Intro to iOS Tutorial
ATLAS Mobile Makers

Open Xcode
Create a new Project
Choose iOS Application → Single View Application
Product name: helloworld
Organization name: Your name (Used in copyright)
Organization identifier: Your name (Xcode uses this to create a unique bundle identifier)
Language: Swift
Device: iPhone
Leave Use Core Data Unchecked
Choose a folder to hold all of your projects
Leave create local git repository unchecked → Create
Now you are ready to explore Xcode and create your first app!

Xcode and Xcode Map
Xcode can be overwhelming when you first open it… there is a lot going on
Check out the Xcode map to see the main areas you will be working with
https://github.com/ATLASMobileMakers/AppResources/blob/master/xcodeMap.pdf

Adding a Label
Click on Main.storyboard on the left-hand side of Xcode (Project Navigator)
Find the one view and click on it somewhere in the middle
Now, over on the lower right-hand side of Xcode, you will find a square icon with a circle (the Object Library)
Search for a label and drag it from the Object Library onto the view you found on the Main.storyboard
Double click the label and type something into it like Hello World! or Jessie is awesome!
You can now build and run your app using the simulator or your own phone by clicking the green arrow in the top left corner of Xcode.
You will see it's not centered on the screen. This is because the storyboard isn't exactly what you would see on a real device.
To fix this, go back to Main.storyboard → click on the File Inspector → deselect Use Size Classes
Chose Keep size class data for: iPhone and click Disable Size Classes
Now reposition the label to the center of the view.
The attributes section on the right-hand side of Xcode allows you to change font size, font color, background color, and other properties of the label.

Let's Create an Action
Click on Main.storyboard from the Project Navigator if you are not already there
Go over to the Object Library (lower right-hand corner) and drag and drop a button from the Object Library onto the view
Double click the button to change the text to say something like "Click me!"
You have a button, but it doesn't do anything yet. We have to connect the interface and the code.
Go to Main.Storyboard and click on the view
You need to open the double view so you can see the storyboard and the code, so click on the 'Show 2 views' icon (top right corner, middle button). This should show the ViewController.swift file
Now click on the button and hold down the control key
Then click and drag from the button to the swift file

You should notice a blue line following your cursor. This line connects the button to the code so we can add an action to it
Move your cursor between the curly braces for the class
When you see a grey box appear release the mouse button
This window lets you set up the connection between the button and your code
Connection: Action
Name: clickMe
Type: UIButton
Event: Touch Up Inside (the standard event to use for buttons)
Arguments: Sender Now
Connect
You should now see in the swift file
@IBAction func clickMe(sender: UIButton) { }
This is a method called clickMe that will be called when the user taps the button
Now let's connect the label
Control-click from the label to the swift file as we did with the button
Connection: Outlet
Name: myText
Type:UILabel
Leave storage as weak
Connect
Notice this created in the swift file
@IBOutlet weak var myText: UILabel!

Now let's write some code
Find the @IBAction function you created earlier

```
@IBAction func clickMe(sender: UIButton) {
     myText.text="Jessie is pretty cool!"
}
```

This changes the string from whatever you had before to the new text, "Jessie is pretty cool!"
You can now run the app!

Resources
Xcode map https://github.com/ATLASMobileMakers/AppResources/blob/master/xcodeMap.pdf
Looking for more tutorials? Check us out at https://github.com/ATLASMobileMakers