

2.5.9 AlertDialog(对话框)详解

分类 [Android 基础入门教程](#)

Android 基础入门教程(Q群号 : 153836263)

本节引言：

本节继续给大家带来是显示提示信息的第三个控件AlertDialog(对话框)，同时它也是其他 Dialog的父类！比如ProgressDialog，TimePickerDialog等，而AlertDialog的父类是：Dialog！另外，不像前面学习的Toast和Notification，AlertDialog并不能直接new出来，如果你打开 AlertDialog的源码，会发现构造方法是protected的，如果我们要创建AlertDialog的话，我们需要使用到该类中的一个静态内部类：public static class **Builder**，然后来调用AlertDialog 里的相关方法，来对AlertDialog进行定制，最后调用show()方法来显示我们的AlertDialog对话框！好的，下面我们就来学习AlertDialog的基本用法，以及定制我们的AlertDialog！官方文档：[AlertDialog](#)

1.基本使用流程

Step 1：创建AlertDialog.Builder对象；

Step 2：调用setIcon()设置图标，setTitle()或setCustomTitle()设置标题；

Step 3：设置对话框的内容：setMessage()还有其它方法来指定显示的内容；

Step 4：调用setPositive/Negative/NeutralButton()设置：确定，取消，中立按钮；

Step 5：调用create()方法创建这个对象，再调用show()方法将对话框显示出来；

2.几种常用的对话框使用示例

运行效果图：

- 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)



核心代码：

MainActivity.java：

```
public class MainActivity extends AppCompatActivity implements
View.OnClickListener {

    private Button btn_dialog_one;
    private Button btn_dialog_two;
    private Button btn_dialog_three;
    private Button btn_dialog_four;

    private Context mContext;
    private boolean[] checkItems;

    private AlertDialog alert = null;
    private AlertDialog.Builder builder = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mContext = MainActivity.this;
        bindView();
    }

    private void bindView() {
        btn_dialog_one = (Button) findViewById(R.id.btn_dialog_
one);
        btn_dialog_two = (Button) findViewById(R.id.btn_dialog_
two);
        btn_dialog_three = (Button) findViewById(R.id.btn_dialo
g_three);
```

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

2.4.7 ListView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

```

        btn_dialog_four = (Button) findViewById(R.id.btn_dialog
_four);

        btn_dialog_one.setOnClickListener(this);
        btn_dialog_two.setOnClickListener(this);
        btn_dialog_three.setOnClickListener(this);
        btn_dialog_four.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {
        switch (v.getId()) {
            //普通对话框
            case R.id.btn_dialog_one:
                alert = null;
                builder = new AlertDialog.Builder(mContext);
                alert = builder.setIcon(R.mipmap.ic_icon_fish)
                    .setTitle("系统提示: ")
                    .setMessage("这是一个最普通的AlertDialog,
\n带有三个按钮, 分别是取消, 中立和确定")
                    .setNegativeButton("取消", new DialogInterface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface
dialog, int which) {
                            Toast.makeText(mContext, "你点
击了取消按钮~", Toast.LENGTH_SHORT).show();
                        }
                    })
                    .setPositiveButton("确定", new DialogInterface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface
dialog, int which) {
                            Toast.makeText(mContext, "你点
击了确定按钮~", Toast.LENGTH_SHORT).show();
                        }
                    })
                    .setNeutralButton("中立", new DialogInt
erface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface
dialog, int which) {
                            Toast.makeText(mContext, "你点
击了中立按钮~", Toast.LENGTH_SHORT).show();
                        }
                    })
                    .create(); //创建AlertDia
log对象

                alert.show(); //显示对话框
                break;
            //普通列表对话框
            case R.id.btn_dialog_two:
                final String[] lesson = new String[] {"语文", "

```

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

```

数学", "英语", "化学", "生物", "物理", "体育"};

        alert = null;
        builder = new AlertDialog.Builder(mContext);
        alert = builder.setIcon(R.mipmap.ic_icon_fish)
            .setTitle("选择你喜欢的课程")
            .setItems(lesson, new DialogInterface.O

onClickListener() {

            @Override
            public void onClick(DialogInterface

dialog, int which) {

                Toast.makeText(getApplicationContextCo
ntext(), "你选择了" + lesson[which], Toast.LENGTH_SHORT).show();
            }
        }).create();
        alert.show();
        break;
//单选列表对话框
        case R.id.btn_dialog_three:
            final String[] fruits = new String[]{"苹果", "
雪梨", "香蕉", "葡萄", "西瓜"};
            alert = null;
            builder = new AlertDialog.Builder(mContext);
            alert = builder.setIcon(R.mipmap.ic_icon_fish)
                .setTitle("选择你喜欢的水果，只能选一个哦~"

                .setSingleChoiceItems(fruits, 0, new Di
alogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface

dialog, int which) {

                        Toast.makeText(getApplicationContextCo
ntext(), "你选择了" + fruits[which], Toast.LENGTH_SHORT).show();
                    }
                }).create();
                alert.show();
                break;
//多选列表对话框
        case R.id.btn_dialog_four:
            final String[] menu = new String[]{"水煮豆腐", "
萝卜牛腩", "酱油鸡", "胡椒猪肚鸡"};
            //定义一个用来记录个列表项状态的boolean数组
            checkItems = new boolean[]{false, false, false,
false};

            alert = null;
            builder = new AlertDialog.Builder(mContext);
            alert = builder.setIcon(R.mipmap.ic_icon_fish)
                .setMultiChoiceItems(menu, checkItems,
new DialogInterface.OnMultiChoiceClickListener() {
                    @Override
                    public void onClick(DialogInterface

dialog, int which, boolean isChecked) {

                        checkItems[which] = isChecked;
                    }
                })

```

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...

```
        .setPositiveButton("确定", new DialogInterface.OnClickListener() {  
            @Override  
            public void onClick(DialogInterface dialog, int which) {  
                String result = "";  
                for (int i = 0; i < checkItems.  
length; i++) {  
                    if (checkItems[i])  
                        result += menu[i] + " "  
;   
                }  
                Toast.makeText(getApplicationContext()  
ntext(), "客官你点了:" + result, Toast.LENGTH_SHORT).show();  
            }  
        })  
        .create();  
        alert.show();  
        break;  
    }  
}
```

布局就是四个简单的按钮，这里就不贴出来了，用法非常简单~无非就是创建一个Builder对象后，进行相关设置，然后create()生成一个AlertDialog对象，最后调用show()方法将AlertDialog 显示出来而已！另外，细心的你可能发现我们点击对话框的外部区域，对话框就会消失，我们可以为builder设置setCancelable(false)即可解决这个问题！

3.通过Builder的setView()定制显示的AlertDialog

我们可以自定义一个与系统对话框不同的布局，然后调用setView()将我们的布局加载到 AlertDialog上，上面我们来实现这个效果：

运行效果图：



关键代码：

首先是两种不同按钮的selector的drawable文件：

btn_selctor_exit.xml：

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:state_pressed="true" android:drawable="@mipmap/iv_icon_exit_pressed"/>
    <item android:drawable="@mipmap/iv_icon_exit_normal"/>
</selector>
```

btn_selctor_choose.xml：

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:state_pressed="true" android:drawable="@mipmap/bg_btn_pressed"/>
    <item android:drawable="@mipmap/bg_btn_normal"/>
</selector>
```

接着是自定义的Dialog布局：**view_dialog_custom.xml**：

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/RelativeLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
```

<RelativeLayout

```
    android:id="@+id/titlelayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:background="#53CC66"
    android:padding="5dp">
```

<TextView

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:text="提示信息"
    android:textColor="#ffffff"
    android:textSize="18sp"
    android:textStyle="bold" />
```

<Button

```
    android:id="@+id/btn_cancle"
    android:layout_width="30dp"
    android:layout_height="30dp"
    android:layout_alignParentRight="true"
    android:background="@drawable/btn_selctor_exit" />
```

</RelativeLayout>**<LinearLayout**

```
    android:id="@+id/ly_detail"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/titlelayout"
    android:layout_centerInParent="true"
    android:orientation="vertical">
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginTop="20dp"
    android:text="通过setView()方法定制AlertDialog"
    android:textColor="#04AEDA"
    android:textSize="18sp" />
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginTop="10dp"
    android:text="作者:Coder-pig"
    android:textColor="#04AEDA"
    android:textSize="18sp" />
```

```
</LinearLayout>
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ly_detail"
    android:layout_marginTop="10dp"
    android:orientation="horizontal">
```

```
<Button
```

```
    android:id="@+id/btn_blog"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:layout_margin="5dp"
    android:layout_weight="1"
    android:background="@drawable/btn_selector_choose"
    android:text="访问博客"
    android:textColor="#ffffff"
    android:textSize="20sp" />
```

```
<Button
```

```
    android:id="@+id/btn_close"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:layout_margin="5dp"
    android:layout_weight="1"
    android:background="@drawable/btn_selector_choose"
    android:text="关闭"
    android:textColor="#ffffff"
    android:textSize="20sp" />
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

最后是MainActivity.java :

```
public class MainActivity extends AppCompatActivity {

    private Button btn_show;
    private View view_custom;
    private Context mContext;
    private AlertDialog alert = null;
    private AlertDialog.Builder builder = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mContext = MainActivity.this;
        btn_show = (Button) findViewById(R.id.btn_show);

        //初始化Builder
```




```
builder = new AlertDialog.Builder(mContext);

//加载自定义的那个View,同时设置下
final LayoutInflater inflater = MainActivity.this.getLayoutInflater();
view_custom = inflater.inflate(R.layout.view_dialog_custom, null, false);
builder.setView(view_custom);
builder.setCancelable(false);
alert = builder.create();

view_custom.findViewById(R.id.btn_cancel).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        alert.dismiss();
    }
});

view_custom.findViewById(R.id.btn_blog).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(getApplicationContext(), "访问博客", Toast.LENGTH_SHORT).show();
        Uri uri = Uri.parse("http://blog.csdn.net/coder_pig");
        Intent intent = new Intent(Intent.ACTION_VIEW, uri);
        startActivity(intent);
        alert.dismiss();
    }
});

view_custom.findViewById(R.id.btn_close).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(getApplicationContext(), "对话框已关闭~", Toast.LENGTH_SHORT).show();
        alert.dismiss();
    }
});

btn_show.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        alert.show();
    }
});
}
```

4.示例代码下载

[AlertDialogDemo.zip](#)

[AlertDialogDemo1.zip](#)

本节小结：

好的，本节给大家介绍了一下AlertDialog的基本使用，写了几个调用AlertDialog的例子，最后还通过setView方法自定义了一下我们的AlertDialog！是不是还意犹未尽呢？但这说不上真正的自定义控件，我们把自定义控件放到进阶系列，到时后会有个专题来和大家探讨下自定义控件~敬请期待~就说这么多，谢谢~

← 2.5.8 Notification(状态栏通知)详解 2.6.0 其他几种常用对话框基本使用 →



在线实例

- HTML 实例
- CSS 实例
- JavaScript 实例
- Ajax 实例
- jQuery 实例
- XML 实例
- Java 实例

字符集&工具

- HTML 字符集设置
- HTML ASCII 字符集
- HTML ISO-8859-1
- HTML 实体符号
- HTML 拾色器
- JSON 格式化工具

最新更新

- PHP接收 json，并...
- Foundation CSS ...
- Foundation CSS ...
- Foundation 图标...
- Foundation 网格...
- Foundation 块状...
- Foundation 网格...

站点信息

- 意见反馈
- 免责声明
- 关于我们
- 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