**Android 自定义控件\_圆形头像:**

链接地址：<http://www.cnblogs.com/Joanna-Yan/p/4758283.html>

**自定义控件的步骤：**

1. 设计需要的属性-----atts.xml文件中
2. 实现View------新建一个类继承需要的控件
3. 引用View------在layout布局文件中引用

**绘图：**

链接地址：<http://www.jcodecraeer.com/a/anzhuokaifa/androidkaifa/2012/1212/703.html>

Android画图最基本的三个对象（Color（颜料），Paint（画笔），Canvas（画布））

三个类都存放在 android.graphics包下

**Color :颜色对象**

1. 可以通过 Color.颜色名，来获取颜色，应为是静态的，返回一个整数值
2. 调用静态的 argb方法，可以调出个性的颜色
3. alpha 是设置透明度的；red 设置红色； green是设置绿色； blue设置蓝色

**Paint : 画笔对象**

链接地址：<http://www.cnblogs.com/lqminn/archive/2012/11/30/2796101.html>

**作用：**绘制几何、文本和位图

 setAntiAlias: 设置画笔的锯齿效果。   
   setColor: 设置画笔颜色   
   setARGB:  设置画笔的a,r,p,g值。   
   setAlpha:  设置Alpha值   
   setTextSize: 设置字体尺寸。   
   setStyle:  设置画笔风格，空心或者实心。   
   setStrokeWidth: 设置空心的边框宽度。   
   getColor:  得到画笔的颜色   
   getAlpha:  得到画笔的Alpha值。

**Canvas ：画布对象**

**区域：**drawRect(RectF rect, Paint paint) //绘制区域，参数一为RectF一个区域

**路径：**drawPath(Path path, Paint paint) //绘制一个路径，参数一为Path路径对象

**贴图：**drawBitmap(Bitmap bitmap, Rect src, Rect dst, Paint paint)  //贴图，参数一就是我们常规的**Bitmap：**对象，参数二是源区域(这里是bitmap)，参数三是目标区域(应该在canvas的位置和大小)，参数四是Paint画刷对象，因为用到了缩放和拉伸的可能，当原始Rect不等于目标Rect时性能将会有大幅损失。

**线：**drawLine(float startX, float startY, float stopX, float stopY, Paintpaint) //画线，参数一起始点的x轴位置，参数二起始点的y轴位置，参数三终点的x轴水平位置，参数四y轴垂直位置，最后一个参数为Paint 画刷对象。

**点：**drawPoint(float x, float y, Paint paint) //画点，参数一水平x轴，参数二垂直y轴，第三个参数为Paint对象。

**文本：**drawText(String text, float x, floaty, Paint paint)  //渲染文本，Canvas类除了上面的还可以描绘文字，参数一是String类型的文本，参数二x轴，参数三y轴，参数四是Paint对象。

**椭圆：**[drawOval](E:/android/android/android-sdk-windows/docs/reference/android/graphics/Canvas.html" \l "drawOval(android.graphics.RectF, android.graphics.Paint)" \t "http://blog.csdn.net/rhljiayou/article/details/_blank)([RectF](E:/android/android/android-sdk-windows/docs/reference/android/graphics/RectF.html" \t "http://blog.csdn.net/rhljiayou/article/details/_blank) oval, [Paint](E:/android/android/android-sdk-windows/docs/reference/android/graphics/Paint.html" \t "http://blog.csdn.net/rhljiayou/article/details/_blank) paint)//画椭圆，参数一是扫描区域，参数二为paint对象；

**圆：**[drawCircle](E:/android/android/android-sdk-windows/docs/reference/android/graphics/Canvas.html" \l "drawCircle(float, float, float, android.graphics.Paint)" \t "http://blog.csdn.net/rhljiayou/article/details/_blank)(float cx, float cy, float radius,[Paint](E:/android/android/android-sdk-windows/docs/reference/android/graphics/Paint.html" \t "http://blog.csdn.net/rhljiayou/article/details/_blank) paint)// 绘制圆，参数一是中心点的x轴，参数二是中心点的y轴，参数三是半径，参数四是paint对象；

**弧：**[drawArc](E:/android/android/android-sdk-windows/docs/reference/android/graphics/Canvas.html" \l "drawArc(android.graphics.RectF, float, float, boolean, android.graphics.Paint)" \t "http://blog.csdn.net/rhljiayou/article/details/_blank)([RectF](E:/android/android/android-sdk-windows/docs/reference/android/graphics/RectF.html" \t "http://blog.csdn.net/rhljiayou/article/details/_blank) oval, float startAngle, float sweepAngle, boolean useCenter, [Paint](E:/android/android/android-sdk-windows/docs/reference/android/graphics/Paint.html" \t "http://blog.csdn.net/rhljiayou/article/details/_blank) paint)//画弧，

参数一是RectF对象，一个矩形区域椭圆形的界限用于定义在形状、大小、电弧，参数二是起始角(度)在电弧的开始，

参数三扫描角(度)开始顺时针测量的，参数四是如果这是真的话,包括椭圆中心的电弧,并关闭它,如果它是假这将是一个弧线,参数五是Paint对象；

**setXfermode两个图层之间的显示：**

链接地址：<http://www.cnblogs.com/sank615/archive/2013/03/12/2955675.html>