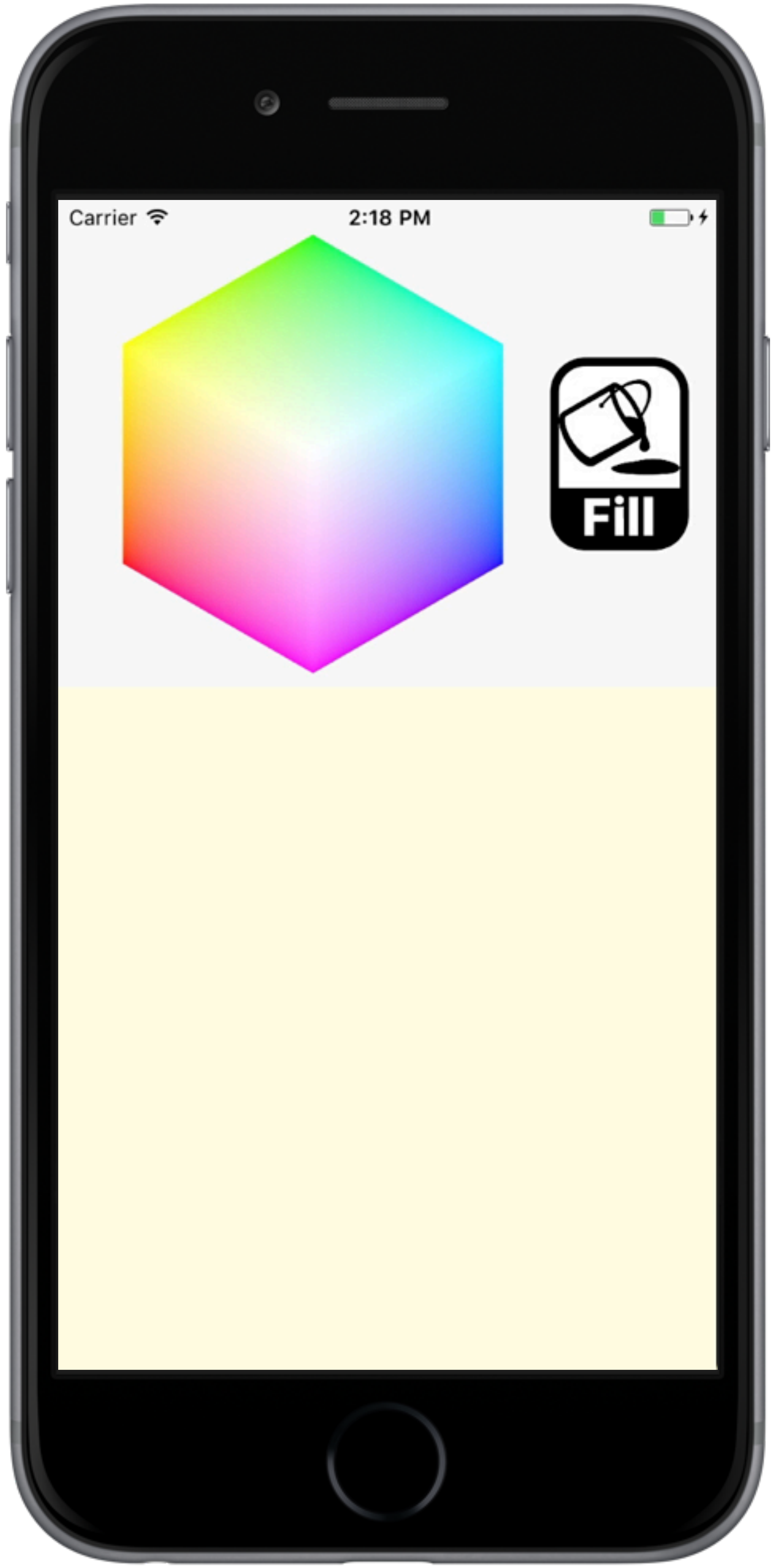
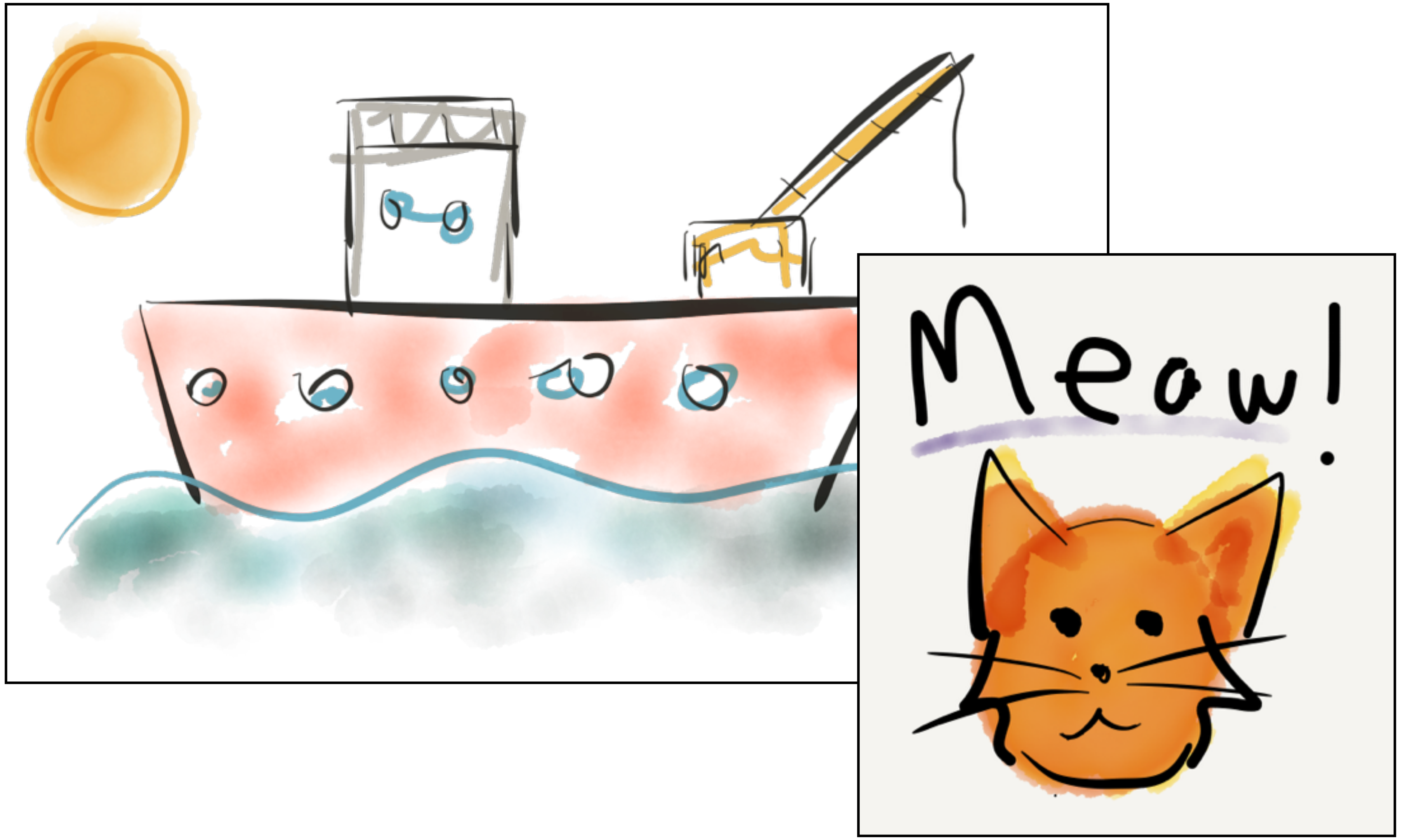
CUSTOM CONTROLS IN iOS

Part 10: Core Image & Core Graphics

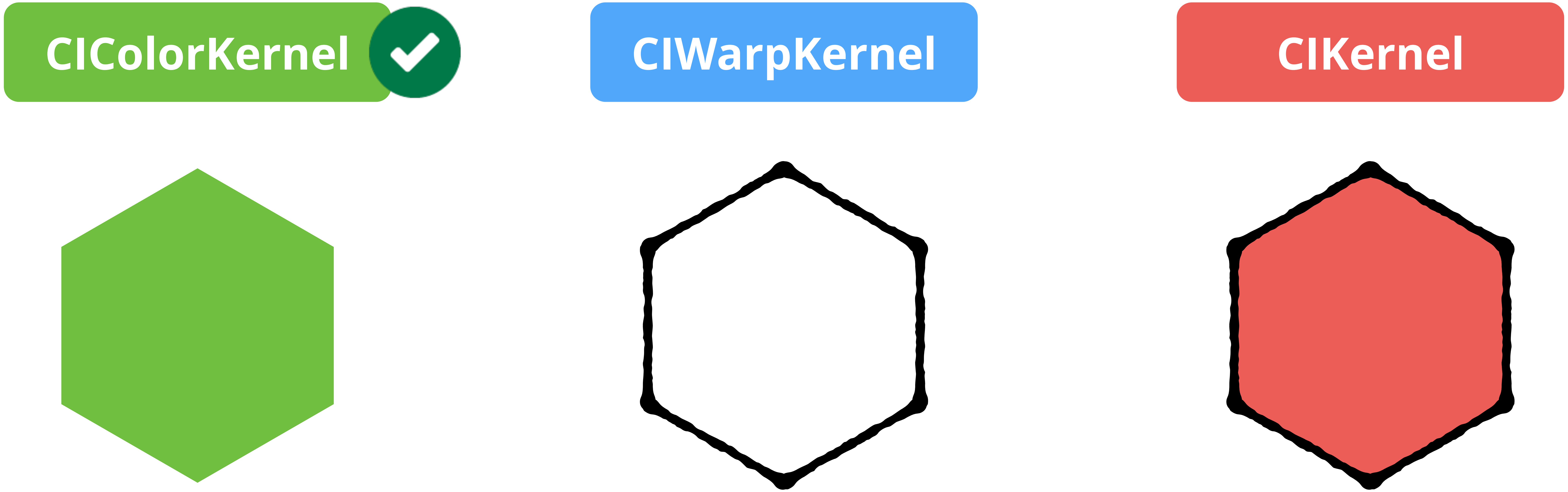


Carrier
2:18 PM
Fill

Core Graphics



Core Image

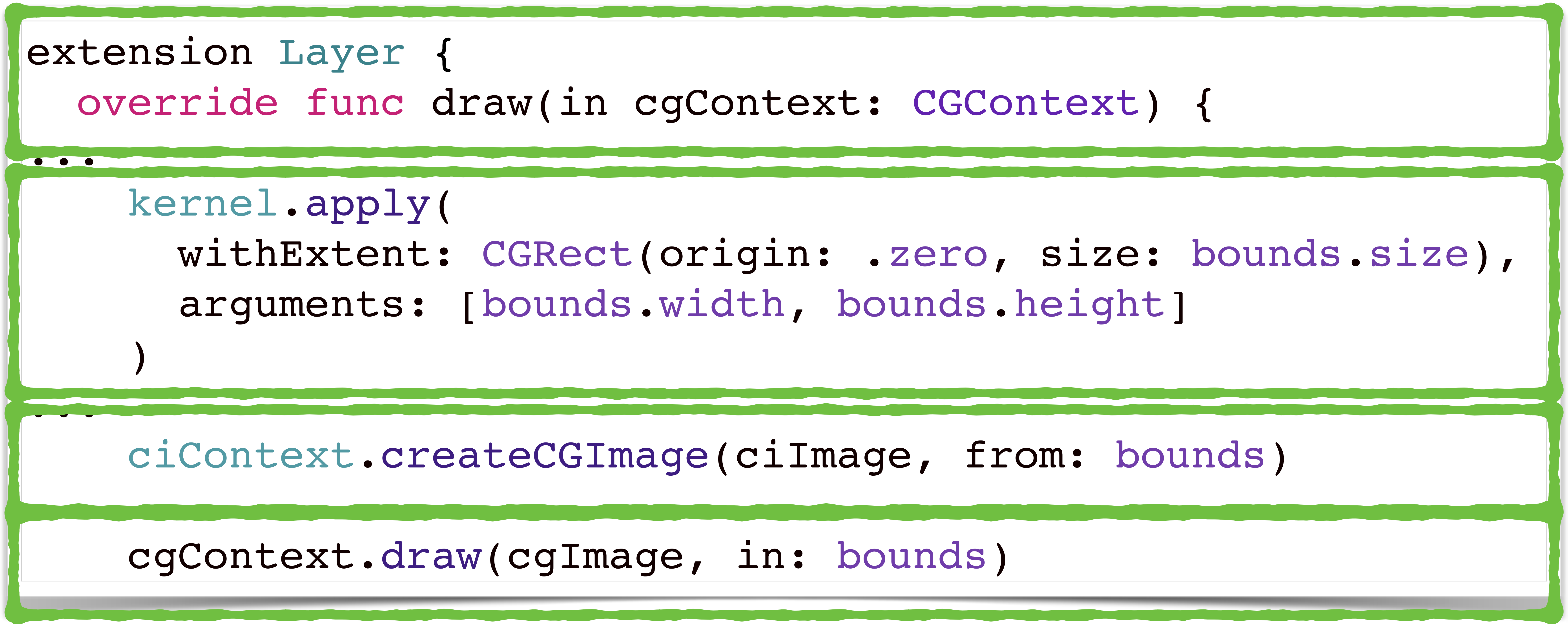


CIColorKernel ✓

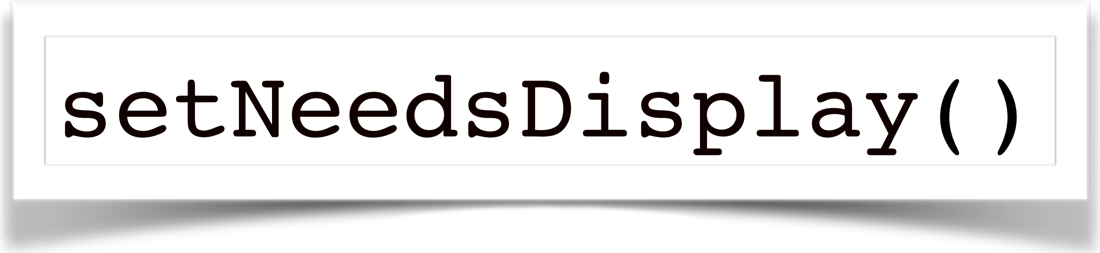
CIWarpKernel

CIKernel

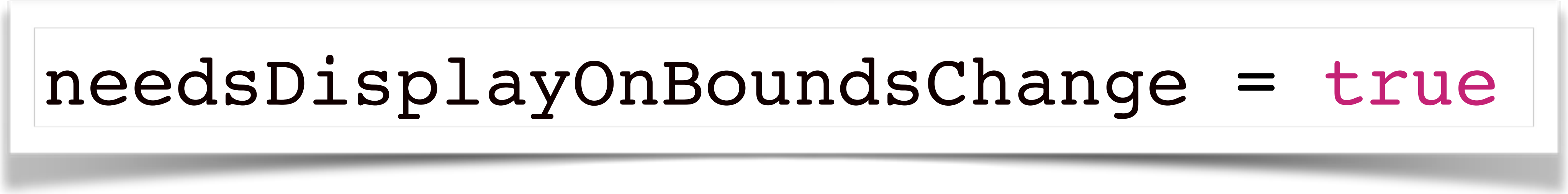
Rendering the Colorgon



```
extension Layer {
  override func draw(in cgContext: CGContext) {
...
    kernel.apply(
      withExtent: CGRect(origin: .zero, size: bounds.size),
      arguments: [bounds.width, bounds.height]
    )
...
    ciContext.createCGImage(ciImage, from: bounds)
    cgContext.draw(cgImage, in: bounds)
```

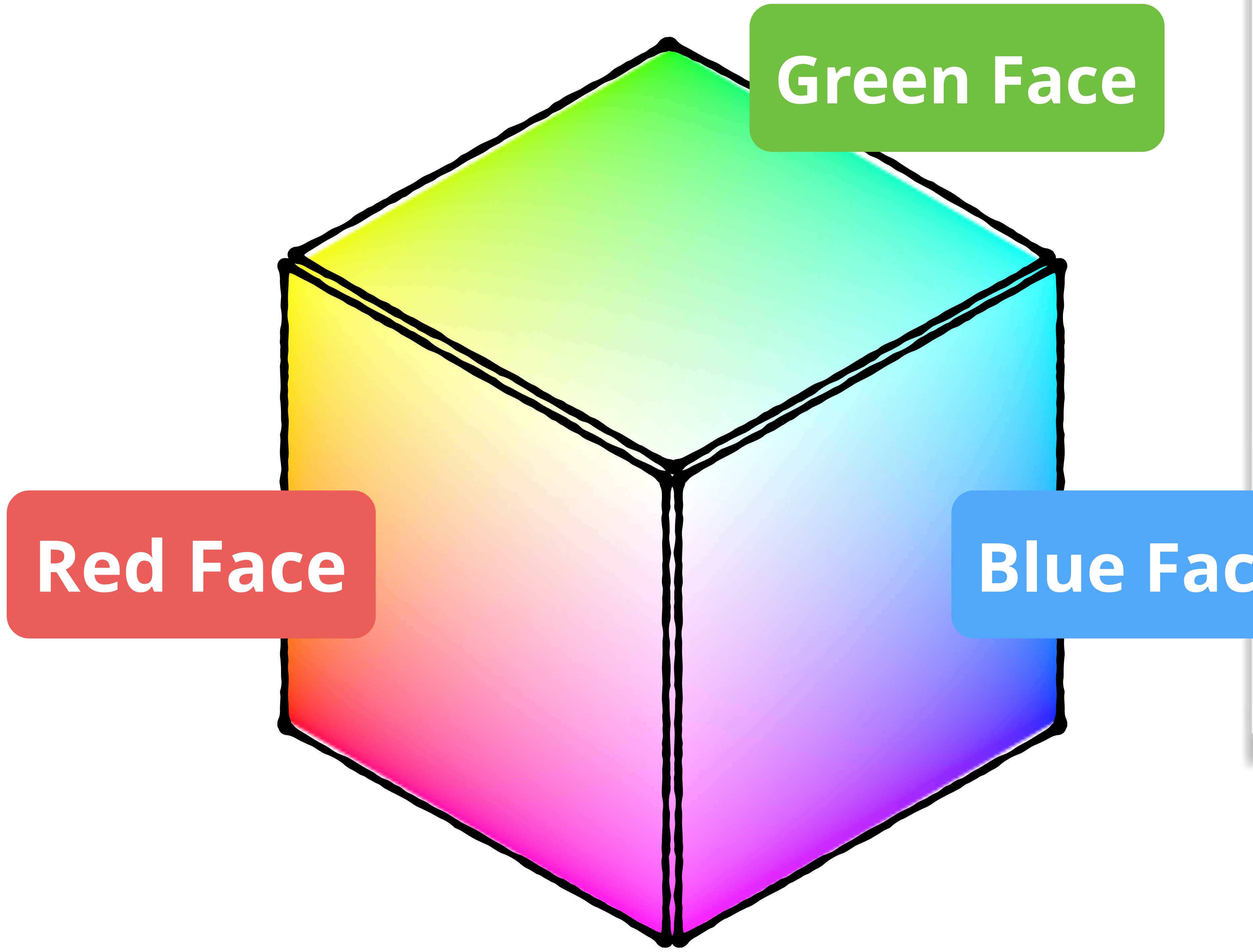


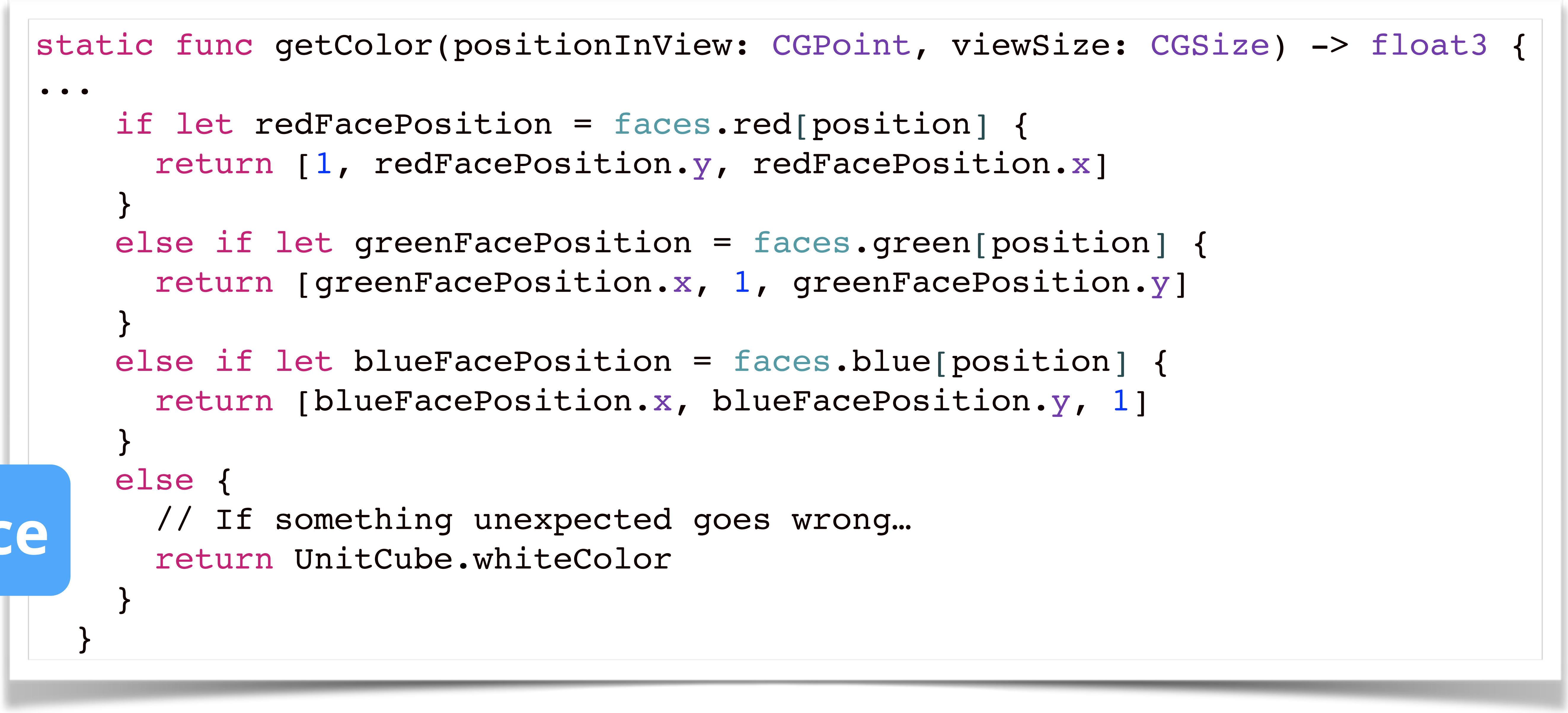
```
setNeedsDisplay()
```



```
needsDisplayOnBoundsChange = true
```

Challenge Time!





```
static func getColor(positionInView: CGPoint, viewSize: CGSize) -> float3 {
...
    if let redFacePosition = faces.red[position] {
      return [1, redFacePosition.y, redFacePosition.x]
    }
    else if let greenFacePosition = faces.green[position] {
      return [greenFacePosition.x, 1, greenFacePosition.y]
    }
    else if let blueFacePosition = faces.blue[position] {
      return [blueFacePosition.x, blueFacePosition.y, 1]
    }
    else {
      // If something unexpected goes wrong…
      return UnitCube.whiteColor
    }
  }
```