从动画、绘图、视图三方面介绍Android自定义控件相关知识，
配以翔实的案例讲解每个知识点，既适合系统学习，又可以用于查漏补缺。

Broadview®
www.broadview.com.cn

Android
自定义控件开发入门与实战

启舰◎著

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

Android
自定义控件开发入门与实战

启舰◎著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

在 Android 中，官方提供的控件是非常有限的，而我们所面临的需求却是多样的。大家在工作中难免会接触到自定义控件的需求，但系统讲解自定义控件知识的书籍却少之又少。这不仅因为自定义控件涉及的知识丰富、繁杂，而且与动画和色彩相关的知识很难在纸张上表现出来。

本书从自定义控件的动画、绘图、视图三方面入手，分别讲解与自定义控件相关的各种知识，给大家系统地梳理相关知识点，并且通过翔实的案例讲解每个知识点在现实工作中所能实现的功能。

本书不仅适合 Android 初、中级水平从业者学习，也适合高水平从业者查漏补缺使用，还可以作为高校 Android 自定义控件方面的入门级教材。

图书在版编目（CIP）数据

Android 自定义控件开发入门与实战 / 启舰著. —北京：电子工业出版社，2018.7

ISBN 978-7-121-34556-2

Ⅰ. ①A… Ⅱ. ①启… Ⅲ. ①移动终端－应用程序－程序设计 Ⅳ. ①TN929.53

中国版本图书馆 CIP 数据核字（2018）第 135168 号

策划编辑：付　睿
责任编辑：牛　勇　　特约编辑：赵树刚
印　　刷：三河市良远印务有限公司
装　　订：三河市良远印务有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编：100036
开　　本：787×1092　1/16　　印张：31.5　　字数：806.4 千字
版　　次：2018 年 7 月第 1 版
印　　次：2018 年 7 月第 1 次印刷
定　　价：99.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件到 dbqq@phei. com.cn。

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

前言

在我刚入门 Android 的时候，就被各种自定义控件所吸引，但当真正想要自己去制作时，由于涉及的知识太多，所以根本无从下手。而且我在搜索网页时也发现，与自定义控件相关的知识非常少，大都是一些例子的源码，讲解的内容非常有限。从那时起，我便想，如果我学会了自定义控件，就要写一系列博文，把相关的知识梳理清楚，供后来者参考。

从 2015 年起，我便着重积累这方面的知识。从 2016 年 1 月起，我基本保持每两周一篇博文的频率在 CSDN 上公开发表。在不知不觉间，我已经连续更新了二十几篇博文，得到了很多朋友的喜欢和赞扬。我当初更新博客的目的很简单，一方面，能够梳理知识点，以防自己忘记；另一方面，能为后来者做一点事，希望大家在学习自定义控件时，不必像我这样费劲。

后来，电子工业出版社的付睿编辑联系到我，想让我把与自定义控件相关的知识整理成书。刚开始我是犹豫的，因为纸质媒介很难表现出自定义控件所特有的动画和色彩。为此，电子工业出版社给予了我很大的支持，在官网上添加博客功能，方便我的动态图片上传到后台，进而将图片地址制作成二维码，供大家扫描观看。这一突破性的想法解决了我的后顾之忧。非常感谢他们的支持！

我在阿里巴巴工作的时候，在时间上是非常紧张的，从每天早上 9 点到晚上 9 点是正常的上下班时间。为了写书，我每天早上保证 7 点到公司，写到 9 点，然后再回工位上班；周末基本上也都在准备资料、写代码、写书中度过。这让我原本非常紧张的生活变得更加紧张。

然而我又是一个不安分的人，我于 2017 年 4 月从阿里巴巴辞职，跟朋友一起去创业。创业路上的艰辛是我所没有预见的，原本不多的时间被瓜分得更是少之又少，只能每天熬夜写作。

非常感谢我的妻子聂倩，在这两年里，基本上没有时间陪你，是你的宽容与支持才有了这本书的成稿。同时，也要感谢我的小公主雯雯，如果不是你的到来，我就不会体会到为人父的快乐，是你让我在工作中充满了力量。感谢灰灰，从创业开始就随我四处奔波，不离不弃。感谢博哥，在公司最困难的时候，选择留下来共渡难关。感谢你们在公司走上正轨后，为我承担了工作中大部分的责任，让我能安心地完成本书。

本书开篇主要讲解了入门自定义控件所需的一些必备知识；在动画篇中，详细讲解了在 Android 中制作动画的几种方法；在绘图篇中，具体讲解了与绘图相关的知识；在视图篇中，

主要讲解了控件本身所涉及的一些知识。

在写作过程中，我尽量做到两点。第一，讲通、讲透。以我的理解，讲解出相关的知识所涉及的方方面面，力争让大家不再需要自己找资料，就可以全面理解这些知识。当然，本书中的有些内容在网上是找不到的，都是根据个人经验而得出的结论，难免有所偏失，如有不足，还望指正。第二，实例交织。我尽量在每个知识点中都加入一些实战中的例子，方便大家理解。

为了做到这两点，本书内容非常多，我把相对不重要的内容迁移到网上，大家可以到网上继续阅读。同时，本书的前后章节是经过严格推敲的，大家切勿跳章学习，必须按照顺序逐步进行。虽然我会给大家提供源码，但是请大家自己把代码敲一遍，因为只有动手写过的东西，才真正是自己的。

我在更新博客时，喜欢在每篇博文前加一句序言来激励自己。在本书中，我在每章前仍会加一句序言。本书第 1 章的序言是我非常喜欢的一句话，送给大家：迷茫，本就是青春该有的样子，但不要让未来的你讨厌现在的自己。

目录

绘 图 篇

视 图 篇

开　篇

自定义控件是 Android 系统中一个非常重要的特性，通过自定义控件可以使生硬的界面变得生动活泼。每个应用或多或少都会有几个乃至几十个自定义控件。

想要自如地自定义控件，必须系统地掌握动画、绘图、Android 原生控件和系统的重要特性等方面的知识。有时候，必须从源码的角度来进行分析，才能达到知其所以然的境界。

本书将分别在绘图、动画、系统特性、原生控件等方面讲述自定义控件的方法，并且会在不同的章节中根据当前所学内容举一些现实中的例子来加深理解。现在我们就一步步走入自定义控件的世界吧。

第 1 章

绘图基础

迷茫，本就是青春该有的样子，但不要让未来的你讨厌现在的自己。

本章作为开篇第 1 章，主要讲解有关自定义控件系列的一些基础知识，是后续各章节的根基。

由于本书初始成稿时已经近 700 页，而这么厚的书定价比较高，因此对第 1 章的内容进行了删减，只保留核心的部分。不过本章完整版本提供了网络下载，读者可登录博文视点本书页面自行下载，其中详细介绍了有关绘图基础知识的内容。

1.1 基本图形绘制

1.1.1 概述

我们平时画图需要两个工具：纸和笔。在 Android 中，Paint 类就是画笔，而 Canvas 类就是纸，在这里叫作画布。

所以，凡是跟画笔设置相关的，比如画笔大小、粗细、画笔颜色、透明度、字体的样式等，都在 Paint 类里设置；同样，凡是要画出成品的东西，比如圆形、矩形、文字等，都要调用 Canvas 类里的函数生成。

下面通过一个自定义控件的例子来看一下如何生成自定义控件，以及 Paint 和 Canvas 类的用法。

（1）新建一个工程，然后写一个类派生自 View。

```
package com.harvic.PaintBasis;

import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.util.AttributeSet;
```

```
import android.view.View;
public class BasisView extends View {
    public BasisView(Context context) {
        super(context);
    }

    public BasisView(Context context, AttributeSet attrs) {
        super(context, attrs);
    }

    public BasisView(Context context, AttributeSet attrs, int defStyle) {
        super(context, attrs, defStyle);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        super.onDraw(canvas);

        //设置画笔的基本属性
        Paint paint=new Paint();
        paint.setColor(Color.RED);                  //设置画笔颜色
        paint.setStyle(Paint.Style.STROKE);         //设置填充样式
        paint.setStrokeWidth(50);                   //设置画笔宽度

        //画圆
        canvas.drawCircle(190, 200, 150, paint);
    }
}
```

代码很简单，首先，写一个类派生自 View。派生自 View 表示当前是一个自定义控件，类似 Button、TextView 这些控件都是派生自 View 的。如果我们想像 LinearLayout、RelativeLayout 这样生成一个容器，则需要派生自 ViewGroup。有关 ViewGroup 的知识，我们会在后面的章节中讲述。

其次，重写 onDraw(Canvas canvas)函数。可以看到，在该函数中，入参是一个 Canvas 对象，也就是当前控件的画布，所以我们只要调用 Canvas 的绘图函数，效果就可以直接显示在控件上了。

在 onDraw(Canvas canvas)函数中，我们设置了画笔的基本属性。

```
Paint paint=new Paint();
paint.setColor(Color.RED);                  //设置画笔颜色
paint.setStyle(Paint.Style.STROKE);         //设置填充样式
paint.setStrokeWidth(50);                   //设置画笔宽度
```

在这里，我们将画笔设置成红色，填充样式为描边，并且将画笔的宽度设置为 50px（有关这些属性的具体含义，会在后面一一讲述）。

最后，我们利用 canvas.drawCircle(190, 200, 150, paint);语句画了一个圆。需要注意的是，画圆所用的画笔就是我们在这里指定的。

（2）使用自定义控件。

我们可以直接在主布局中使用自定义控件（main.xml）。

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
              android:orientation="vertical"
              android:layout_width="fill_parent"
              android:layout_height="fill_parent">

    <com.harvic.PaintBasis.BasisView
              android:layout_width="match_parent"
              android:layout_height="match_parent"/>
</LinearLayout>
```

可以看到，在 XML 中使用自定义控件时，需要使用完整的包名加类包的方式来引入。注意，这里的布局方式使用的全屏方式。后面会讲到如何给自定义控件使用 wrap_content 属性，而目前我们全屏显示控件即可。

效果如下图所示。

从这里可以看到，只需要先创建一个派生自 View 的类，再重新在 onDraw()函数中设置 Paint 并调用 Canvas 的一些绘图函数，就可以画出我们想要的图形。由此看来，自定义控件并不复杂。下面分别来看如何设置画笔，以及 Canvas 中一些常用的绘图函数。

1.1.2 画笔的基本设置

下面初步讲一下 1.1.1 节的示例中所涉及的几个函数。

1. setColor()

该函数的作用是设置画笔颜色，完整的函数声明如下：

```
void setColor(int color)
```

我们知道，一种颜色是由红、绿、蓝三色合成出来的，所以参数 color 只能取 8 位的 0xAARRGGBB 样式颜色值。

其中：

- A 代表透明度（Alpha），取值范围是 0～255（对应十六进制数 0x00～0xFF），取值越小，透明度越高，图像也就越透明。当取 0 时，图像完全不可见。
- R 代表红色值（Red），取值范围是 0～255（对应十六进制数 0x00～0xFF），取值越小，红色越少。当取 0 时，表示红色完全不可见；当取 255 时，红色完全显示。