

PA #6

DMSBA 5061

ANDREW NALUNDASAN

11/13/2020

(1) (a)

```
class Rectangle(object):
```

```
    """ Rectangle class for PO#6 """
```

```
    def __init__(self, height, width):
```

```
        """ Construct a rectangle with height  
        + width dimensions """
```

```
        self.height = height
```

```
        self.width = width
```

```
>>> rect = Rectangle(1.0, 2.5)
```

(b)

```
class Rectangle(object):
```

```
    """
```

```
    def __repr__(self):
```

```
        return "Rectangle(%.1f, %.1f)" %  
               (self.height, self.width)
```

```
>>> rect = Rectangle(1.0, 2.5)
```

```
>>> repr(rect)
```

```
'Rectangle(1.0, 2.5)'
```

c) class Rectangle (object):

```
def area(self):  
    return self.height * self.width
```

```
>>> rect = Rectangle(1.0, 2.5)  
>>> rect.area()  
2.5
```

```
def perimeter(self):  
    return (self.height * 2) + (self.width * 2)
```

```
>>> rect = Rectangle(1.0, 2.5)  
>>> rect.perimeter()  
7
```

$1 \cdot 2 = 2$
 $2 \cdot 5 = 5$
 $2 + 5 = 7$

d) >>> rect = Rectangle(2.0, 2.5)

```
>>> repr(rect)  
'Rectangle(2.0, 2.5)'  
>>> rect.area()
```

```
5.0  
>>> rect.perimeter()  
9
```

$2 \cdot 2 = 4$
 $2 \cdot 5 = 5$
 $4 + 5 = 9$

$1 \cdot 2.5 = 2.5$
 $2 \cdot 2.5 = 5.0$
 $2.5 + 5.0 = 7.5$

(2) class Rectangle(object):
 """ create a rectangle \pm height + width dimensions """

 def __init__(self, height, width):
 """ initialize rectangle dimensions """
 self.height = height
 self.width = width

 def __repr__(self):
 """ return representation of details of the rectangle's dimensions """
 return "Rectangle(%.1f, %.1f)" % (self.height, self.width)
 # use %.1f because inserting a float

 def area(self):
 """ area of rectangle is: area = height * width """
 return self.height * self.width

 def perimeter(self):
 """ perimeter = $2(\text{height} + \text{width})$ """
 return (self.height * 2) + (self.width * 2)