**VIEWS**

**A view (UIView subclass) represents a rectangular area.**

* **For drawing, handling touch events and defines a coordinate space.**
* **Can have one – var superview: UIView? And many – var subviews: [UIView]**

**Adding views programmatically,**

**addSubview(aView: UIView)**

**removeFromSuperView();**

**How to draw inside UIView coordinate system?**

**CGFloat – convert double to CGFloat – let cfg = CGFloat(aDouble)**

**CGPoint – two CGFloats in it: x and y**

**var point = CGPoint(x: 37.0, 55.2)**

**point.x -= 30**

**point.x += 20.0**

**CGSize – two CGFloats in it: width and height**

**var size = CGSize(width: 100.0, height: 50.0)**

**size.width += 42.5**

**size.height += 75**

**CGRect**

**struct CGRect**

**{**

**var origin: CGPoint**

**var size: CGSize**

**}**

**let rec = CGRect(origin: aCGFloat, size: aCGSize);**

**COORDINATE SYSTEM IN XCODE**

1. **Origin is upper left**
2. **Units are points, not pixels – but if you care about pixels, we can get the number of pixels per point – var contentScaleFactor: CGFloat**
3. **FRAME VS CENTER**

**Var center: CGPoint – the superView’s center**

**Var frame: CGRect – rect containing the superViews’s system**

**CREATING VIEWS**

**let labelRect = CGRect(x: 20, y: 20, width: 40, height: 50)**

**let label = UILabel(frame: labelRect)**

**label.text = Hello**

**view.addSubView(label)**

**WHY DO WE WANT TO CREATE CUSTOM VIEW?**

**Because we might want to draw custom drawing and also we need to use gestures.**

**To draw, just create a UIView subclass(CGRect) and override drawRect**

**Override func drawRect(regionThatNeedsToBeDrawn: CGRect)**

**HOW DO I IMPLEMENT drawRect?**

1. **Gives a context – UIGraphicsGetCurrentContext()**
2. **Create paths(out of lines, arcs, etc…)**
3. **Set drawing attributes like colors, fonts, textures, linewidths, etc…**

**UIBezierPath**

**It has methods to draw except color, text, image.**

**DEFINING A PATH(DRAWING A TRAIANLGE)**

**Let path = UIBezierPath()**

**path.moveToPoint(CGPoint(x: 80, y:50))**

**path.addLineToPoint(CGPoint(x: 140, y: 150)**

**path.addLineToPoint(CGPoint(x:10, y: 150))**

**path.closePath()**

**Nothing happens until we set the set attributes and stroke/fill**

**UIColor.greenColor().setFill;**

**TRANSPERANCY**

**Hide a view – var hidden: Bool**

**For example, we want a view to appear when a button is on a certain state.**

**VERY IMPORTANT MIGHT BE ON THE FINAL!!!!!!!!**

**PUT TEXT ON SCREEN WITHOUT ADDING A LABEL AS A SUVIEW**

**To put a text in drawRect, use NSAttributedString**

**let text = NSAttributedString(“hello”);**

**text.drawAtPoint(aCGPoint)**

**let textSize: CGSize = text.size**

**DRAWING IMAGES**

**To put an image in drawRect, use UILabel-equivalent**

1. **Create a UIImage object – let image: UIImage? = UIImage(name: “foo”)**

**Image.drawAtPoint(aCGPoint)**

**image.drawInRect(aCGRect)**

**image.drawAsPatternInRect(aCGRect)**