Day 3 - Assignments (Part 1)

1. Basic Class

1.1 Class `Point`

Create a Python class `Point` which contains 2 attributes, `x` and `y`, representing x and y coordinate of the point.

- Implement initializer method which initialize `x` and `y`.
- Implement a instance method `dist_to_origin()` which return distance from origin using formular `math.sqrt(x**2 + y**2)`.
- Implement `__str__()` which returns "(x,y)", e.g. "(3.0,4.0)"

Sample Output

```
(3.0, 4.0)
5.0
```

In [ ]:

```
class Point:

    def __init__(self, x = 1, y = 1):
        self.x = x
        self.y = y

    def dist_to_origin(self):
        import math
        return math.sqrt(self.x**2 + self.y**2)

    def __str__(self):
        return '({},{})'.format(self.x, self.y)

p = Point(3, 4)
print(p)
d = p.dist_to_origin()
print(d)
```

1.2 Class `Rectangle`

Create a Python class `Rectangle` which contains 3 attributes, `width`, `height` and `corner`. The `corner` is of `Point` type, which gives coordinate of bottom left corner of the rectangle.

- Implement initializer method which initialize `width`, `height` and `corner`.
- Implement a instance method `get_centre()` which returns a `Point` boject representing centre point of the rectance.

- Implement a instance method `scale(val)` which scale width and height by `val` times.

Sample Output

```
(12.0, 22.0)  # print return value from `get_centre()`
Rectangle(20, 40) at point (2.0, 2.0)   # return value of `str()`
```

In [ ]:

```
class Rectangle:

    def __init__(self, w, h, p = Point(0,0)):
        self.width = w
        self.height = h
        self.corner = p

    def get_centre(self):
        x = self.width / 2 + self.corner.x
        y = self.height / 2  + self.corner.y
        return Point(x, y)

    def scale(self, val):
        self.width = self.width * val
        self.height = self.height * val

    def __str__(self):
        return 'Rectangle({}, {}) at point {}'.format(self.width, self.height

r = Rectangle(20, 40, Point(2,2))
print(r.get_centre().dist_to_origin())
```