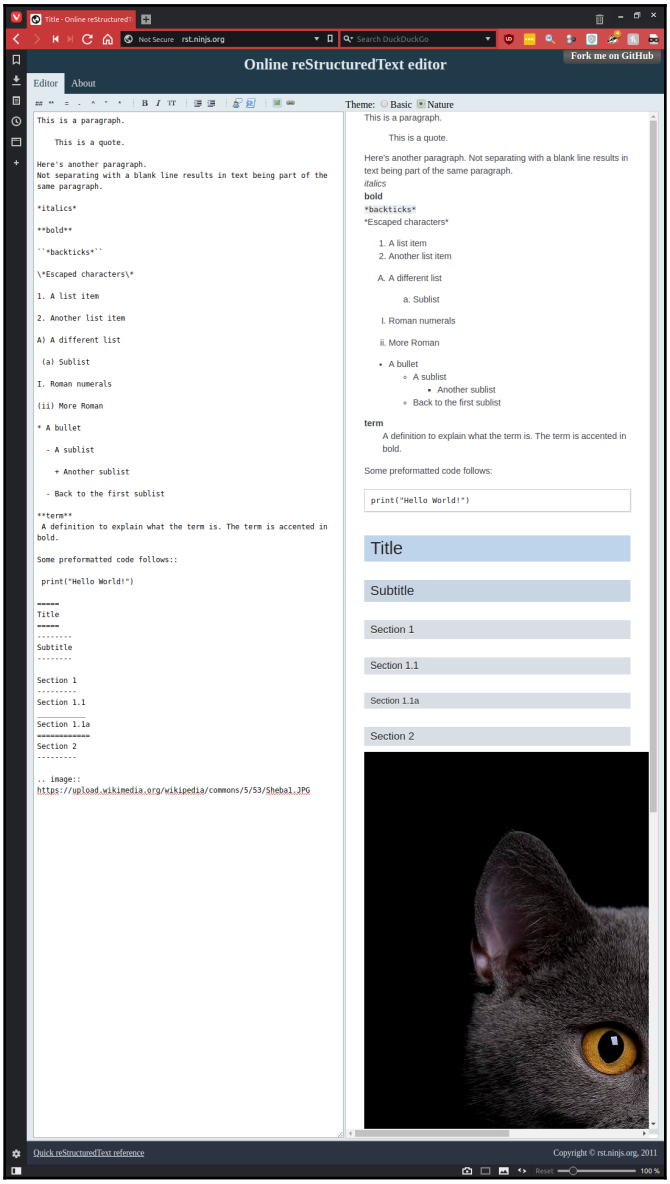

Pydoc: Index of Modules
localhost:45223
Search DuckDuckGo
Python 3.6.5 [default, GCC 5.4.0 20160609]
Linux-4.13.0-39-generic-x86_64-with-glibc2.9
Module Index : Topics : Keywords
Get
Search
Index of Modules
Built-in Modules
_ast _codecs _collections _functools _imp _io _locale _operator
_signal _socket _sre _ssl _stat _string _symtable _thread
_tracemalloc _warnings _weakref atexit builtins errno faulthandler gc
itertools marshal posix pwd sys time xxsubtype zipimport
config (package) conftest doc (package)
goal (package) interpreter (package) module (package)
objspace (package) pytest-A sandbox (package)
test_all testrunner_cfg tool (package)
/home/cody/.local/share/virtualenvs/pypy-T8QDpaxb/lib/python36.zip
/home/cody/.local/share/virtualenvs/pypy-T8QDpaxb/lib/python3.6
__future__ _bootlocale _collections_abc _dummy_thread _weakrefset abc base64 bisect codecs collections (package) copy copyreg distutils (package)
encodings (package) enum fnmatch functools genericpath hashlib heapq hmac imp importlib (package) io keyword linecache
locale ntpath operator os posixpath random re reprlib rlcompleter shutil site sre_compile sre_constants
sre_parse stat struct tarfile tempfile token tokenize types warnings weakref
/home/cody/.local/share/virtualenvs/pypy-T8QDpaxb/lib/python3.6/lib-dynload
_asyncio _bisect _blake2 _bz2 _dbm _decimal _elementtree _gdbm
_sha3 _sha512 _socket _sqlite3 mmap nis ossaudiodev parser
Reset
100 %

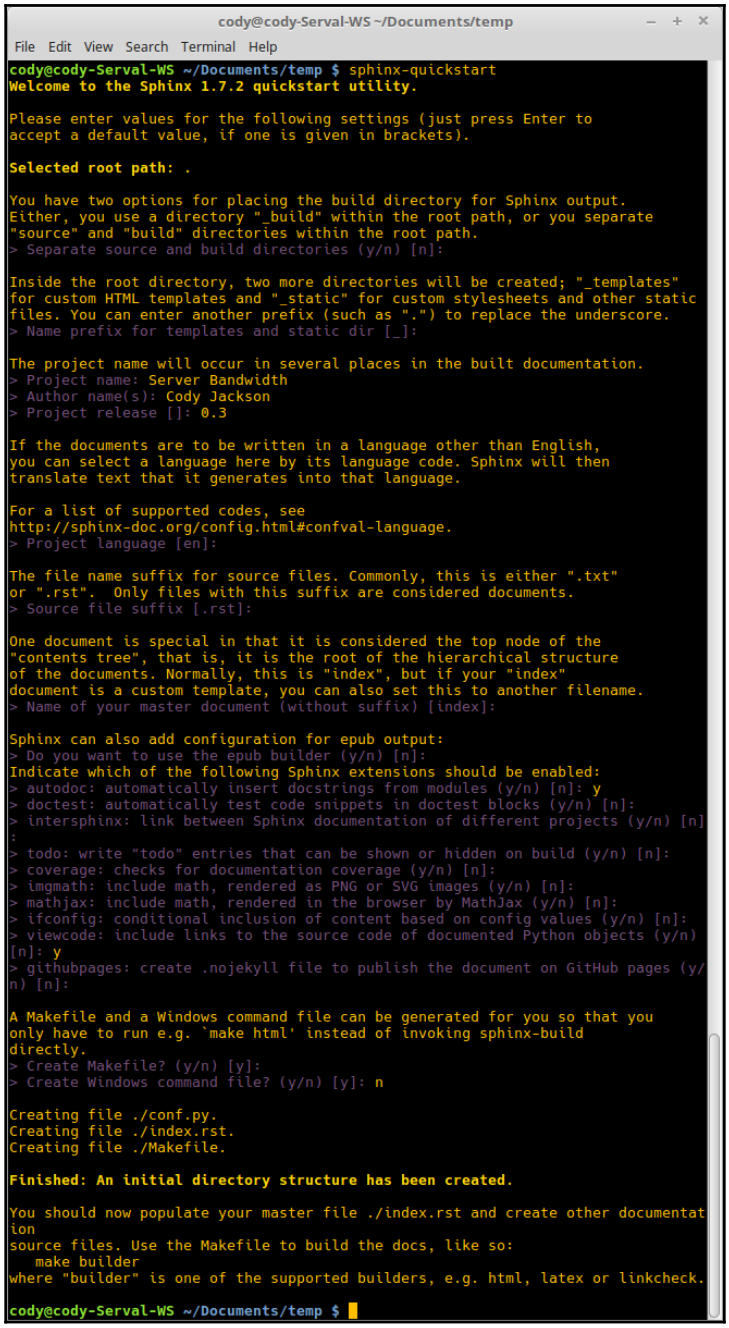

pydoc
Python documentation server at
http://localhost:7464/
open browser
quit serving
Search for

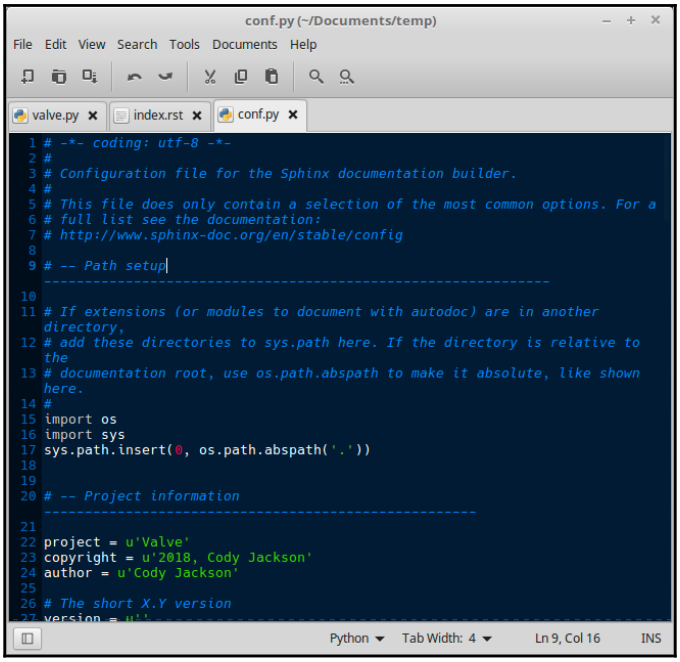

pydoc
Python documentation server at
http://localhost:7464/
open browser
quit serving
Searching for "multiprocessing"...
stop
multiprocessing - (no description)
multiprocessing.connection - (no description)
multiprocessing.dummy - (no description)
multiprocessing.dummy.connection - (no description)
multiprocessing.forking - (no description)
multiprocessing.heap - (no description)
multiprocessing.managers - (no description)
multiprocessing.pool - (no description)
go to selected
hide results

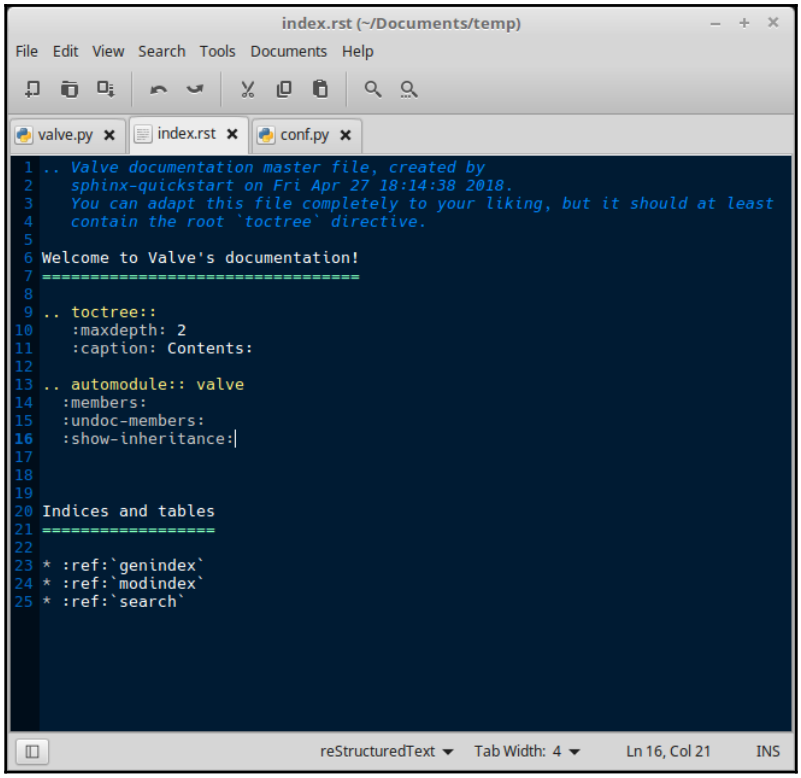

Python: module multiproce
localhost:7464/multiprocessing.pool.html
Search DuckDuckGo
multiprocessing.pool
index
/home/cody/anaconda3/lib/python2.7/multiprocessing/pool.py
Module Docs
Module providing the `Pool` class for managing a process pool
#
multiprocessing/pool.py
#
Copyright (c) 2006-2008, R Oudkerk
All rights reserved.
#
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
#
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of author nor the names of any contributors may be
used to endorse or promote products derived from this software
without specific prior written permission.
#
THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
#
Modules
Queue
collections
itertools
threading
time
Classes
__builtin__.object
Pool
class Pool(__builtin__.object)
Class which supports an async version of the `apply()` builtin
Methods defined here:
__init__(self, processes=None, initializer=None, initargs=(), maxtasksperchild=None)
Reset
100 %

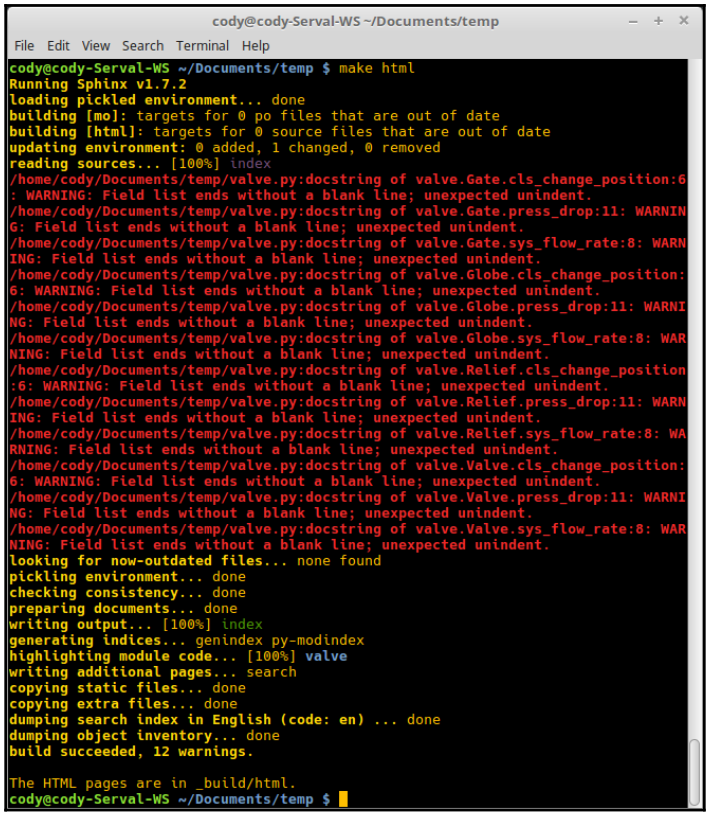

Online reStructuredText editor
Fork me on GitHub
Editor
About
Theme: Basic Nature
This is a paragraph.
This is a quote.
Here's another paragraph. Not separating with a blank line results in text being part of the same paragraph.
italics
bold
``*backticks*``
Escaped characters
1. A list item
2. Another list item
A) A different list
(a) Sublist
I. Roman numerals
(ii) More Roman
* A bullet
- A sublist
+ Another sublist
- Back to the first sublist
term
A definition to explain what the term is. The term is accented in bold.
Some preformatted code follows::
print("Hello World!")
Title
Subtitle
Section 1
Section 1.1
Section 1.1a
Section 2
.. image::
https://upload.wikimedia.org/wikipedia/commons/5/53/Sheba1.JPG
Quick reStructuredText reference
Copyright © rst.ninjs.org, 2011

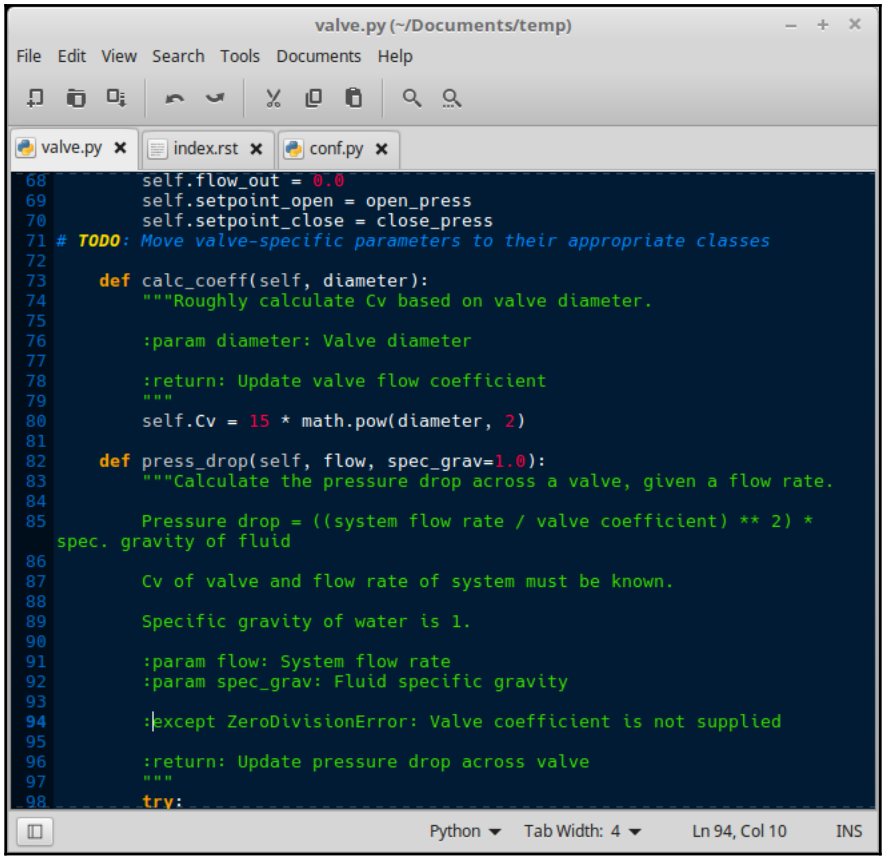

Online reStructuredText editor
Fork me on GitHub
Editor
About
Theme: Basic Nature
This is a paragraph.
This is a quote.
Here's another paragraph. Not separating with a blank line results in text being part of the same paragraph.
italics
bold
backticks
Escaped characters
1. A list item
2. Another list item
A. A different list
a. Sublist
I. Roman numerals
ii. More Roman
A bullet
A sublist
Another sublist
Back to the first sublist
term
A definition to explain what the term is. The term is accented in bold.
Some preformatted code follows:
print("Hello World!")
Title
Subtitle
Section 1
Section 1.1
Section 1.1a
Section 2
Quick reStructuredText reference
Copyright © rst.ninjs.org, 2011

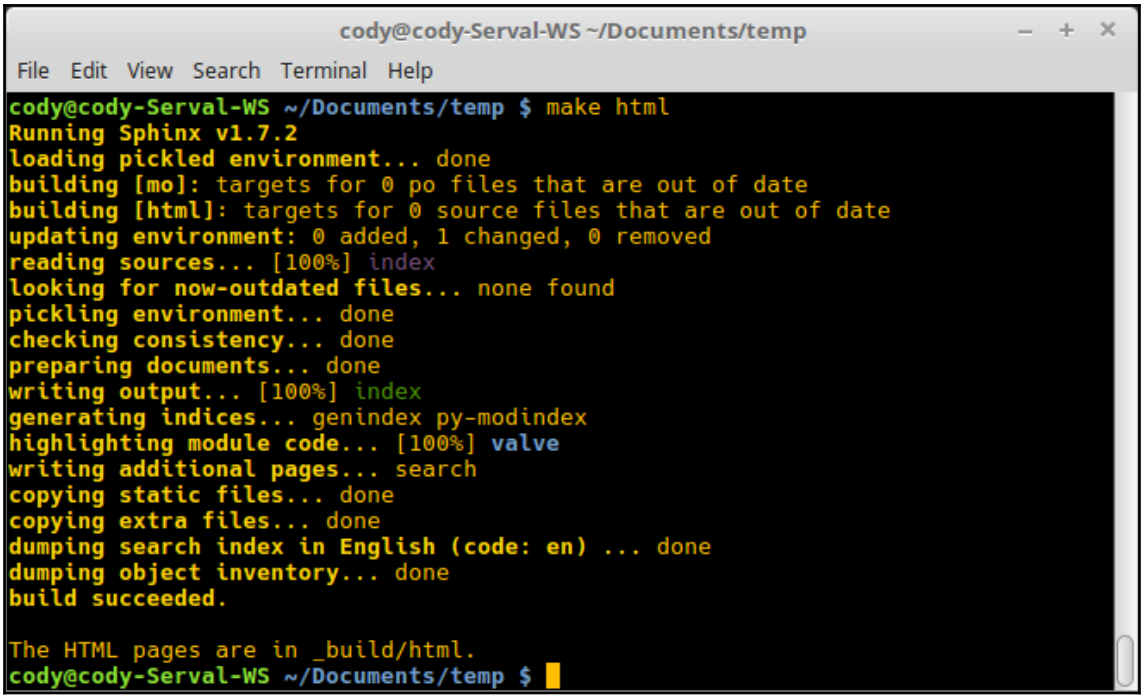

cody@cody-Serval-WS ~/Documents/temp
File Edit View Search Terminal Help
cody@cody-Serval-WS ~/Documents/temp $ sphinx-quickstart
Welcome to the Sphinx 1.7.2 quickstart utility.
Please enter values for the following settings (just press Enter to
accept a default value, if one is given in brackets).
Selected root path: .
You have two options for placing the build directory for Sphinx output.
Either, you use a directory "_build" within the root path, or you separate
"source" and "build" directories within the root path.
> Separate source and build directories (y/n) [n]:
Inside the root directory, two more directories will be created; "_templates"
for custom HTML templates and "_static" for custom stylesheets and other static
files. You can enter another prefix (such as ".") to replace the underscore.
> Name prefix for templates and static dir [_]:
The project name will occur in several places in the built documentation.
> Project name: Server Bandwidth
> Author name(s): Cody Jackson
> Project release []: 0.3
If the documents are to be written in a language other than English,
you can select a language here by its language code. Sphinx will then
translate text that it generates into that language.
For a list of supported codes, see
http://sphinx-doc.org/config.html#confval-language.
> Project language [en]:
The file name suffix for source files. Commonly, this is either ".txt"
or ".rst". Only files with this suffix are considered documents.
> Source file suffix [.rst]:
One document is special in that it is considered the top node of the
"contents tree", that is, it is the root of the hierarchical structure
of the documents. Normally, this is "index", but if your "index"
document is a custom template, you can also set this to another filename.
> Name of your master document (without suffix) [index]:
Sphinx can also add configuration for epub output:
> Do you want to use the epub builder (y/n) [n]:
Indicate which of the following Sphinx extensions should be enabled:
> autodoc: automatically insert docstrings from modules (y/n) [n]: y
> doctest: automatically test code snippets in doctest blocks (y/n) [n]:
> intersphinx: link between Sphinx documentation of different projects (y/n) [n]
:
> todo: write "todo" entries that can be shown or hidden on build (y/n) [n]:
> coverage: checks for documentation coverage (y/n) [n]:
> imgmath: include math, rendered as PNG or SVG images (y/n) [n]:
> mathjax: include math, rendered in the browser by MathJax (y/n) [n]:
> ifconfig: conditional inclusion of content based on config values (y/n) [n]:
> viewcode: include links to the source code of documented Python objects (y/n)
[n]: y
> githubpages: create .nojekyll file to publish the document on GitHub pages (y/
n) [n]:
A Makefile and a Windows command file can be generated for you so that you
only have to run e.g. `make html' instead of invoking sphinx-build
directly.
> Create Makefile? (y/n) [y]:
> Create Windows command file? (y/n) [y]: n
Creating file ./conf.py.
Creating file ./index.rst.
Creating file ./Makefile.
Finished: An initial directory structure has been created.
You should now populate your master file ./index.rst and create other documentat
ion
source files. Use the Makefile to build the docs, like so:
make builder
where "builder" is one of the supported builders, e.g. html, latex or linkcheck.
cody@cody-Serval-WS ~/Documents/temp $

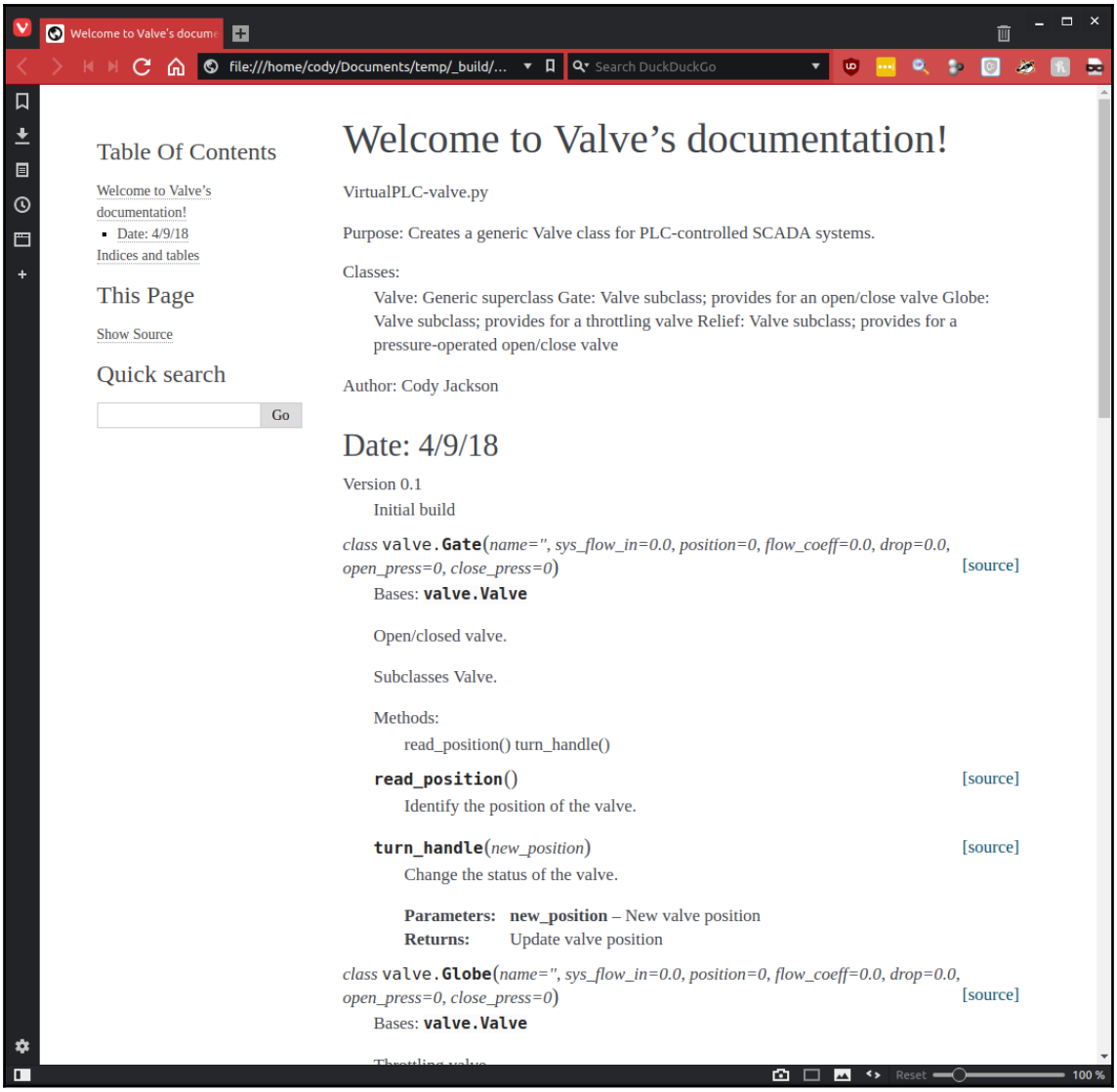

conf.py (~/Documents/temp)
File Edit View Search Tools Documents Help
valve.py
index.rst
conf.py
-*- coding: utf-8 -*-
#
Configuration file for the Sphinx documentation builder.
#
This file does only contain a selection of the most common options. For a
full list see the documentation:
http://www.sphinx-doc.org/en/stable/config
-- Path setup
If extensions (or modules to document with autodoc) are in another directory,
add these directories to sys.path here. If the directory is relative to the
documentation root, use os.path.abspath to make it absolute, like shown here.
#
import os
import sys
sys.path.insert(0, os.path.abspath('.'))
# -- Project information
project = u'Valve'
copyright = u'2018, Cody Jackson'
author = u'Cody Jackson'
# The short X.Y version
Python
Tab Width: 4
Ln 9, Col 16
INS

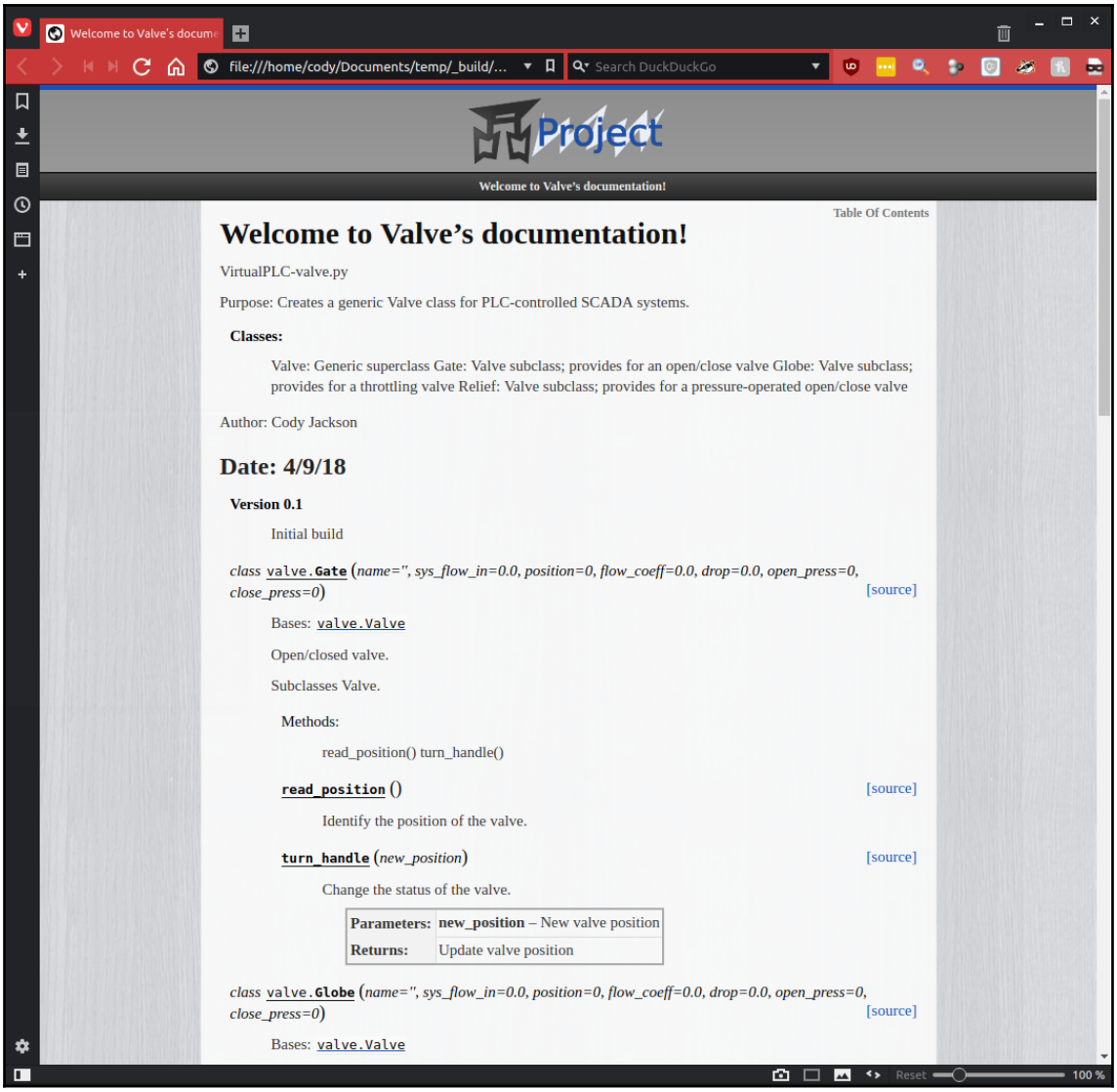

index.rst (~/Documents/temp)
File Edit View Search Tools Documents Help
valve.py
index.rst
conf.py
.. Valve documentation master file, created by
sphinx-quickstart on Fri Apr 27 18:14:38 2018.
You can adapt this file completely to your liking, but it should at least
contain the root `toctree` directive.
Welcome to Valve's documentation!
.. toctree::
:maxdepth: 2
:caption: Contents:
.. automodule:: valve
:members:
:undoc-members:
:show-inheritance:
Indices and tables
* :ref:`genindex`
* :ref:`modindex`
* :ref:`search`
reStructuredText
Tab Width: 4
Ln 16, Col 21
INS

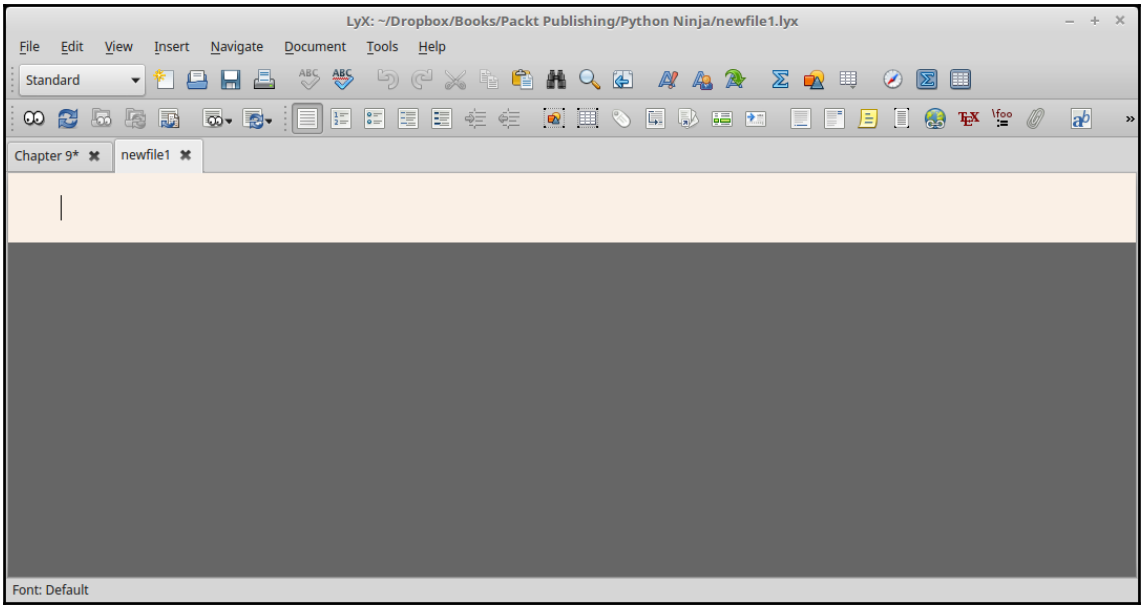

cody@cody-Serval-WS ~/Documents/temp
File Edit View Search Terminal Help
cody@cody-Serval-WS ~/Documents/temp $ make html
Running Sphinx v1.7.2
loading pickled environment... done
building [mo]: targets for 0 po files that are out of date
building [html]: targets for 0 source files that are out of date
updating environment: 0 added, 1 changed, 0 removed
reading sources... [100%] index
/home/cody/Documents/temp/valve.py:docstring of valve.Gate.cls_change_position:6
: WARNING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Gate.press_drop:11: WARNIN
G: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Gate.sys_flow_rate:8: WARN
ING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Globe.cls_change_position:
6: WARNING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Globe.press_drop:11: WARNI
NG: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Globe.sys_flow_rate:8: WAR
NING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Relief.cls_change_position
:6: WARNING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Relief.press_drop:11: WARN
ING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Relief.sys_flow_rate:8: WA
RNING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Valve.cls_change_position:
6: WARNING: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Valve.press_drop:11: WARNI
NG: Field list ends without a blank line; unexpected unindent.
/home/cody/Documents/temp/valve.py:docstring of valve.Valve.sys_flow_rate:8: WAR
NING: Field list ends without a blank line; unexpected unindent.
looking for now-outdated files... none found
pickling environment... done
checking consistency... done
preparing documents... done
writing output... [100%] index
generating indices... genindex py-modindex
highlighting module code... [100%] valve
writing additional pages... search
copying static files... done
copying extra files... done
dumping search index in English (code: en) ... done
dumping object inventory... done
build succeeded, 12 warnings.
The HTML pages are in _build/html.
cody@cody-Serval-WS ~/Documents/temp $

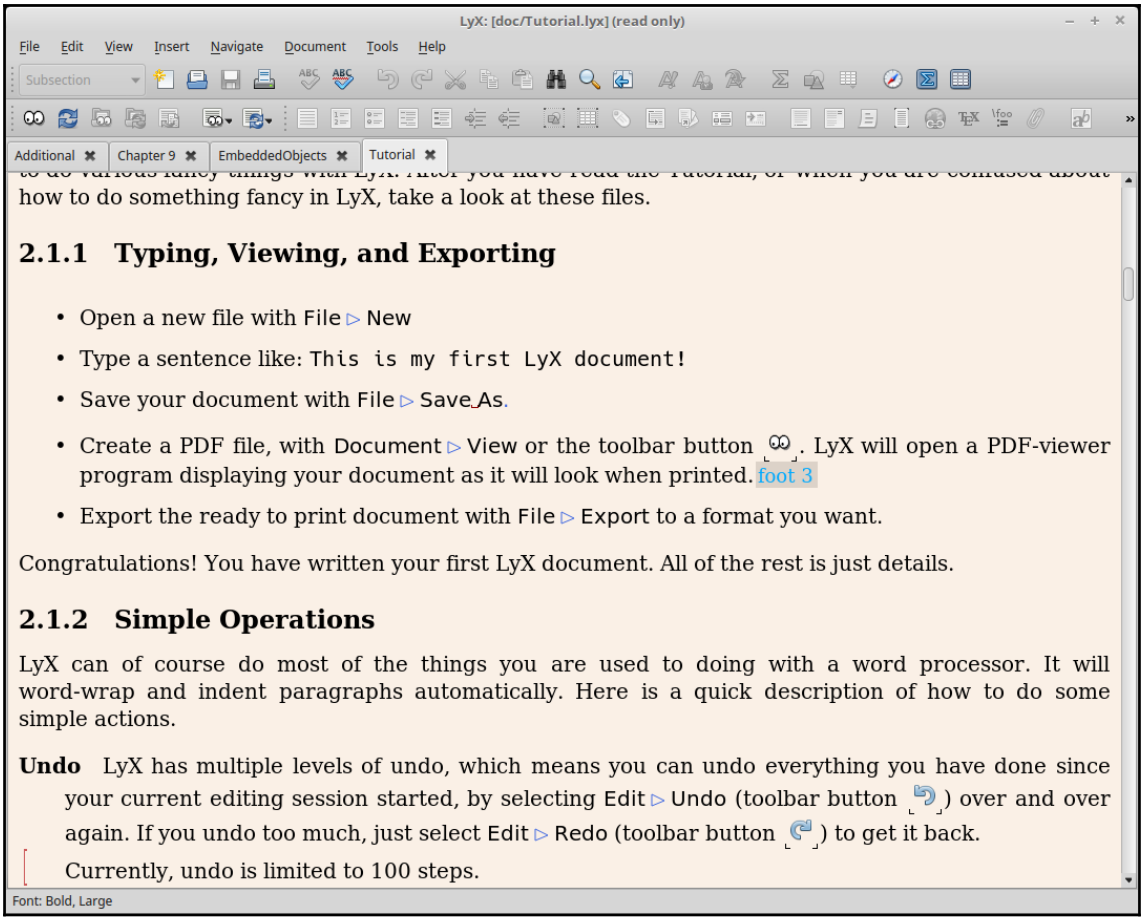

valve.py (~/Documents/temp)
File Edit View Search Tools Documents Help
valve.py
index.rst
conf.py
68 self.flow_out = 0.0
69 self.setpoint_open = open_press
70 self.setpoint_close = close_press
71 # TODO: Move valve-specific parameters to their appropriate classes
72
73 def calc_coeff(self, diameter):
74 """Roughly calculate Cv based on valve diameter.
75
76 :param diameter: Valve diameter
77
78 :return: Update valve flow coefficient
79 """
80 self.Cv = 15 * math.pow(diameter, 2)
81
82 def press_drop(self, flow, spec_grav=1.0):
83 """Calculate the pressure drop across a valve, given a flow rate.
84
85 Pressure drop = ((system flow rate / valve coefficient) ** 2) *
spec. gravity of fluid
86
87 Cv of valve and flow rate of system must be known.
88
89 Specific gravity of water is 1.
90
91 :param flow: System flow rate
92 :param spec_grav: Fluid specific gravity
93
94 :except ZeroDivisionError: Valve coefficient is not supplied
95
96 :return: Update pressure drop across valve
97 """
98 try:
Python
Tab Width: 4
Ln 94, Col 10
INS

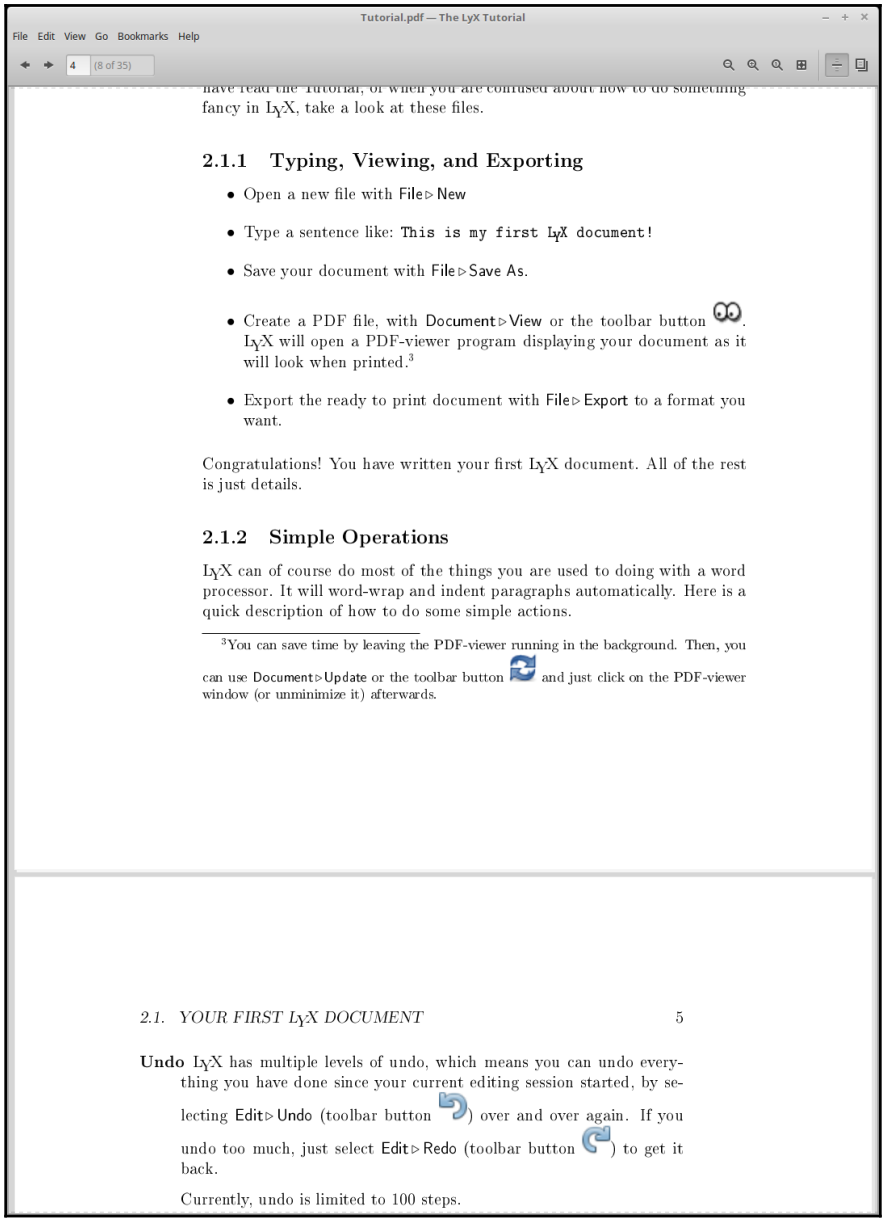

cody@cody-Serval-WS ~/Documents/temp
File Edit View Search Terminal Help
cody@cody-Serval-WS ~/Documents/temp $ make html
Running Sphinx v1.7.2
loading pickled environment... done
building [mo]: targets for 0 po files that are out of date
building [html]: targets for 0 source files that are out of date
updating environment: 0 added, 1 changed, 0 removed
reading sources... [100%] index
looking for now-outdated files... none found
pickling environment... done
checking consistency... done
preparing documents... done
writing output... [100%] index
generating indices... genindex py-modindex
highlighting module code... [100%] valve
writing additional pages... search
copying static files... done
copying extra files... done
dumping search index in English (code: en) ... done
dumping object inventory... done
build succeeded.
The HTML pages are in _build/html.
cody@cody-Serval-WS ~/Documents/temp $

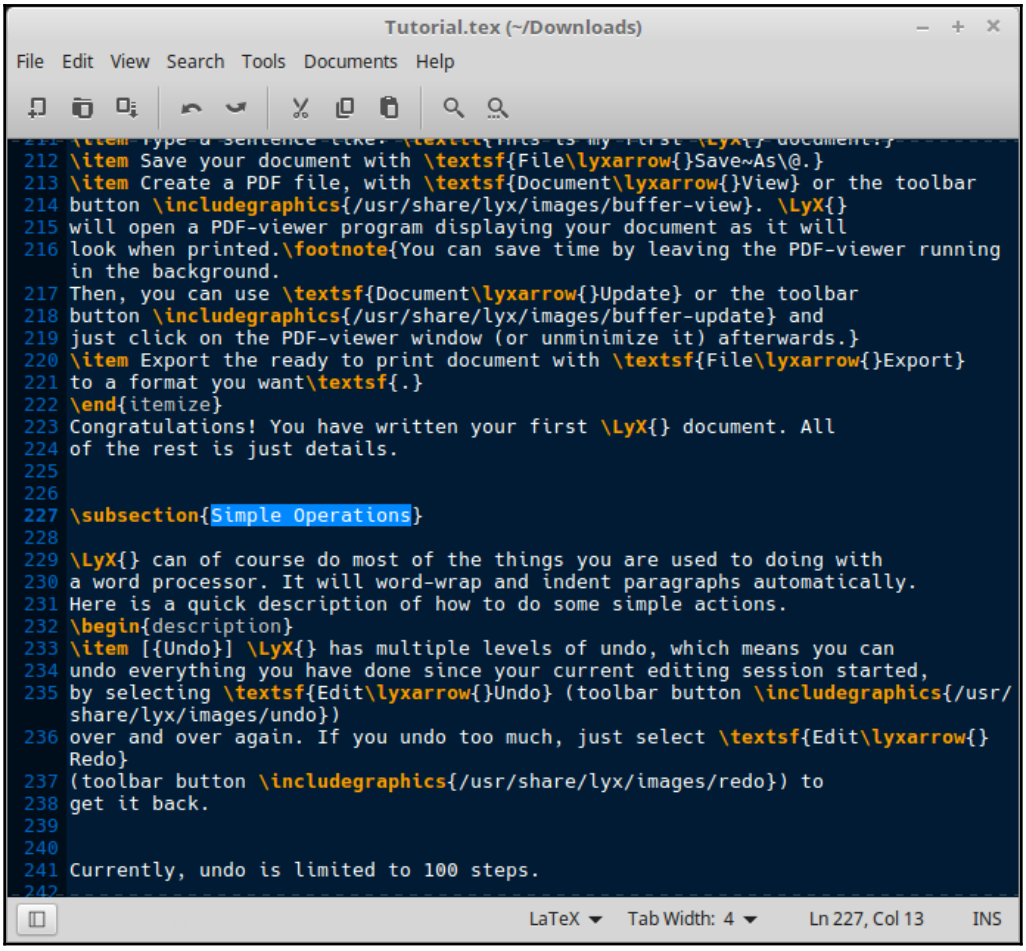

Welcome to Valve's docum
file:///home/cody/Documents/temp/_build/...
Search DuckDuckGo
Table Of Contents
Welcome to Valve's documentation!
Date: 4/9/18
Indices and tables
This Page
Show Source
Quick search
Go
Welcome to Valve's documentation!
VirtualPLC-valve.py
Purpose: Creates a generic Valve class for PLC-controlled SCADA systems.
Classes:
Valve: Generic superclass Gate: Valve subclass; provides for an open/close valve Globe: Valve subclass; provides for a throttling valve Relief: Valve subclass; provides for a pressure-operated open/close valve
Author: Cody Jackson
Date: 4/9/18
Version 0.1
Initial build
class valve.Gate(name='', sys_flow_in=0.0, position=0, flow_coeff=0.0, drop=0.0, open_press=0, close_press=0) [source]
Bases: valve.Valve
Open/closed valve.
Subclasses Valve.
Methods:
read_position() turn_handle()
read_position() [source]
Identify the position of the valve.
turn_handle(new_position) [source]
Change the status of the valve.
Parameters: new_position – New valve position
Returns: Update valve position
class valve.Globe(name='', sys_flow_in=0.0, position=0, flow_coeff=0.0, drop=0.0, open_press=0, close_press=0) [source]
Bases: valve.Valve
Reset
100 %

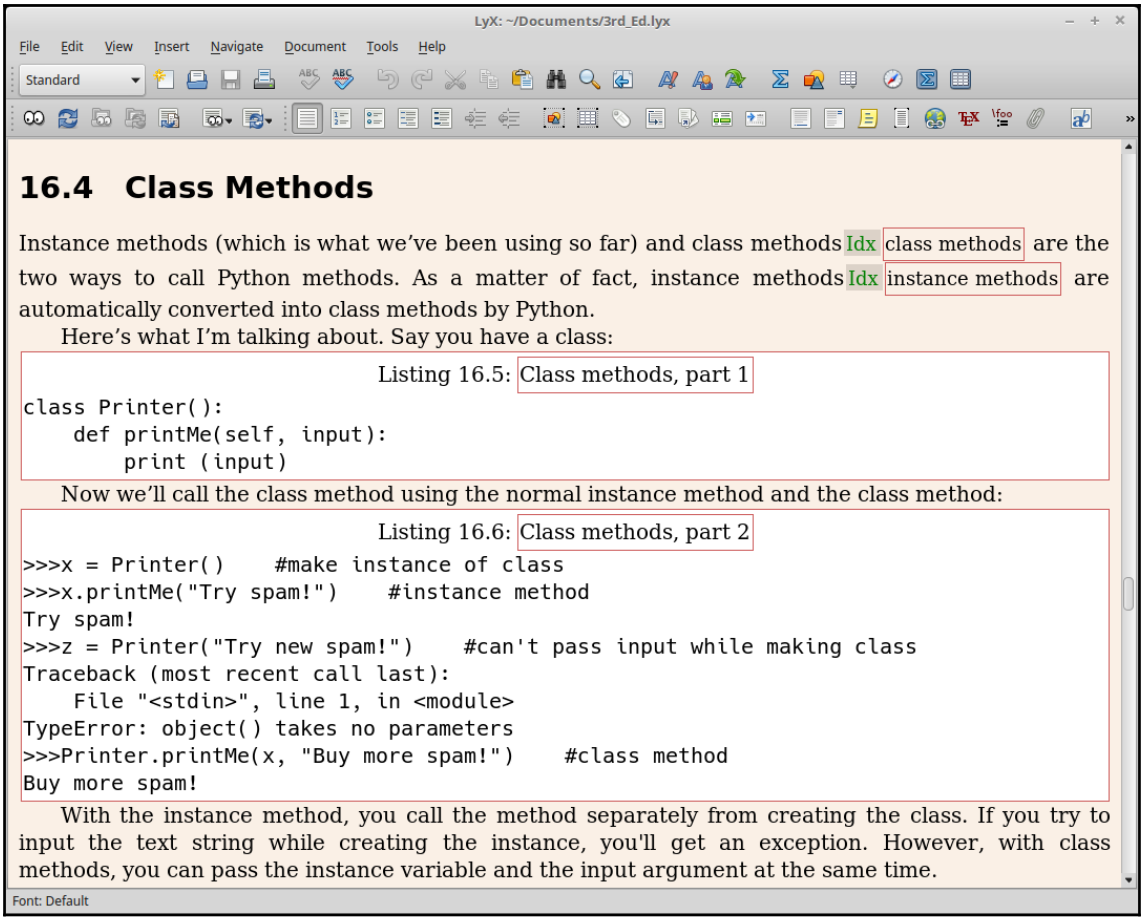

Welcome to Valve's documentation!
Table Of Contents
Welcome to Valve's documentation!
VirtualPLC-valve.py
Purpose: Creates a generic Valve class for PLC-controlled SCADA systems.
Classes:
Valve: Generic superclass Gate: Valve subclass; provides for an open/close valve Globe: Valve subclass; provides for a throttling valve Relief: Valve subclass; provides for a pressure-operated open/close valve
Author: Cody Jackson
Date: 4/9/18
Version 0.1
Initial build
class valve.Gate (name='', sys_flow_in=0.0, position=0, flow_coeff=0.0, drop=0.0, open_press=0, close_press=0) [source]
Bases: valve.Valve
Open/closed valve.
Subclasses Valve.
Methods:
read_position() turn_handle()
read_position () [source]
Identify the position of the valve.
turn_handle (new_position) [source]
Change the status of the valve.
Parameters: new_position – New valve position
Returns: Update valve position
class valve.Globe (name='', sys_flow_in=0.0, position=0, flow_coeff=0.0, drop=0.0, open_press=0, close_press=0) [source]
Bases: valve.Valve

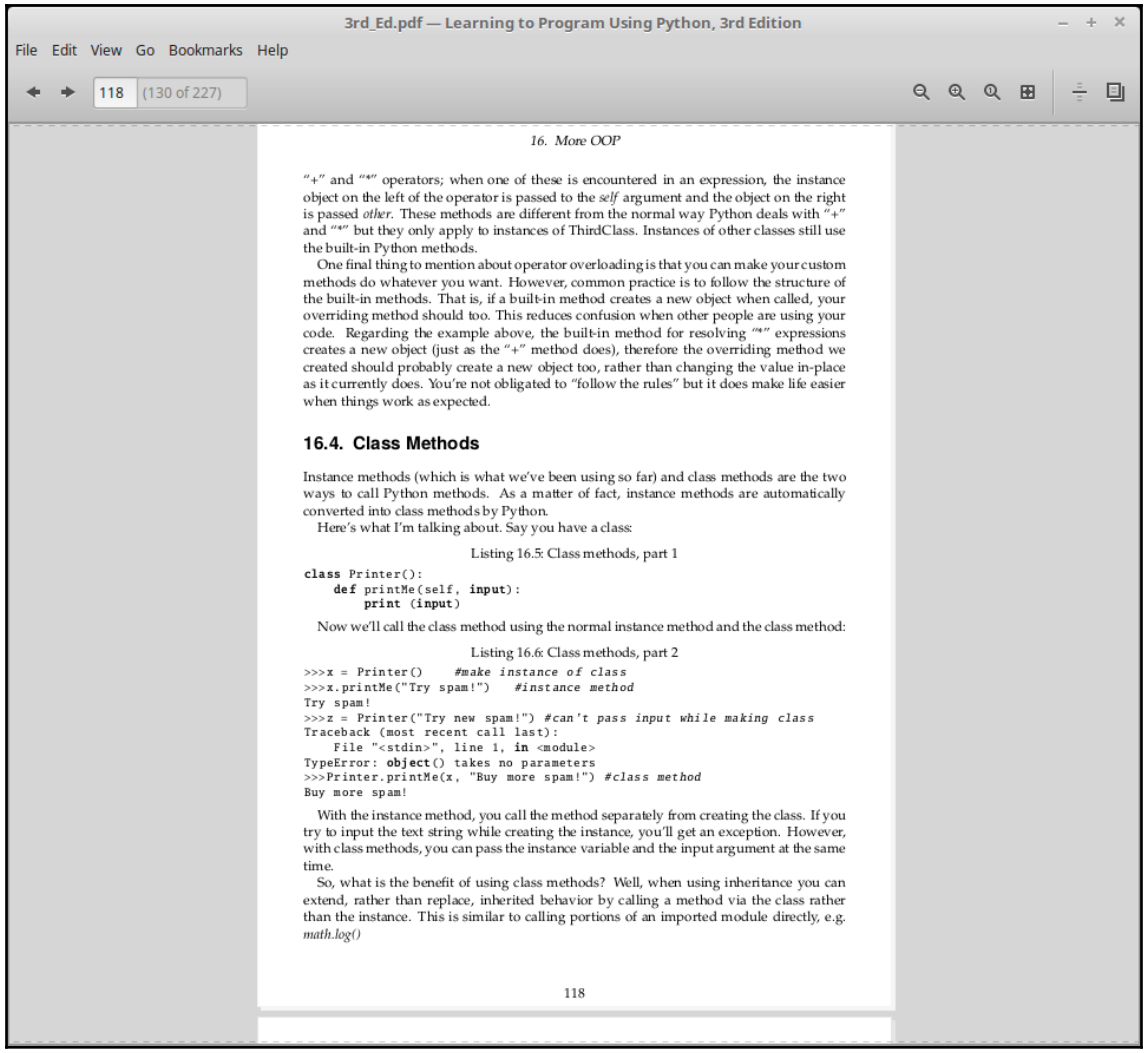

LyX: ~/Dropbox/Books/Packt Publishing/Python Ninja/newfile1.lyx
File Edit View Insert Navigate Document Tools Help
Standard
Chapter 9*
newfile1
Font: Default


LyX: [doc/Tutorial.lyx] (read only)
File Edit View Insert Navigate Document Tools Help
Subsection
Additional
Chapter 9
EmbeddedObjects
Tutorial
how to do something fancy in LyX, take a look at these files.
2.1.1 Typing, Viewing, and Exporting
• Open a new file with File ▷ New
• Type a sentence like: This is my first LyX document!
• Save your document with File ▷ Save As.
• Create a PDF file, with Document ▷ View or the toolbar button . LyX will open a PDF-viewer program displaying your document as it will look when printed. foot 3
• Export the ready to print document with File ▷ Export to a format you want.
Congratulations! You have written your first LyX document. All of the rest is just details.
2.1.2 Simple Operations
LyX can of course do most of the things you are used to doing with a word processor. It will word-wrap and indent paragraphs automatically. Here is a quick description of how to do some simple actions.
Undo LyX has multiple levels of undo, which means you can undo everything you have done since your current editing session started, by selecting Edit ▷ Undo (toolbar button) over and over again. If you undo too much, just select Edit ▷ Redo (toolbar button) to get it back.
Currently, undo is limited to 100 steps.
Font: Bold, Large


Tutorial.pdf — The LyX Tutorial
File Edit View Go Bookmarks Help
4 (8 of 35)
have read the Tutorial, or when you are confused about how to do something
fancy in LyX, take a look at these files.
2.1.1 Typing, Viewing, and Exporting
Open a new file with File▷New
Type a sentence like: This is my first LyX document!
Save your document with File▷Save As.
Create a PDF file, with Document▷View or the toolbar button . LyX will open a PDF-viewer program displaying your document as it will look when printed.[3]
Export the ready to print document with File▷Export to a format you want.
Congratulations! You have written your first LyX document. All of the rest is just details.
2.1.2 Simple Operations
LyX can of course do most of the things you are used to doing with a word processor. It will word-wrap and indent paragraphs automatically. Here is a quick description of how to do some simple actions.
[3]You can save time by leaving the PDF-viewer running in the background. Then, you can use Document▷Update or the toolbar button and just click on the PDF-viewer window (or unminimize it) afterwards.
2.1. YOUR FIRST LyX DOCUMENT 5
Undo LyX has multiple levels of undo, which means you can undo everything you have done since your current editing session started, by selecting Edit▷Undo (toolbar button) over and over again. If you undo too much, just select Edit▷Redo (toolbar button) to get it back.
Currently, undo is limited to 100 steps.


Tutorial.tex (~/Downloads)
File Edit View Search Tools Documents Help
\item Save your document with \textsf{File\lyxarrow{}Save~As\@.}
\item Create a PDF file, with \textsf{Document\lyxarrow{}View} or the toolbar
button \includegraphics{/usr/share/lyx/images/buffer-view}. \LyX{}
will open a PDF-viewer program displaying your document as it will
look when printed.\footnote{You can save time by leaving the PDF-viewer running
in the background.
Then, you can use \textsf{Document\lyxarrow{}Update} or the toolbar
button \includegraphics{/usr/share/lyx/images/buffer-update} and
just click on the PDF-viewer window (or unminimize it) afterwards.}
\item Export the ready to print document with \textsf{File\lyxarrow{}Export}
to a format you want\textsf{.}
\end{itemize}
Congratulations! You have written your first \LyX{} document. All
of the rest is just details.
\subsection{Simple Operations}
\LyX{} can of course do most of the things you are used to doing with
a word processor. It will word-wrap and indent paragraphs automatically.
Here is a quick description of how to do some simple actions.
\begin{description}
\item [{Undo}] \LyX{} has multiple levels of undo, which means you can
undo everything you have done since your current editing session started,
by selecting \textsf{Edit\lyxarrow{}Undo} (toolbar button \includegraphics{/usr/
share/lyx/images/undo})
over and over again. If you undo too much, just select \textsf{Edit\lyxarrow{}
Redo}
(toolbar button \includegraphics{/usr/share/lyx/images/redo}) to
get it back.
Currently, undo is limited to 100 steps.
LaTeX
Tab Width: 4
Ln 227, Col 13
INS


LyX: ~/Documents/3rd_Ed.lyx
File Edit View Insert Navigate Document Tools Help
Standard
16.4 Class Methods
Instance methods (which is what we've been using so far) and class methods Idx class methods are the two ways to call Python methods. As a matter of fact, instance methods Idx instance methods are automatically converted into class methods by Python.
Here's what I'm talking about. Say you have a class:
Listing 16.5: Class methods, part 1
class Printer():
def printMe(self, input):
print (input)
Now we'll call the class method using the normal instance method and the class method:
Listing 16.6: Class methods, part 2
>>>x = Printer() #make instance of class
>>>x.printMe("Try spam!") #instance method
Try spam!
>>>z = Printer("Try new spam!") #can't pass input while making class
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
TypeError: object() takes no parameters
>>>Printer.printMe(x, "Buy more spam!") #class method
Buy more spam!
With the instance method, you call the method separately from creating the class. If you try to input the text string while creating the instance, you'll get an exception. However, with class methods, you can pass the instance variable and the input argument at the same time.
Font: Default



"+" and "*" operators; when one of these is encountered in an expression, the instance object on the left of the operator is passed to the *self* argument and the object on the right is passed *other*. These methods are different from the normal way Python deals with "+" and "*" but they only apply to instances of ThirdClass. Instances of other classes still use the built-in Python methods.

One final thing to mention about operator overloading is that you can make your custom methods do whatever you want. However, common practice is to follow the structure of the built-in methods. That is, if a built-in method creates a new object when called, your overriding method should too. This reduces confusion when other people are using your code. Regarding the example above, the built-in method for resolving "*" expressions creates a new object (just as the "+" method does), therefore the overriding method we created should probably create a new object too, rather than changing the value in-place as it currently does. You're not obligated to "follow the rules" but it does make life easier when things work as expected.

16.4. Class Methods

Instance methods (which is what we've been using so far) and class methods are the two ways to call Python methods. As a matter of fact, instance methods are automatically converted into class methods by Python.

Here's what I'm talking about. Say you have a class:

Listing 16.5: Class methods, part 1

```
class Printer():
    def printMe(self, input):
        print (input)
```

Now we'll call the class method using the normal instance method and the class method:

Listing 16.6: Class methods, part 2

```
>>>x = Printer()     #make instance of class
>>>x.printMe("Try spam!")    #instance method
Try spam!
>>>z = Printer("Try new spam!") #can't pass input while making class
Traceback (most recent call last):
    File "<stdin>", line 1, in <module>
TypeError: object() takes no parameters
>>>Printer.printMe(x, "Buy more spam!") #class method
Buy more spam!
```

With the instance method, you call the method separately from creating the class. If you try to input the text string while creating the instance, you'll get an exception. However, with class methods, you can pass the instance variable and the input argument at the same time.

So, what is the benefit of using class methods? Well, when using inheritance you can extend, rather than replace, inherited behavior by calling a method via the class rather than the instance. This is similar to calling portions of an imported module directly, e.g. *math.log()*