一、UIButton和UIImageView的区别

1.显示图片

1> UIImageView只能显示一种图片(图片默认会填充整个UIImageView) image\setImage:

2> UIButton能显示2种图片

\* 背景 (背景会填充整个UIButton) setBackroungImage:forState:

\* 前置（覆盖在背景上面的图片，按照之前的尺寸显示） setImage:forState:

\* 还能显示文字

2.点击事件

1> UIImageView默认是不能响应点击事件

2> UIButton能响应点击事件 : addTarget:action:forControlEvents:

3.使用场合

1> UIImageView : 只显示图片，不监听点击，点击了图片后不做任何反应

2> UIButton : 既显示图片，又监听点击，点击了图片后做一些其他事情

4.继承结构

1> UIButton之所以能添加监听器来监听事件，是因为它继承自UIControl

2> UIImagevIew之所以不能添加监听器来监听事件，是因为它直接继承自UIView

二、UIImageView的帧动画

@property(nonatomic,copy) NSArray \*animationImages; // 设置需要播放的图片（到时会按照数组顺序播放）

@property(nonatomic) NSTimeInterval animationDuration; // 动画的持续时间

@property(nonatomic) NSInteger animationRepeatCount; // 动画的执行次数（默认情况下是无限重复执行）

- (void)startAnimating; // 开始动画

- (void)stopAnimating; // 停止动画

- (BOOL)isAnimating; // 是否正在执行动画

三、格式符补充

%03d ： 每个整数占据3个位置，多出的位置用0填充

比如：

\* [NSString stringWithFormat:@"%03d", 0]; 返回的是@"000"

\* [NSString stringWithFormat:@"%03d", 7]; 返回的是@"007"

\* [NSString stringWithFormat:@"%03d", 15]; 返回的是@"015"

\* [NSString stringWithFormat:@"%03d", 134]; 返回的是@"134"

四、加载图片的两种方式

1.有缓存

UIImage \*image = [UIImage imageNamed:@"a.png"]

2.无缓存

// 全路径

NSString \*path = [[NSBundle mainBundle] pathForResource:@"a.png" ofType:nil];

// path是a.png的全路径

UIImage \*image = [[UIImage alloc] initWithContentsOfFile:path]