Python Tutor: Visualize code in Python, JavaScript, C, C++, and Java

Python 3.6
(known limitations)

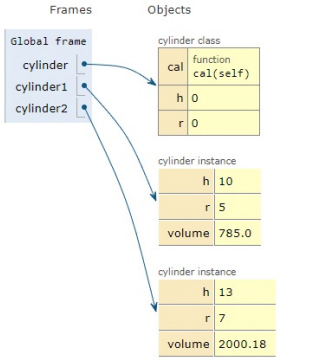
```
1  class cylinder():
2      r=0
3      h=0
4      def cal(self):
5          self.volume=3.14*(self.r*self.r)*self.h
6          return self.volume
7
8  cylinder1=cylinder()
9  cylinder1.r=5
10 cylinder1.h=10
11
12
13
14 cylinder2=cylinder()
15 cylinder2.r=7
16 cylinder2.h=13
17
18 print(str(cylinder1.cal()))
19 print(str(cylinder2.cal()))
```

Edit this code

line that just executed
next line to execute

Print output (drag lower right corner to resize)

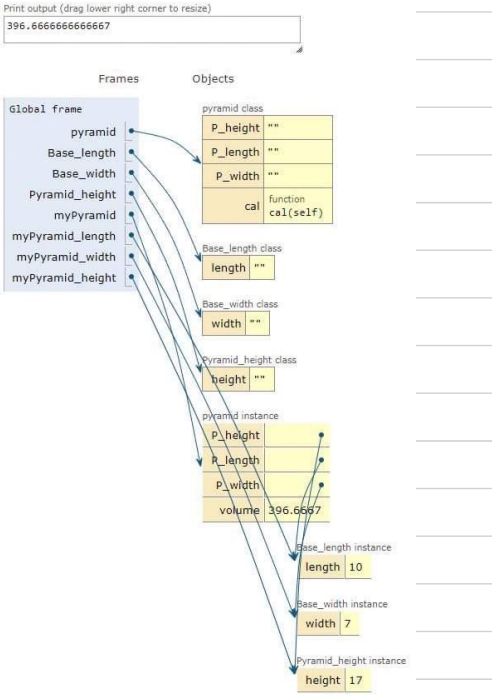
```
785.0
2000.1800000000003
```




```
1  class pyramid():
2      P_length=''
3      P_width=''
4      P_height=''
5      def cal(self):
6          self.volume=(self.P_length.length*self.P_width.width*self.P_height
7          return self.volume
8
9  class Base_length():
10     length=''
11
12 class Base_width():
13     width=''
14
15 class Pyramid_height():
16     height=''
17
18 myPyramid=pyramid()
19 myPyramid_length=Base_length()
20 myPyramid_width=Base_width()
21 myPyramid_height=Pyramid_height()
22
23 myPyramid_length.length=10
24 myPyramid_width.width=7
25 myPyramid_height.height=17
26
27 myPyramid.P_length=myPyramid_length
28 myPyramid.P_width=myPyramid_width
29 myPyramid.P_height=myPyramid_height
30
31 print(myPyramid.cal())
```

Print output (drag lower right corner to resize)

```
396.6666666666667
```



```python
class Node:
    def __init__(self, dataval=None):
        self.dataval = dataval
        self.poiter = None

class SLinkedList:
    def __init__(self):
        self.header = None

list = SLinkedList()

e1 = Node("44")
e2 = Node("36")
e3 = Node("90")
e4 = Node("10")
e5 = Node("60")
e6 = Node("99")

list.header = e1
e1.poiter = e2
e2.poiter = e3
e3.poiter = e4
e4.poiter = e5
e5.poiter = e6
```

header → 44 → 36 → 90 → 10 → 60 → 99

```python
class Node:
    def __init__(self, dataval=None):
        self.dataval = dataval
        self.poiter = None

class SLinkedList:
    def __init__(self):
        self.header = None

list = SLinkedList()

e1 = Node("44")
e2 = Node("36")
e3 = Node("90")
e4 = Node("10")
e5 = Node("60")
e6 = Node("99")
e7 = Node("104")

list.header = e1
e1.poiter = e2
e2.poiter = e3
e3.poiter = e4
e4.poiter = e5
e5.poiter = e6

e7.poiter = e1
list.header = e7
```

header → 44 → 36 → 90 → 10 → 60 → 99

↑

104

header → 44 → 36 → 90 → 10 → 60 → 99

↑

104

header → 104 → 44 → 36 → 90 → 10 → 60 → 99

```python
class Node:
    def __init__(self, dataval=None):
        self.dataval = dataval
        self.poiter = None

class SLinkedList:
    def __init__(self):
        self.header = None

list = SLinkedList()

e1 = Node("44")
e2 = Node("36")
e3 = Node("90")
e4 = Node("10")
e5 = Node("60")
e6 = Node("99")
e7 = Node("104")
e8 = Node("57")

list.header = e1
e1.poiter = e2
e2.poiter = e3
e3.poiter = e4
e4.poiter = e5
e5.poiter = e6

e7.poiter = e1
list.header = e7

e8.poiter = e6
```

header → 104 → 44 → 36 → 90 → 10 → 60 → 99

header → 104 → 44 → 36 → 90 → 10 → 60 → 99 → 57

```python
class Node:
    def __init__(self, dataval=None):
        self.dataval = dataval
        self.poiter = None

class SLinkedList:
    def __init__(self):
        self.header = None

list = SLinkedList()

e1 = Node("44")
e2 = Node("36")
e3 = Node("90")
e4 = Node("10")
e5 = Node("60")
e6 = Node("99")
e7 = Node("104")
e8 = Node("57")

list.header = e1
e1.poiter = e2
e2.poiter = e3
e3.poiter = e4
e4.poiter = e5
e5.poiter = e6

e7.poiter = e1
list.header = e7

e8.poiter = e6

e3.poiter = e5
```

header → 104 → 44 → 36 → 90 → 10 → 60 → 99 → 57

header → 104 → 44 → 36 → 90 → 60 → 99 → 57