

Day 01



Trịnh Minh Cường



Lập trình – Objective C

Đây là nội dung chính của khoá học này

Cocoa Touch

- Storyboards
- Documents
- Gesturing
- Multitasking
- Notifications
- UIKit Framework

Media Layer

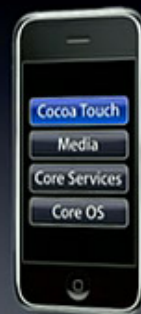
- Graphic Technologies
- Audio Technologies
- Video Technologies
- AirPlay

Core Services Layer

- iCloud
- In-App purchases
- SQLite
- Core Data
- Core Location

Core OS Layer

- Bluetooth
- External Accessories
- Accelerator Framework



Cocoa Touch

Multi-Touch Events
Multi-Touch Controls
Accelerometer
View Hierarchy
Localization

Alerts
Web View
People Picker
Image Picker
Camera

Những chủ đề cần học #1

Objective-C

- Control sequence
- Class – Object Oriented
- iVar, local var, property
- Category – Protocol – Blocks
- Memory Management
- C – C++ in Objective-C
- Collection Programming

GUI #1

- Native Control
- Target Action – Delegate vs Block
- UIView – Animation
- Gesture Recognizer
- UIViewController – Navigation Controller – TabBar Controller – SplitViewController
- Landscape vs Portrait Mode

GUI #2

- UIScrollView, UITextView, UIWebView – Hybrid App
- UIAlertView – UIActionSheet. Use Block to replace Delegate
- UITableView – Customize table view cell, search
- UICollectionView

GUI #3

- StoryBoard vs XIB vs Code
- Auto Layout vs Manual Layout
- Develop Universal app
- Using custom control
- Localization

Database

- CoreData
- SQLite
- CoreData wrapper
- Migration Data when model change
- Term Project

Những chủ đề cần học #2

Networking

- AFNetworking and extend AFNetworking (XML, JSON, SOAP, REST)
- Upload – Download
- Core Foundation Networking - Bonjour

Concurrency

- Grand Central Dispatch
- NSOperation
- Background Tasking (play audio, download)

CoreGraphics

- Basic line – shape
- Bezier Path
- Hand Drawing
- Make custom control
- Customize UI

AVFoundation

- Basic Audio player - recoder
- Basic Video player
- AVPlayer
- Subtitle
- Mix Audio

Những chủ đề cần học #3

Map – Location

- MKMapKit
- CLCoreLocation
- Foursquare service

Camera

- Core Image
- GPU Image
- Create new image filter

CoreMotion

- Detect shaking
- Accelerometer
- Push vs Polling
- Game using CoreMotion

Test & Publish App

- Device, Certificate, Provisioning
- Dev Provision vs Distribution Provision
- Test Flight
- Crash Log

Misc

- File Operation
- Google Drive SDK
- DropBox SDK
- FaceBook SDK
- iCloud

Những chủ đề nâng cao

Core
Animation

Core Audio

OpenGL ES 2.0

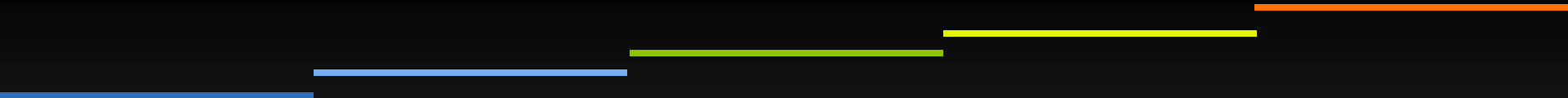
iCloud

StoreKit

Game Center

Hoặc chúng ta sẽ học về lập trình phần server cho ứng dụng iOS:

- NodeJS dễ học
- Phalcon PHP dễ học
- Hoặc PlayFramework khó nhưng chịu tải rất tốt



**Học nhiều không hay bằng biết
cách tìm kiếm, hệ thống**

Cứ tham gia dự án thật khắc biết !



Công cụ lập trình



Utilities



Developer



Dashcode



Instruments



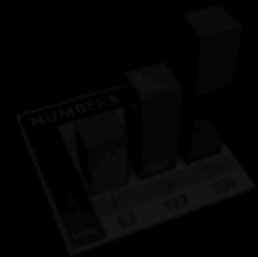
Quartz Composer



Xcode



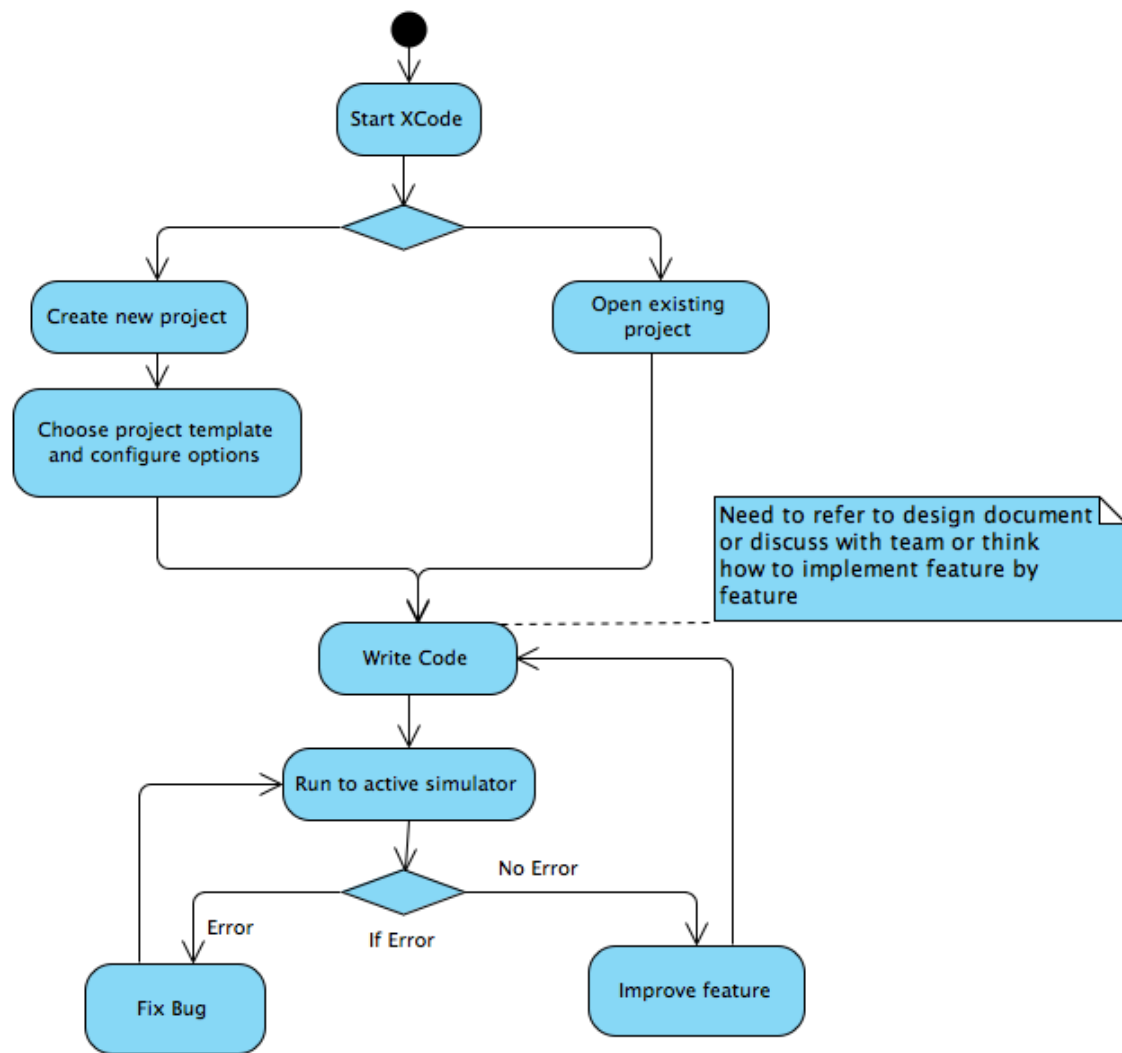
Keynote

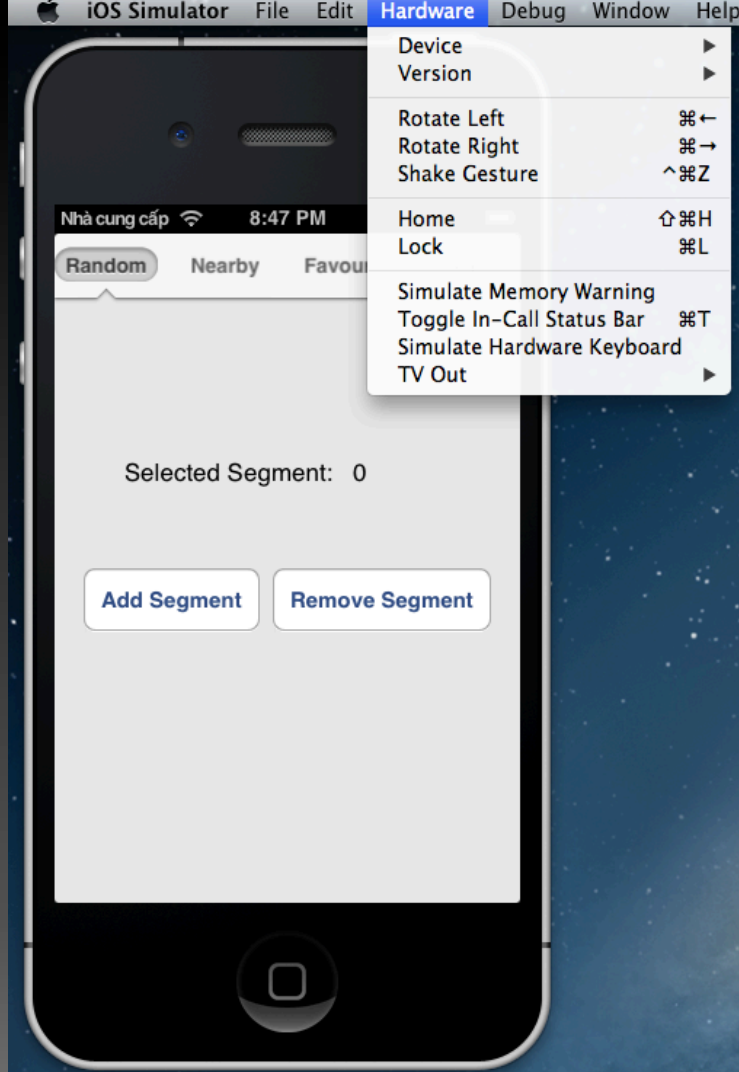


Numbers



Chúng ta học luôn trên XCode 5 và iOS 7

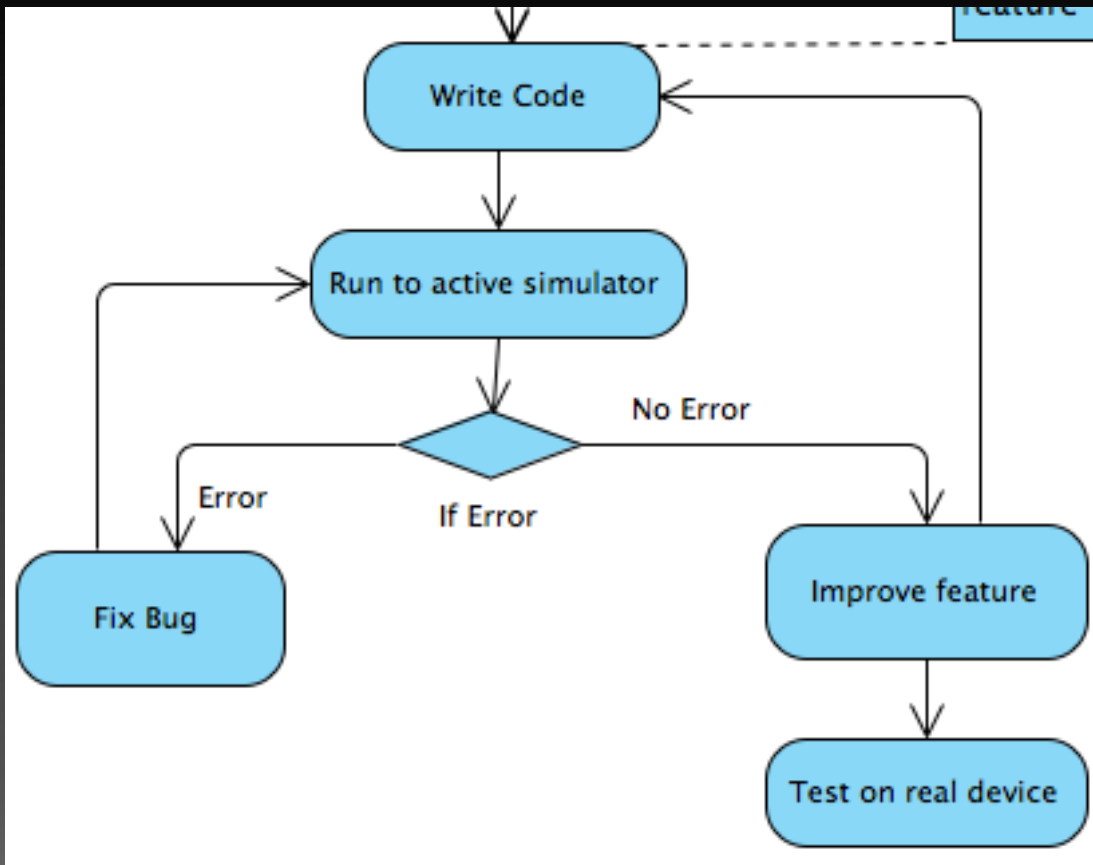




Lập trình di động luôn cần có simulator (giả lập)

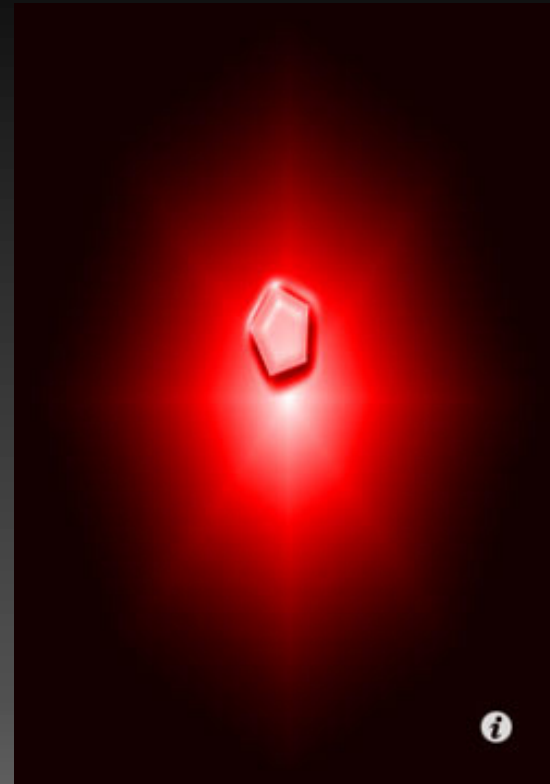
- Thiết bị thật có rất nhiều chủng loại, lập trình viên không thể sở hữu tất cả model thiết bị để thử nghiệm
- Ứng dụng di động chạy kiểu sandbox rất phù hợp giả lập
- Gỡ rối, kiểm tra trạng thái biến dễ dàng
- Nhiều tính năng của thiết bị thật, simulator không giả lập được 😞

Kiểm thử trên máy thật rất cần



- Đội dev iOS chuyên nghiệp cần có khoảng 4-8 thiết bị khác nhau
- Đối với Android, số lượng thiết bị test còn lớn hơn nhiều

Ứng dụng I Am Rich, bán \$999,99



Phân tích thiết kế

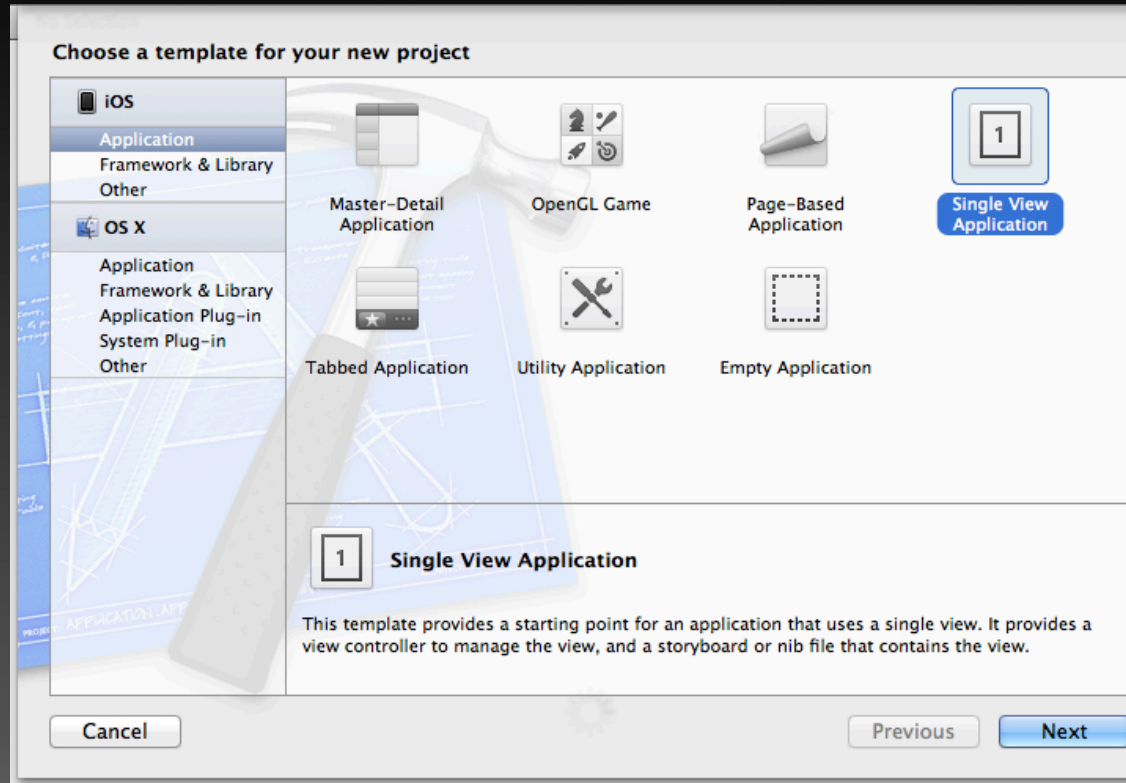
- Đây là ứng dụng rất đơn giản, chỉ có duy nhất một màn hình.
- Người dùng bật ứng dụng sẽ thấy một viên kim cương.
- Chạm ngón tay vào viên kim cương, hiển thị dòng chữ “I am Rich”
- Nâng cao: kim cương mờ rồi đậm rồi lại mờ theo chu kỳ





Tạo ứng dụng I am Rich

- Bật Xcode
- File > Project. Chọn SingleViewApplication





Cấu hình cho dự án mới

Product Name: HelloWorld

Organization Name: TechMaster

Company Identifier: vn.techmaster

Bundle Identifier: vn.techmaster.HelloWorld

Class Prefix: XYZ

Devices: iPhone

☒ Use Storyboards

☒ Use Automatic Reference Counting

☐ Include Unit Tests

Product Name: Tên ứng dụng

Organization Name: Tên công ty

Company Identifier: Viết ngược đảo tên miền của web site công ty

Class Prefix: để phân biệt, gom các class tác giả viết với mã nguồn khác. Có thể để trống

Device: iPhone, iPad, hoặc Universal cho cả 2 thiết bị

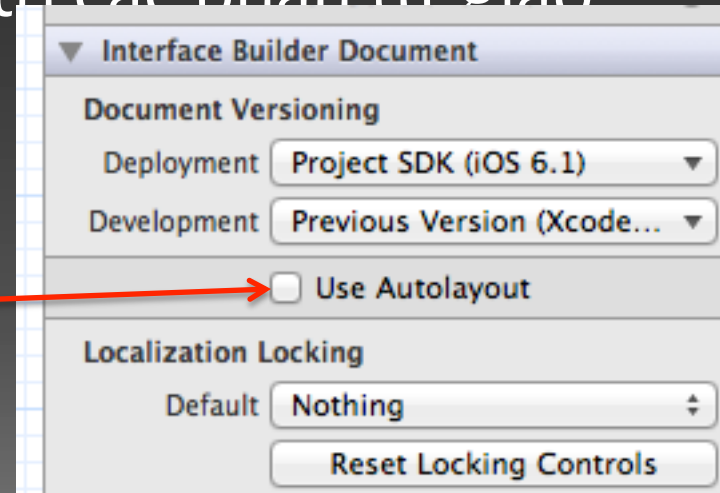
Use Storyboards: thiết kế giao diện mới có trong iOS 5

Use Automatic Reference Counting: bật chế độ quản lý bộ nhớ ARC, mới có từ iOS5



Mấy điểm chú ý

- Trong Mac hầu hết các tác vụ chỉ cần single click
- Auto layout chỉ có từ phiên bản iOS 6, nếu bật ở cho ứng dụng chạy ở iOS 5 sẽ bị crash
- Đối với ứng dụng cần lập trình vị trí các phần tử giao diện nên tăng Auto layout





Màn hình thường vs Retina

3.5 inch vs 4.0 inch

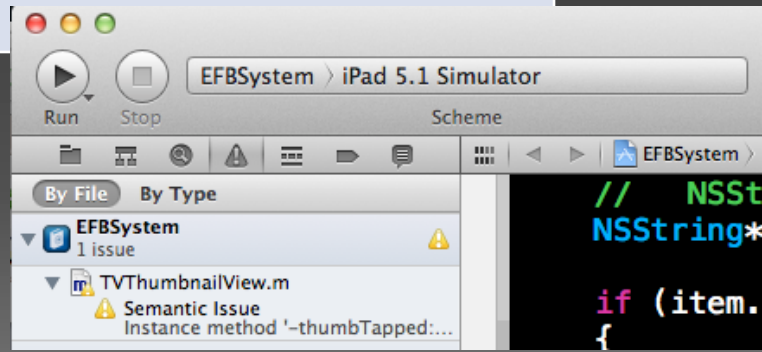
Thiết kế với
4.0 inch

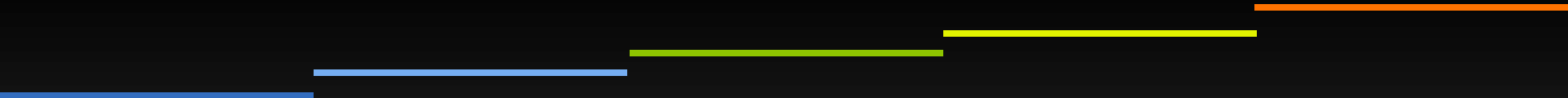


Thực tế trên
3.5 inch

Phiên bản nào của iOS chạy thử được trên phiên bản nào của Mac

	Mac OSX 10.7.x	Mac OSX 10.8.x
iOS 4.x	Xcode 4.4.1. OK	Cannot
iOS 5.x	Xcode 4.4.1. OK	XCode 4.4.1 & XCode 4.5
iOS 6.x	Cannot	Xcode 4.5

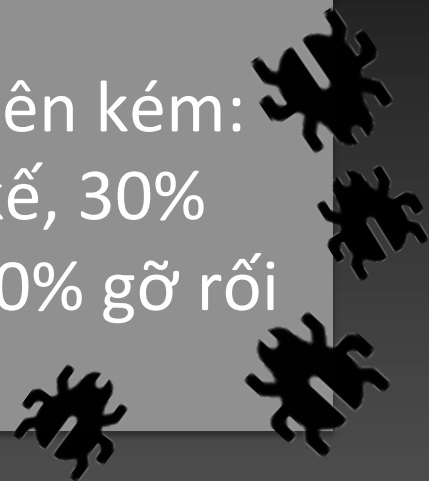




Lập trình viên giỏi:
30% thiết kế, 60%
lập trình, 20 % gỡ rối



Lập trình viên kém:
10% thiết kế, 30%
lập trình, 60% gỡ rối



Debug – Gỡ rối ứng dụng

Debug – Gỡ rối

Ứng dụng có lỗi, chạy không như mong muốn, đang chạy thì bị crash. Làm thế nào?

- Đặt breakpoint (điểm ngừng) để kiểm tra các biến – trạng thái
- In ra màn hình console
- Ứng dụng multi-thread cần biết dòng lệnh hiện đang ở thread nào

Xem
các
thread
đang
chạy

Break point

Xem các biến
cục bộ

Màn hình
console, xem
log

The screenshot shows the Xcode IDE with the following components:

- Top Bar:** Run, Stop, Scheme, Breakpoints, and a status bar indicating "Running HelloWorld on iPhone 6.0 Simulator".
- Left Sidebar:** A project navigator showing a folder named "HellWorld" containing a "Thread 1" and several other threads (Thread 2, Thread 3, Thread 4, Thread 5, Thread 6, Thread 7, Thread 8).
- Editor:** A Swift file named "ViewController.m" with the following code:

```
@interface ViewController ()  
  
@end  
  
@implementation ViewController  
  
- (void)viewDidLoad  
{  
    [super viewDidLoad];  
    // Do any additional setup after loading the view, typically from a nib.  
}  
  
- (void)didReceiveMemoryWarning  
{  
    [super didReceiveMemoryWarning];  
    // Dispose of any resources that can be recreated.  
}  
  
- (IBAction)helloWorldClick:(id)sender {  
    NSLog(@"Hello World");  
    UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"Thong bao"  
                                                                message:@"Hello  
                                                                World"  
                                                                delegate:self  
                                                                cancelButtonTitle:@"Cancel"  
                                                                otherButtonTitles: nil];  
    [alertView show];  
}  
  
@end
```
- Right Sidebar:** An "Identity and Type" panel showing the file name "ViewController.m", file type "Default - Objective-C...", location "Relative to Group", and full path "/Users/techmaster/Desktop/HellWorld/HellWorld/ViewController.m". Below it is an "Objects" panel listing various UIKit classes like "View Controller", "Table View Controller", "Collection View Controller", "Navigation Controller", "Tab Bar Controller", "Page View Controller", "GLKit View Controller", and "Object".
- Bottom Panel:** A breakpoint bar showing a breakpoint set at "Thread 1" at "0 - [ViewController helloWorldClick:]". Below it is a "Variables" panel showing the current state of variables:

```
self = (ViewController *) 0x07199620  
sender = (UIRoundedRectButton *) 0x07466aa0
```
- Console:** A console window at the bottom right showing the output of the NSLog statement:

```
2012-10-24 20:59:20.025  
HellWorld[1653:c07] Hello World  
(lldb)
```


Running HelloWorld on iPhone 6.0 Simulator

ViewController.m

Thread 1
com.apple.main-thread
0 -[ViewController helloWorldClick:]
1 -[NSObject performSelector:wit...
20 UIApplicationMain
21 main
Thread 3
com.apple.libdispatch-manager
Thread 5 WebThread
Thread 6
Thread 7
Thread 8

```
@interface ViewController ()  
  
@end  
  
@implementation ViewController  
  
- (void)viewDidLoad  
{  
    [super viewDidLoad];  
    // Do any additional setup after loading the view, typically from a nib.  
}  
  
- (void)didReceiveMemoryWarning  
{  
    [super didReceiveMemoryWarning];  
    // Dispose of any resources that can be recreated.  
}  
  
- (IBAction)helloWorldClick:(id)sender {  
    NSLog(@"Hello World");  
    UIAlertView *alertView = [[UIAlertView alloc] initWithTitle:@"Thong bao"  
                                                                message:@"Hello  
                                                                World"  
                                                                delegate:self  
                                                                cancelButtonTitle:@"Cancel"  
                                                                otherButtonTitles: nil];  
  
    [alertView show];  
}  
  
@end
```

Thread 1: breakpoint 2.1

self = (ViewController *) 0x07199620
sender = (UIRoundedRectButton *) 0x07466aa0

2012-10-24 20:59:20.025
HelloWorld[1653:c07] Hello World
(lldb)

Identity and Type
File Name: ViewController.m
File Type: Default - Objective-C...
Location: Relative to Group
ViewController.m
Full Path: /Users/techmaster/Desktop/HelloWorld/HelloWorld/ViewController.m

Objects
View Controller - A controller that supports the fundamental view-management model in iPhone OS.
Table View Controller - A controller that manages a table view.
Collection View Controller - A controller that manages a collection view.
Navigation Controller - A controller that manages navigation through a hierarchy of views.
Tab Bar Controller - A controller that manages a set of view controllers that represent tab bar...
Page View Controller - Presents a sequence of view controllers as pages.
GLKit View Controller - A controller that manages a GLKit view.
Object - Provides a template for objects and controllers not directly available in Interface Builder.
Label - A variably sized amount of static text.
Round Rect Button - Intercepts touch events and sends an action message to a target object when...
Segmented Control - Displays multiple segments, each of which

Nút điều khiển
việc chạy debug

Đặt breakpoint

- Chạy tiếp từng dòng lệnh
- Chạy sâu vào dòng lệnh hiện thời
- Xem biến hiện thời
- Gỡ breakpoint

NSLog là hàm in ra console của Xcode, chứ không phải màn hình thiết bị

```
int x = 10, y= 100;  
    NSLog(@"ALL YOUR BASE ARE BELONG TO US");  
    NSLog(@"x = %d, y = %d", x, y);  
    NSLog(@"%s\t%s\t0x%04x", "hex", "value",  
0x1243);
```

%@ Object

%d, %i signed int

%u unsigned int

%f float/double

%x, %X hexadecimal int

%o octal int

%zu size_t

%p pointer

%e float/double (in scientific notation)

%g float/double (as %f or %e, depending on value)

%s C string (bytes)

%S C string (unichar)

%.*s Pascal string (requires two arguments, pass pstr[0] as the first, pstr+1 as the second)

%c character

%C unichar

%lld long long

%llu unsigned long long

%Lf long double