

Introduction to iPhone Development Session Two

Friday October 23rd, 2009

Session Two Presenters

- Eric Busch
Lead Developer at Mint Apps
- Devon DeNure
Lead of Design and User Experience at
Mint Apps
- Aaron Gehri
VP of Computer Science Club

Topics

- Objective-C Code and Syntax
- Objective-C vs. Java
- Interface Builder Connections
 - Outlets and Actions
- Touch Detection
- Computer Graphics
 - OpenGL
 - iPhone 3GS vs. Others
- Instruments Usage

Objective-C Basics

- There are header (.h) and implementation (.m) files
- Header files are used to declare methods and variables
 - Outline
- Implementation files are used to do the actual work and computation

Java vs. Objective-C

- `public class MyClass() {}`
- `public void myFunc() {}`
- `int number = 5;`
- `String str = "hello";`
- `System.out.println("A");`
- `this.number = num;`
- `@implementation MyClass`
- `-(void)myFunc {}`
- `int number = 5;`
- `NSString *str = @"Hello";`
- `NSLog(@"A");`
- `self.number = num;`

Java vs. Objective-C

- `MyClass class = new MyClass();`
- `MyClass *class = [MyClass new];`
- `class.func();`
- `[class func];`
- `returnVar = class.func(passThis);`
- `returnVar = [class func:passThis];`

Objective-C Memory

- Garbage collection is available
 - Auto-release pools manage your memory automatically
- Otherwise, we must release alloc'd objects
 - [object release];

Objective-C Basics

- Otherwise, Objective-C is largely standard
- `for(int i = 0; i < 5; i++) {}`
- `if(a < b) {} else if(a == b) { }`
- `// Comment /* Another comment */`
- Documentation, Practice, Time

Let's Start An App

- Interface
 - User input text field
 - Submit button
 - Text label
- Button changes the label to text field contents

Demo...

To Recap...

- We created a view-based application
- We added the necessary outlets and actions
- `IBOutlet UITextField *myTextField;`
- `-(IBAction)changeGreeting;`

To Recap...

- Created the UI in Interface Builder
- Attached the appropriate actions and outlets to objects
- Wrote the needed code

One Step Further...

- Touch detection...
 - - (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event;
 - - (void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event;
 - - (void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event;
 - - (void)touchesCancelled:(NSSet *)touches withEvent:(UIEvent *)event;

Touch Usage

- Getting the (x,y) of a touch
 - - (void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event
{
 UITouch *touch = [touches anyObject];
}
- Using this we can get x and y locations
 - [touch locationInView:self.view].x
 - [touch locationInView:self.view].y

Demo...

Another recap...

- Added the image to the project
- Created a new outlet
- Added the image to the UI, connected it to the new outlet
- Setup touchesBegan and touchesMoved
- Set the center of the image

OpenGL ES

- Low-level, lightweight Applications Programmers Interface (API)
- Easy and affordable to offer a variety of advanced 3D graphics
- The most widely adopted cross-platform graphics API

OpenGL ES

- Industry Standard and Royalty Free
- Small footprint & low power consumption
- Well-documented

iPhone 3GS

- iPhone 3GS uses OpenGL ES 2.0
- Other iPhones use OpenGL ES 1.1
- iPhone 3GS also has a faster CPU
- Be aware of differences in device speed

Instruments Features

- Optimization and Analysis
 - Disk
 - Memory
 - CPU
 - FPS
- Find the offending lines of code

Demo...

One More Recap...

- Opened Instruments
- Selected the sample tools we wanted
- Selected which app to launch
 - “Launch Executable”
- Record

Q&A

Eric Busch, Devon DeNure, Aaron Gehri