**Intro to Python**

**Exercise : Dodgeball**

**Challenge**

Modify the game to have multiple balls.

**Suggestion**

This feature will require changes to `DodgeballData` that create multiple balls, move them all, and check each for a collision with the player. It will also require changes to `DodgeballDisplay` to display all of the balls.

**DodgeballData Changes**

In `DodgeballData` we need to create multiple balls. This would be in the `newGame` method. The best way to do this is to create a list of balls, instead of having a separate variable for each ball. We replaced the creation of `self.ball` with an empty list creation:

```python
self.balls = []
```

followed by a `for` loop that creates the desired number of balls, and `append` them to the list.

```python
for i in range(5):
    ball = DodgeballBall(0, self.height/2, 13, 13, i+1, 1)
    self.balls.append(ball)
```

Note that we chose to give each ball a different speed `i+1` so they would not move together as a group.

Now, we need to make `evolve` process each ball in the list in the same way that the individual ball was processed before. We do this by adding a `for` loop around that code. The loop looks like this:

```python
for ball in self.balls:
```

Then, all of the code that processes the ball, needs to be indented to fit inside. Be sure to include the collision detection code.

Next, all of the references to `self.ball` inside that code needs to be replaced with `ball`, so that each of the balls will be processed from the `for` loop.

Finally, the `getBall` method needs to be changed to `getBalls` and to return `self.balls`.

**DodgeballDisplay Changes**

These changes to `DodgeballDisplay` will cause all of the balls to be displayed.

Find the lines of code in the `draw` method that cause the ball to be displayed. Put a `for` loop around them (that means add the `for` loop) and indent the lines of code to be inside of it. The loop should look like this:

```python
for ball in data.getBalls():
```

Now, every place that the code says `data.getBall()` needs to be replaced with `ball`, so that we use the ball from the loop.

Congratulations, you should now have a game with many balls.