Python 3: The Python Environment

Lesson 3, Project 1

Handed in: 4 Jan 2016 02:51:53PM Graded: 5 Jan 2016 12:42:46PM

Here are your instructions:

Create a **Python3\_Homework03** project and assign it to your **Python3\_Homework** working set. In the **Python3\_Homework03/src** folder, create a file named **decoder.py**, which contains an iterator named **alphabator**. When passed a list, simply return objects as-is unless they are integers between 1 and 26, in which case it should convert that number to the corresponding letter. The integer-to-letter correspondence is 1=A, 2=B, 3=C, 4=D, and so on.

You may use any technique you've learned in lesson 3 to execute this project.

Your alphabator iterator must pass the following unit test.

test\_decoder.py:

from string import ascii\_uppercase  
import unittest  
  
from decoder import alphabator  
  
class TestAlpha(unittest.TestCase):  
  
 def test\_easy\_26(self):  
 a = alphabator(range(1,27))  
 self.assertEqual(list(ascii\_uppercase), list(a))  
  
 def test\_upper\_range(self):  
 a = alphabator(range(40,50))   
 self.assertEqual(list(range(40, 50)), list(a))  
  
 def test\_various\_objects(self):  
 l = ['python', object, ascii\_uppercase, 10, alphabator]  
 a = list(alphabator(l))  
 self.assertNotEqual(l[3], a[3])  
 self.assertEqual("J", a[3])  
 self.assertTrue(isinstance(a[1], object))  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 unittest.main()

Submit **decoder.py** and **test\_decoder.py** when they are working to your satisfaction.

##### **Your Comment:**

*no comment given*

##### **Items Handed In**

* [Open Project Handed In](https://students.oreillyschool.com/student/project/?/.handin/147-6586-1/com.ost.mboyd.147.6586.1.Python3Homework03.zip)

### **Overall Comments:**

Excellent.  
  
I remembered your email about the error so fixed it for you.  
  
"""  
decoder.py: go through a list, and if a value is an integer  
 between 1 and 26, replace it with a letter  
"""  
from string import ascii\_uppercase  
  
def alphabator(lst):  
 for item in lst:  
 if item in range(1, 27):  
 yield list(ascii\_uppercase)[item - 1]  
 else:  
 yield item  
  
  
-Kirby

### **Grade:**