

Hello UI!

เราจะทำ App แบบง่ายๆ กัน ใน app ประกอบด้วย ปุ่มและข้อความ เมื่อกดปุ่ม ข้อความจะเปลี่ยนเป็น "Morning!"

จุดประสงค์

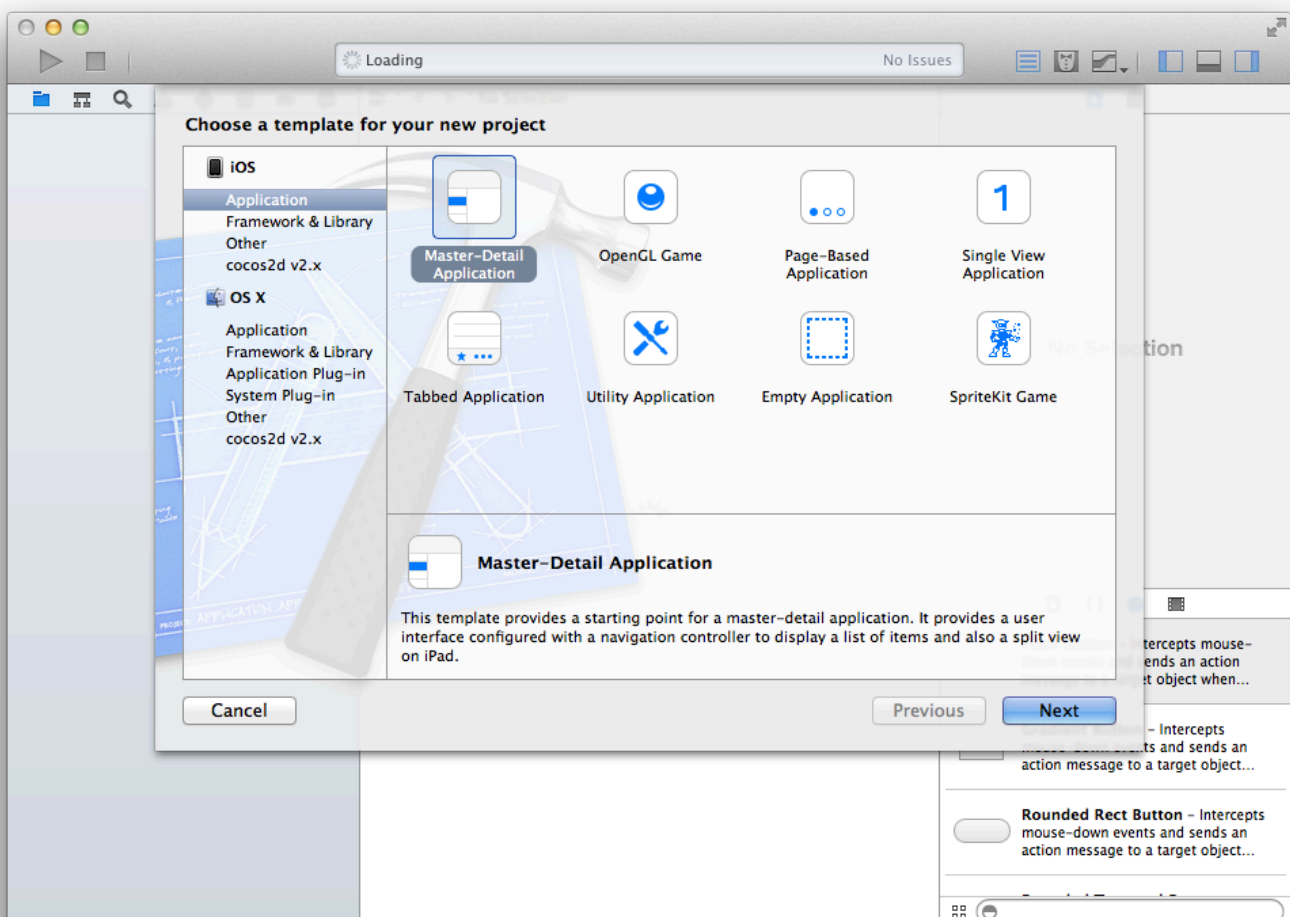
ทำความเข้าใจกับระบบ Project ของ Xcode, การปรับ Property ต่างๆใน Interface Builder, และการเชื่อมความสัมพันธ์ระหว่าง Code กับ UI พื้นฐาน

สร้าง Project ใน Xcode

เครื่องมือที่เราจะใช้กันคือ Xcode ซึ่งเป็น IDE ของ Apple เอง. นอกจากเขียน iOS app แล้ว ยังใช้สำหรับเขียน Native app บน Mac ได้อีกด้วย

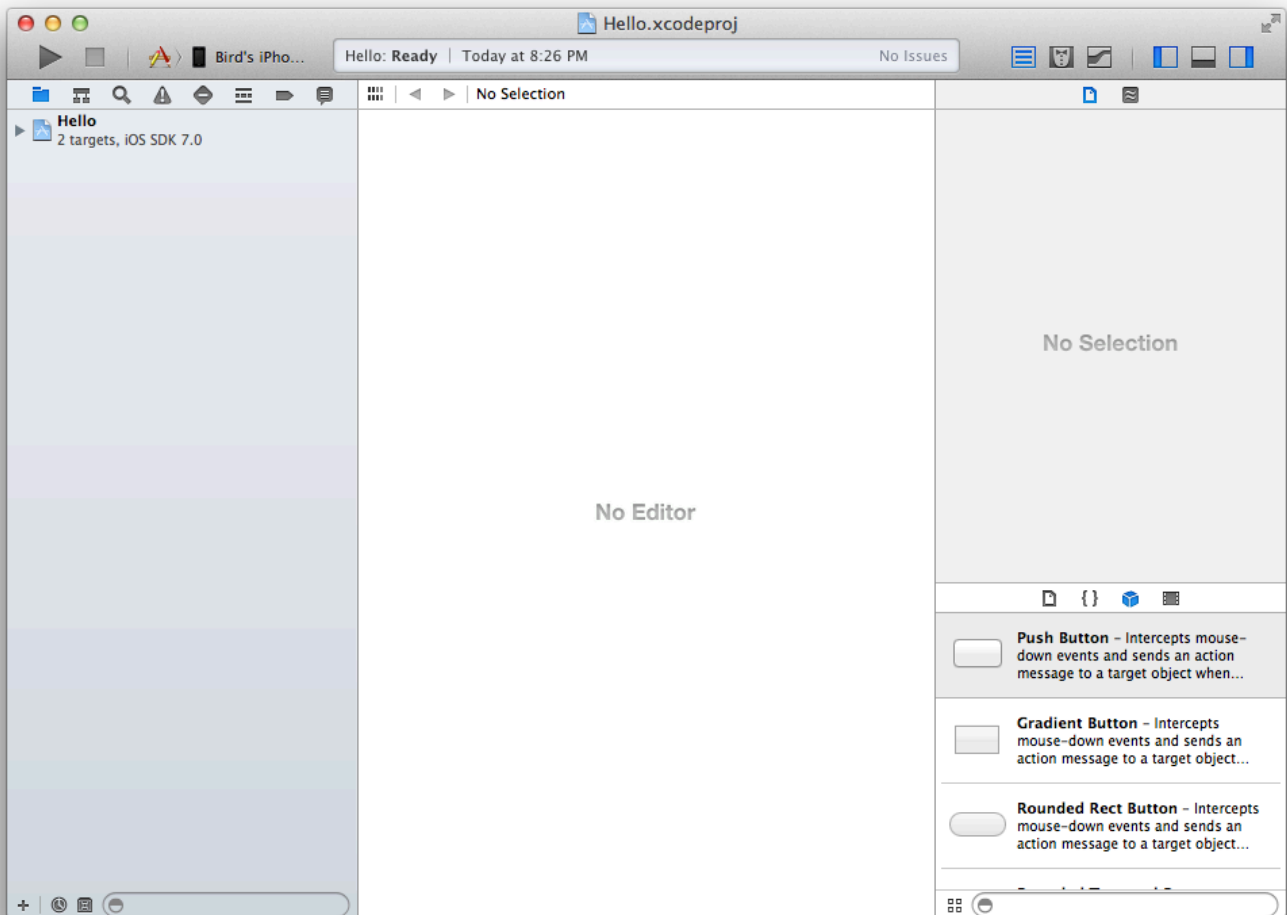
สร้าง Project ใหม่...

1. เปิด Xcode ขึ้นมา
2. สร้าง Project ใหม่ โดยเลือก File > New Project. จะได้หน้าจอแบบนี้



3. ด้านซ้ายเลือก iOS Application
4. เลือก Single View Application แล้วกด Next

5. ใส่ Hello เป็น Product name, Company Identifier ใช้เป็น Reverse domain name เช่น th.ac.ku
6. Device Family เลือกเป็น iPhone
7. กด Next, แล้วเลือกที่ Save project
8. กด Save แล้วจะได้หน้าจอแบบนี้:



ลอง Run ใน Simulator...

1. เลือก Scheme ที่ด้านซ้ายบนเป็น iPhone Retina (4-inch) ใน iOS Simulator
2. กดปุ่ม Play ที่เมนูด้านซ้ายบน

iOS Simulator จะขึ้นมา เราควรจะได้เห็นจอเปล่าๆ แบบนี้ เราจะลองเข้าไปเปลี่ยนสี Background ดู ตอนนี้ปิด Simulator ไปก่อน

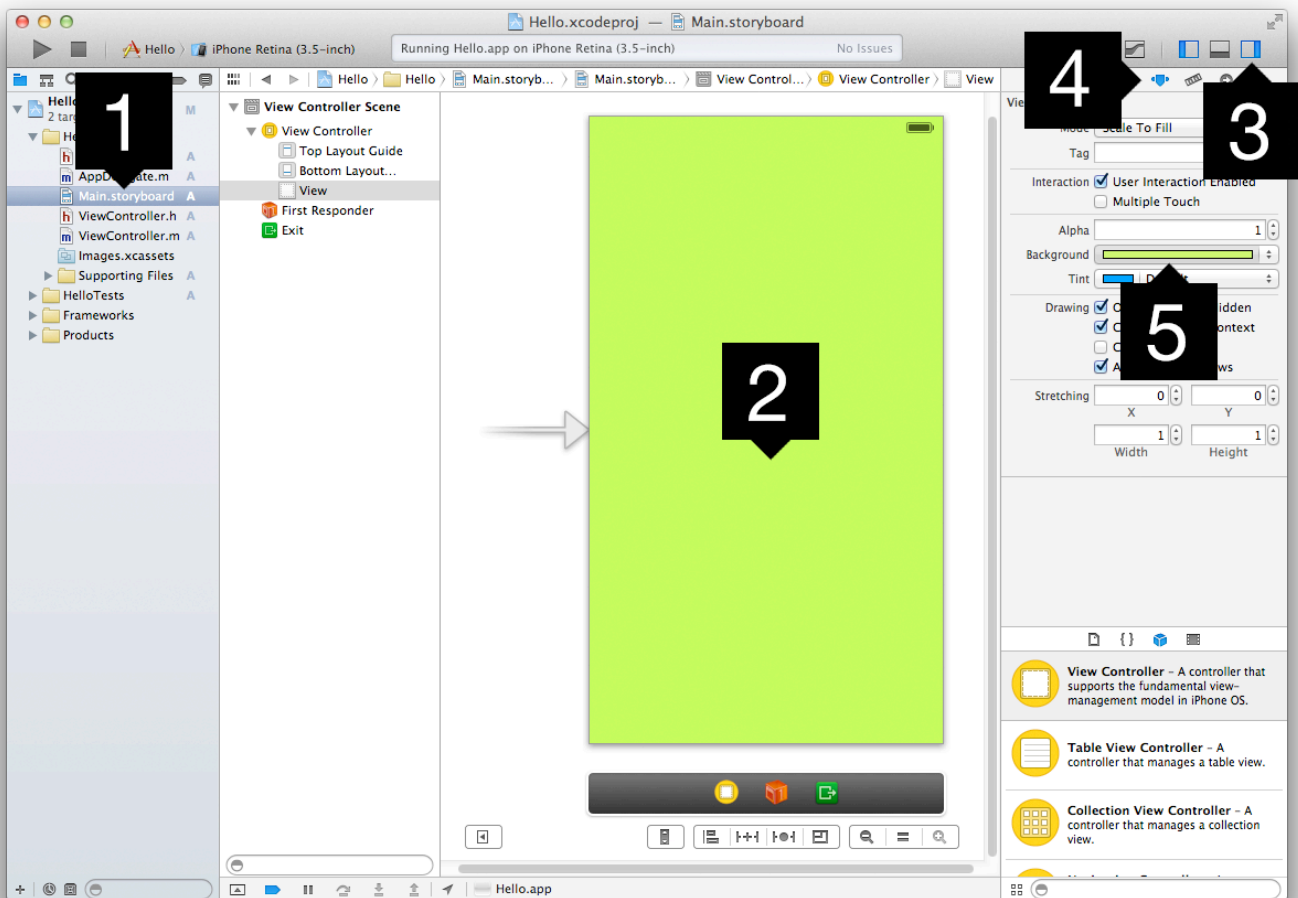
ปิด Simulator...

1. กด Stop ใน Xcode



เปลี่ยนสี Background

1. เข้าไปใน Interface Builder ด้วยการ เลือก Main.storyboard จากเมนูด้านซ้าย



2. เลือก view ตรงกลางให้มีขอบสีน้ำเงินขึ้นมา
3. เปิดเมนูด้านขวา
4. เรียก Attribute Inspector ทางด้านซ้ายมือของไม้บรรทัด
5. เลือก Background > Other แล้วปรับสีที่ต้องการ
6. รัน Simulator ขึ้นมา

Checkpoint

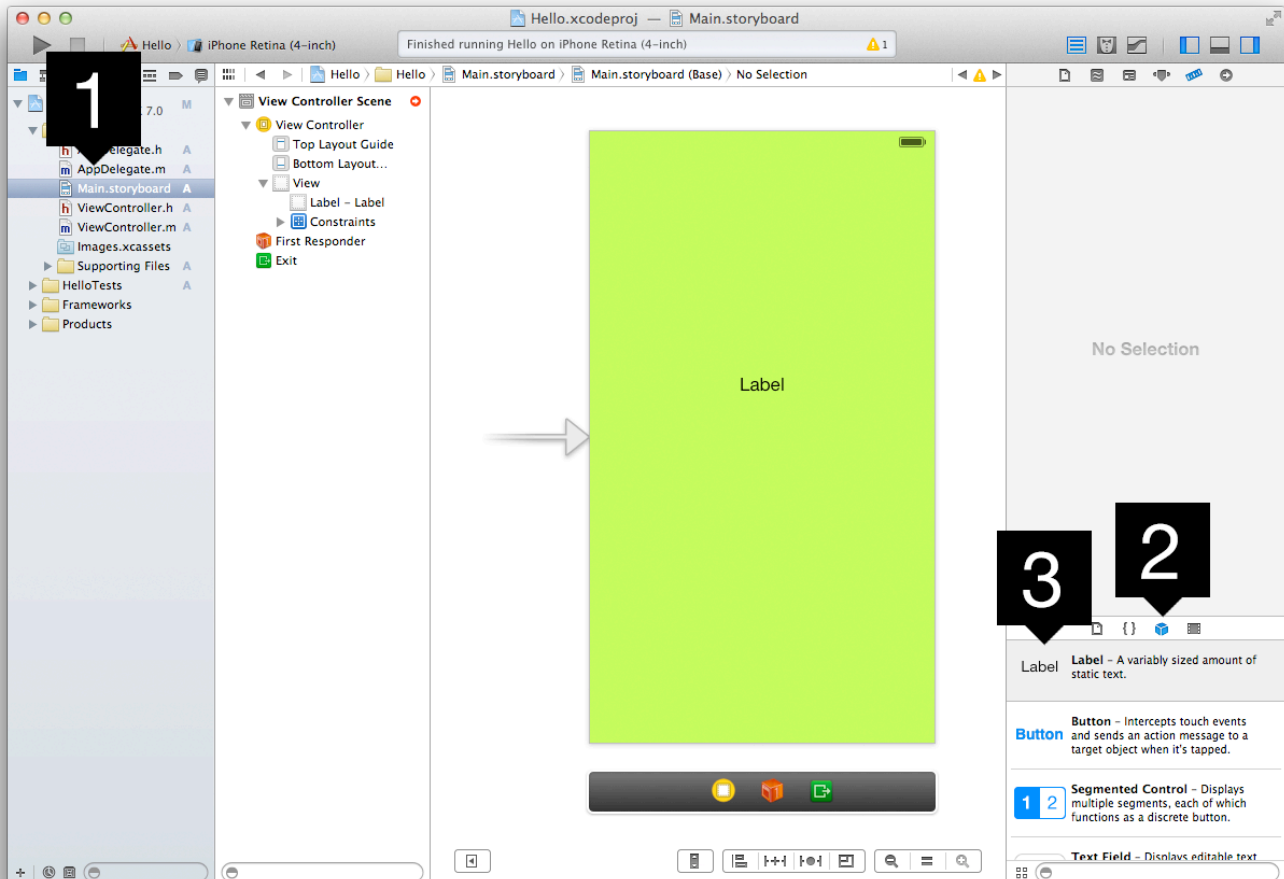
ถึงจุดนี้ App ที่ run ขึ้นมาควรเป็นสีตามที่เราเปลี่ยนไว้ หากยังไม่ใช่ ลองตรวจสอบดูว่าได้แก้ไขที่ ไฟล์ Main.storyboard หรือไม่

เพิ่ม User Interface

เราเพิ่ม ปุ่ม, กล่องข้อความ ด้วยการ drag จาก Object library ลงไปใน view

เพิ่ม Control ลงใน nib file..

1. ใน Xcode, เลือก Main.storyboard



2. เปิดแถบเมนูด้านขวา เลือก Object Library (รูปลูกบาศก์) ขึ้นมา

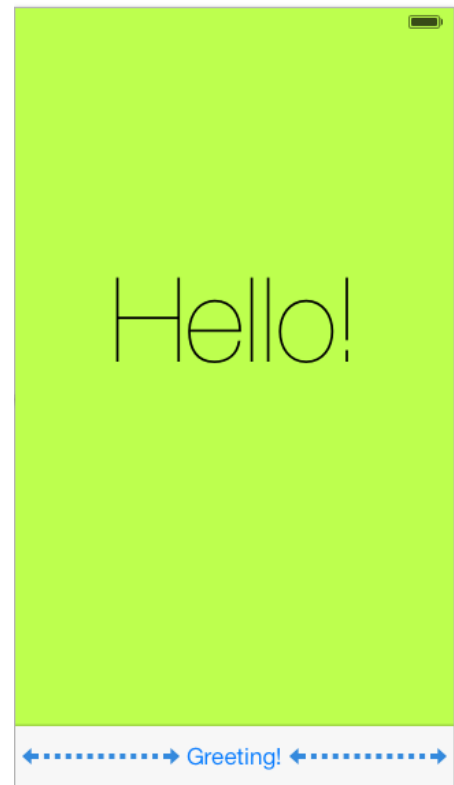
3. ลาก Label (UILabel), Toolbar (UIToolbar) เข้าไปใน view

4. ลาก Flexible Space Bar Button Item ใส่เข้าไปใน toolbar ทั้งสองข้างของปุ่ม เพื่อให้ปุ่มอยู่ตรงกลาง
5. จัด Layout ให้เรียบร้อย
 - ปรับสี Background
 - ปรับขนาด Label ให้กว้างขึ้น แก่ Text จาก Label เป็น Hello!
 - ปรับ Font ของ Label ให้ใหญ่ขึ้น, ใช้ inspector ปรับ Alignment เป็น Center
 - เปลี่ยน Title ของ Toolbar Button ให้เป็น Greeting !

เสร็จแล้ว ควรจะได้หน้าต่างตามด้านขวามือ แบบนี้ :

Checkpoint

รันด้วย Simulator เราควรจะได้เห็น control เรียงไว้สวยงาม ตามที่จัด กดปุ่ม Greeting! ตอนนี้จะมี highlight ลึกครุ่นึงเท่านั้น



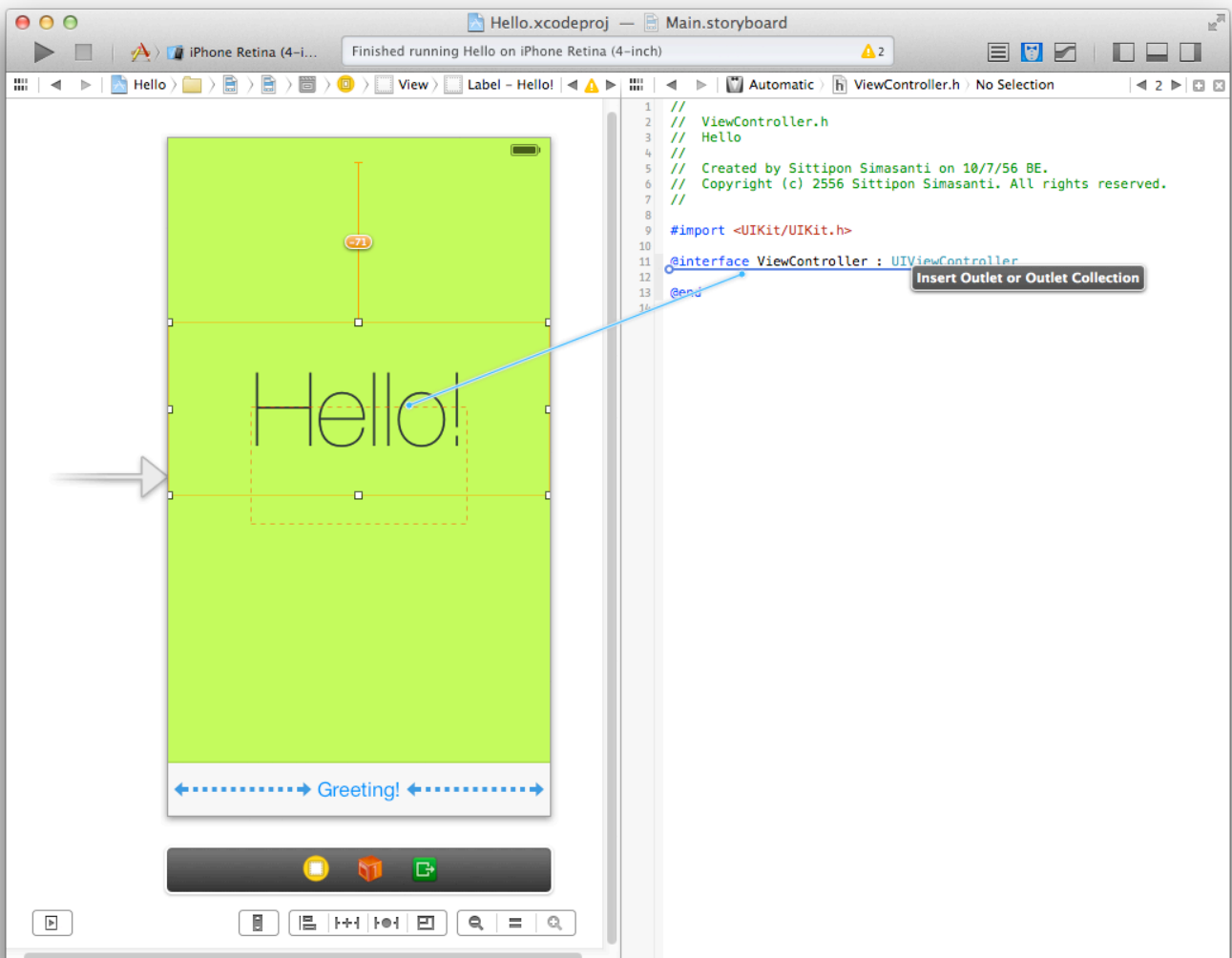
Outlets and Actions

เรากำลังจะทำให้ ViewController รู้จัก label ที่อยู่ใน Storyboard และสร้าง action `buttonDidClick:` เพื่อลองรับการกดปุ่ม Greeting!

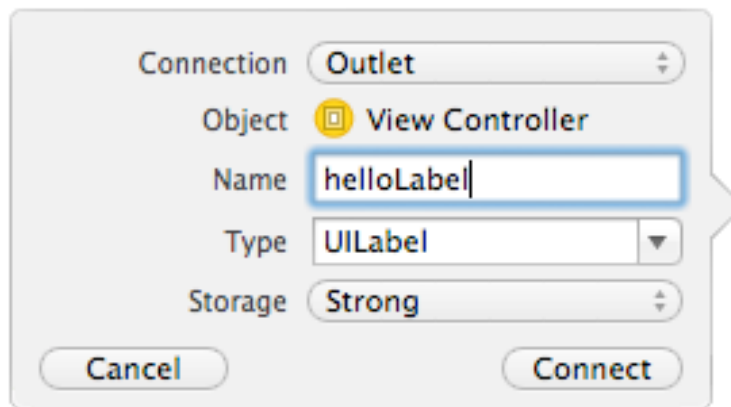
ใน Xcode 5 เราสามารถ drag control เข้าไปใน code เพื่อสร้าง Outlet และ Action ได้โดยตรง

ทำให้ HelloWorldViewController รู้จัก Label..

1. ใน Xcode เลือก Storyboard (Main.storyboard)
2. เปิด Assistant editor (รูปสากทักชีโด)
3. ให้ Assistant แสดง header file ขึ้นมา (ViewController.h)
4. Mouse ขวา drag label เข้าไปใน header ไฟล์ ก่อน `@end`



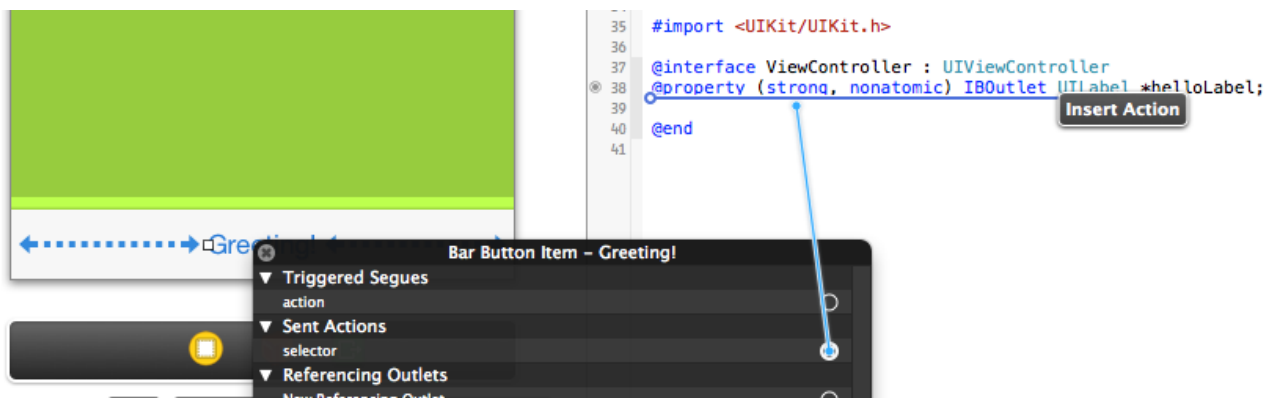
5. ตั้งค่าของ connection เป็นดังนี้



6. กด Connect

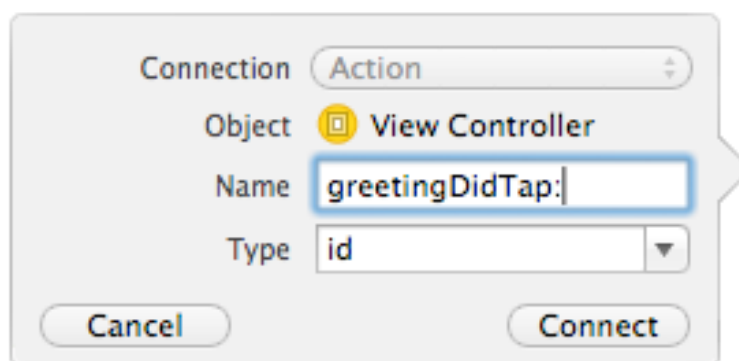
เพิ่ม Action ให้ปุ่ม...

1. เลือกปุ่ม Greeting! จากนั้นกด Mouse ขวา
2. Outlet ของ Bar Button Item จะแสดงขึ้นมา Click ที่วงกลมด้านขวา ของ selector แล้วลากเข้าไป ก่อน @end



3. ตั้งค่า connection เป็นตามนี้

4. กด Connect



Checkpoint

ตอนนี้ Xcode ทำการสร้าง connection ระหว่าง Storyboard และ code ภายใน ViewController.h และ ViewController.m ให้เรียบร้อยแล้ว

ถ้าเข้าไปดู ViewController.m เราจะเห็น method เปล่าๆ เตรียมไว้ให้เราเขียน

```
- (IBAction)greetingDidTap: (id)sender {  
}
```

greetingDidTap:

ตอนปุ่ม Greeting ! ถูกกด ViewController จะได้รับ message greetingDidTap: ตามที่เราตั้งไว้จาก nib file เราจะเปลี่ยนให้ข้อความ Hello! เปลี่ยนเป็น Morning! แทน

เขียน greetingDidTap: ...

1. เปิด ViewController.m ขึ้นมา
2. เติม คำสั่งต่อไปนี้ใน greetingDidTap:

```
- (IBAction)greetingDidTap: (id)sender {  
    [self.helloLabel setText: @"Morning!"];  
}
```

Checkpoint

เสร็จแล้ว ลอง run ใน Simulator เมื่อกดปุ่มข้อความ “Hello!” ควรจะเปลี่ยนเป็น “Morning!”

ณ จุดนี้เราได้เขียน Application แรกของเราเสร็จเรียบร้อยแล้ว เราได้ทำความคุ้นเคยกับการใช้งาน Xcode, การผูก User Interface เข้ากับ code พื้นฐาน ระบบ UI ต่างๆของ UIKit เป็นที่เรียบร้อยแล้ว

ยินดีด้วยครับ !

แต่นี้เป็นเพียงจุดเริ่มต้นเท่านั้น ..

Mission 5

สร้างโปรแกรมเครื่องคิดเลขใน Project ใหม่ชื่อ Calc โปรแกรมจะต้องประกอบด้วย: -

1. ปุ่ม (UIButton) เลข 0-9
2. กล่องข้อความแสดงผลลัพธ์ (UILabel)
3. ปุ่ม บวก, ลบ, คูณ,หาร และ ปุ่ม เท่ากับ
4. ปุ่ม Clear สำหรับลบผลลัพธ์ทั้งหมด
5. ปุ่ม +/- สำหรับกลับเครื่องหมาย

Optional:

6. ปุ่มจุด สำหรับเลขทศนิยม

Code Listings

สามารถ Download source code ของ กิจกรรมนี้ได้ที่นี่

1. เปิด Terminal ขึ้นมา
2. พิมพ์คำสั่งต่อไปนี้ เพื่อทำการ git clone project ลงมาในเครื่อง

```
git clone git://github.com/programmerbird/ios-helloui.git
```

สรุปเมื่อแบบฝึกนี้เสร็จแล้ว จะได้ code ตามนี้

ViewController

Header file: ViewController.h

```
#import <UIKit/UIKit.h>

@interface ViewController : UIViewController
@property (strong, nonatomic) IBOutlet UILabel *helloLabel;
- (IBAction)greetingDidTap:(id)sender;

@end
```

Implementation File: ViewController.m

```
#import "ViewController.h"

@interface ViewController ()
@end

@implementation ViewController

- (void)viewDidLoad {
    [super viewDidLoad];
}

- (void)didReceiveMemoryWarning {
    [super didReceiveMemoryWarning];
}

- (IBAction)greetingDidTap:(id)sender {
    [self.helloLabel setText: @"Greeting!"];
}

@end
```