EXERCISES

MULTIPLE INHERITANCE & ABSTRACT CLASSES

- define a class *Cylinder* derived from both *Cube* and *Circle*
- base has radius r from circle
- height h is *side* from cube
- write code for its volume and (surface) area

$$\text{Volume} = \pi r^2 h$$

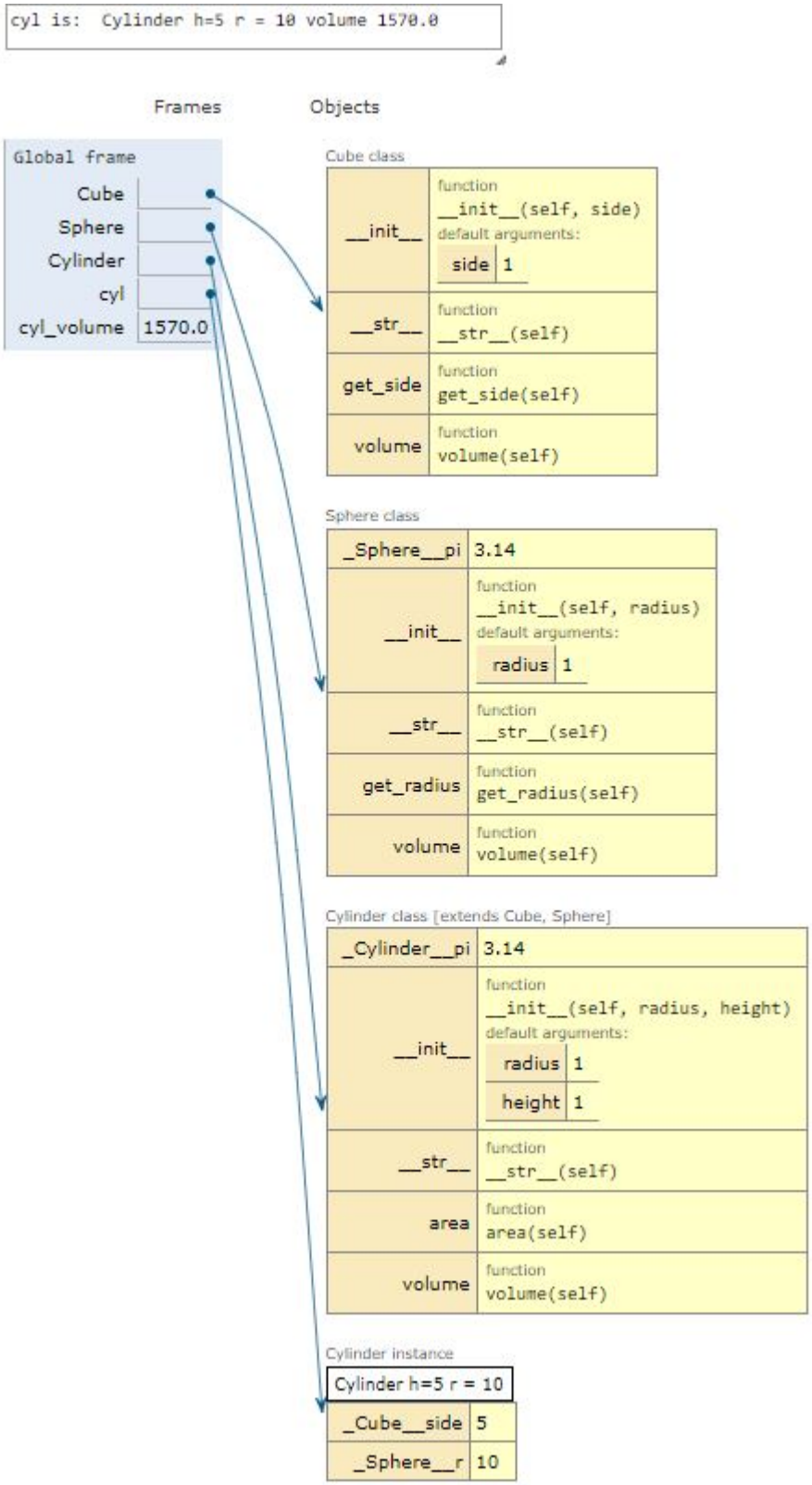
$$\text{Area} = 2\pi r(r + h)$$

Solution:

```
class Cube:
    def __init__(self, side = 1):
        self.__side = side

    def __str__(self):
        return 'cube with side {}'\
        .format(self.__side)

    def get_side(self):
        return self.__side

    def volume(self):
        return self.__side ** 3

class Sphere():
    __pi = 3.14

    def __init__(self, radius = 1):
        self.__r = radius

    def __str__(self):
        return 'sphere with radius {}'\
```

```
            .format(self.__r)

    def get_radius(self):          # accessor
        return self.__r            # mutator

    def volume(self):
        return 4 * Sphere.__pi*self.__r**3 / 3

class Cylinder(Cube, Sphere):
    __pi = 3.14
    def __init__(self, radius = 1, height =1):
        Cube.__init__(self, height)
        Sphere.__init__(self, radius)

    def __str__(self):
        r = self.get_radius()
        h = self.get_side()
        return "Cylinder h={} r = {}".format(h,r)

    def volume(self):
        r = self.get_radius()
        h = self.get_side()
        return self.__pi * (r**2) * h
```

```
    def area(self):
        r = self.get_radius()
        h = self.get_side()
        return 2 * self.__pi * r * (r + h)

cyl = Cylinder(10, 5)
cyl_volume = cyl.volume()
print("cyl is: ", cyl, "volume", cyl_volume)
```


cyl is: Cylinder h=5 r = 10 volume 1570.0
Frames
Objects
Global frame
Cube
Sphere
Cylinder
cyl
cyl_volume
1570.0
Cube class
__init__
function
__init__(self, side)
default arguments:
side 1
__str__
function
__str__(self)
get_side
function
get_side(self)
volume
function
volume(self)
Sphere class
_Sphere__pi
3.14
__init__
function
__init__(self, radius)
default arguments:
radius 1
__str__
function
__str__(self)
get_radius
function
get_radius(self)
volume
function
volume(self)
Cylinder class [extends Cube, Sphere]
_Cylinder__pi
3.14
__init__
function
__init__(self, radius, height)
default arguments:
radius 1
height 1
__str__
function
__str__(self)
area
function
area(self)
volume
function
volume(self)
Cylinder instance
Cylinder h=5 r = 10
_Cube__side
5
_Sphere__r
10