Smalltalk Programming
Lesson 24

In the last lesson you added the ability to keep score when an enemy is destroyed. The shooter game is not yet complete though – there is only 1 enemy and your game is supposed to have 3 enemies. Today, you will make changes to have 3 enemies like the game design planned for.

1. Where do you think the code change for today will be? Look through your class methods if you need to.

2. You may have remembered or found that `ShooterGame>>initializeEnemies` is used to create enemies for the game. This will be where your code changes will need to be made. Make the following changes and then save them.

initializeEnemies

```
| startingPoints |
startingPoints := OrderedCollection with: 75 @ 75 with: 150 @ 150 with: 225 @ 225.
startingPoints
    do: [:aPoint |
        | enemy |
        enemy := Enemy new.
        enemy position: self position + aPoint.
        self addMorph: enemy]
```

3. Here, you are using an *OrderedCollection,* which is one of the different kinds of collections in Smalltalk. A collection is like a list or group of things. It can hold multiple items (such as numbers, objects, or anything else) that you can access or use in your code. For example, you could have a list of names or a collection of location points for enemies in a game.

4. The code creates a collection of three points where enemies will appear. It then creates an enemy for each point, places each enemy at one of those points, and adds them to `ShooterGame`.

5. Notice in the *do:* block that there is ":aPoint |" and "| enemy |". What is *:apoint*? It functions like a temporary variable, but it is different from *enemy*. You might think you could declare both *aPoint* and *enemy* as temporary variables in the same way, like "| aPoint enemy |", but that is not the case. The *parameter :aPoint* is special because it is receiving a value from the *OrderedCollection*. In the *do:* block, *aPoint* is used to temporarily hold each point from the collection as the loop runs. This is why *aPoint* must be handled separately. This is also why it begins with a ":" and has its own "|". On the other hand, "| enemy |" declares a temporary variable that is used to hold a newly created enemy object.

6. Test your changes. Does it work?

7. What else might be nice to have for your shooter game?

8. Save and Quit your Smalltalk image.