

2.4.2 Date & Time组件(上)

分类 [Android 基础入门教程](#)

Android 基础入门教程(Q群号 : 153836263)

本节引言：

本节给大家带来的是Android给我们提供的显示时间的几个控件，他们分别是：TextClock，AnalogClock，Chronometer，另外其实还有个过时的DigitalClock就不讲解了！好的，开始本节内容！

1.TextClock(文本时钟)

TextClock是在Android 4.2(API 17)后推出的用来替代DigitalClock的一个控件！

TextClock可以以字符串格式显示当前的日期和时间，因此推荐在Android 4.2以后使用TextClock。

这个控件推荐在24进制的android系统中使用，TextClock提供了两种不同的格式，一种是在24进制中显示时间和日期，另一种是在12进制中显示时间和日期。大部分人喜欢默认的设置。

可以通过调用：TextClock提供的is24HourModeEnabled()方法来查看，系统是否在使用24进制时间显示！在24进制模式中：

如果没获取时间，首先通过getFormat24Hour()返回值；

获取失败则通过getFormat12Hour()获取返回值；

以上都获取失败则使用默认；

另外他给我们提供了下面这些方法，对应的还有get方法：

Attribute Name	Related Method	Description
android:format12Hour	setFormat12Hour(CharSequence)	设置12时制的格式
android:format24Hour	setFormat24Hour(CharSequence)	设置24时制的格式
android:timeZone	setTimeZone(String)	设置时区

其实更多的时间我们是花在时间形式定义上，就是里面这个CharSequence！这里提供下常用的写法以及结果：

```
<TextClock
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:format12Hour="MM/dd/yy h:mmaa"/>
</TextClock>
```

1.0 Android基础入门教程

1.0.1 2015年最新Android基...

1.1 背景相关与系统架构分析

1.2 开发环境搭建

1.2.1 使用Eclipse + ADT + S...

1.2.2 使用Android Studio开...

1.3 SDK更新不了问题解决

1.4 Genymotion模拟器安装

1.5.1 Git使用教程之本地仓...

1.5.2 Git之使用GitHub搭建...

1.6 .9(九妹)图片怎么玩

1.7 界面原型设计

1.8 工程相关解析(各种文件...

1.9 Android程序签名打包

1.11 反编译APK获取代码&...

2.1 View与ViewGroup的概念

2.2.1 LinearLayout(线性布局)

2.2.2 RelativeLayout(相对布...

2.2.3 TableLayout(表格布局)

2.2.4 FrameLayout(帧布局)

2.2.5 GridLayout(网格布局)

2.2.6 AbsoluteLayout(绝对...

2.3.1 TextView(文本框)详解

2.3.2 EditText(输入框)详解

2.3.3 Button(按钮)与ImageB...

2.3.4 ImageView(图像视图)

2.3.5.RadioButton(单选按钮...

2.3.6 开关按钮ToggleButton...

2.3.7 ProgressBar(进度条)

2.3.8 SeekBar(拖动条)

2.3.9 RatingBar(星级评分条)

2.4.1 ScrollView(滚动条)

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:format12Hour="MMM dd, yyyy h:mmaa"/>

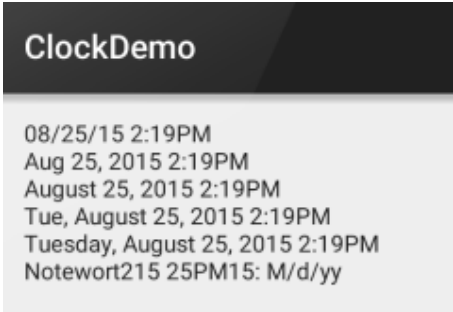
<TextClock
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:format12Hour="MMMM dd, yyyy h:mmaa"/>

<TextClock
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:format12Hour="E, MMMM dd, yyyy h:mmaa"/>

<TextClock
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:format12Hour="EEEE, MMMM dd, yyyy h:mmaa"/>

<TextClock
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:format12Hour="Noteworthy day: 'M/d/yy"/>
```

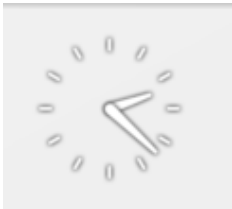
运行结果：



PS:另外minsdk 要大于或者等于17哦！

2.AnalogClock(模拟时钟)

就是下图这种：



官网中我们可以看到这样三个属性：

XML Attributes
Attribute Name
android:dial
android:hand_hour
android:hand_minute

依次是：表背景，表时针，分时针的图片，我们可以自行定制：

示例代码如下：

2.4.2 Date & Time组件(上)

2.4.3 Date & Time组件(下)

2.4.4 Adapter基础讲解

2.4.5 ListView简单实用

2.4.6 BaseAdapter优化

2.4.7ListView的焦点问题

2.4.8 ListView之checkbox错...

2.4.9 ListView的数据更新问题

2.5.0 构建一个可复用的自定...

2.5.1 ListView Item多布局的...

2.5.2 GridView(网格视图)的...

2.5.3 Spinner(列表选项框)...

2.5.4 AutoCompleteTextVie...

2.5.5 ExpandableListView(...

2.5.6 ViewFlipper(翻转视图)...

2.5.7 Toast(吐司)的基本使用

2.5.8 Notification(状态栏通...

2.5.9 AlertDialog(对话框)详解

2.6.0 其他几种常用对话框基...

2.6.1 PopupWindow(悬浮框...

2.6.2 菜单(Menu)

2.6.3 ViewPager的简单使用

2.6.4 DrawerLayout(官方侧...

3.1.1 基于监听的事件处理机制

3.2 基于回调的事件处理机制

3.3 Handler消息传递机制浅析

3.4 TouchListener PK OnTo...

3.5 监听EditText的内容变化

3.6 响应系统设置的事件(Co...

3.7 AsyncTask异步任务

3.8 Gestures(手势)

4.1.1 Activity初学乍练

4.1.2 Activity初窥门径

4.1.3 Activity登堂入室

4.2.1 Service初涉

4.2.2 Service进阶

4.2.3 Service精通

4.3.1 BroadcastReceiver牛...

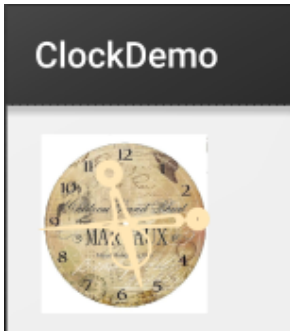
4.3.2 BroadcastReceiver庖...

4.4.1 ContentProvider初探

<AnalogClock

```
android:layout_width="100dp"
android:layout_height="100dp"
android:dial="@mipmap/ic_c_bg"
android:hand_hour="@mipmap/zhen_shi"
android:hand_minute="@mipmap/zhen_fen" />
```

运行结果：



3.Chronometer(计时器)

如题，就是一个简单的计时器，我们直接上使用示例吧：

使用示例：

实现代码：

布局代码：

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res
/android"
```

```
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".MainActivity">
```

<Chronometer

```
android:id="@+id/chronometer"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:gravity="center"
android:textColor="#ff0000"
android:textSize="60dip" />
```

<LinearLayout

```
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:layout_margin="10dip"
android:orientation="horizontal">
```

<Button

```
android:id="@+id/btnStart"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
```

4.4.2 ContentProvider再探...

4.5.1 Intent的基本使用

4.5.2 Intent之复杂数据的传递

5.1 Fragment基本概述

5.2.1 Fragment实例精讲—...

5.2.2 Fragment实例精讲—...

5.2.3 Fragment实例精讲—...

5.2.4 Fragment实例精讲—...

5.2.5 Fragment实例精讲—...

6.1 数据存储与访问之——文...

6.2 数据存储与访问之——S...

6.3.1 数据存储与访问之——...

6.3.2 数据存储与访问之——...

7.1.1 Android网络编程要学...

7.1.2 Android Http请求头与...

7.1.3 Android HTTP请求方...

7.1.4 Android HTTP请求方...

7.2.1 Android XML数据解析

7.2.2 Android JSON数据解析

7.3.1 Android 文件上传

7.3.2 Android 文件下载 (1)

7.3.3 Android 文件下载 (2)

7.4 Android 调用 Webservice

7.5.1 WebView(网页视图)基...

7.5.2 WebView和JavaScrip...

7.5.3 Android 4.4后WebVie...

7.5.4 WebView文件下载

7.5.5 WebView缓存问题

7.5.6 WebView处理网页返...

7.6.1 Socket学习网络基础准备

7.6.2 基于TCP协议的Socket...

7.6.3 基于TCP协议的Socket...

7.6.4 基于UDP协议的Socke...

8.1.1 Android中的13种Draw...

8.1.2 Android中的13种Draw...

8.1.3 Android中的13种Draw...

8.2.1 Bitmap(位图)全解析 P...

8.2.2 Bitmap引起的OOM问题

8.3.1 三个绘图工具类详解

8.3.2 绘图类实战示例

```

        android:layout_weight="1"
        android:text="开始计时" />

<Button
    android:id="@+id/btnStop"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="停止计时" />

<Button
    android:id="@+id/btnReset"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="重置" />

<Button
    android:id="@+id/btn_format"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="格式化" />

</LinearLayout>

</LinearLayout>

```

MainActivity.java

```

public class MainActivity extends AppCompatActivity implements
    View.OnClickListener, Chronometer.OnChronometerTickListener {

    private Chronometer chronometer;
    private Button btn_start, btn_stop, btn_base, btn_format;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initView();
    }

    private void initView() {
        chronometer = (Chronometer) findViewById(R.id.chronomet
er);

        btn_start = (Button) findViewById(R.id.btnStart);
        btn_stop = (Button) findViewById(R.id.btnStop);
        btn_base = (Button) findViewById(R.id.btnReset);
        btn_format = (Button) findViewById(R.id.btn_format);

        chronometer.setOnChronometerTickListener(this);
        btn_start.setOnClickListener(this);
        btn_stop.setOnClickListener(this);
        btn_base.setOnClickListener(this);
    }
}

```

8.3.3 Paint API之—— Mask...

8.3.4 Paint API之—— Xferm...

8.3.5 Paint API之—— Xferm...

8.3.6 Paint API之—— Xferm...

8.3.7 Paint API之—— Xferm...

8.3.8 Paint API之—— Xferm...

8.3.9 Paint API之—— Color...

8.3.10 Paint API之—— Colo...

8.3.11 Paint API之—— Colo...

8.3.12 Paint API之—— Path...

8.3.13 Paint API之—— Sha...

8.3.14 Paint几个枚举/常量值...

8.3.15 Paint API之——Type...

8.3.16 Canvas API详解(Part 1)

8.3.17 Canvas API详解(Part...

8.3.18 Canvas API详解(Part...

8.4.1 Android动画合集之帧...

8.4.2 Android动画合集之补...

8.4.3 Android动画合集之属...

8.4.4 Android动画合集之属...

9.1 使用SoundPool播放音...

9.2 MediaPlayer播放音频与...

9.3 使用Camera拍照

9.4 使用MediaRecord录音

10.1 TelephonyManager(电...

10.2 SmsManager(短信管理...

10.3 AudioManager(音频管...

10.4 Vibrator(振动器)

10.5 AlarmManager(闹钟服务)

10.6 PowerManager(电源服...

10.7 WindowManager(窗口...

10.8 LayoutInflater(布局服务)

10.9 WallpaperManager(壁...

10.10 传感器专题(1)——相...

10.11 传感器专题(2)——方...

10.12 传感器专题(3)——加...

10.12 传感器专题(4)——其...

10.14 Android GPS初涉

11.0 《2015最新Android基...

```
        btn_format.setOnClickListener(this);

    }

    @Override
    public void onClick(View v) {
        switch (v.getId()){
            case R.id.btnStart:
                chronometer.start();// 开始计时
                break;
            case R.id.btnStop:
                chronometer.stop();// 停止计时
                break;
            case R.id.btnReset:
                chronometer.setBase(SystemClock.elapsedRealtime
格式
()); // 复位
                break;
            case R.id.btn_format:
                chronometer.setFormat("Time: %s");// 更改时间显示
                break;
        }
    }

    @Override
    public void onChronometerTick(Chronometer chronometer) {
        String time = chronometer.getText().toString();
        if(time.equals("00:00")){
            Toast.makeText(MainActivity.this, "时间到了~", Toast.L
ENGTH_SHORT).show();
        }
    }
}
```

运行截图：



本节小结：

本节跟大家简单的介绍了TextClock，AnalogClock，Chronometer这三个组件，从篇幅就可以看出 其实这几个东西用得并不多，几乎是没用过...知道下就好，用法也超简单... 就这样吧，本节就到这里~谢谢

[← 2.4.1 ScrollView\(滚动条\)](#)[2.4.3 Date & Time组件\(下\) →](#)

最省心的支付SDK
3行代码实现APP/网页支付



在线实例

- [HTML 实例](#)
- [CSS 实例](#)
- [JavaScript 实例](#)

字符集&工具

- [HTML 字符集设置](#)
- [HTML ASCII 字符集](#)

最新更新

- [PHP接收 json，并...](#)
- [Foundation CSS ...](#)

站点信息

- [意见反馈](#)
- [免责声明](#)
- [关于我们](#)

<ul style="list-style-type: none">· Ajax 实例· jQuery 实例· XML 实例· Java 实例	<ul style="list-style-type: none">· HTML ISO-8859-1· HTML 实体符号· HTML 拾色器· JSON 格式化工具	<ul style="list-style-type: none">· Foundation CSS ...· Foundation 图标...· Foundation 网格...· Foundation 块状...· Foundation 网格...	<ul style="list-style-type: none">· 文章归档
--	---	--	--

关注微信



Copyright © 2013-2015 菜鸟教程 **runoob.com** All Rights Reserved. 备案号：闽ICP备15012807号-1