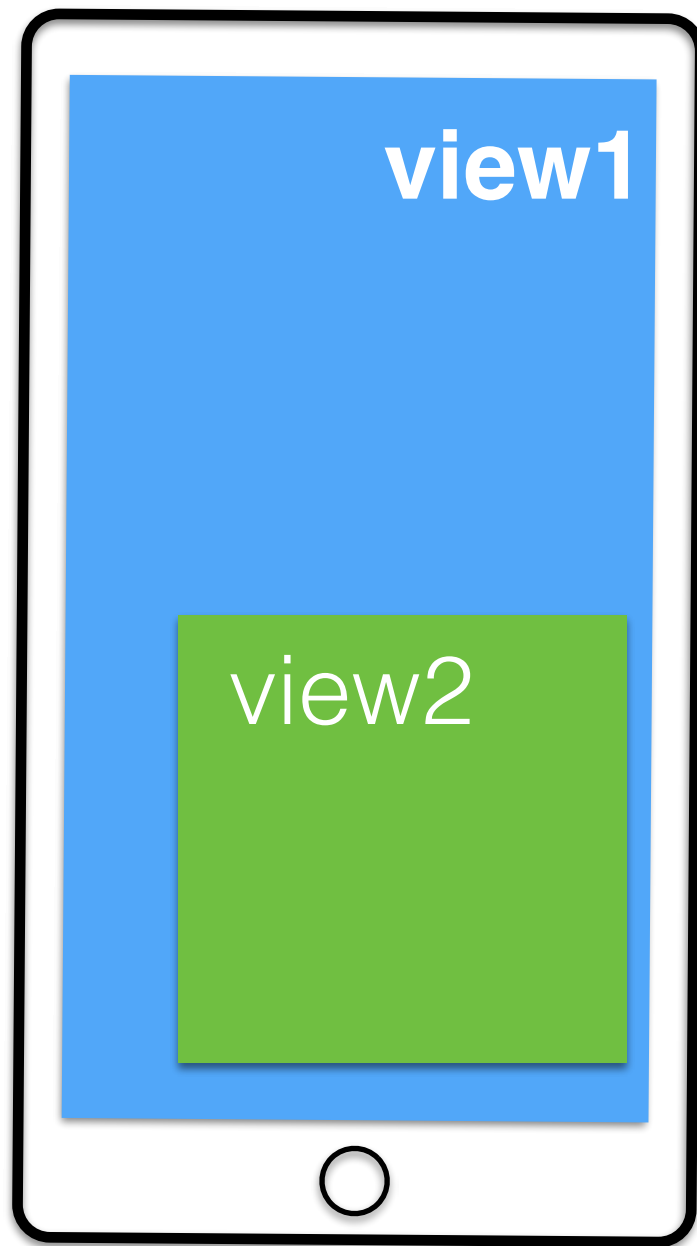


UIView

UIView hierarchy

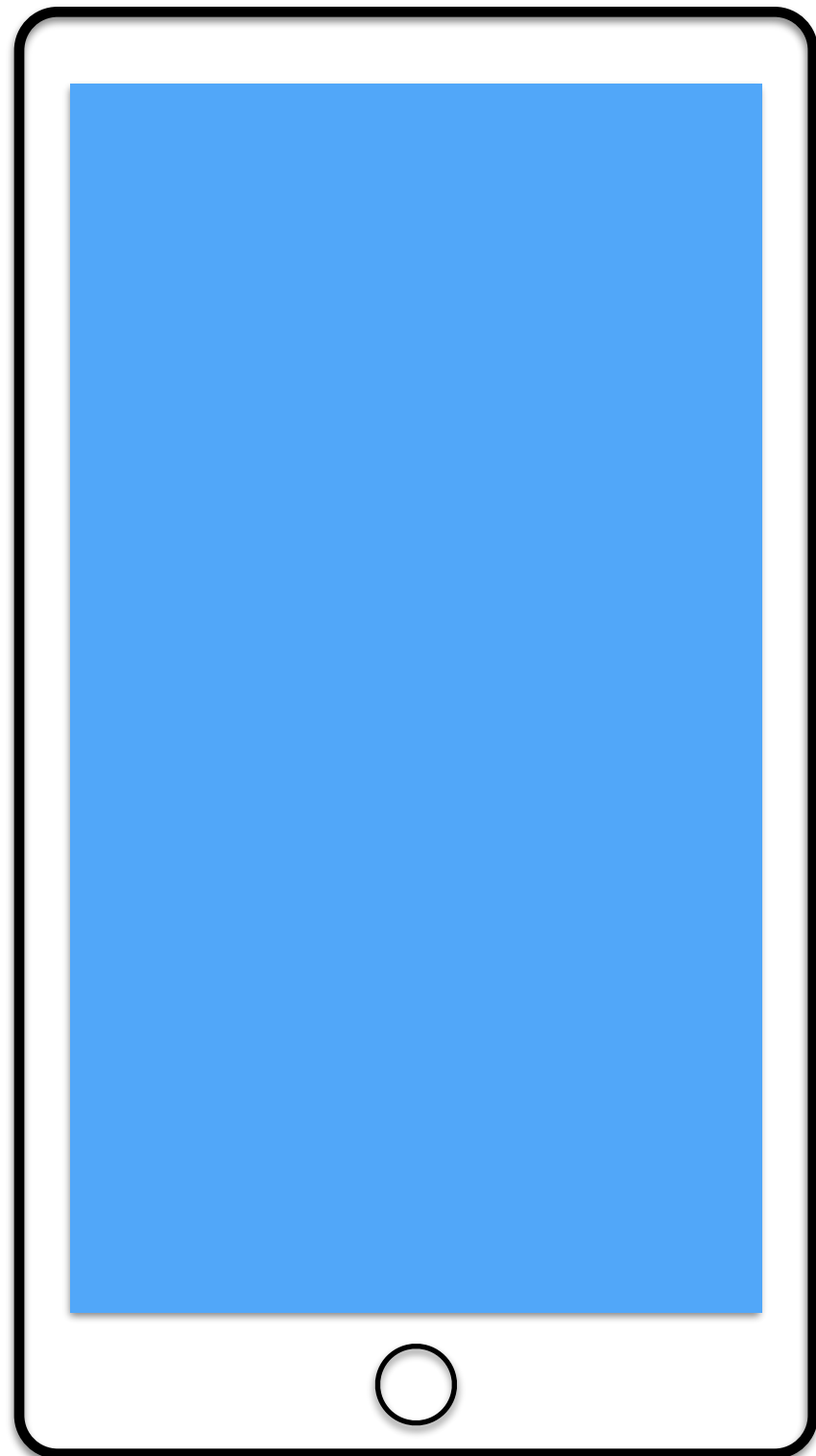


UIView hierarchy



```
view1.addSubview(view2)
```

```
view2.removeFromSuperview()
```



UIWindow

Only one for iOS application

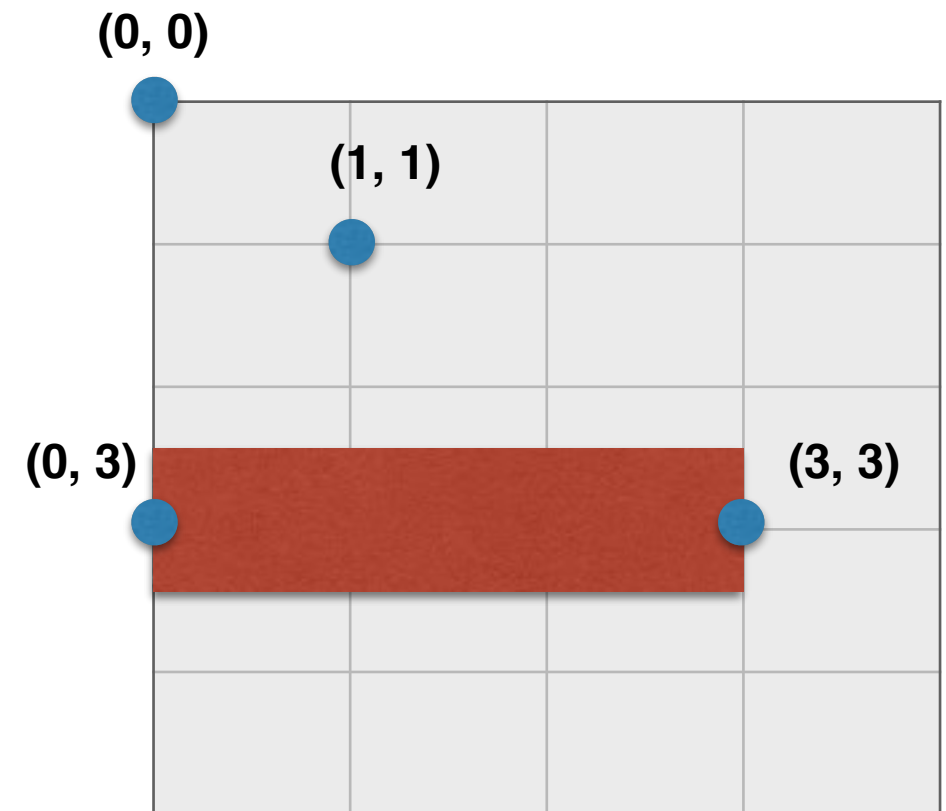
if view is on screen,
`view.window != nil`

Core Graphics(CG)

```
struct CGPoint {  
    var x, y: CGFloat  
    init(x: CGFloat, y: CGFloat)  
}
```

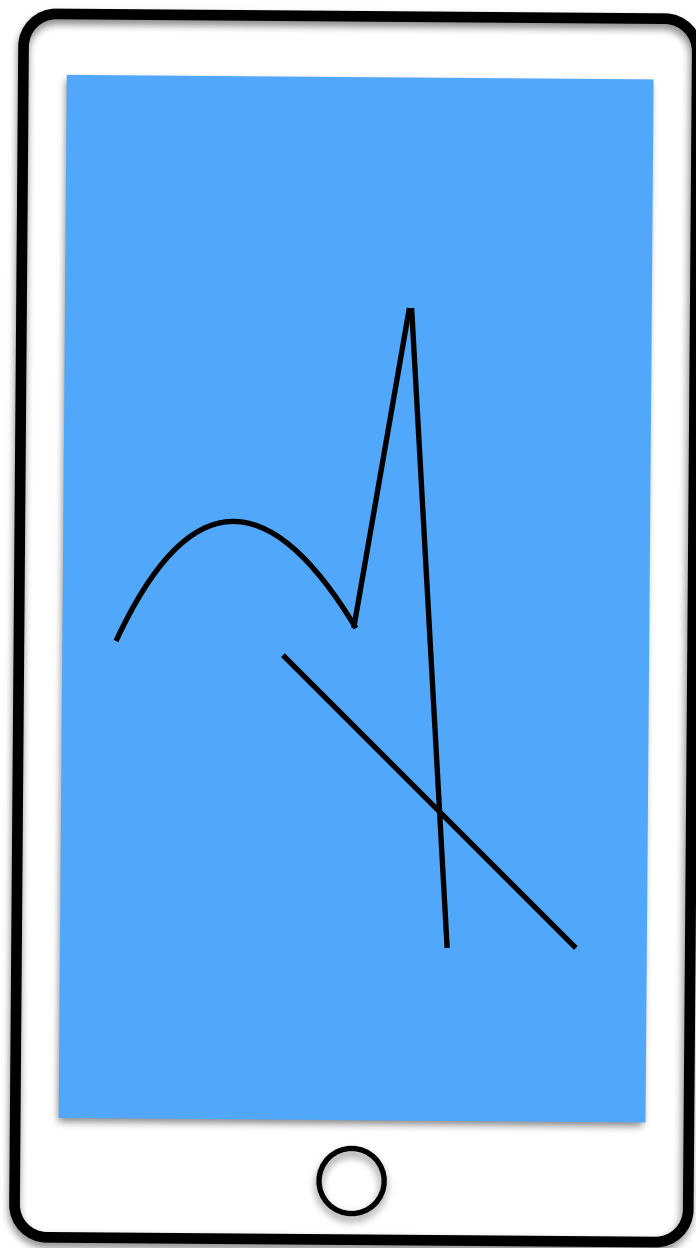
```
struct CGSize {  
    var width, height: CGFloat  
    init(width: CGFloat, height: CGFloat)  
}
```

```
struct CGRect {  
    var origin: CGPoint  
    var size: CGSize  
    init(origin: CGPoint, size: CGSize)  
    init(x: CGFloat, y: CGFloat, width: CGFloat, height: CGFloat)  
}
```



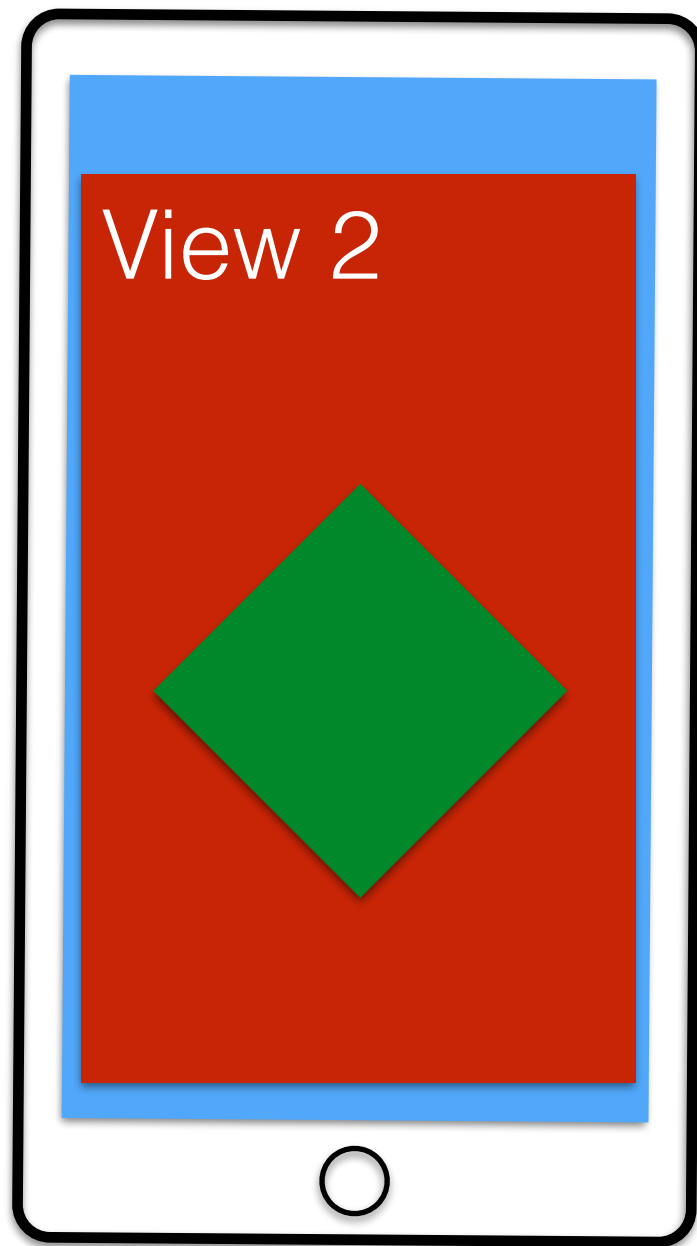
Line from (0, 3) to (3, 3) with lineWidth == 1

Drawing inside UIView



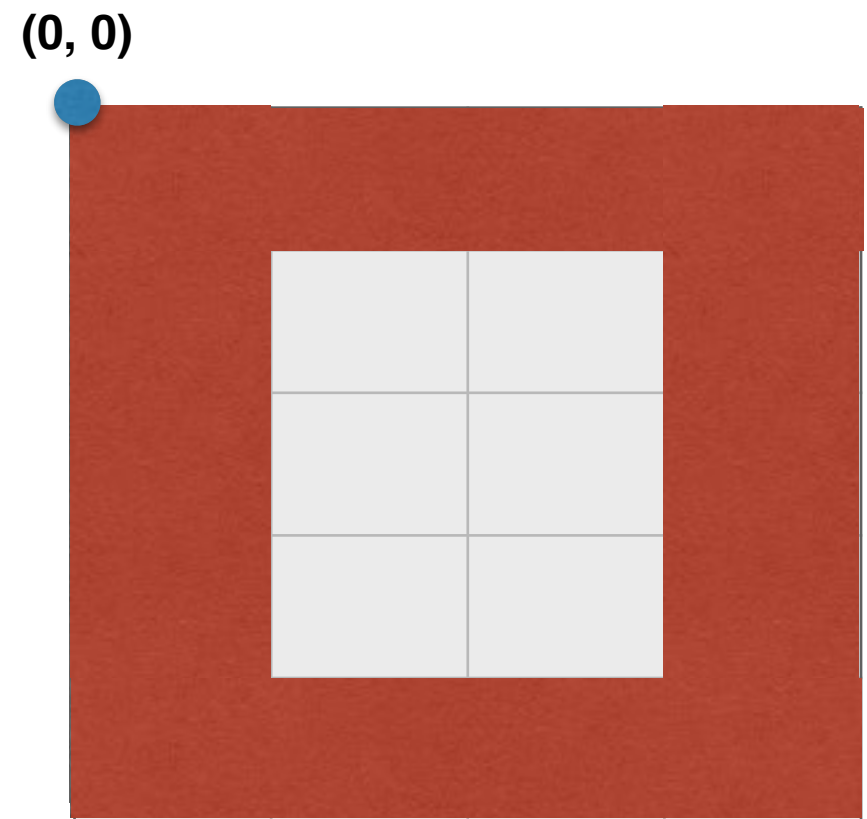
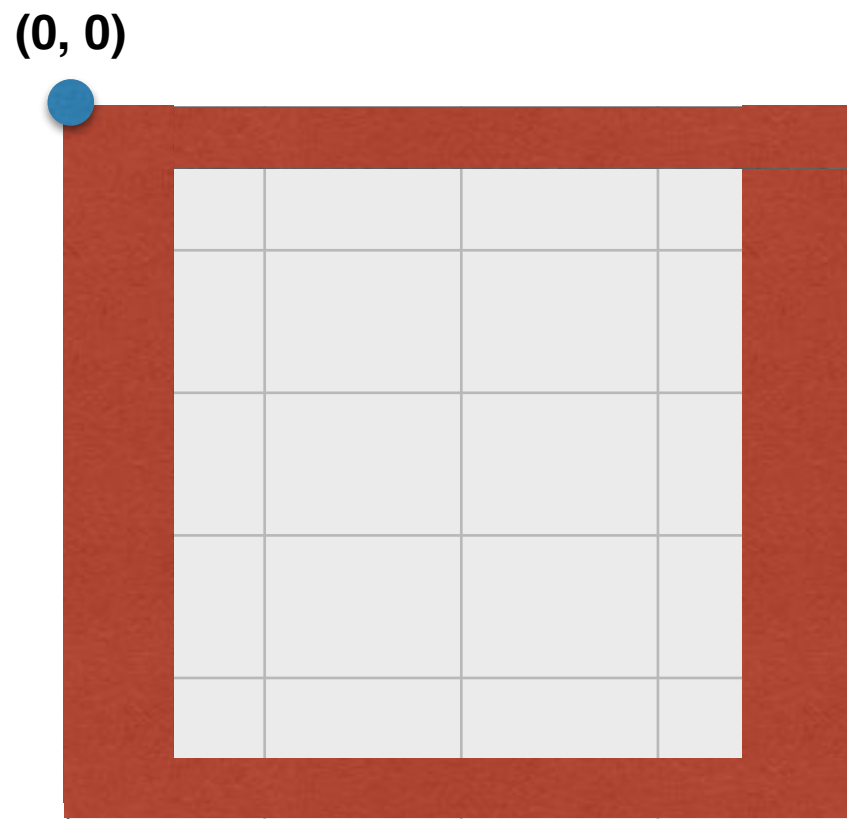
```
override func drawRect(rect: CGRect) {  
    UIColor.greenColor().setStroke()  
    UIColor.redColor().setFill()  
    let bezierPath = UIBezierPath()  
    bezierPath.fill()  
    bezierPath.stroke()  
}
```

Where is a view?



```
var frame: CGRect  
var bounds: CGRect  
var transform: CGAffineTransform
```

Be careful



```
let rect = CGRect(x: 0, y: 0, width: 4, height: 5)  
let path = UIBezierPath(rect: rect)
```

Which one is the right one?

UIView creation

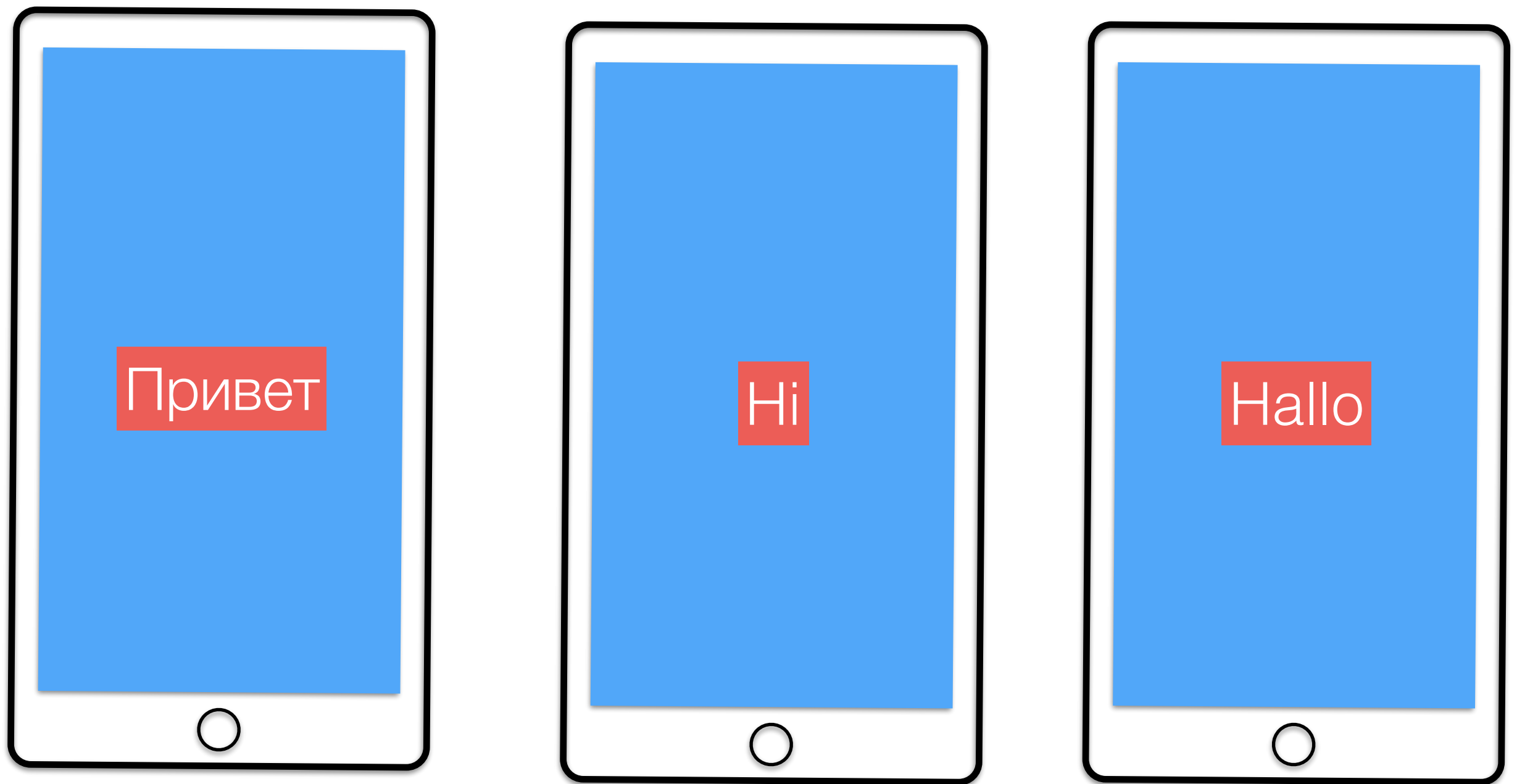
- `let view = UIView()`
- `let view = UIView(frame: frame)`
- `let frame = CGRect(x: 0, y: 0, width: 100, height: 100)`

UIButton example

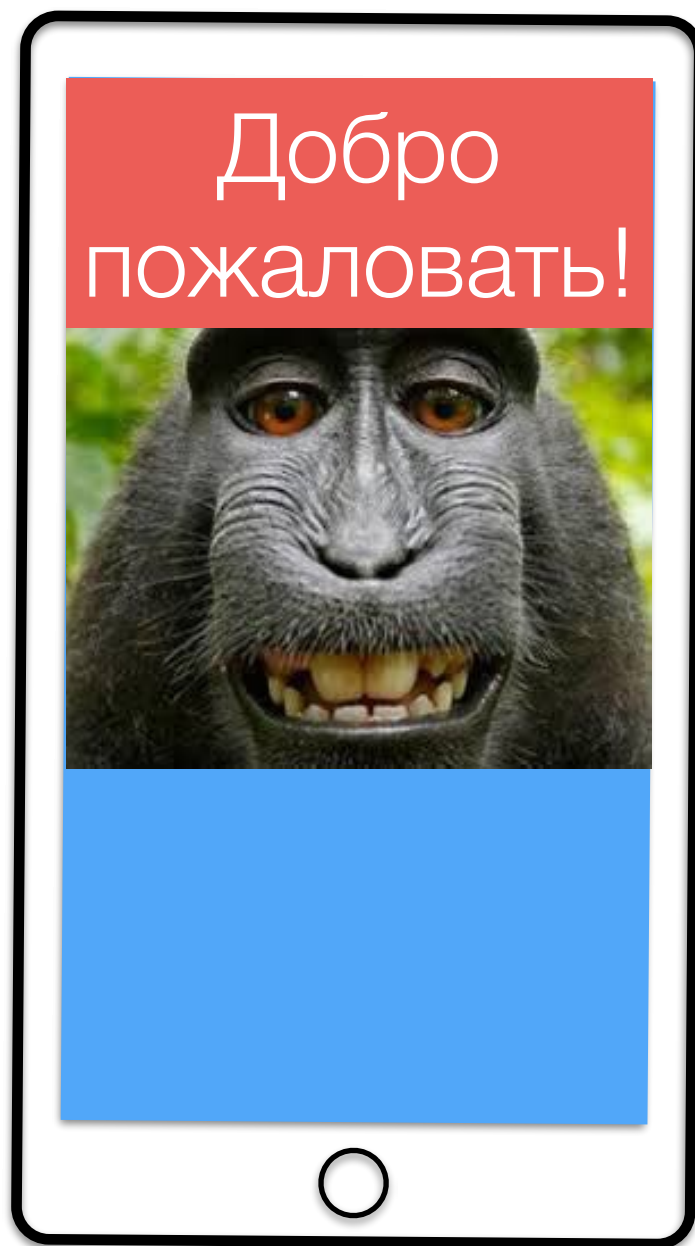


Drawing inside views can
be used to archive custom
buttons

Autolayout



Autolayout



Why Autolayout?

- Localizations
- Screen orientations
- Screen orientations

What is Autolayout?

- Set of linear equations defining how UIView will be displayed(related to another UIView)
- UIView's frame will be calculated according to the equations

$$x2 = b * x1 + d$$

$$y2 \geq b * x1 + d$$

Possible x values:

- leading, trailing, center, width

Possible y values:

- top, bottom, center, height