android:绘图
View：组件，理解为画布
Drawable:所有可见对象的描述，理解为：素材类
Bitmap：图片类
Canvas：画笔
Paint：画笔样式与颜色、特效的集合

近期很多网友对 Android 用户界面的设计表示很感兴趣，对于 Android UI 开发自绘控件和游戏制作而言掌握好绘图基础是必不可少的。本次专题分 10 节来讲述，有关 OpenGL ES 相关的可能将放到以后再透露。本次主要涉及以下四个包的相关内容:

android.content.res 资源类
android.graphics 底层图形类
android.view 显示类
android.widget 控件类

一、android.content.res.Resources

对于 Android 平台的资源类 android.content.res.Resources 可能很多网友比较陌生，一起来看看 SDK 上是怎么介绍的吧，Contains classes for accessing application resources, such as raw asset files, colors, drawables, media or other other files in the package, plus important device configuration details (orientation, input types, etc.) that affect how the application may behave.平时用到的二进制源文件 raw、颜色 colors、图形 drawables 和多媒体文件 media 的相关资源均通过该类来管理。

int getColor(int id) 对应 res/values/colors.xml
Drawable getDrawable(int id) 对应 res/drawable/
XmlResourceParser getLayout(int id) 对应 res/layout/
String getString(int id) 和 CharSequence getText(int id) 对应 res/values/strings.xml
InputStream openRawResource(int id) 对应 res/raw/
void parseBundleExtra (String tagName, AttributeSet attrs, Bundle outBundle) 对应 res/xml/

String[] getStringArray(int id) res/values/arrays.xml
float getDimension(int id) res/values/dimens.xml

二、android.graphics.Bitmap

作为位图操作类，Bitmap 提供了很多实用的方法，常用的我们总结如下:

boolean compress(Bitmap.CompressFormat format, int quality, OutputStream stream) 压缩一个 Bitmap 对象根据相关的编码、画质保存到一个 OutputStream 中。其中第一个压缩格式目前有 JPG 和 PNG
void copyPixelsFromBuffer(Buffer src) 从一个 Buffer 缓冲区复制位图像素
void copyPixelsToBuffer(Buffer dst) 将当前位图像素内容复制到一个 Buffer 缓冲区

我们看到创建位图对象 createBitmap 包含了 6 种方法在目前的 Android 2.1 SDK 中，当然他们使用的是 API Level 均为 1，所以说从 Android 1.0 SDK 开始就支持了，所以大家可以放心使用。

static Bitmap createBitmap(Bitmap src)
static Bitmap createBitmap(int[] colors, int width, int height, Bitmap.Config config)
static Bitmap createBitmap(int[] colors, int offset, int stride, int width, int height, Bitmap.Config config)
static Bitmap createBitmap(Bitmap source, int x, int y, int width, int height, Matrix m, boolean filter)
static Bitmap createBitmap(int width, int height, Bitmap.Config config)
static Bitmap createBitmap(Bitmap source, int x, int y, int width, int height)
static Bitmap createScaledBitmap(Bitmap src, int dstWidth, int dstHeight, boolean filter) //创建一个可以缩放的位图对象

final int getHeight() 获取高度
final int getWidth() 获取宽度
final boolean hasAlpha() 是否有透明通道
void setPixel(int x, int y, int color) 设置某像素的颜色
int getPixel(int x, int y) 获取某像素的颜色，android 开发网提示这里返回的 int 型是 color 的定义

三、android.graphics.BitmapFactory

作为 Bitmap 对象的 I/O 类，BitmapFactory 类提供了丰富的构造 Bitmap 对象的方法，比如从一个字节数组、文件系统、资源 ID、以及输入流中来创建一个 Bitmap 对象，下面本类的全部成员，除了 decodeFileDescriptor 外其他的重载方法都很常用。

static Bitmap decodeByteArray(byte[] data, int offset, int length) //从字节数组创建

static Bitmap decodeByteArray(byte[] data, int offset, int length, BitmapFactory.Options opts)

static Bitmap decodeFile(String pathName, BitmapFactory.Options opts) //从文件创建，路径要写全

static Bitmap decodeFile(String pathName)

static Bitmap decodeFileDescriptor(FileDescriptor fd, Rect outPadding, BitmapFactory.Options opts) //从输入流句柄创建

static Bitmap decodeFileDescriptor(FileDescriptor fd)

static Bitmap decodeResource(Resources res, int id) //从 Android 的 APK 文件资源中创建，android123 提示是从/res/的 drawable 中

static Bitmap decodeResource(Resources res, int id, BitmapFactory.Options opts)

static Bitmap decodeResourceStream(Resources res, TypedValue value, InputStream is, Rect pad, BitmapFactory.Options opts)

static Bitmap decodeStream(InputStream is) //从一个输入流中创建

static Bitmap decodeStream(InputStream is, Rect outPadding, BitmapFactory.Options opts)

四、android.graphics.Canvas

从 J2ME MIDLET 时我们就知道 Java 提供了 Canvas 类，而目前在 Android 平台中，它主要任务为管理绘制过程，The Canvas class holds the "draw" calls. To draw something, you need 4 basic components: A Bitmap to hold the pixels, a Canvas to host the draw calls (writing into the bitmap), a drawing primitive (e.g. Rect, Path, text, Bitmap), and a paint (to describe the colors and styles for the drawing).

该类主要提供了三种构造方法，分别为构造一个空的 Canvas、从 Bitmap 中构造和从 GL 对象中创建，如下

Canvas()
Canvas(Bitmap bitmap)
Canvas(GL gl)

同时 Canvas 类的一些字段保存着重要的绘制方法定义，比如 Canvas.HAS_ALPHA_LAYER_SAVE_FLAG 保存时需要 alpha 层，对于 Canvas 类提供的方法很多，每个都很重要，下面我们一一作介绍

boolean clipPath(Path path)
boolean clipPath(Path path, Region.Op op)
boolean clipRect(float left, float top, float right, float bottom)
boolean clipRect(Rect rect)
boolean clipRect(float left, float top, float right, float bottom, Region.Op op)
boolean clipRect(Rect rect, Region.Op op)
boolean clipRect(RectF rect)
boolean clipRect(RectF rect, Region.Op op)
boolean clipRect(int left, int top, int right, int bottom)
boolean clipRegion(Region region, Region.Op op)
boolean clipRegion(Region region)

void concat(Matrix matrix)

void drawARGB(int a, int r, int g, int b)

void drawArc(RectF oval, float startAngle, float sweepAngle, boolean useCenter, Paint paint)

void drawBitmap(Bitmap bitmap, Matrix matrix, Paint paint)
void drawBitmap(int[] colors, int offset, int stride, float x, float y, int width, int height, boolean hasAlpha, Paint paint)
void drawBitmap(Bitmap bitmap, Rect src, Rect dst, Paint paint)
void drawBitmap(Bitmap bitmap, float left, float top, Paint paint)
void drawBitmap(int[] colors, int offset, int stride, int x, int y, int width, int height, boolean hasAlpha, Paint paint)
void drawBitmap(Bitmap bitmap, Rect src, RectF dst, Paint paint)
void drawBitmapMesh(Bitmap bitmap, int meshWidth, int meshHeight, float[] verts, int vertOffset, int[] colors, int colorOffset, Paint paint)

void drawCircle(float cx, float cy, float radius, Paint paint)
void drawColor(int color)
void drawColor(int color, PorterDuff.Mode mode)
void drawLine(float startX, float startY, float stopX, float stopY, Paint paint)
void drawLines(float[] pts, Paint paint)
void drawLines(float[] pts, int offset, int count, Paint paint)
void drawOval(RectF oval, Paint paint)
void drawPaint(Paint paint)
void drawPath(Path path, Paint paint)

void drawPicture(Picture picture, RectF dst)
void drawPicture(Picture picture, Rect dst)
void drawPicture(Picture picture)
void drawPoint(float x, float y, Paint paint)
void drawPoints(float[] pts, int offset, int count, Paint paint)
void drawPoints(float[] pts, Paint paint)
void drawPosText(char[] text, int index, int count, float[] pos, Paint paint)
void drawPosText(String text, float[] pos, Paint paint)
void drawRGB(int r, int g, int b)
void drawRect(RectF rect, Paint paint)
void drawRect(float left, float top, float right, float bottom, Paint paint)
void drawRect(Rect r, Paint paint)
void drawRoundRect(RectF rect, float rx, float ry, Paint paint)
void drawText(String text, int start, int end, float x, float y, Paint paint)
void drawText(char[] text, int index, int count, float x, float y, Paint paint)
void drawText(String text, float x, float y, Paint paint)
void drawText(CharSequence text, int start, int end, float x, float y, Paint paint)
void drawTextOnPath(String text, Path path, float hOffset, float vOffset, Paint paint)
void drawTextOnPath(char[] text, int index, int count, Path path, float hOffset, float vOffset, Paint paint)
void drawVertices(Canvas.VertexMode mode, int vertexCount, float[] verts, int vertOffset, float[]

texs, int texOffset, int[] colors, int colorOffset, short[] indices, int indexOffset, int indexCount, Paint paint)
static void freeGlCaches()
boolean getClipBounds(Rect bounds)
final Rect getClipBounds()
int getDensity()
DrawFilter getDrawFilter()
GL getGL()
int getHeight()
void getMatrix(Matrix ctm)
final Matrix getMatrix()
int getSaveCount()
int getWidth()
boolean isOpaque()
boolean quickReject(Path path, Canvas.EdgeType type)
boolean quickReject(float left, float top, float right, float bottom, Canvas.EdgeType type)
boolean quickReject(RectF rect, Canvas.EdgeType type)
void restore()
void restoreToCount(int saveCount)
final void rotate(float degrees, float px, float py)
void rotate(float degrees)
int save()
int save(int saveFlags)
int saveLayer(float left, float top, float right, float bottom, Paint paint, int saveFlags)
int saveLayer(RectF bounds, Paint paint, int saveFlags)
int saveLayerAlpha(float left, float top, float right, float bottom, int alpha, int saveFlags)
int saveLayerAlpha(RectF bounds, int alpha, int saveFlags)
final void scale(float sx, float sy, float px, float py)
void scale(float sx, float sy)
void setBitmap(Bitmap bitmap)
void setDensity(int density)
void setDrawFilter(DrawFilter filter)
void setMatrix(Matrix matrix)
void setViewport(int width, int height)
void skew(float sx, float sy)
void translate(float dx, float dy)

五、android.graphics.Color

有关 Android 平台上表示颜色的方法有很多种，Color 提供了常规主要颜色的定义比如 Color.BLACK 和 Color.GREEN 等等，我们平时创建时主要使用以下静态方法

static int argb(int alpha, int red, int green, int blue) 构造一个包含透明对象的颜色

static int rgb(int red, int green, int blue)　构造一个标准的颜色对象
static int parseColor(String colorString)　解析一种颜色字符串的值，比如传入 Color.BLACK

本类返回的均为一个整形类似　绿色为 0xff00ff00，红色为 0xffff0000。我们将这个 DWORD 型看做 AARRGGBB，AA 代表 Aphla 透明色，后面的就不难理解，每个分成 WORD 整好为 0-255。

今天我们继续介绍 Android 平台底层绘图类的相关内容，在 Android UI 开发专题(一) 之界面设计中我们介绍了有关 Android 平台资源使用以及 Bitmap 相关类的操作，接下来将会以实例的方式给大家演示各种类的用处以及注意点。今天我们继续了解 android.graphics 包中比较重要的绘图类。

一、 android.graphics.Matrix

有关图形的变换、缩放等相关操作常用的方法有:

void　reset() // 重置一个 matrix 对象。
void　set(Matrix src) //复制一个源矩阵，和本类的构造方法 Matrix(Matrix src)　一样
boolean　isIdentity() //返回这个矩阵是否定义(已经有意义)

void setRotate(float degrees) //指定一个角度以 0,0 为坐标进行旋转

void　setRotate(float degrees, float px, float py)　//指定一个角度以 px,py 为坐标进行旋转

void　setScale(float sx, float sy)　// 缩放

void　setScale(float sx, float sy, float px, float py)　//以坐标 px,py 进行缩放

void setTranslate(float dx, float dy) //平移

void setSkew (float kx, float ky, float px, float py) //以坐标 px,py 进行倾斜

void setSkew (float kx, float ky) //倾斜

二、android.graphics.NinePatch

NinePatch 是 Android 平台特有的一种非矢量图形自然拉伸处理方法，可以帮助常规的图形在拉伸时不会缩放，实例中 Android 开发网提示大家对于 Toast 的显示就是该原理，同时 SDK 中提供了一个工具名为 Draw 9-Patch，有关该工具的使用方法可以参考我们经发布的 Draw 9-Patch 使用方法介绍一文。由于该类提供了高质量支持透明的缩放方式，所以图形格式为 PNG，文件命名方式为.9.png 的后缀比如 android123.9.png。

三、android.graphics.Paint

Paint 类我们可以理解为画笔、画刷的属性定义，本类常用的方法如下:

void reset() //重置
void setARGB(int a, int r, int g, int b) 或 void setColor(int color) 均为设置 Paint 对象的颜色

void setAntiAlias(boolean aa) // 是否抗锯齿，需要配合 void setFlags (Paint.ANTI_ALIAS_FLAG) 来帮助消除锯齿使其边缘更平滑。

Shader setShader(Shader shader) //设置阴影，Shader 类是一个矩阵对象，如果为 NULL 将清除阴影。
void setStyle(Paint.Style style) //设置样式，一般为 FILL 填充，或者 STROKE 凹陷效果。
void setTextSize(float textSize) //设置字体大小
void setTextAlign(Paint.Align align) //文本对齐方式
Typeface setTypeface(Typeface typeface) //设置字体，通过 Typeface 可以加载 Android 内部的字体，一般为宋体对于中文，部分 ROM 可以自己添加比如雅黑等等
void setUnderlineText(boolean underlineText) //是否设置下划线，需要撇和 void setFlags (Paint.UNDERLINE_TEXT_FLAG) 方法。

四、android.graphics.Rect

Rect 我们可以理解为矩形区域，类似的还有 Point 一个点，Rect 类除了表示一个矩形区域位置描述外，android123 提示主要可以帮助我们计算图形之间是否碰撞(包含)关系，对于 Android 游戏开发比较有用，其主要的成员 contains 包含了三种重载方法，来判断包含关系

boolean contains(int left, int top, int right, int bottom)
boolean contains(int x, int y)
boolean contains(Rect r)

五、android.graphics.Region

Region 在 Android 平台中表示一个区域和 Rect 不同的是，它表示的是一个不规则的样子，可以是椭圆、多边形等等，而 Rect 仅仅是矩形。同样 Region 的 boolean contains(int x, int y) 成员可以判断一个点是否在该区域内

六、android.graphics.Typeface

Typeface 类是帮助描述一个字体对象，在 TextView 中通过使用 setTypeface 方法来制定一个输出文本的字体，其直接构造调用成员 create 方法可以直接指定一个字体名称和样式，比如

static Typeface create(Typeface family, int style)
static Typeface create(String familyName, int style)

同时使用 isBold 和 isItalic 方法可以判断出是否包含粗体或斜体的字型。

final boolean isBold()
final boolean isItalic()

该类的创建方法还有从 apk 的资源或从一个具体的文件路径，其具体方法为

static Typeface createFromAsset(AssetManager mgr, String path)
static Typeface createFromFile(File path)
static Typeface createFromFile(String path)

有关 Android 平台的图形、图像我们在前两节中已经整理出来，下次我们将首先讲述下 NinePatch 的实例应用。