

Errata for
Python Programming and Visualization for Scientists

A. J. DeCaria
October 5, 2018

This page documents errors that were discovered after the book went to press. Those items marked with an asterisk * have been corrected in the second printing. To reduce the potential for confusion, instructors assigning this book are encouraged to distribute this page to their students. An up-to-date copy can be found at www.sundogpublishing.com/PythonErrata.pdf

If you find errors not listed here, please send a message describing the error to feedback@sundogpublishing.com.

- * **p. 14:** In Sec. 2.1.5 in the examples for showing the use of continuation lines, there are some single quotes missing and/or some erroneous spaces. The examples should read

```
y = 'The quick brown fox jumped over the lazy dog.'  
y = 'The quick brown fox jumped \  
over the lazy dog.'  
y = 'The quick brown fox jumped' \  
    ' over the lazy dog.'
```

- * **p. 20:** The first sentence in Sec. 2.5 should be changed to read “If a variable is created but not explicitly assigned a value or object, then it is of the None data type.”

- * **p. 44:** In Sec. 4.7.5 the fifth bullet, discussing the `np.log(x, b)` function for base b logarithms, should be deleted.

p. 44: In Sec. 4.7.5 the description of `np.log1p(x)` should read “returns $\ln(x + 1)$, and is more accurate than using `np.log(x + 1)`.”

- * **p. 50:** In the code description at the bottom of the page, the colons are missing at the end of the `elif` statements for *condition 2* and *condition 3*.

- * **p. 52:** In the code example in Sec. 5.3.3, the first print statement should all be on one line, and the argument to the `len()` function should be `mylist`. The line should read:

```
print('The list has {0:d} elements.'.format(len(mylist)))
```

- * **p. 54:** The code description showing the use of `enumerate()` contains an erroneous equals sign. It should read:

```
for i, val in enumerate(itobj):  
    [code block]
```

- * **p. 95:** Delete the equals sign ‘=’ in the first code template.

- * **p. 104:** In the code example delete the empty parentheses ‘()’ after ‘float’.

- * **p. 106:** In the code example, for the definition of the `volume(self)` method, the line after the `return` statement should be on the same line as the `return` statement.

pp. 107-108: In Sec. 9.5 the entire code example should be replaced with

```
class moist_parcel(dry_parcel):
    '''A class for moist air parcel objects.'''

    def __init__(self,p,T,m,r):
        dry_parcel.__init__(self,p,T,m)
        self.r = float(r) # Mixing ratio (g/kg)
        self.q = (r)/(1 + r/1000.) # Specific hum (g/kg)
    def moles(self):
        '''Returns number of moles in parcel.'''
        q = self.q/1000. # dimensionless specific humidity
        return self.mass*(1/Md - q/Md + q/Mw)

    def Tv(self):
        '''Returns virtual temperature in Kelvin'''
        q = self.q/1000. # dimensionless specific humidity
        return self.temperature*(1 + 0.61*q)
```

* **p. 152:** In Sec. 12.2.3 under the subheading ‘Colors’, the first sentence should read ‘The colors of the contour lines are controlled using the colors keyword.’

* **p. 176:** In the code sample following “We examine their attributes as shown:”, the colons in the print () statements should be inside of the single quotes.

```
print(a, ':', Latitude.attrs[a])
print(a, ':', TC.attrs[a])
```

* **p. 203:** The caption on the table should be ‘Table 16.1’, not ‘Fig. 16.1’.

* **p. 203:** In the last and third-to-last entries of the table the equals sign should be deleted.

* **p. 209:** In Sec. 16.3.8, midway down, in the sentence that begins
“The syntax `expr1(?!=expr2)...`”,
the equals sign after the exclamation should be deleted.

* **p. 211:** In Sec. 16.5.1, the pattern in the last code example should be `r'[A-Za-z]'`.

p. 219: In Sec. 17.3.1 where `today()` is described, `today()` is actually a *class method* of the date class, not a function of the datetime module. The usage example should be

```
> now = dt.date.today()
```

* **p. 219:** In Sec. 17.3.1 where the `replace()` method is described, `replace()` actually returns a new date object. It does not modify the original object.

* **p. 221:** In the first sentence at the top of the page the arguments to the `dt.timedelta()` function should be *days* and *seconds*, not hours and seconds.

* **p. 224-225:** The arguments of exponentials in (18.1), (18.2), (18.6), and (18.7) should be divided by N . In the forward transforms (18.1) and (18.6) this should be $\exp(-i2\pi jm/N)$, while the inverse transforms (18.2) and (18.7) should be $\exp(i2\pi jm/N)$.