Intro

In this step we're going to create classes to add the model layer. The model layer is where we're going to manage our data.

Creating Classes

The first class we're going to create is an activity class. Create a new file called *Activity* with a type "Cocoa Touch Class". Choose "NSObject" from the "Subclass of" dropdown.

Now let's add a couple of properties. They will allow me to assign the type of activity and the time that was involved with this activity. To declare properties, I'm going to use our variable syntax, because properties are basically variables.

```
var activityName:String
var totalTime:Int
```

Now create an initializer so that when we create this activity class you can also give it the activity name and the total time:

```
init(activityName:String, totalTime:Int){
    self.activityName = activityName
    self.totalTime = totalTime
}
```

When these values come into this initializer method, I'm going to assign them to our class properties. `self` is used to differentiate between the parameters that I'm passing in and the class properties. `self` is referencing itself rather than those parameters.

Now I want to create an activity manager. Add another class and call it "ActivityManager". Create it the same as we did our `Activity` class. `ActivityManager` will be used to access our activities.

```
class ActivityManager: NSObject {
    static var activities = [Activity]()
}
```

This static variable allows me to access activities throughout the application. That way I don't have to create instances of the activity manager, because it is creating instances of the Activity for me right here.

ViewDidLoad Function

Now I want to go into our `PerformanceViewController`. Let's create our array and assign it to the activities on the `ActivityManager`.

`viewDidLoad` is a function that gets fired one time when the scene loads. That's really what I want, because we don't want to keep reassigning the array over and over. Once this loads, it's going to assign to our array.

```
override func viewDidLoad() {
    super.viewDidLoad()
    var act1 = Activity(activityName: "Running", totalTime: 10)
    var act2 = Activity(activityName: "Swimming", totalTime: 20)
    var act3 = Activity(activityName: "Jogging", totalTime: 15)
    ActivityManager.activities.append(act1)
    ActivityManager.activities.append(act2)
    ActivityManager.activities.append(act3)
}
```

Now we've got now two model classes.