

一、理解图层与UIView的关系

1. UIView能显示在屏幕上，是因为里面有个图层，UIView的DrawRect方法，会将绘制的东西绘在图层上，当要显示的时候，将图层拷贝到屏幕上进行显示

二、CALayer的基本属性

1. 尺寸(bounds)

2. 背景颜色(background-color)

3. 位置(position)

4. 边框颜色(border-color)、边框长度(border-width) "配合使用才有效果"

5. 圆角(cornerRadius)

6. 内容对象(contents) 通常设置图片，

*如果要显示圆角效果，必须设置图层的masksToBounds为YES，意思是把多余的剪切掉

7. 阴影属性

*阴影颜色(shadow-color)、阴影透明度(shadow-opacity) "要配合使用才有效果"

*阴影起始位置(shadow-offset)

*如果图片添加了圆角并设置了剪切掉多余的部分，阴影效果不会出现

*如果又想图片有圆角效果，又想有阴影，那图片只能是本来就有圆角效果的，不用代码设置

8. transform(CATransform3D)属性

*缩放CATransform3DMakeScale

*旋转CGAffineTransformMakeRotation,理解xyz方向的意思

*平移CATransform3DMakeTranslation

*熟练transform里的key Paths 如transform.scale transform.rotation transform.translation

*查看文档，搜索CATransform3D key paths，了解更多key paths属性

三、UIView与CALayer的细节

1. CALayer属于QuartzCore框架 与 CGColor所属于CoreGraphics框架

QuartzCore框架与CoreGraphics框架都是可跨平台使用，能在iOS和Mac OS X上都能使用

2. UIView比CALayer多了一个事件处理的功能

四、position和anchorPoint

/*目标

*掌握position和anchorPoint的关系

*/

五、隐式动画

"什么是隐式动画，直接改变属性就会有动画效果"

1. 根层与非根层，

*控件的layer属性是根层

*控件的layer属性的子层就是非根层

2. 非根层有隐式动画，根层是没有隐式动画的

3. 隐藏动画怎么禁止

```
/*
[CATransaction begin];
[CATransaction setDisableActions:YES];
//设置隐式动画动画时间
self.myview.layer.position = CGPointMake(10, 10);
[CATransaction commit];
*/
```

六、自定义图层

1. 自定义图层实现【drawInContext:】绘制自己想要的东西到图层

2. 如果自定义图层，要显示时要调用图层的setNeedsDisplay方法

//只有显示的调用setNeedsDisplay方法才会调用 drawInContext:方法

-(void)drawInContext:(CGContextRef)ctx{

CGContextSetRGBStrokeColor(ctx, 1, 0, 0, 1);

```
CGContextAddEllipseInRect(ctx, CGRectMake(0, 0, 50, 50));
CGContextFillPath(ctx);
```

七、核心动画

```
/*目标:
 *掌握核心动画的类结构
 *
 */
```

1、CABasicAnimation

①相关属性

```
*keyPath 动画类型
/*如
 *平移动画: position/transform.translation.x
 *旋转动画: transform.rotation/transform.rotation.x/transform.rotation/rotation.y
 *缩放动画: bounds/transform.scale/transform.scale.x/transform.scale.y
 */
*fromValue 动画开始位置
*toValue 动画结束位置
*byValue 动画增加值

*removedOnCompletion 动画完成后不要删除
*fillMode 保持最新状态
/*
 *默认动画执行后, 会恢复到动画之前的状态
 *如果想保存动画之后的状态设置动画的removedOnCompletion=NO/fillMode=kCAFillModeForwards
 */
```

2、CAKeyframeAnimation帧动画

```
*values 动画轨迹
*timingFunction 动画速度
*duration 动画时间
*repeatCount 动画重复次数
//kCAMediaTimingFunctionLinear 线性匀速
//kCAMediaTimingFunctionEaseIn 先慢后快
//kCAMediaTimingFunctionEaseOut 先快后慢
//kCAMediaTimingFunctionEaseInEaseOut 先慢后快再慢、中间快两边慢

*path 动画路径
//如果设置了path values被忽略
/*如添加一个圆的路径
*
CGMutablePathRef path = CGPathCreateMutable();
CGPathAddEllipseInRect(path, NULL, CGRectMake(10, 30, 300, 300));
animation.path = path;
CFRelease(path);
*/
```

"图标抖动案例"

步骤:

1>添加图片到控制器view

2>监听抖动按钮

```
//创建帧动画对象
CAKeyframeAnimation *animation = [CAKeyframeAnimation animation];
animation.keyPath = @"transform.rotation";
//旋转浮动度数
CGFloat angle = M_PI_4 * 0.1;
//抖动轨迹
animation.values = @[@(-angle),@(angle),@(-angle)];
//动画次数
animation.repeatCount = MAXFLOAT;
//执行动画
[self.iconBtn.layer addAnimation:animation forKey:@"shake"];
```

3>监听停止按钮

```
//停止动画
[self.iconBtn.layer removeAnimationForKey:@"shake"];
```

3、CATransition(转场动画)

使用步骤

①创建CATransition对象

②设置type

```
/**
*****
type:动画类型(比如:滴水效果,翻转效果...)
-----
fade kCATransitionFade 交叉淡化过渡
moveIn kCATransitionMoveIn 新视图移到旧视图上面
push kCATransitionPush 新视图把旧视图推出去
reveal kCATransitionReveal 将旧视图移开,显示下面的新视图
pageCurl 向上翻一页
pageUnCurl 向下翻一页
rippleEffect 滴水效果
suckEffect 收缩效果,如一块布被抽走
cube 立方体效果
oglFlip 上下左右翻转效果
rotate 旋转效果
cameraIrisHollowClose 相机镜头关上效果(不支持过渡方向)
cameraIrisHollowOpen 相机镜头打开效果(不支持过渡方向)

*****/
③设置子类型subtype
/**
*****
subtype: 动画方向(比如说是从左边进入,还是从右边进入...)
-----
kCATransitionFromRight;
kCATransitionFromLeft;
kCATransitionFromTop;
kCATransitionFromBottom;

当 type 为@"rotate"(旋转)的时候,它也有几个对应的 subtype,分别为:
90cw 逆时针旋转 90°
90ccw 顺时针旋转 90°
180cw 逆时针旋转 180°
180ccw 顺时针旋转 180°
**/
```

4、组动画

旋转 + 缩放 + 平移

组动画的使用步骤

//1.创建组动画

```
CAAnimationGroup *group = [CAAnimationGroup animation];
```

//2.设置动画数组[平移动画、旋转动画]

```
group.animations = @[transAni,rotationAni,scaleAni];
```

//3.添加动画

```
[self.imageView.layer addAnimation:group forKey:nil];
```

"车小弟补充案例"

1> 透明度动画 CABasicAnimation *opacity

2> 旋转动画 CABasicAnimation *rotation (transform.rotation)

3> 缩放动画CAKeyframeAnimation *scale ("transform.scale)

```
if (self.imageView.alpha == 0) {//打开
```

```
self.imageView.alpha = 1;
```

```
//由看不见到看得见 opacity.fromValue = @(0); opacity.toValue = @(1);
```

```

//由-45度到0rotation.fromValue = @(-M_PI_4); rotation.toValue = @(0);

//缩放scale.values = @[@(0),@(1.2),@(1)];

}
//组动画
CAAnimationGroup *group = [CAAnimationGroup animation];
group.animations = @[opacity,rotation,scale];

```

"底部旋转菜单案例"

八、动画总结(了解)

1.UIView封闭的动画

```

/*
[UIView beginAnimations:nil context:nil];
.....写要执行动画的代码
[UIView commitAnimations];*/

```

2.UIView animateWithDuration静态方法的动画

3.UIView的转场动画transitionWithView:duration:options:animations:completion:

4.图层的动画 如果要监听动画执行完成,要设置代理,监听animationDidStop方法

5.CALayer的动画, position未真正改变

6.能用UIView动画尽量用UIView

|