Classes: Going Deeper with Encapsulation

Recap: Sequences & Classes

Classes (continued)

- Example: distance based collision check
- Example: Using an existing ball class in pSpy: make multiple bouncing balls
  - Store the balls
  - check for collisions (using a list)
- Challenge: given a ball class in Maya, make multiples
- Extra Challenge: alter the code to make gravity controlled by keyframes
- A closer look at basic class structure: make a rectangle class together in IDLE

```python
from turtle import *
from random import uniform

class Rectangle:
    def __init__(self, x, y, w, h):
        self.x = x
        self.y = y
        self.w = w
        self.h = h
    def draw(self):
        goto(self.x, self.y)
        down()
        forward(self.w)
        right(90)
        forward(self.h)
        right(90)
        forward(self.w)
        right(90)
        forward(self.h)
        right(90)
        up()
    def scale(self, w, h):
        self.w = w
        self.h = h
    def relativeScale(self, dw, dh):
        self.w = self.w * dw
        self.h = self.h * dh
    def move(self, x, y):
        self.x = x
        self.y = y
    def relativeMove(self, dx, dy):
        self.x += dx
        self.y += dy
```

- Review how to make a list of classes

```python
rects = []
for i in range(8):
```
rects.append( Rectangle( 0, i*50, uniform(0,50), uniform(0,50) ) )

for r in rects:
    r.draw()
    r.relativeMove(-120,0)
    r.relativeScale(2,1)
    r.draw()

**Next time:** AI and crowd sim!