Code Outline

Home/Opening Page

* Collection view used to display each circuit (all of which are stored in an array)

Circuit Page (selected from Home Page collection view)

* ‘Back’ button takes user back to Home Page for a different circuit
* Each circuit is an instance of the Circuit class
  + Each Circuit has:
    - Name
    - Color (like GS Green. If no color, hard code one)
    - Dictionary of sequentially numbered keys, responding to the name, duration/reps, and visual (if applicable) for each exercise
* 3 Buttons:
  + Back: Either restart the timer on the exercise or go back to the previous exercise
  + Play/Pause: Either start or pause the timer.
    - For non-timed exercises, omit this button (or have it do nothing if lazy)
  + Forward: Next exercise. If completed, show ‘completed’ message/picture

Making different pages:

Simply go to your storyboard file, drag and drop as many view controllers as you want. Once done, create two buttons in your “main menu” and ctrl + drag them to each individually to the view controller you desire (i.e. with button 1, ctrl drag onto “fun fact” view controller and with button 2 ctrl drag onto “other” view controller)

Collection view:

(To create a subclass, use cocoatouch)

* Overall need 2 protocalls:
  + Delegate
    - Helps pick up interactions with cells
    - 1 function required: Use collectionView DidSelectItemAt to produce a function that is called when the user taps on a cell
  + Data Source
    - 2 functions required:
    - collectionView NumberOfItems returns the number of cells in a given section
    - collectionView CellForItemAt returns a cell for the given item; can change the type of return expected from UICollectionViewCell to custom
* To Create a cell:
  + Create a CocoaTouchClass as a subclass of UICollectionViewCell. This Class is a custom implementation of a UICollectionViewCell
    - !!!: Create the XIB file in order to create IBOutlets for the collection view cells
  + In the custom implementation of UICollectionViewCell, a public function that returns a UINib will tell the collection view to use the cell we created
    - “register” the cell through the nib or the class name (nib in this case)
  + In the XIB file, set the Reuse Identifier to the custom UICollectionViewCell name
  + Outlet: Right-click the MyCollectionViewCell in the view outline and add outlet to the cell
  + In ViewDidLoad: collectionView.register a nib for a cellwithreuseidentifier and MyCollectionViewCell and name MyCollectionViewCell as the parameters

Here is what is wrong –

* The one cell that should be displayed is either not displaying or the color/text changes are not taking effect
* Printing “test” in the viewDidLoad method will not print

1/23 – Try saving all of the code somewhere else, then deleting enough to start the ViewController class all over