

«MyView» Swift handmade implementation

Fabrice.Kordon@lip6.fr



As an introduction...



Storyboard is not coming back

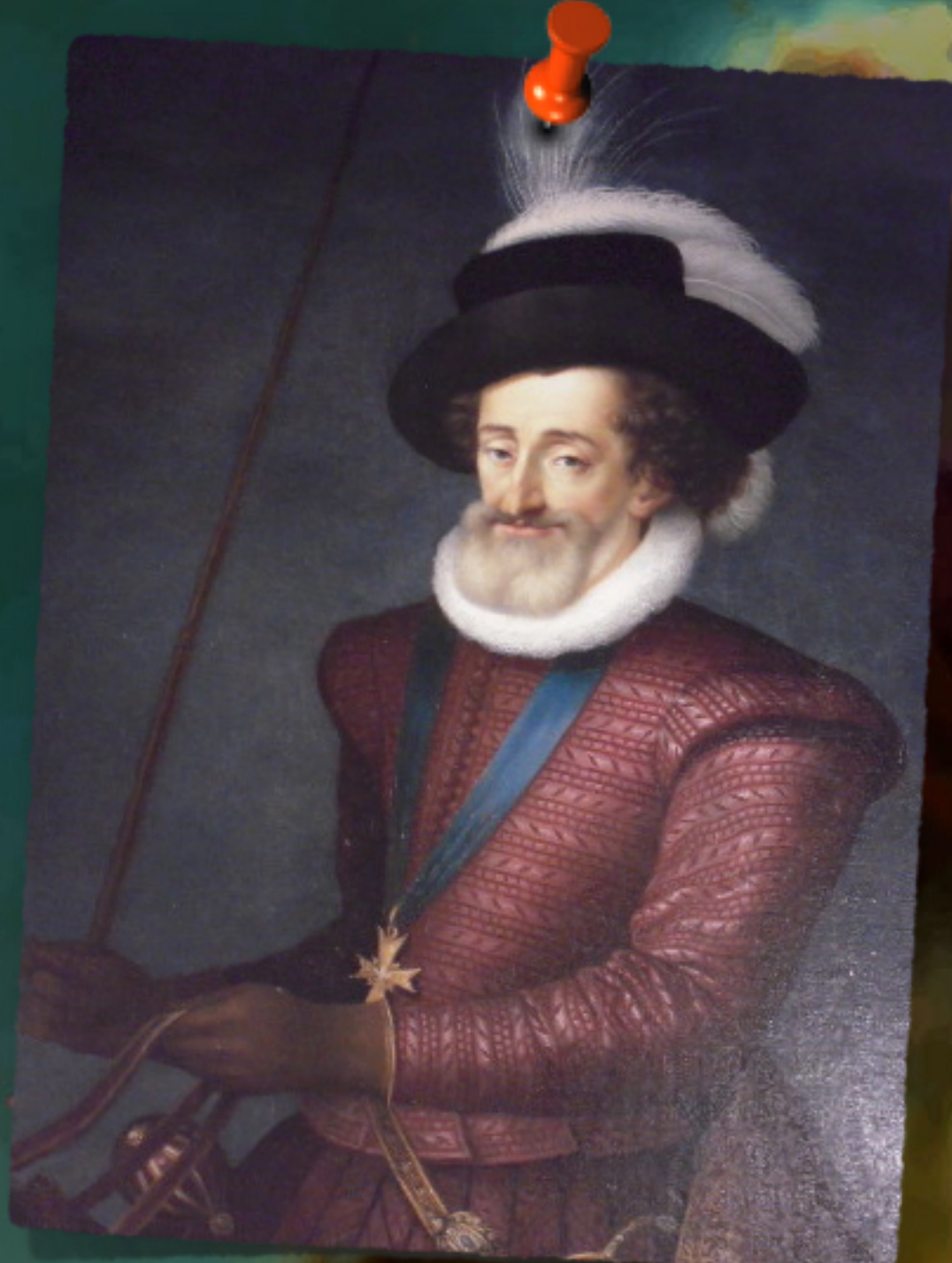
Bye!

Programmatic implementation

- Swift
- Explicit handling of coordinates
 - ▶ The «increment method»
- No «model» here
 - ▶ Far too simple



Let's go!



Small demo (4" screen)



ViewController.swift

```
//  
// ViewController.swift  
// MyView-swift  
//  
// Created by Fabrice Kordon on 28/09/2018.  
// Copyright © 2018 Sorbonne Université. All rights reserved.  
//  
  
import UIKit  
  
class ViewController: UIViewController {  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
        // Do any additional setup after loading the view, typically from a nib.  
        let screen = UIScreen.main  
        let rect = screen.bounds  
        let v = MyView(frame: rect)  
        self.view = v  
    }  
}
```

MyView.swift

```
import UIKit

class MyView: UIView {

    fileprivate let myDevice = UIDevice.current
    fileprivate let screen = UIScreen.main

    fileprivate let deviceData = UILabel()
    fileprivate let model = UILabel()
    fileprivate let orientation = UILabel()

    fileprivate var top = 0
    fileprivate var incr = 0

    // Required by Xcode (but unused)
    required init?(coder aDecoder: NSCoder) {
        super.init(coder: aDecoder)
        fatalError("init(coder:) has not been implemented")
    }
}
```

MyView.swift

```
override init (frame: CGRect) {
    deviceData.textAlignment = .center
    model.textAlignment = .center
    orientation.textAlignment = .center
    model.text = myDevice.model
    super.init(frame: frame)
    // Handled after the call to "super" (we need self here)
    if myDevice.userInterfaceIdiom == .phone && screen.scale == 3.0 {
        deviceData.text = "Large iPhone (\(myDevice.systemName), \
(myDevice.systemVersion))"
        incr = 50
    } else if myDevice.userInterfaceIdiom == .phone {
        deviceData.text = "iPhone/iPod (\(myDevice.systemName), \
(myDevice.systemVersion))"
        incr = 40
    } else {
        deviceData.text = "iPad (\(myDevice.systemName), \
(myDevice.systemVersion))"
        incr = 100
    }
    // Let's deal with the view's elements
    // Handled after the call to "super" (these are super's properties)
    self.backgroundColor = UIColor.white
    self.addSubview(deviceData)
    self.addSubview(model)
    self.addSubview(orientation)
}
```

MyView.swift

```
override func draw(_ rect: CGRect) {
    if myDevice.orientation == UIDeviceOrientation.portrait ||
        myDevice.orientation == UIDeviceOrientation.portraitUpsideDown {
        orientation.text = "Portrait orientation"
        top = 100
    } else if myDevice.orientation == UIDeviceOrientation.landscapeLeft ||
        myDevice.orientation == UIDeviceOrientation.landscapeRight {
        orientation.text = "Landscape orientation"
        top = 50
    } else {
        orientation.text = "Device is laid flat"
    }
    // Setting up orientation information
    deviceData.frame = CGRect(x: CGFloat(rect.size.width / 2.0 - 150.0),
                              y: CGFloat(top), width: 300.0, height: 21.0)
    model.frame = CGRect(x: CGFloat(rect.size.width / 2.0 - 150.0),
                          y: CGFloat(top + incr), width: 300.0, height: 21.0)
    orientation.frame = CGRect(x: CGFloat(rect.size.width / 2.0 - 150.0),
                                y: CGFloat(top + 2 * incr),
                                width: 300.0, height: 21.0)
}
```

MyView.swift

5

```
// Drawing circles
let ctx = UIGraphicsGetCurrentContext()
let table = [(0.9,70.0), (0.8,60.0), (0.7,50.0), (0.6,40.0),
             (0.5,30.0), (0.3,20.0), (0.0,10.0),]
for (vert,radius) in table {
    UIColor(red: 1.0, green: CGFloat(vert), blue: 0.0, alpha: 1.0).setFill()
    ctx?.addArc(center: CGPoint(x:rect.size.width / 2.0,
                                y:rect.size.height / 2.0 + 70.0),
                radius: CGFloat(radius), startAngle: 0.0,
                endAngle: CGFloat(Double.pi * 2), clockwise: true)
    ctx?.fillPath();
}
```


As a conclusion

Easy si'n't it?

- No, you really do not need StoryBoard
 - ▶ A first way to deal with orientation



- Keep thinking a bit

