

iOS App Development

1.4 UIView

Views

- UIView subclass
 - UIButton, UILabel, UIImageView, ...
- Rectangular Area
- Drawing & Event Handling
- Hierarchical: superview, subviews
- UIWindow: Top Most UIView

View Coordinates

- CGFloat
- CGPoint: x and y
- CGSize: width and height
- CGRect: origin and size



View Coordinates

(0, 0): Upper Left

- Points Unit (Not Pixels)
- Location and Size Properties
 - `CGRect bounds; // internal drawing space`
 - `CGPoint center; // superview's coordinate`
 - `CGRect frame; // superview's coordinate`

Creating Views

- Storyboard
- Programmatically

```
CGRect frame = CGRectMake( 10, 10, 40, 30 );
UILabel *label = [[UILabel alloc] initWithFrame:frame];
label.text = @"Hi there!";
[self.view addSubview:label];
```

Subclassing UIView

- Custom View
- Custom Drawing
 - Override:
 - - (void) drawRect:(CGRect) rect;
 - Request redraw
 - [customView setNeedsDisplay];
 - **-drawRect: is internally called**

-drawRect:

- Must not be called directly
- Core Graphics Framework
 - C API
 - Graphics Context is **Given** every time called
 - CGContextRef **context** = UIGraphicsGetCurrentContext();
 - CGContextFillRect(**context**, rect);

Core Graphics

• To the Context

```
// Define Path  
CGContextBeginPath( context );  
CGContextMoveToPoint( context, x, y );  
CGContextAddLineToPoint( context, x, y );  
...  
CGContextClosePath( context );  
  
// Fill & Stroke Colors  
[[UIColor greenColor] setFill];  
[[UIColor redColor] setStroke];  
  
// Draw  
CGContextDrawPath( context, kCGPathFillStroke );
```

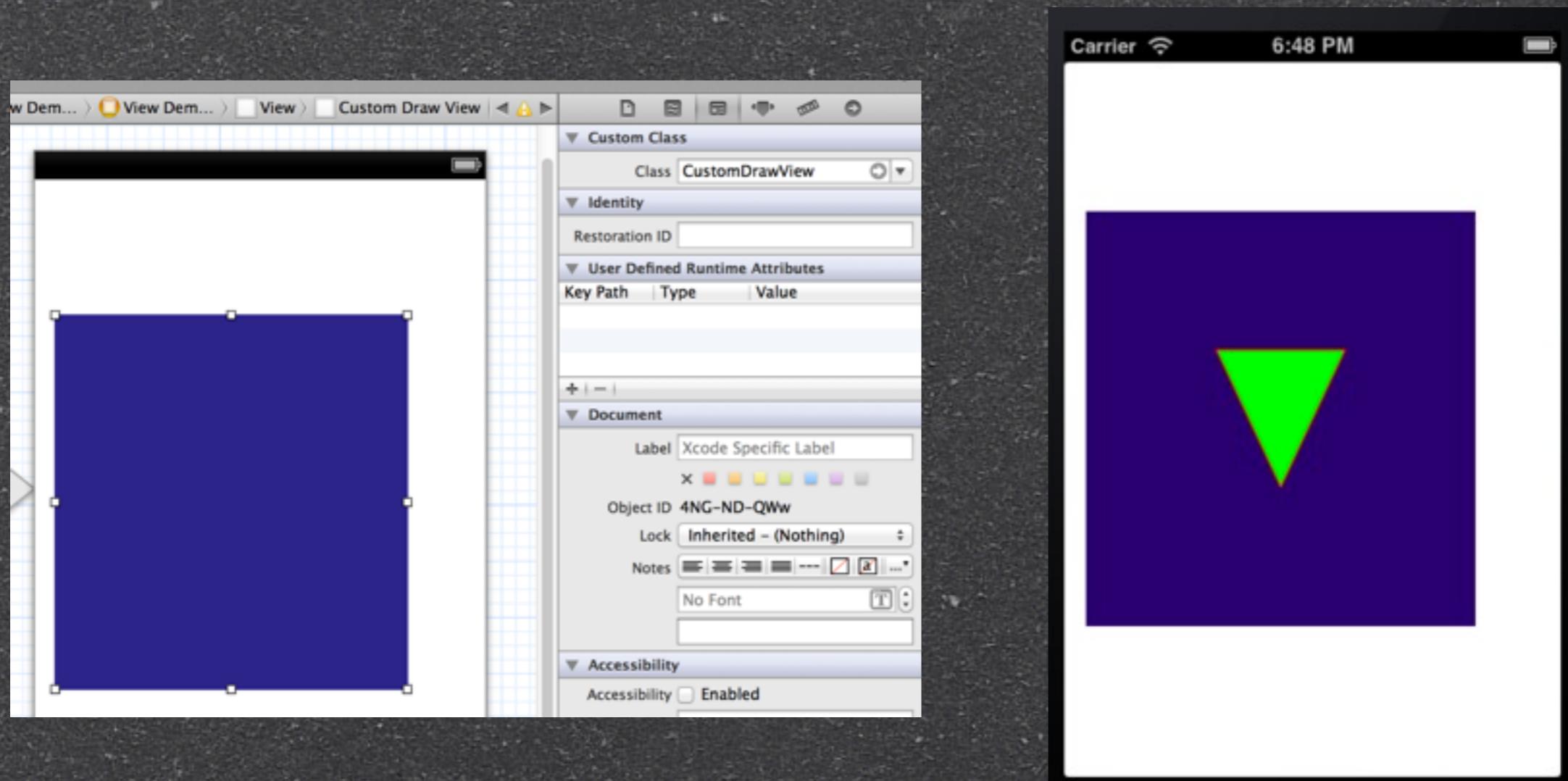
Drawing

- Text
 - `NSString *text = @”Hi”;`
 - `[text drawAtPoint:p withFont:font];`
- UIImage
 - `[image drawInRect:r];`
 - `[image drawAtPoint:p];`

View Visibility

- view.alpha = 0.5; // transparency
- view.hidden = YES; // not visible

Sample Custom View



Sample Custom View

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

@interface CustomDrawView : UIView

@end
-----
// CustomDrawView.m
#import "CustomDrawView.h"

@implementation CustomDrawView

- (id)initWithFrame:(CGRect)frame {
//...
}

- (void)drawRect:(CGRect)rect
{
//... Upside Down Triangle ...
}
@end
```

Sample Custom View

```
CGFloat x1 = rect.origin.x + rect.size.width /3;
CGFloat x2 = rect.origin.x + rect.size.width /3 * 2;
CGFloat x3 = rect.origin.x + rect.size.width /2;
CGFloat y1 = rect.origin.y + rect.size.height / 3;
CGFloat y2 = rect.origin.y + rect.size.height / 3;
CGFloat y3 = rect.origin.y + rect.size.height /3 * 2 ;

- (void)drawRect:(CGRect)rect
{
    // 1. Get the context
    CGContextRef context = UIGraphicsGetCurrentContext();

    // 2. Define a path
    CGContextBeginPath(context);
    ...
    CGContextMoveToPoint(context, x1, y1);
    CGContextAddLineToPoint(context, x2, y2);
    CGContextAddLineToPoint(context, x3, y3 );

    CGContextClosePath(context);

    [[UIColor greenColor] setFill];      // Green as Fill Color
    [[UIColor redColor] setStroke];      // Red as Stroke Color

    CGContextDrawPath(context, kCGPathFillStroke);
}
```