**Procedure to Develop the First Application**

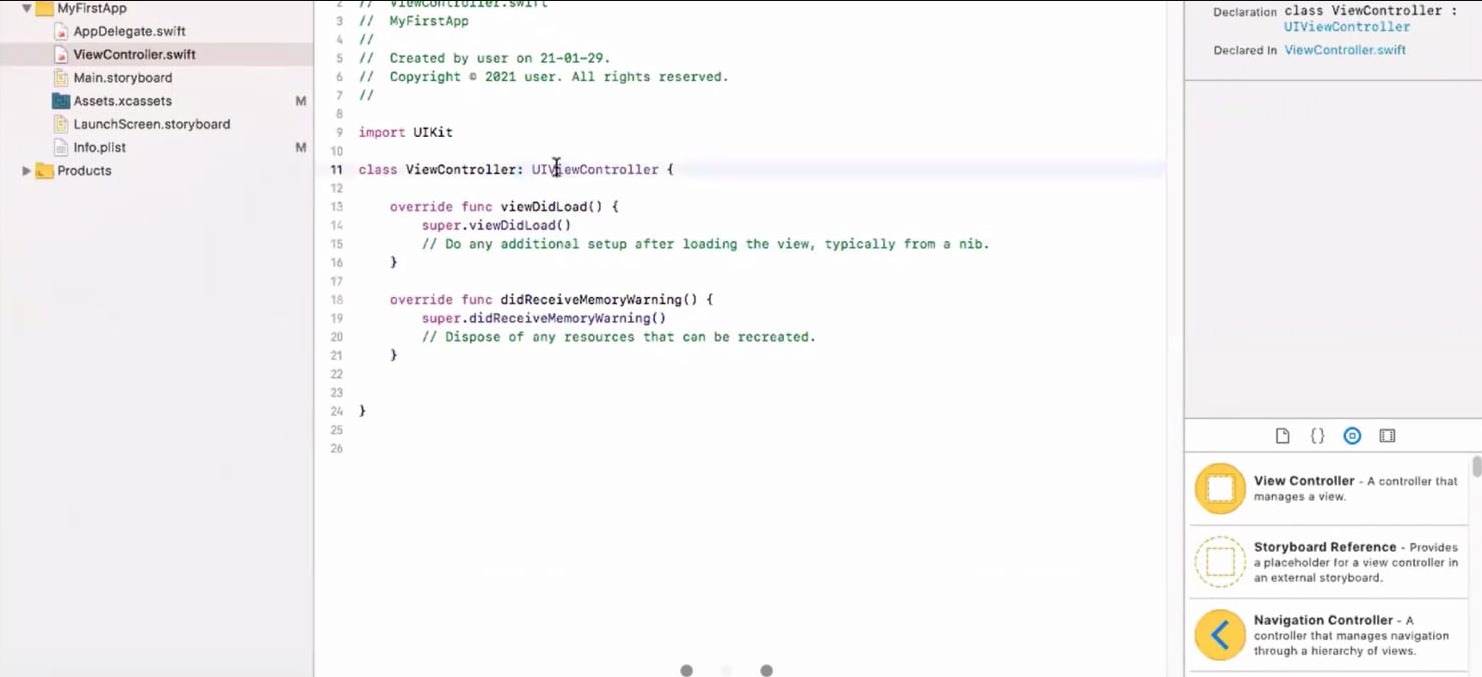
Requirements:

* + Mac Book or a Virtual Machine for MacBook
  + MAC OS
  + XCODE (Latest Version)
  + If you have older version of MAC OS then make sure to download the XCODE for older versions by navigating to developer.apple.com

Installation: Just Click on Get App Icon in the App Store and it should install the app automatically in your MAC.

Developing the first Application:

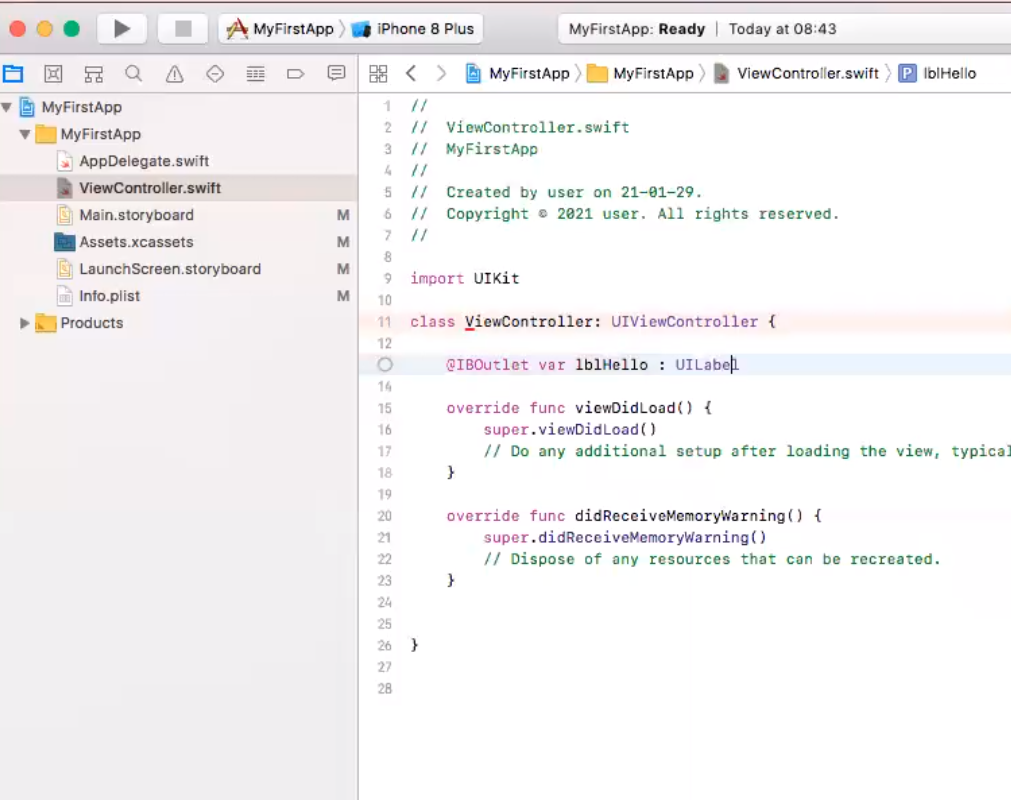
* + Open the Xcode Application from launchpad.
  + Select “Create new Xcode Project” option.
  + From the template we need to select the ‘Single View’ option and click on Next.
  + Please ensure to enter the product name while initiating the Program.
  + Now we have an option to choose the language. It can be either swift or objective C.
  + Now we have an option to connect our project to GitHub.
  + In Identity option we need to enter the Display Name for our Project.
  + Once it is done, we have to click on Launch Screen Story Board option in the Left pane.
  + Now we will be able to see the First screen of our project. It is also called as splash Screen.
  + On the Left pane itself you will find the options to find, debug, errors, Hierarchy, breakpoints, logs and Run Commands.
  + There are 2 things called Launch Screen and Main Story Board.  
    Launch Screen is like Splash screen in Android.  
    Main Story Board is like Activity in Android.
  + We will also have assets file which is equivalent to Res file in android.
  + The Other file we have in the hierarchy is AppDelegate.swift.  
    It is like Android Lifecycle in android.
  + On the Right pane we will have the options to add label, buttons, scrollbar etc. in the toolkit.
  + Like android we can simply drag and drop the desired options into our story board or launch screen.
  + Once we drag and drop any element into our launch screen or story board, we will get the properties of the element on the Right pane to adjust the element as per the requirements.



**Code Explanation:**

**@IBOUTLET:** it is a connection from an Interface Builder to user interface component. So, it acts as a connector between these two. It makes the code trigger the UI Component in the application.

Example – when you are declaring a variable for the User interface then you need add @IBOutlet. (@IBOutlet var x: UILabel)



**@IBACTION:** It is used for making storyboard layouts trigger the code. It makes the Storyboard layout trigger the Code.

Example - @IbAtion func sayHello(sender:Any){

lblHello.text="hello Gaspesie!"

