SCIENCES SORBONNE UNIVERSITÉ

Views, additional elements

Fabrice.Kordon@lip6.fr

LIP6

F. Kordon - Sorbonne Université - CC2018

A few useful stuff

- **Change the status bar**
  - Light/Dark
- **Change the angle of a view**
  - Allow an object to rotate

F. Kordon - Sorbonne Université - CC2018

The status bar...

Light? Dark?

F. Kordon - Sorbonne Université - CC2018

Changing the status bar

In a view controller

Objective-C

- Overload a method
  - **- (UIStatusBarStyle)preferredStatusBarStyle**
- Values to return
  - **UIStatusBarStyleDefault, UIStatusBarStyleLightContent**

Swift

- Override a property
  - **var preferredStatusBarStyle: UIStatusBarStyle { get }**
- Values
  - **Default, LightContent**

If you change during execution?

- setNeedsStatusBarAppearanceUpdate()

F. Kordon - Sorbonne Université - CC2018

Display a view with an angle

Rotate some display

- Particular case of wider forms of transformations
- Handled by Core Graphics

CGAffineTransform

- Transforming an image based on a 3x3 matrix
- Complex

But to change the angle, it's easier

- Layout the view from its center
  - **center property**
- Apply some transformation
  - **CGAffineTransformMakeRotation**
  - **Angle ∈ [0, 2π]**

F. Kordon - Sorbonne Université - CC2018

Demo

F. Kordon - Sorbonne Université - CC2018

Code from the example


```
private let toolBar = UIToolbar()
private let asteroid = UIImageView(image: UIImage(named: "comete"))
private let label = UILabel()
private var posX = 10
private var posY = 50
private var angle = CGFloat(0.0)
private let asteroidSize = UIImage(named: "comete")!.size.width

convenience init(frame : CGRect) {
    self.init()
    let v = UIView(frame: frame)
    v.backgroundColor = UIColor.black
    self.view = v
...
    label.text = "Tchourioumov-Guerassimenko"
    label.textAlignment = .center
    label.textColor = UIColor.white
    v.addSubview(asteroid)
    v.addSubview(label)
    v.addSubview(toolBar)
    self.displayView(frame.size)
    self .displayAsteroid()
}
```

Code from the example


```
@objc func myAction() {
    posX = posX + 4
    posY = posY + 4
    self.displayAsteroid()
}
func displayView(_ size: CGSize) {
    label.frame = CGRect(x: size.width / 2.0 - 150.0,
                         y: size.height - 80.0 ,
                         width: 300.0, height: 20.0)
    if UIDevice.current.userInterfaceIdiom == .phone &&
        (size.width >= 812 || size.height >= 812) {
        // iPhone X, Xs, Xs max, Xr (with a bevel)
        toolBar.frame = CGRect(x: 0.0, y: 0.0 ,
                               width: size.width, height: 80.0)
    } else {
        toolBar.frame = CGRect(x: 0.0, y: 0.0 ,
                               width: size.width, height: 60.0)
    }
    self.displayAsteroid()
}
func displayAsteroid () {
    angle = angle + 0.0628
    if angle > 6.28 {
        angle = 0.0
    }
    asteroid.transform = CGAffineTransform(rotationAngle: angle)
    asteroid.center = CGPoint(x: CGFloat(posX), y: CGFloat(posY))
}
```

Code from the example


```
@objc func myAction() {
    posX = posX + 4
    posY = posY + 4
    self.displayAsteroid()
}
```

In Objective-C (Only angle management)

```
- (void) displayAtX:(CGFloat)x andY:(CGFloat)y andAngle:(float)a {
    CGAffineTransform transformation = CGAffineTransformMakeRotation(a);
    [view setCenter:CGPointMake(x, y)];
    [view setTransform:transformation];
}
```

```
        } else {
            toolBar.frame = CGRect(x: 0.0, y: 0.0 ,
                                   width: size.width, height: 60.0)
        }
        self.displayAsteroid()
    }
    func displayAsteroid () {
        angle = angle + 0.0628
        if angle > 6.28 {
            angle = 0.0
        }
        asteroid.transform = CGAffineTransform(rotationAngle: angle)
        asteroid.center = CGPoint(x: CGFloat(posX), y: CGFloat(posY))
    }
```

As a conclusion…

Observation

- The controller is linked to its view
  - **And the related data manager (MVC)**
- View can be «ported»
  - **By means of the MVC group**

Yet much to discover

- 3D effects
- «layers» in a view
- Moves & animations
- etc.

F. Kordon - Sorbonne Université - CC2018