**Topic: More UI Elements**

Intro

* Review due dates on Moodle and talk about progress

Overview

UI Elements:

* Image view
* Slider
* Text fields – one set for text, one for numeric values only
* Segmented control: toggles between two values
* Switch
* Button – an iOS button that looks like buttons did before iOS 7.
* Action Sheets and Alerts

UI Elements

* Images
  + ImageView
    - Image, from Resources
    - View, Mode options
    - Using images on the Storyboard
  + Using images as app icons
* TextFields
  + Limit input by selecting a particular type of pop-up keyboard, like numeric for integers.
  + Touching the background to close the keyboard (pg. 110)
    - The view property points to an instance of UIView that acts as a container for all the items in our user interface, also called a *container view*.
      * A UIView can have a *gesture recognizer* added to it, we want a particular gesture recognizer, the *tap gesture recognizer*
        + Do this in Xamarin Studio by going to the Story Board and selecting *Tap Gesture Recognizer* in the Tool Box. Drag it onto the main View. It will show up in the Story Board as a dot in the bar along the bottom of the view.
      * Or, alternatively, change the UIView class to UIControl  
        Using Interface Builder, we can change the class of the object that the View points to so that its underlying class is UIControl instead of UIView. Because UIControl is a subclass of UIView, it is perfectly appropriate for us to connect our view property to an instance of UIControl. Remember that when a class subclasses another object, it is just a more specialized version of that class, so a UIControl is a UIView. If we simply change the instance that is created from UIView to UIControl, we gain the ability to trigger action methods. Before we do that, though, we need to create an action method that will be called when the background is tapped. We need to add one more action to our controller class. Add the following method to your ViewController.swift file, right after textFieldDoneEditing():  
          
        @IBAction func backgroundTap(sender: UIControl) {  
         nameField.resignFirstResponder()  
         numberField.resignFirstResponder()  
        }