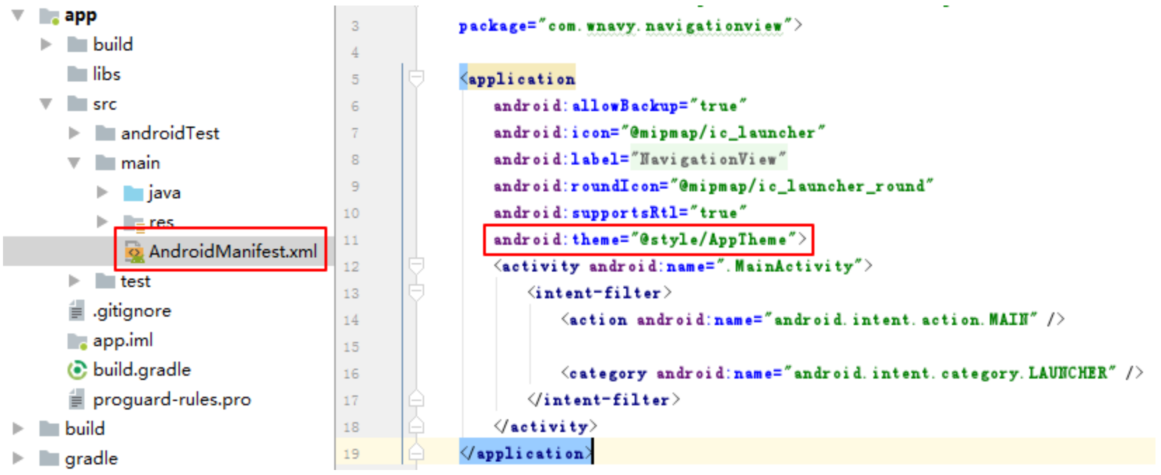
Material Design 学习笔记

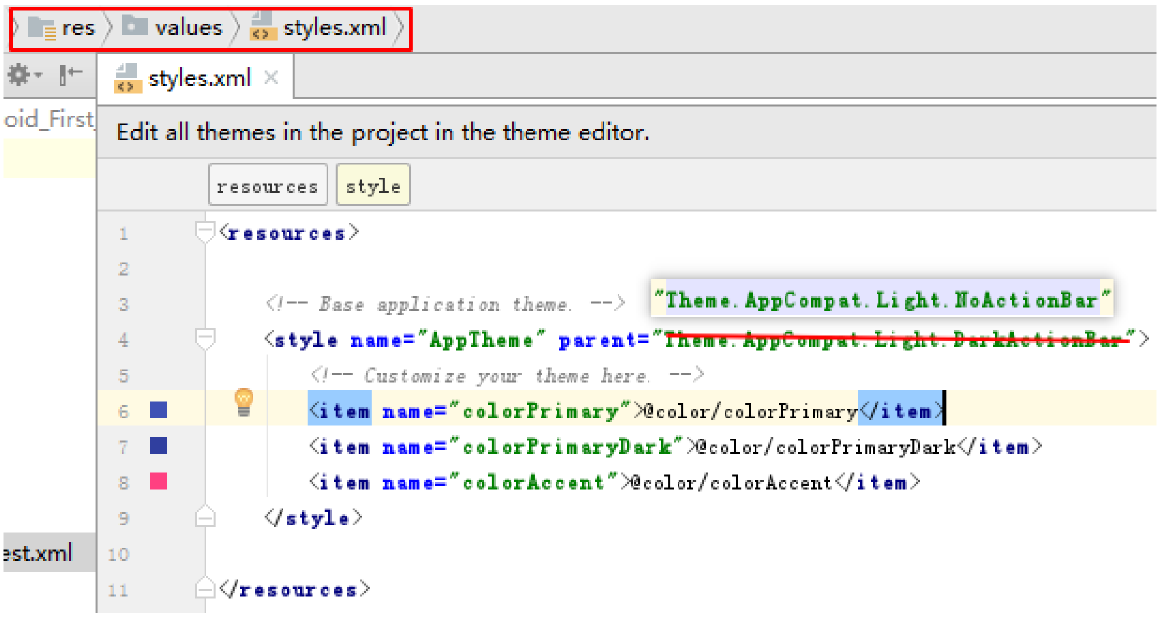
2018 年 01 月 25 日
上午 09:19

ToolBar

1,为 AppTheme 指定不带 ActionBar 的主题:



```
package="com.wnavy.navigationview">

<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="NavigationView"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".MainActivity">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</application>
```



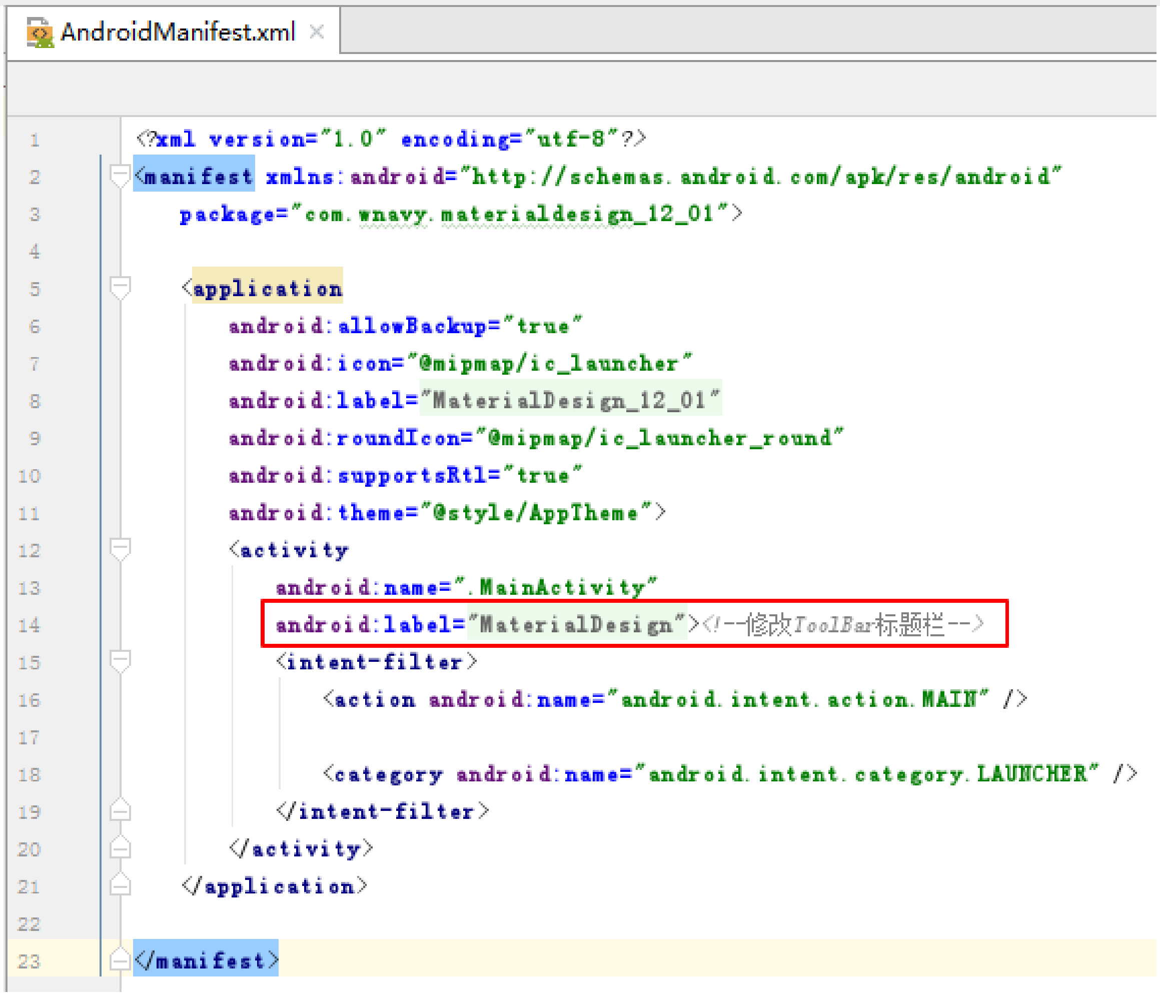
Edit all themes in the project in the theme editor.

```
<resources>

    <!-- Base application theme. -->  "Theme.AppCompat.Light.NoActionBar"
    <style name="AppTheme" parent="~~Theme.AppCompat.Light.DarkActionBar~~">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>

</resources>
```

2,使用 ToolBar 代替 ActionBar:

```
<!--main_activity.xml-->
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <android.support.v7.widget.Toolbar
        android:id="@+id/toolbar"

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="?attr/colorPrimary"
        android:minHeight="?attr/actionBarSize"
        android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
        app:popupTheme="@style/ThemeOverlay.AppCompat.Light" />
</FrameLayout>
```

```
public class MainActivity extends AppCompatActivity{
    Toolbar toolBar;


    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        toolBar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolBar);
    }
}
```



AndroidManifest.xml

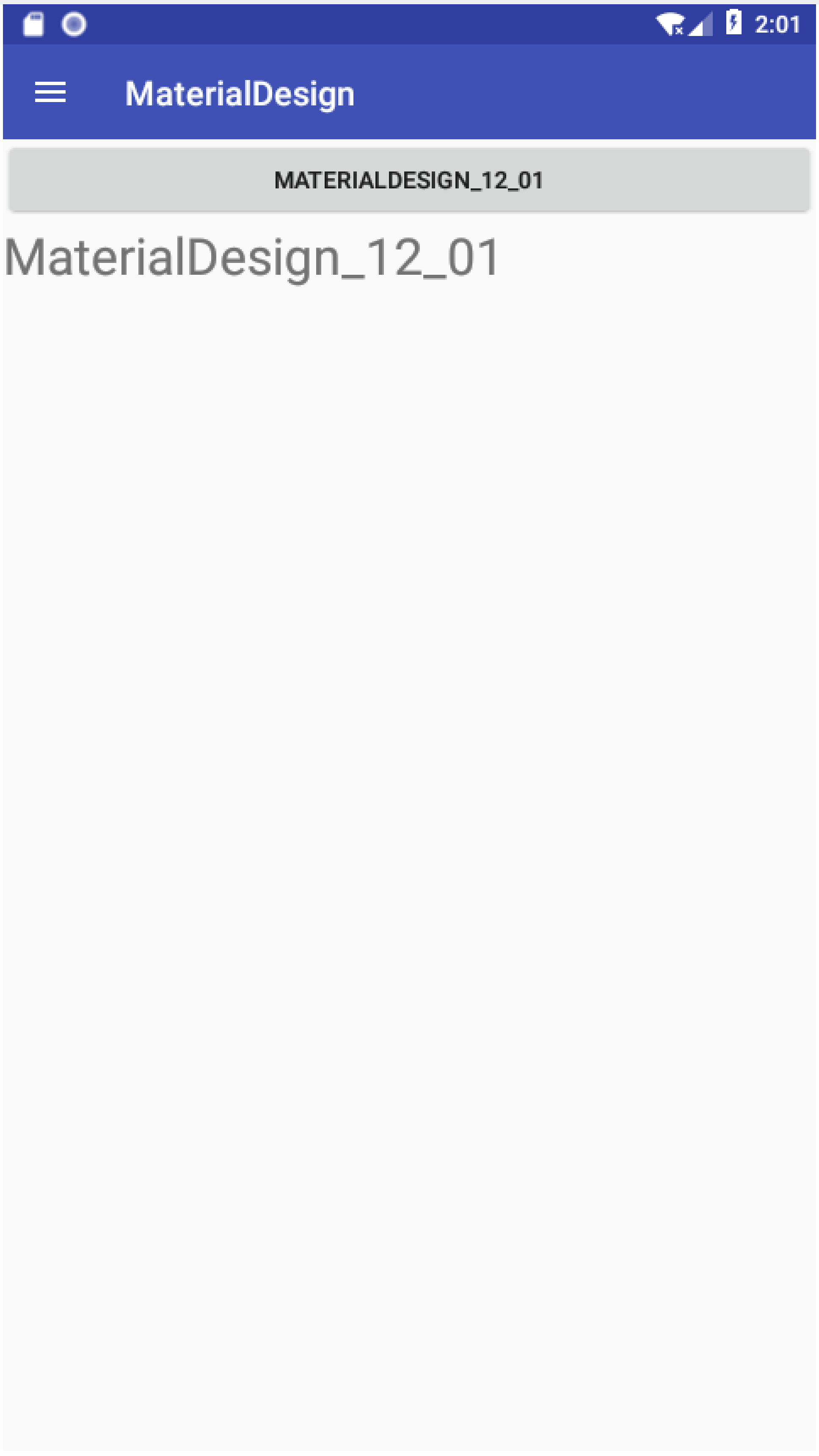
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.wnavy.materialdesign_12_01">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="MaterialDesign_12_01"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="MaterialDesign"><!--修改ToolBar标题栏-->
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

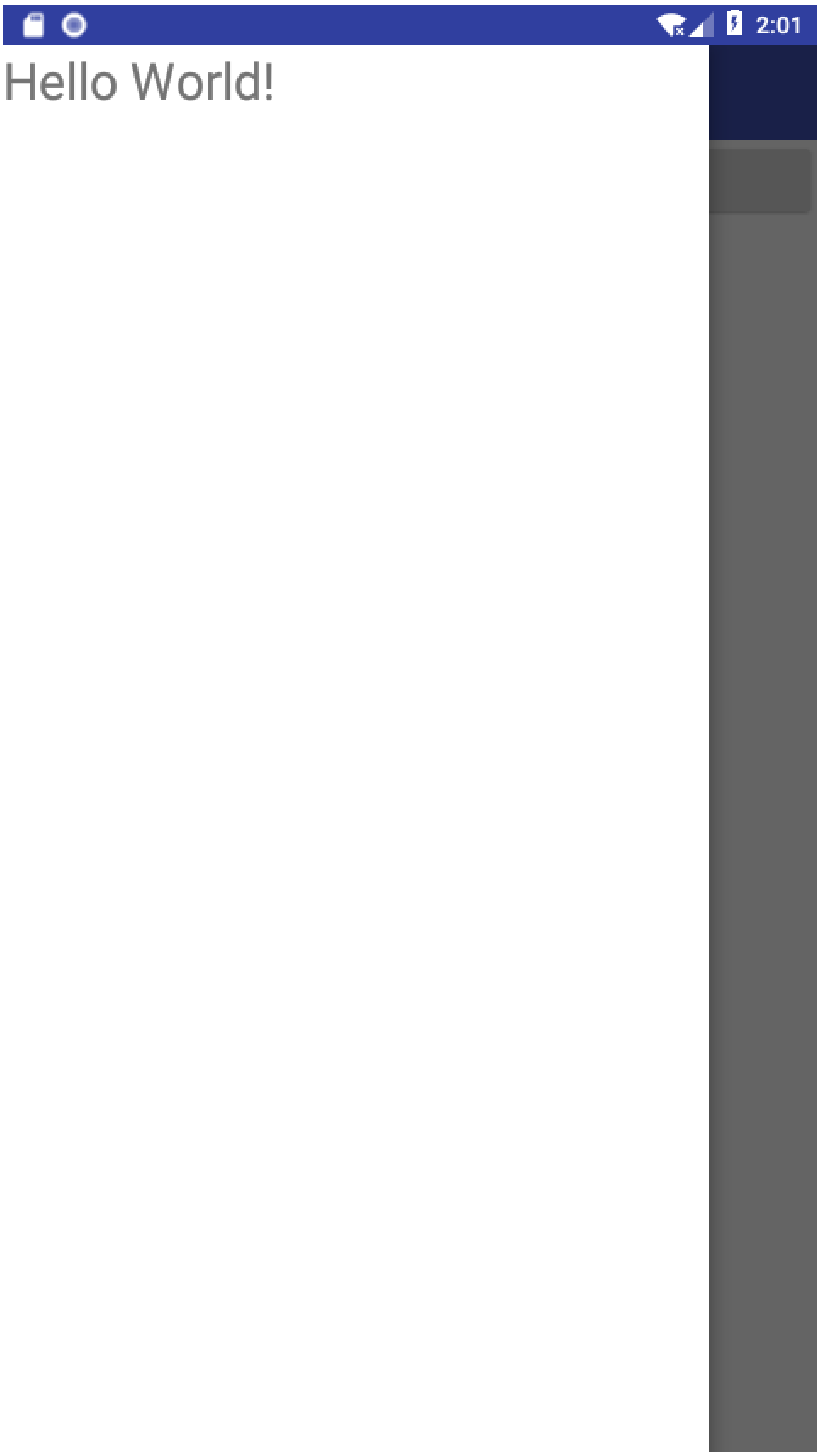
DrawerLayout(滑动菜单)

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <!--DrawerLayout 包含两个直接子控件(Layout)-->
    <!--第一个用于主屏幕显示的内容,其中包含 ToolBar 和一个 Button,一个 TextView-->
    <!--Content-->
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <android.support.v7.widget.Toolbar
            android:id="@+id/toolbar"

            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="?attr/colorPrimary"
            android:minHeight="?attr/actionBarSize"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
            app:popupTheme="@style/ThemeOverlay.AppCompat.Light" />

        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/app_name" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/app_name"
            android:textSize="30sp" />
    </LinearLayout>

    <!--第二个用于滑动菜单中显示的内容,这里用一个 TextView 代替-->
    <!--滑动菜单中的 layout_gravity 属性必须指定,用于告诉 DrawerLayout 滑动菜单从屏幕边缘滑出的方向
    start 表示会根据系统语言判断,如果系统语言是从左往右,则滑动菜单位于屏幕左侧,会从左侧滑出-->
    <!--Menu-->
    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:background="#ffffffff"
        android:text="Hello World!"
        android:textSize="30sp" />

</android.support.v4.widget.DrawerLayout>
```

```
public class MainActivity extends AppCompatActivity {
    private Toolbar toolBar;
    private DrawerLayout drawerLayout;
    private ActionBarDrawerToggle arrowBtn;

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

        toolBar = (Toolbar) findViewById(R.id.toolbar);
        drawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);
        setSupportActionBar(toolBar);

        //得到 ActionBar 的实例,其实这个 ActionBar 是由 ToolBar 实现的
        ActionBar actionBar = getSupportActionBar();
        if (actionBar != null)
        {
            actionBar.setHomeButtonEnabled(true); //设置返回键可用
            actionBar.setDisplayHomeAsUpEnabled(true);//在 ToolBar 左侧显示一个导航按钮

            //给导航按钮设置图标,默认是一个返回箭头
            //actionBar.setHomeAsUpIndicator(android.R.drawable.ic_menu_view);
        }

        arrowBtn = new ActionBarDrawerToggle(MainActivity.this, drawerLayout, toolBar,
            R.string.drawer_toggle_open, R.string.drawer_toggle_close);
        arrowBtn.syncState();//设置导航按钮显示为三横杠

        //添加菜单拖动监听事件  根据菜单的拖动距离,折算成导航按钮旋转角度
        drawerLayout.addDrawerListener(arrowBtn);
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item)
    {
        switch (item.getItemId())
        {
        //处理导航按钮的点击事件,点击导航按钮,滑出滑动菜单
        case android.R.id.home://导航按钮的 ID 永远是 android.R.id.home
            drawerLayout.openDrawer(GravityCompat.START);
```

```
                break;
            default:
                break;
            }
            return super.onOptionsItemSelected(item);
        }
}
```



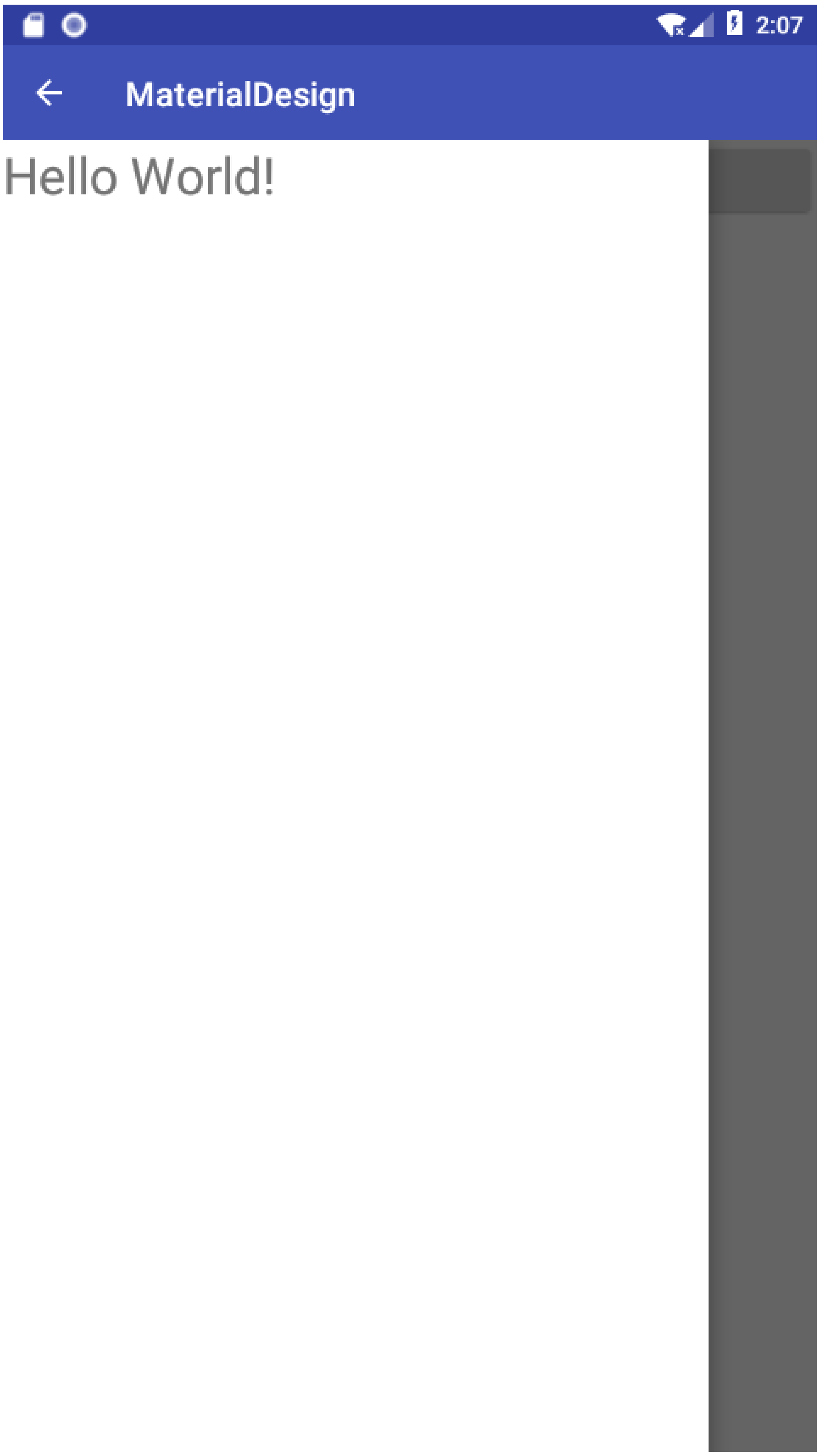


Hello World!

另外一种布局方式:

```
<!----------------------------------------main_activity.xml--------------------------------------->
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
```

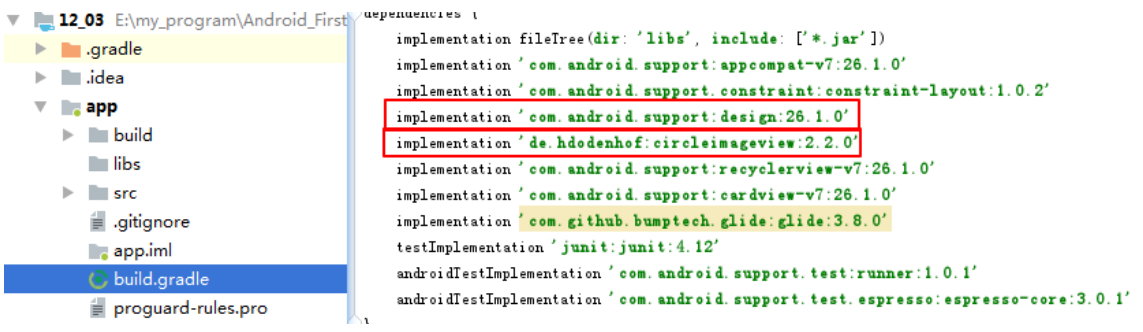
```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <!--DrawerLayout 不再包含 Toolbar-->
    <include layout="@layout/layout_toolbar" />

    <include layout="@layout/layout_drawer" />

</LinearLayout>

<!--把 Toolbar 和 DrawerLayout 独立出来,方便重用-->
<!--------------------------------layout_toolbar.xml----------------------------------------->
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.Toolbar xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/toolbar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="?attr/colorPrimary"
    android:minHeight="?attr/actionBarSize"
    android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
    app:popupTheme="@style/ThemeOverlay.AppCompat.Light">

</android.support.v7.widget.Toolbar>

<!--------------------------------layout_drawer.xml------------------------------------------->
<?xml version="1.0" encoding="utf-8"?>
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <!--Content-->
    <LinearLayout
        android:id="@+id/ll_content"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/app_name" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/app_name" />
    </LinearLayout>

    <!--Menu-->
    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:background="#ffffffff"
        android:text="Hello World!"
        android:textSize="30sp" />

</android.support.v4.widget.DrawerLayout>
```


2:07
MaterialDesign
Hello World!

NavigationView:

1,NavigationView 是 Design Support 库中提供的控件,所以需要添加 Design Support 库的引用:

```
implementation 'com.android.support:design:26.1.0'//design support 库
implementation 'de.hdodenhof:circleimageview:2.2.0'//用于图片圆形化的第三方库
```



```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:appcompat-v7:26.1.0'
    implementation 'com.android.support.constraint:constraint-layout:1.0.2'
    implementation 'com.android.support:design:26.1.0'
    implementation 'de.hdodenhof:circleimageview:2.2.0'
    implementation 'com.android.support:recyclerview-v7:26.1.0'
    implementation 'com.android.support:cardview-v7:26.1.0'
    implementation 'com.github.bumptech.glide:glide:3.8.0'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.1'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.1'
}
```

2,为 NavigationView 准备两个布局文件:menu.xml(用于显示菜单)和 header.xml(用于显示头像)

```
<!----------------------menu.xml---------------------->
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <group android:checkableBehavior="single">
        <item
            android:id="@+id/menu_call"
            android:icon="@drawable/ic_menu_call"
            android:title="@string/menu_call_title" />
        <item
            android:id="@+id/menu_friends"
            android:icon="@drawable/ic_menu_friends"
            android:title="@string/menu_friends_title" />
        <item
            android:id="@+id/menu_address"
            android:icon="@drawable/ic_menu_home"
            android:title="@string/menu_address_title" />
        <item
            android:id="@+id/menu_photo"
            android:icon="@drawable/ic_menu_photo"
            android:title="@string/menu_photo_title" />
    </group>
</menu>
```

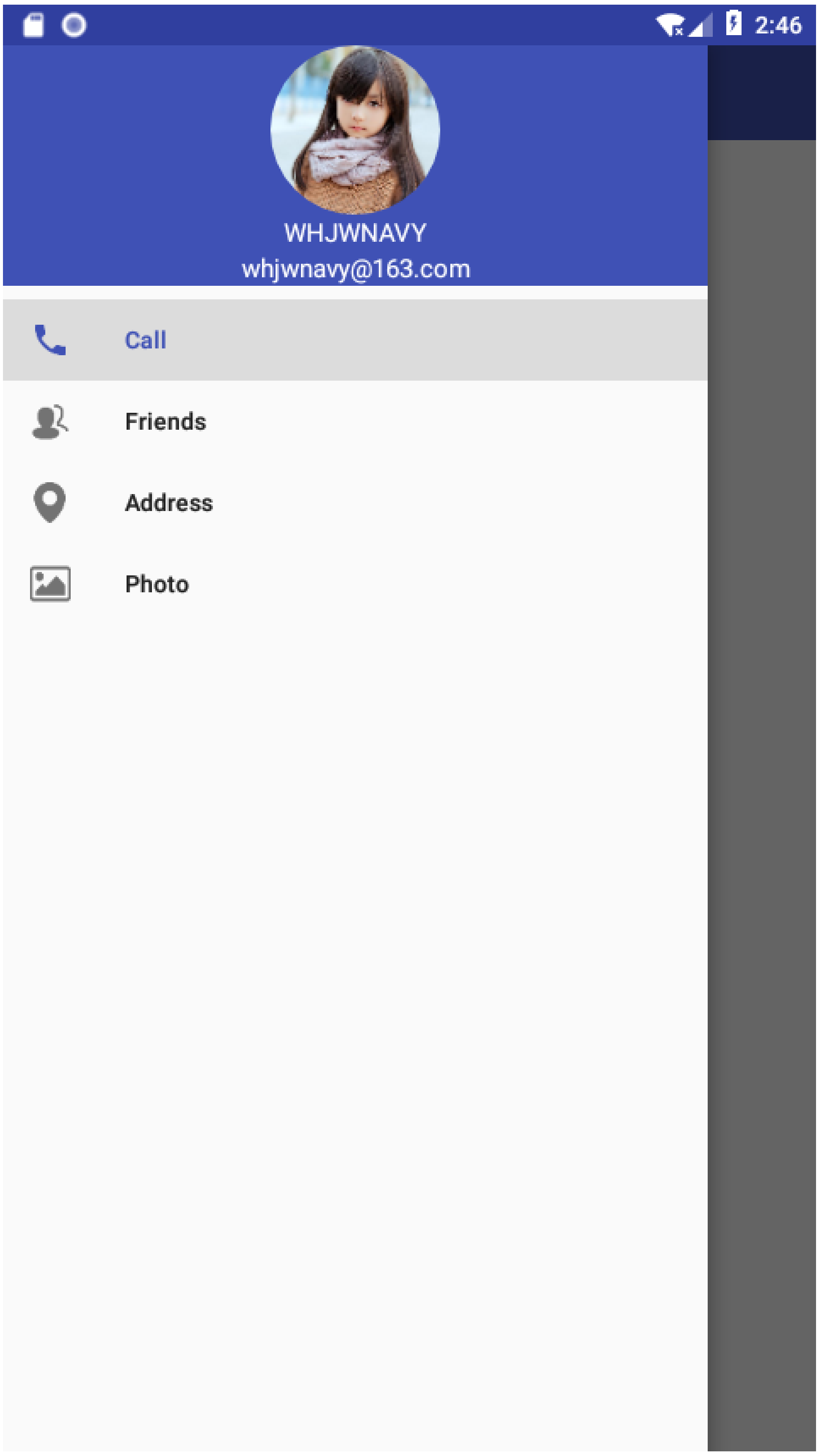
```
<!----------------------header_layout.xml---------------------->
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="?attr/colorPrimary"
    android:gravity="center"
    android:orientation="vertical">
    <!--顶部显示一个圆形头像-->
    <de.hdodenhof.circleimageview.CircleImageView
        android:id="@+id/header_image"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:src="@drawable/ic_header_photo" />

    <TextView
        android:id="@+id/header_usrname"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/header_user_name"
        android:textAlignment="center"
        android:textColor="@color/colorTextPrimary"
        android:textSize="@dimen/textSizePrimary" />

    <TextView
        android:id="@+id/header_email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/header_email"
        android:textAlignment="center"
        android:textColor="@color/colorTextPrimary"
        android:textSize="@dimen/textSizePrimary" />

</LinearLayout>
```

3,将 DrawerLayout 中原来的菜单(TextView)替换为 NavigationView:

```
<!----------------------main_activity.xml---------------------->
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <android.support.v7.widget.Toolbar
            android:id="@+id/tool_bar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            android:minHeight="?attr/actionBarSize"
```

```xml
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
            app:popupTheme="@style/Base.ThemeOverlay.AppCompat.Light" />

    </FrameLayout>

    <!--将 DrawerLayout 中原来的菜单(TextView)替换为 NavigationView-->
    <android.support.design.widget.NavigationView
        android:id="@+id/navigation_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        app:headerLayout="@layout/layout_header"<!--为 NavigationView 指定 header-->
        app:menu="@menu/layout_menu" /><!--为 NavigationView 指定菜单项-->

</android.support.v4.widget.DrawerLayout>
```

```java
public class MainActivity extends AppCompatActivity{
    Toolbar toolbar;
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle actionBarDrawerToggle;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        drawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);
        toolbar = (Toolbar) findViewById(R.id.tool_bar);
        navigationView = (NavigationView) findViewById(R.id.navigation_view);

        MainActivity.this.setSupportActionBar(toolbar);

        ActionBar actionBar = MainActivity.this.getSupportActionBar();
        if (actionBar != null)
        {
            actionBar.setHomeButtonEnabled(true);//设置返回键可用
            actionBar.setDisplayHomeAsUpEnabled(true);//在 ToolBar 左侧显示一个导航按钮
        }

        navigationView.setCheckedItem(R.id.menu_call);//设置 navigationView 的默认选中项目
        //为 navigationView 的菜单项目(menu)添加点击事件
        navigationView.setNavigationItemSelectedListener(
            new NavigationView.OnNavigationItemSelectedListener() {
                @Override
                public boolean onNavigationItemSelected(@NonNull MenuItem item)
                {
                    //这里仅用于点击菜单项之后关闭滑动菜单
                    drawerLayout.closeDrawers();
                    return true;
                }
            });

        actionBarDrawerToggle = new ActionBarDrawerToggle(MainActivity.this,
                drawerLayout, toolbar, R.string.drawer_toggle_open, R.string.drawer_toggle_close);

        actionBarDrawerToggle.syncState();//设置导航按钮显示为三横杠

        //添加菜单拖动监听事件,根据菜单的拖动距离,折算成导航按钮旋转角度
        drawerLayout.addDrawerListener(actionBarDrawerToggle);
    }
}
```



CoordinatorLayout,FloatingActionButton,Snackbar

FloatingActionButton:悬浮按钮

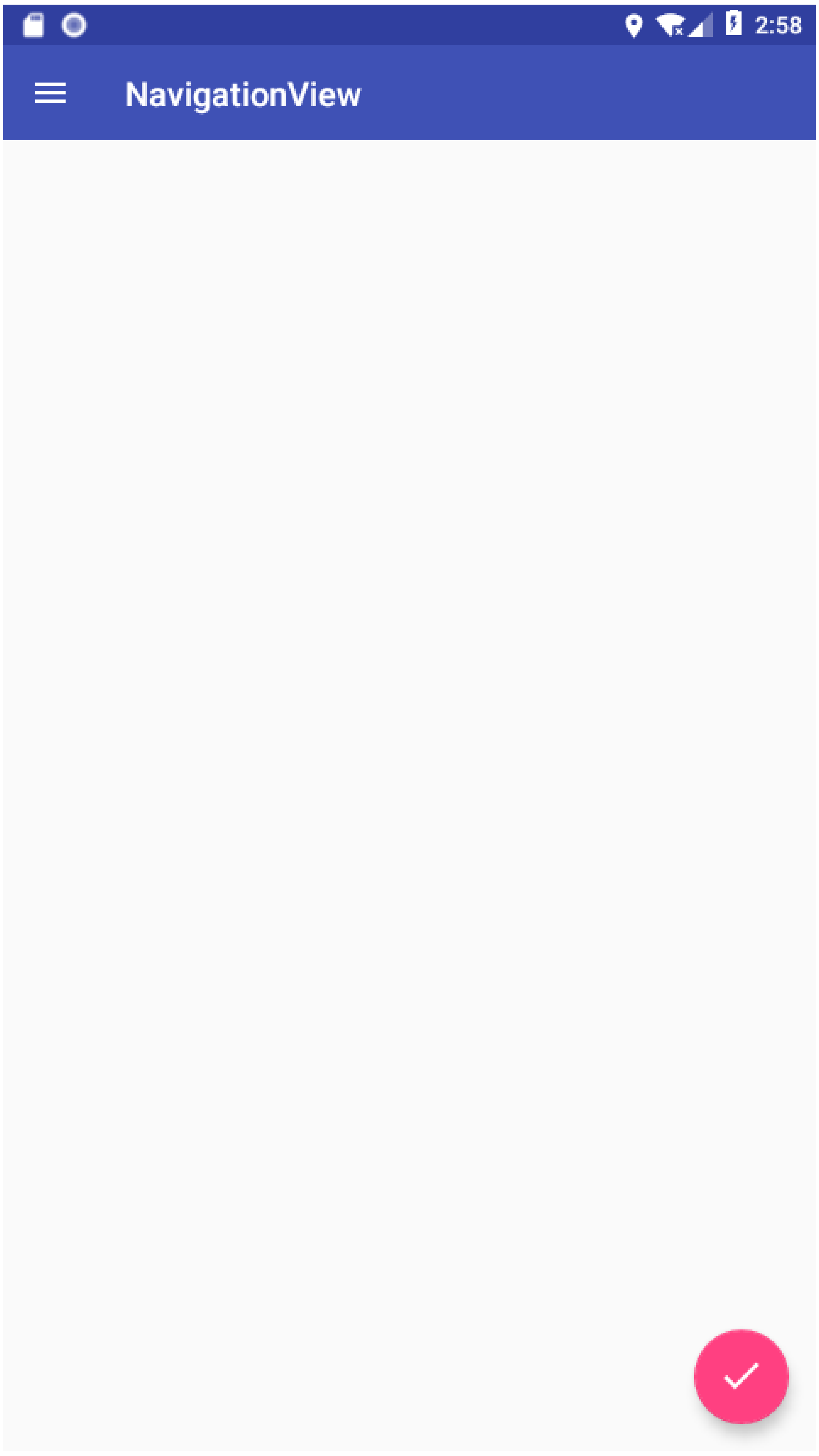
Snackbar:可交互提示

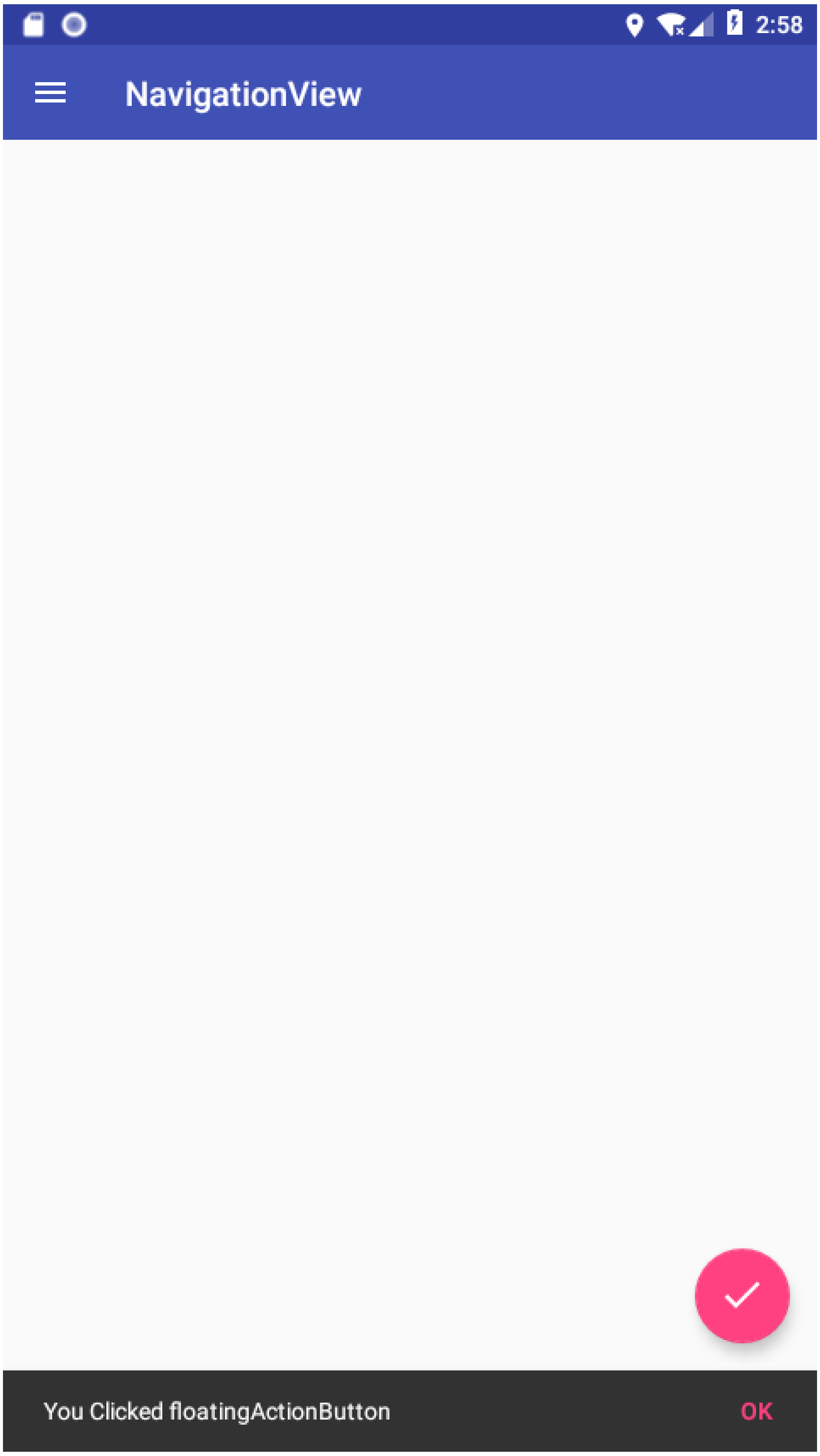
CoordinatorLayout:Snackbar 弹出时会遮挡住悬浮按钮,利用 CoordinatorLayout 布局可以解决这个问题,CoordinatorLayout 会监听到 Snackbar 弹出事件,然后自动将悬浮按钮向上偏移,以确保不会被 Snackbar 遮挡.

```xml
<!----------------------main_activity.xml---------------------->
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <!--将原来的布局替换为 CoordinatorLayout 即可-->
    <android.support.design.widget.CoordinatorLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <android.support.v7.widget.Toolbar
            android:id="@+id/tool_bar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            android:minHeight="?attr/actionBarSize"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
            app:popupTheme="@style/Base.ThemeOverlay.AppCompat.Light" />

        <!--悬浮按钮-->
        <android.support.design.widget.FloatingActionButton
            android:id="@+id/float_action_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="bottom|end"
            android:layout_margin="16dp"
            android:clickable="true"
            android:elevation="10dp"
            android:focusable="true"
            android:src="@drawable/ic_cab_done" />

    </android.support.design.widget.CoordinatorLayout>

    <android.support.design.widget.NavigationView
        android:id="@+id/navigation_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        app:headerLayout="@layout/layout_header"
        app:menu="@menu/layout_menu" />
```

```
</android.support.v4.widget.DrawerLayout>
```

```
public class MainActivity extends AppCompatActivity {

    /* ...... */
    FloatingActionButton floatingActionButton;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        /* ...... */
        floatingActionButton = (FloatingActionButton) findViewById(R.id.float_action_button);

        floatingActionButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                /*Toast.makeText(MainActivity.this, "You Clicked floatingActionButton", Toast.LENGTH_SHO
RT).show();*/
                //创建可交互提示框 Snackbar,并为其绑定点击交互事件
                Snackbar.make(v, "You Clicked floatingActionButton", Snackbar.LENGTH_SHORT).
                        setAction("Ok", new View.OnClickListener() {
                            @Override
                            public void onClick(View v) {
                                Toast.makeText(MainActivity.this, "You Clicked Snackbar OK", Toast.LENGT
H_SHORT).show();
                            }
                        }).show();
            }
        });

       /* ...... */
    }
}
```


2:58
NavigationView



You Clicked floatingActionButton

CardView(卡片式布局)

示例:利用 RecyclerView 展示图片,RecyclerView 的子项使用 CardView 布局.图片使用 Glide 加载.

1,添加依赖库:

```
implementation 'com.android.support:recyclerview-v7:26.1.0'
implementation 'com.android.support:cardview-v7:26.1.0'
implementation 'com.github.bumptech.glide:glide:3.8.0'
```

2,添加 RecvclerView 布局:

```
<!--main_activity.xml-->
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <android.support.design.widget.CoordinatorLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <android.support.v7.widget.Toolbar
            ...... />

        <android.support.v7.widget.RecyclerView
            android:id="@+id/recycle_view"
            android:layout_width="match_parent"
            android:layout_height="match_parent" />

        <android.support.design.widget.FloatingActionButton
            ...... />

    </android.support.design.widget.CoordinatorLayout>

    <android.support.design.widget.NavigationView
        ...... />

</android.support.v4.widget.DrawerLayout>
```

3,新建用于展示图片的类和布局文件:

```
public class Girl {
    private String name;
    private int imageId;

    public Girl(String name, int imageId) {
        this.imageId = imageId;
        this.name = name;
    }
    public String getName() {
        return this.name;
    }

    public int getImageId() {
        return this.imageId;
    }
}
```

```
<!--girl_layout.xml-->
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    app:cardCornerRadius="5dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <ImageView
            android:id="@+id/girl_image"
            android:layout_width="match_parent"
            android:layout_height="100dp"
            android:scaleType="centerCrop" />

        <TextView
            android:id="@+id/girl_name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:layout_margin="5dp"
            android:textSize="@dimen/textSizePrimary" />
    </LinearLayout>

</android.support.v7.widget.CardView>
```

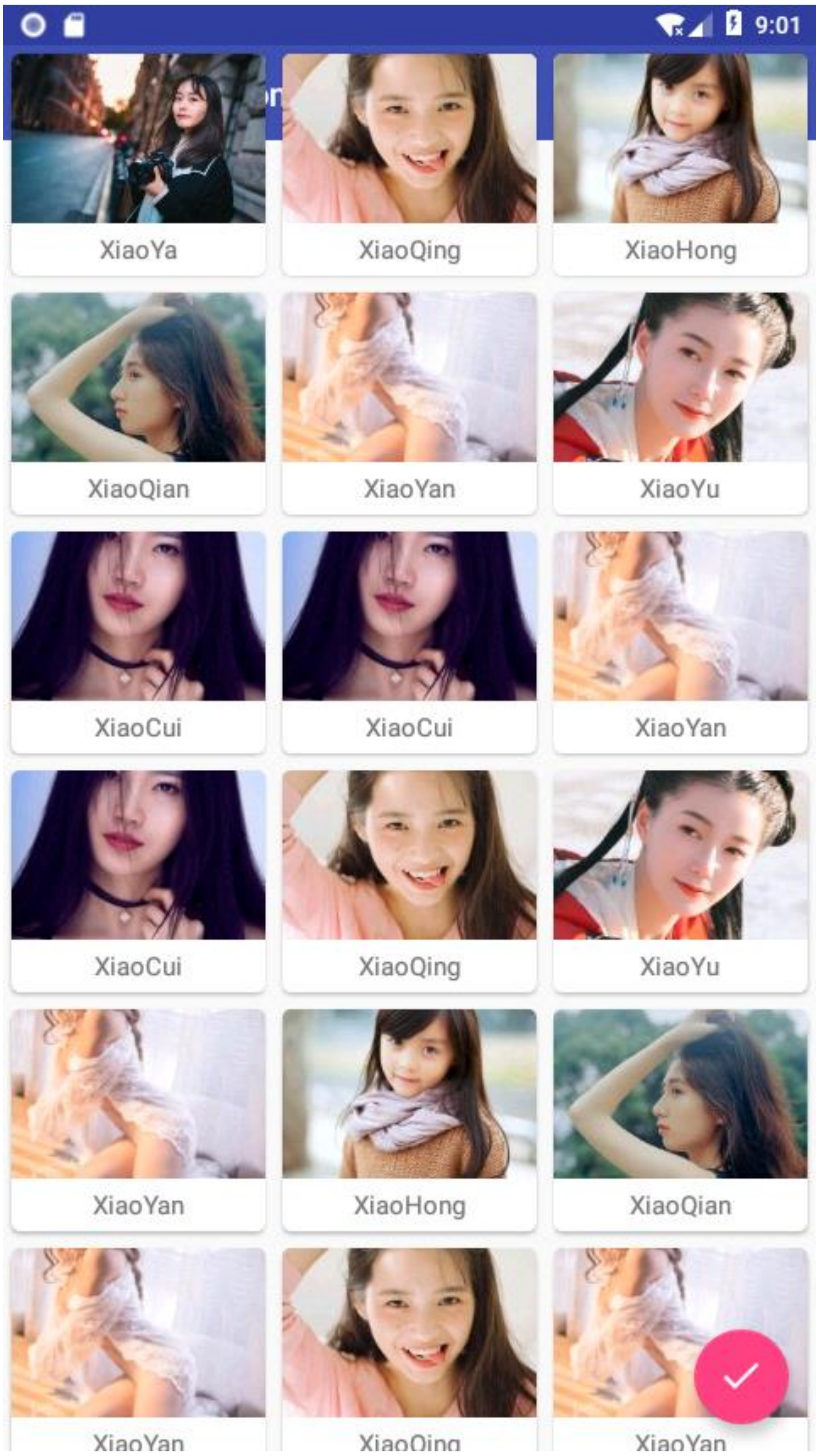
4,创建自定义适配器:

```
public class GirlAdapter extends RecyclerView.Adapter<GirlAdapter.ViewHolder> {
    private Context context;
    private List<Girl> girlList;

    static class ViewHolder extends RecyclerView.ViewHolder {
        CardView cardView;
```

```java
        ImageView imageView;
        TextView textView;

        private ViewHolder(View itemView) {
            super(itemView);
            cardView = (CardView) itemView;
            imageView = (ImageView) itemView.findViewById(R.id.girl_image);
            textView = (TextView) itemView.findViewById(R.id.girl_name);
        }
    }

    public GirlAdapter(List<Girl> girlList) {
        this.girlList = girlList;
    }

    @Override
    public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        if (this.context == null) {
            this.context = parent.getContext();
        }

        View view = LayoutInflater.from(this.context).inflate(R.layout.layout_girl, parent, false);
        return new ViewHolder(view);
    }

    @Override
    public void onBindViewHolder(ViewHolder holder, int position) {
        Girl girl = this.girlList.get(position);
        holder.textView.setText(girl.getName());
        //利用 Glide 加载图片到 ImageView
        Glide.with(this.context).load(girl.getImageId()).into(holder.imageView);
    }

    @Override
    public int getItemCount() {
        return this.girlList.size();
    }
}
```

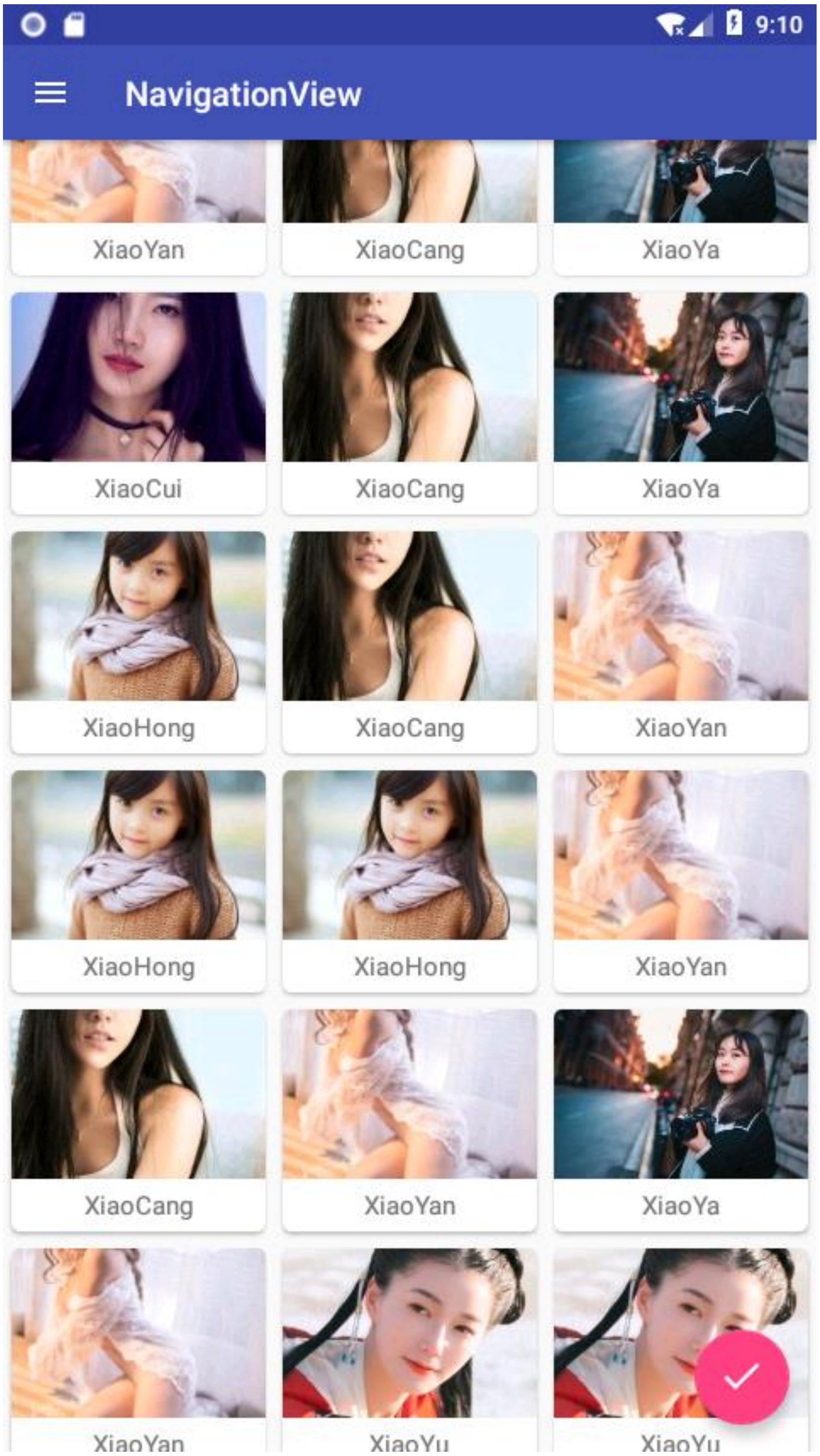
5,修改 MainActivity 代码逻辑:

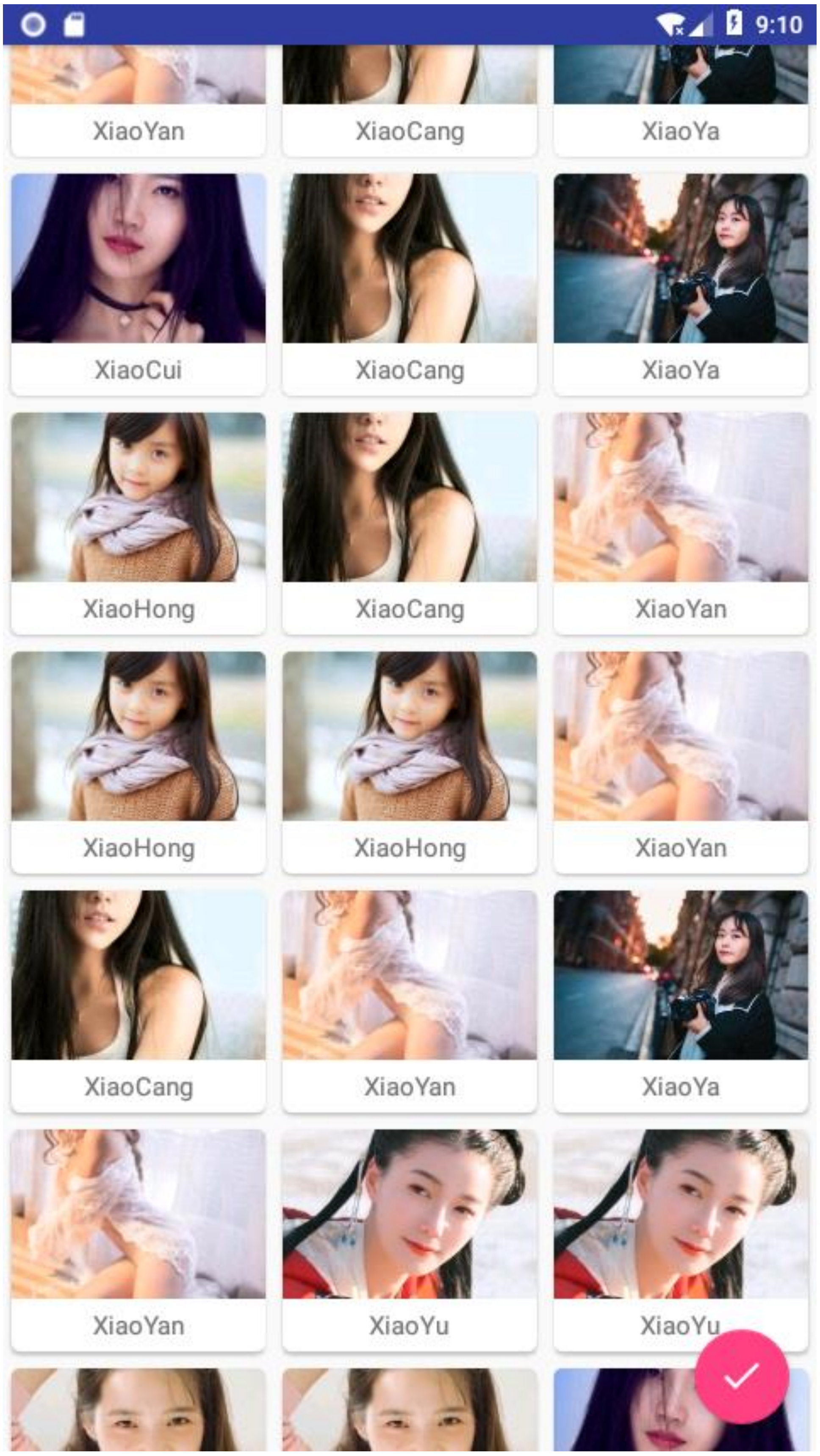
```java
public class MainActivity extends AppCompatActivity {
    /*......*/
    RecyclerView recyclerView;
    GridLayoutManager gridLayoutManager;

    private Girl[] girlArry = {
            new Girl("XiaoHong", R.drawable.image_girl_1),
            new Girl("XiaoQing", R.drawable.image_girl_2),
            new Girl("XiaoYu", R.drawable.image_girl_3),
            new Girl("XiaoCui", R.drawable.image_girl_4),
            new Girl("XiaoCang", R.drawable.image_girl_5),
            new Girl("XiaoYa", R.drawable.image_girl_6),
            new Girl("XiaoQian", R.drawable.image_girl_7),
            new Girl("XiaoYan", R.drawable.image_girl_8),
    };

    private List<Girl> girlList = new ArrayList<>();
    private GirlAdapter girlAdapter;

    private void initGirlList() {
        girlList.clear();
        Random random = new Random();
        int index = 0;
        for (int i = 0; i < 300; i++) {
            index = random.nextInt(girlArry.length);
            girlList.add(girlArry[index]);
        }
    }

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

        recyclerView = (RecyclerView) findViewById(R.id.recycle_view);
        initGirlList();//初始化图片列表

        /*......*/

        gridLayoutManager = new GridLayoutManager(MainActivity.this, 3);

        recyclerView.setLayoutManager(gridLayoutManager);
        girlAdapter = new GirlAdapter(girlList);
        recyclerView.setAdapter(girlAdapter);
    }
}
```



AppbarLayout

```
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    ...... >
    <android.support.design.widget.CoordinatorLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <!--为了解决 RecyclerView 遮挡住 Toolbar 的问题,只要把 TooBar 放在 AppBarLayout 布局中即可-->
        <android.support.design.widget.AppBarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <android.support.v7.widget.Toolbar
                android:id="@+id/tool_bar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                android:background="?attr/colorPrimary"
                android:minHeight="?attr/actionBarSize"
                android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
                <!--给 ToolBar 添加对滚动事件的处理属性,用于滚动列表时自动显示或隐藏 ToolBar-->
                app:layout_scrollFlags="scroll|enterAlways|snap"
                app:popupTheme="@style/Base.ThemeOverlay.AppCompat.Light" />

        </android.support.design.widget.AppBarLayout>

        <android.support.v7.widget.RecyclerView
            android:id="@+id/recycle_view"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            app:layout_behavior="@string/appbar_scrolling_view_behavior"/>

        <android.support.design.widget.FloatingActionButton
            ...... />

    </android.support.design.widget.CoordinatorLayout>

    <android.support.design.widget.NavigationView
        ...... />

</android.support.v4.widget.DrawerLayout>
```


9:10
NavigationView
XiaoYan
XiaoCang
XiaoYa
XiaoCui
XiaoCang
XiaoYa
XiaoHong
XiaoCang
XiaoYan
XiaoHong
XiaoHong
XiaoYan
XiaoCang
XiaoYan
XiaoYa
XiaoYan
XiaoYu
XiaoYu



SwipeRefreshLayout(下拉刷新)

把需要实现下拉刷新的控件放在 SwipeRefreshLayout 布局中即可.

1,修改布局文件:

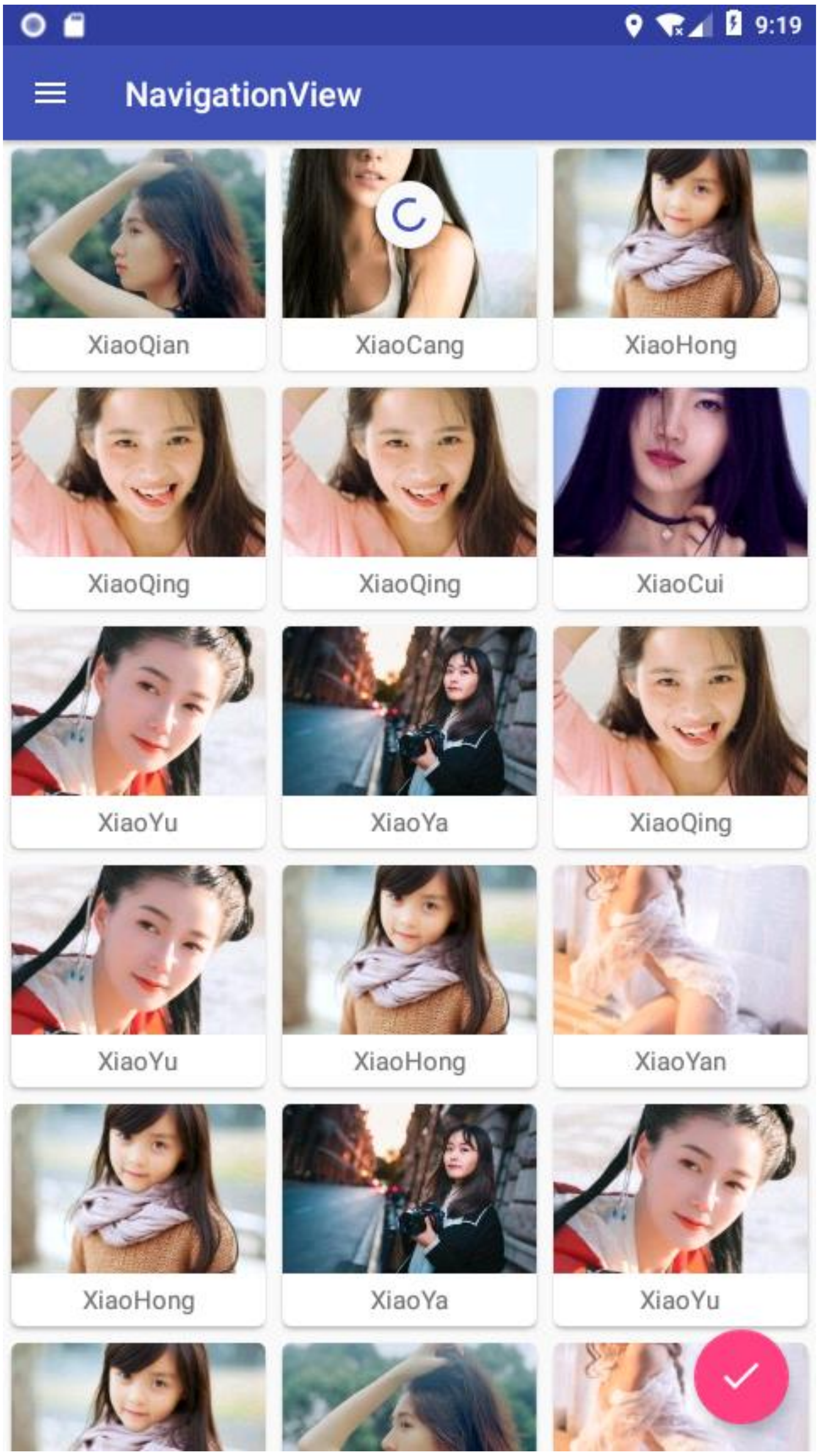
```
<android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
```

```
    ...... >

    <android.support.design.widget.CoordinatorLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <!--为了解决 RecyclerView 遮挡住 Toolbar 的问题,只要把 TooBar 放在 AppBarLayout 布局中即可-->
        <android.support.design.widget.AppBarLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <android.support.v7.widget.Toolbar
                ...... />

        </android.support.design.widget.AppBarLayout>

        <android.support.v4.widget.SwipeRefreshLayout
            android:id="@+id/swip_refresh"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            app:layout_behavior="@string/appbar_scrolling_view_behavior">

            <android.support.v7.widget.RecyclerView
                android:id="@+id/recycle_view"
                android:layout_width="match_parent"
                android:layout_height="match_parent" />
        </android.support.v4.widget.SwipeRefreshLayout>

        <android.support.design.widget.FloatingActionButton
            ...... />

    </android.support.design.widget.CoordinatorLayout>

    <android.support.design.widget.NavigationView
       ...... />

</android.support.v4.widget.DrawerLayout>
```

2,修改 MainActivity 代码:

```
public class MainActivity extends AppCompatActivity {

    /* ...... */

    private SwipeRefreshLayout swipeRefreshLayout;

   /* ...... */

    private void refreshGirlList() {
```

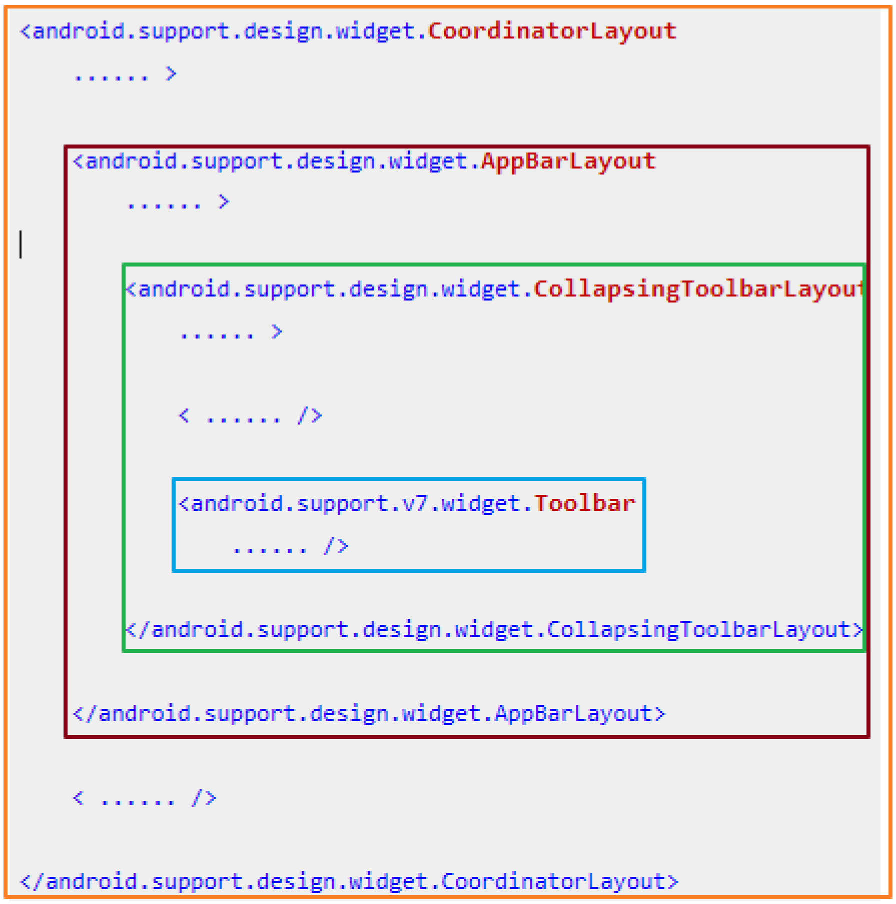
```java
        new Thread(new Runnable() {
            @Override
            public void run() {
                try {
                    Thread.sleep(3000);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }

                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        initGirlList();
                        //通知数据发生了变化
                        girlAdapter.notifyDataSetChanged();
                        //设置刷新事件结束,隐藏进度条
                        swipeRefreshLayout.setRefreshing(false);
                    }
                });
            }
        }).start();
    }

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

        /* ...... */

        swipeRefreshLayout = (SwipeRefreshLayout) findViewById(R.id.swip_refresh);
        //设置下拉刷新进度条的颜色
        swipeRefreshLayout.setColorSchemeResources(R.color.colorPrimary);
        swipeRefreshLayout.setOnRefreshListener(new SwipeRefreshLayout.OnRefreshListener() {
            @Override
            public void onRefresh() {
                refreshGirlList();
            }
        });
    }
}
```



CollapsingToolbarLayout(可折叠式标题栏)

CollapsingToolbarLayout 是一个作用于 ToolBar 之上的布局,它可以让 ToolBar 的效果更丰富,不仅仅是展示一个标题栏,而且能够实现非常华丽的效果(高级版标题栏).但是

CollapsingToolbarLayout 布局不能独立存在,只能作为 AppBarLayout 的直接子布局,而 AppBarLayout 又必须是 CoordinatorLayout 的子布局,如下:



```
<android.support.design.widget.CoordinatorLayout
    ...... >

    <android.support.design.widget.AppBarLayout
        ...... >

        <android.support.design.widget.CollapsingToolbarLayout
            ...... >

            < ...... />

            <android.support.v7.widget.Toolbar
                ...... />

        </android.support.design.widget.CollapsingToolbarLayout>

    </android.support.design.widget.AppBarLayout>

    < ...... />

</android.support.design.widget.CoordinatorLayout>
```

示例:在上例中,继续实现一个点击卡片进入图片详情的页面,包含高级版标题栏.

1,新建图片详情页布局文件 girl_Activity.xml,用 CollapsingToolbarLayout 实现详情页标题布局:

```
<!--girl_Activity.xml-->
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <!--图片详情页标题栏-->
    <android.support.design.widget.AppBarLayout
        android:id="@+id/girl_app_bar"
        android:layout_width="match_parent"
        android:layout_height="250dp"> <!--给 AppBarLayout 指定一个合适的高度-->

        <!--用 CollapsingToolbarLayout 作为 TooBar 的父布局-->
        <android.support.design.widget.CollapsingToolbarLayout
            android:id="@+id/girl_collapsing_toolbar"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"<!--把原来作用于 TooBar 的主题属性移到父布局中来-->
            app:contentScrim="?attr/colorPrimary"<!--用于指定 CollapsingToolbarLayout 在趋于折叠状态时的背景色,
                                                        CollapsingToolbarLayout 折叠之后就是一个普通的 ToolBar-->
            app:layout_scrollFlags="scroll|exitUntilCollapsed"><!--把原来作用于 TooBar 的随着滑动自动移出移入的属性移到父布局中来,
                                                                   exitUntilCollapsed 表示当 CollapsingToolbarLayout 随着滚动完成
                                                                  折叠之后就保留在屏幕上,不再移除屏幕-->

            <!--定义标题栏的具体内容,高级版的标题栏是由普通的 ToolBar 加上图片组合而成-->
            <ImageView
                android:id="@+id/girl_image_view"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:scaleType="centerCrop"<!—图片缩放模式—>
                app:layout_collapseMode="parallax" /><!—指定控件在 CollapsingToolbarLayout 折叠过程中的折叠状态,parallax 表示
                                                      ImageView 会在折叠过程中产生一定的错位偏移—>
            <android.support.v7.widget.Toolbar
                android:id="@+id/girl_tool_bar"
                android:layout_width="match_parent"
                android:layout_height="?attr/actionBarSize"
                app:layout_collapseMode="pin" /><!—指定控件在 CollapsingToolbarLayout 折叠过程中的折叠状态,pin 表示 ToolBar 在
                                                  折叠过程中位置始终保持不变—>

        </android.support.design.widget.CollapsingToolbarLayout>
    </android.support.design.widget.AppBarLayout>
</android.support.design.widget.CoordinatorLayout>
```

2,继续实现图片内容详情部分:

```
<!--girl_Activity.xml-->
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    ...... >

    <!--图片详情页标题栏-->
    <android.support.design.widget.AppBarLayout
        ...... >
        <android.support.design.widget.CollapsingToolbarLayout
            ...... >

            <ImageView
                ...... />
            <android.support.v7.widget.Toolbar
                ...... />
        </android.support.design.widget.CollapsingToolbarLayout>
    </android.support.design.widget.AppBarLayout>

    <!--图片详情页,NestedScrollView 是一个高级版 ScrollView 控件,允许使用滚动的方式来查看屏幕以外的数据,
       NestedScrollView 在 ScrollView 的基础上增加了嵌套响应滚动事件的功能.由于 CoordinatorLayout 本身已
       经可以响应滚动事件了,因此我们在它内部就需要使用 NestedScrollView 或 RecyclerView 这样的布局(来代替
       ScrollView).由于 NestedScrollView(或 ScrollView)布局内部只允许存在一个直接子布局,因此为了放入更多
       的内容,先嵌套一个 LinearLayout 布局,然后在 LinearLayout 中放入具体内容.-->
    <android.support.v4.widget.NestedScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_behavior="@string/appbar_scrolling_view_behavior"><!--指定一个布局行为,避免滚动时遮挡
ToolBar-->

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <!--把要现实的图片详情内容放入 CardView 中-->
            <android.support.v7.widget.CardView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginBottom="15dp"<!--下边距-->
                android:layout_marginLeft="15dp"<!--左边距-->
                android:layout_marginRight="15dp"<!--右边距-->
                android:layout_marginTop="35dp"<!--上边距,为接下来的悬浮按钮预留控件-->
                app:cardCornerRadius="4dp">

                <TextView
                    android:id="@+id/girl_text_view"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_margin="10dp" />
            </android.support.v7.widget.CardView>
        </LinearLayout>
    </android.support.v4.widget.NestedScrollView>

</android.support.design.widget.CoordinatorLayout>
```

3,在界面上添加一个悬浮按钮:

```
<!--girl_Activity.xml-->
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    ...... >

    <!--图片详情页标题栏-->
    <android.support.design.widget.AppBarLayout
```

```
        ...... >
        <android.support.design.widget.CollapsingToolbarLayout
            ...... >

            <ImageView
                ...... />
            <android.support.v7.widget.Toolbar
                ...... />

        </android.support.design.widget.CollapsingToolbarLayout>
    </android.support.design.widget.AppBarLayout>

    <!--图片详情页-->
    <android.support.v4.widget.NestedScrollView
        ...... >

        <LinearLayout
            ...... >

            <android.support.v7.widget.CardView
                ...... >

                <TextView
                    ...... />

            </android.support.v7.widget.CardView>

        </LinearLayout>

    </android.support.v4.widget.NestedScrollView>

    <!--悬浮按钮-->
    <android.support.design.widget.FloatingActionButton
        android:id="@+id/float_action_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="16dp"
        android:src="@drawable/ic_cab_done"
        app:layout_anchor="@id/girl_app_bar"<!--为悬浮按钮指定一个锚点,将锚点设置为 AppBarLayout,这样
                                               悬浮按钮就会出现在标题栏内-->
        app:layout_anchorGravity="bottom|end" /><!--将悬浮按钮定位在标题栏区域右下角-->

</android.support.design.widget.CoordinatorLayout>
```

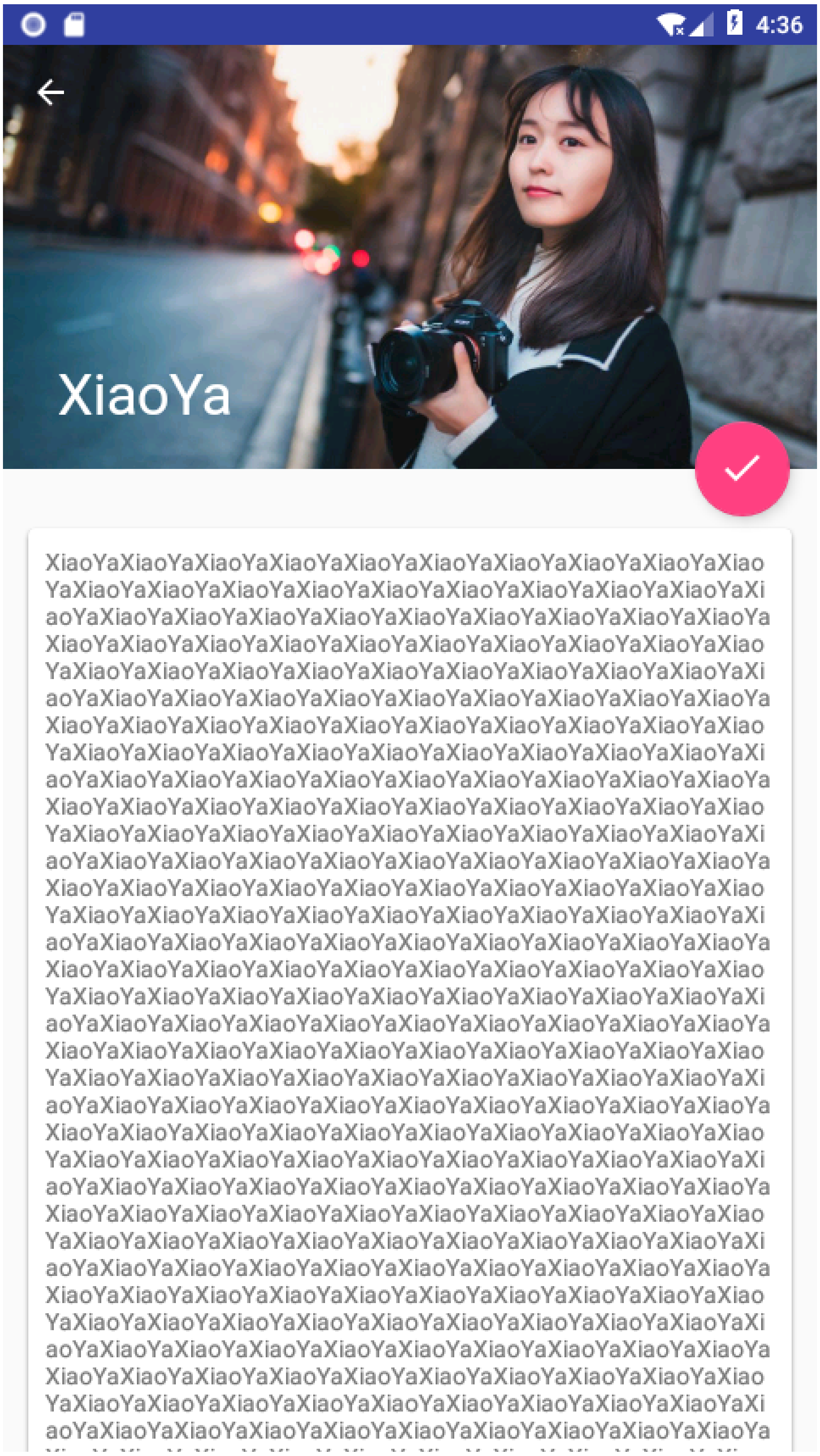
4,创建图片展示详情页 Activity,编写界面逻辑:

```
public class GirlActivity extends AppCompatActivity {

    public static final String GIRL_NAME = "Girl_Name";
```

```java
    public static final String GIRL_IMAGE_ID = "Girl_Image_Id";

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

        //通过 Intent 获取传入的图片名称和图片资源 ID
        Intent intent = getIntent();
        String girlName = intent.getStringExtra(GIRL_NAME);
        int girlImageId = intent.getIntExtra(GIRL_IMAGE_ID, 0);
        //通过 findViewById 拿到各种控件的实例
        Toolbar toolbar = (Toolbar) findViewById(R.id.girl_tool_bar);
        CollapsingToolbarLayout collapsingToolbarLayout =
            (CollapsingToolbarLayout) findViewById(R.id.girl_collapsing_toolbar);
        ImageView girlImageView = (ImageView) findViewById(R.id.girl_image_view);
        TextView girlTextView = (TextView) findViewById(R.id.girl_text_view);

        //设置 ToolBar
        setSupportActionBar(toolbar);
        ActionBar actionBar = getSupportActionBar();
        if (actionBar != null) {
            actionBar.setDisplayHomeAsUpEnabled(true);//ToolBar 显示返回箭头
        }

        //将图片名称设置为页面标题
        collapsingToolbarLayout.setTitle(girlName);

        //为标题栏中的 ImageView 加载图片
        Glide.with(this).load(girlImageId).into(girlImageView);
        //设置图片详情内容文本
        String girlContent = generateGirlContent(girlName);
        girlTextView.setText(girlContent);
    }

    //模拟生成图片详情文本
    private String generateGirlContent(String girlName) {
        StringBuilder stringBuilder = new StringBuilder();
        for (int i = 0; i < 500; i++) {
            stringBuilder.append(girlName);
        }
        return stringBuilder.toString();
    }

    @Override //处理返回箭头的点击事件
    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case android.R.id.home:
                finish();//点击返回箭头结束当前 Activity
                break;
            default:
                break;
        }

        return true;
    }
}
```

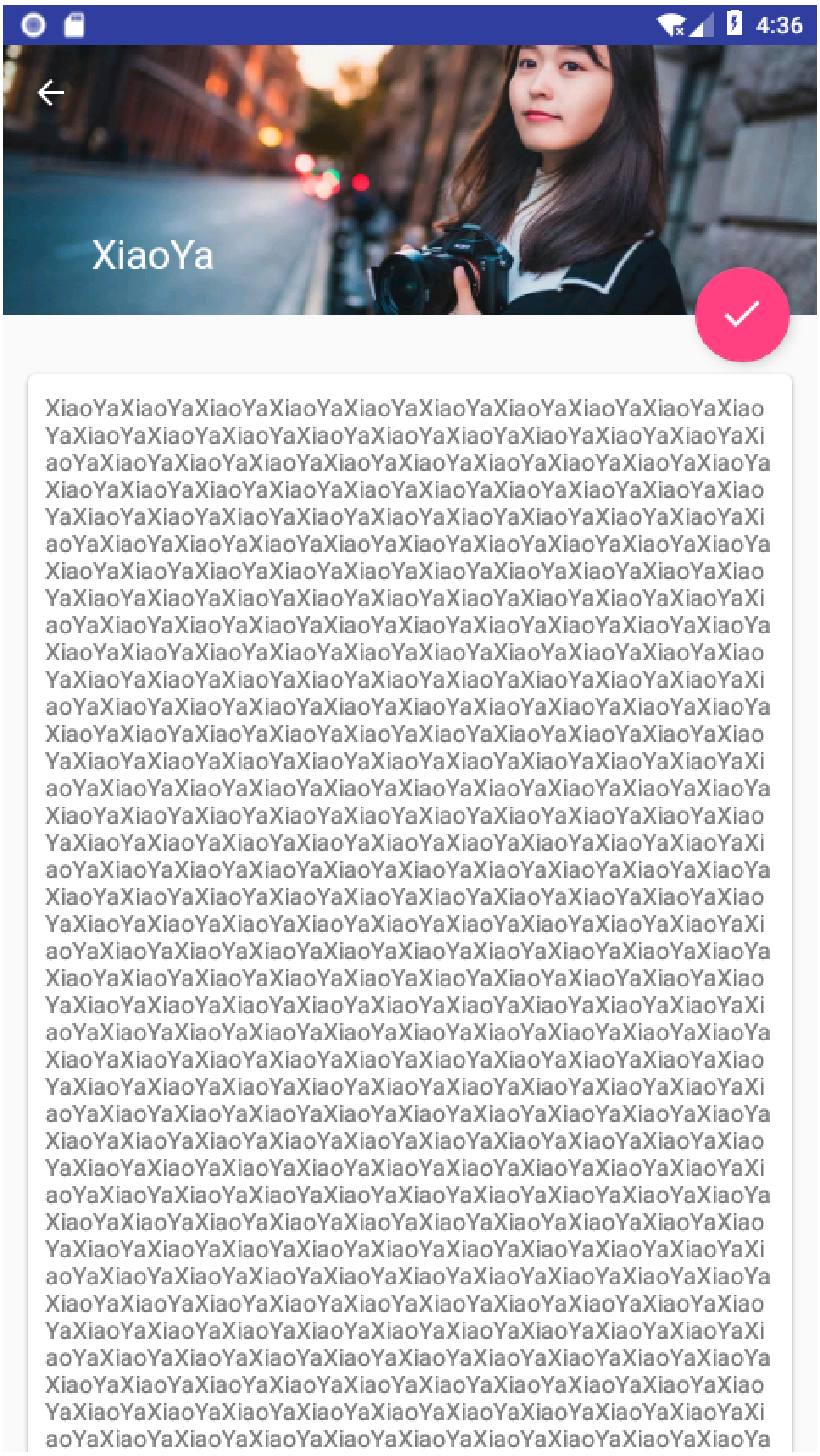
5,添加跳转到详情页的逻辑代码(在 GirlAdapter 类中添加卡片的点击事件):

```
public class GirlAdapter extends RecyclerView.Adapter<GirlAdapter.ViewHolder> {
    private Context context;
    private List<Girl> girlList;

    static class ViewHolder extends RecyclerView.ViewHolder {
        CardView cardView;
        ImageView imageView;
        TextView textView;

        /* ...... */
    }

    /* ...... */

    @Override
    public ViewHolder onCreateViewHolder(ViewGroup parent, final int viewType) {
        if (this.context == null) {
            this.context = parent.getContext();
        }

        View view = LayoutInflater.from(this.context).inflate(R.layout.layout_girl, parent, false);

        final ViewHolder viewHolder = new ViewHolder(view);
        viewHolder.cardView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int position = viewHolder.getAdapterPosition();//获取点击的卡片 id
                Girl girl = girlList.get(position);//获取卡片对应的 Girl 类的实例

                Intent intent = new Intent(context, GirlActivity.class);
                intent.putExtra(GirlActivity.GIRL_NAME, girl.getName());//传递 girl name 参数
                intent.putExtra(GirlActivity.GIRL_IMAGE_ID, girl.getImageId());//传递 girl image id 参数
                context.startActivity(intent);//跳转页面
            }
        });

        return viewHolder;
        //return new ViewHolder(view);
    }

    /* ...... */
}
```

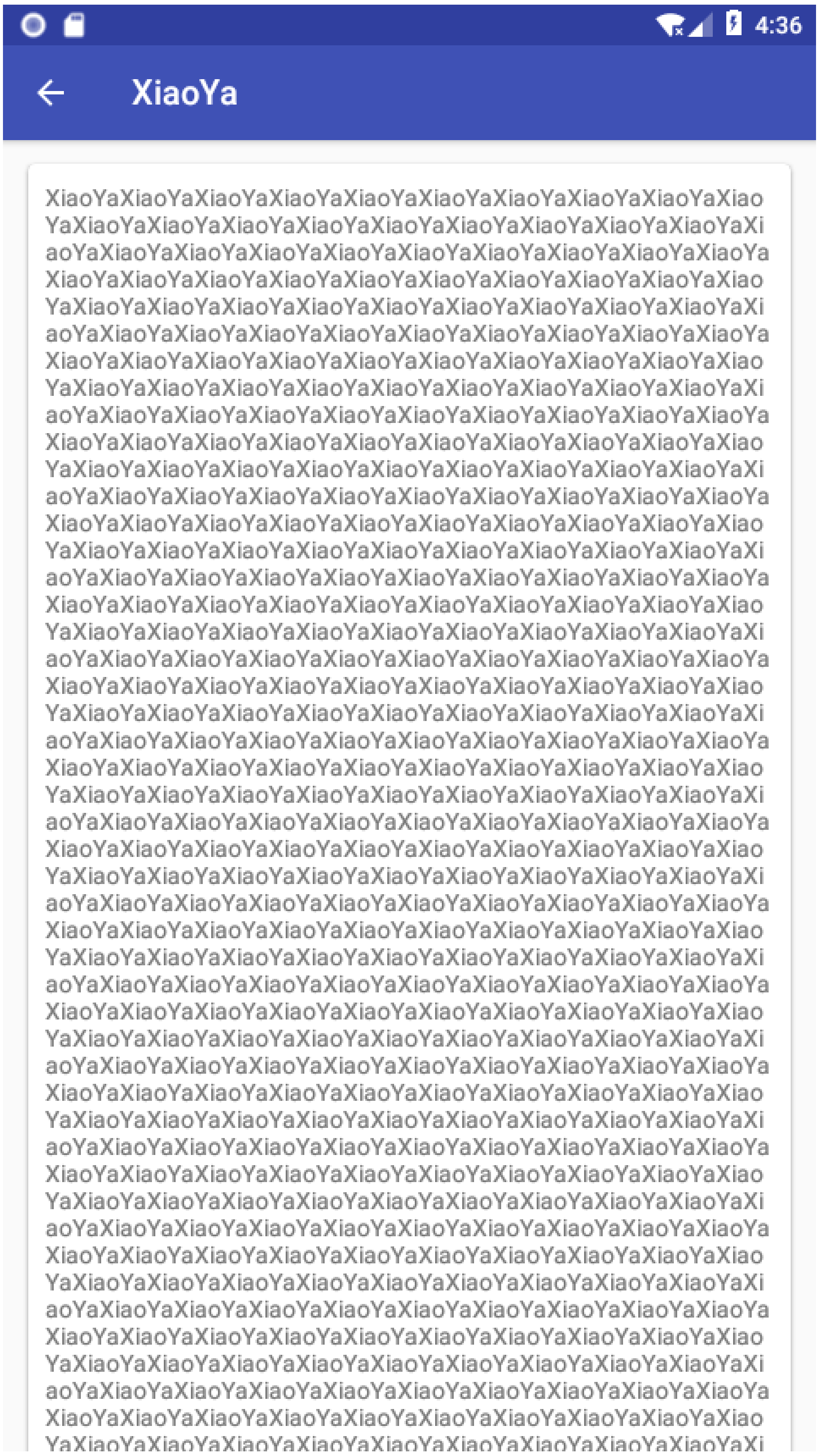


XiaoYa

XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa



XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa



XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi

隐藏状态栏

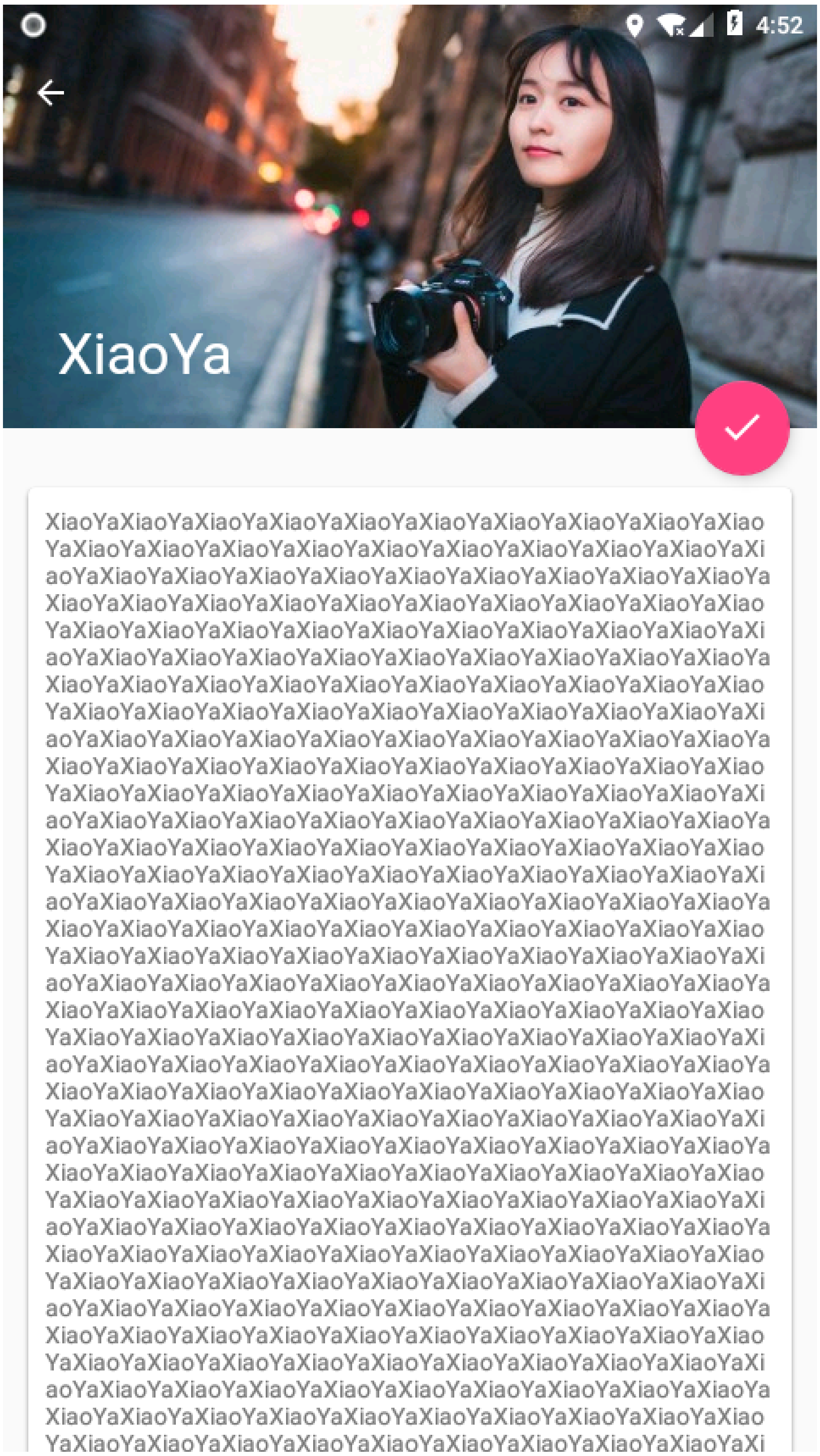
只要为控件或标题添加[**android:fitsSystemWindows="true"**]属性,同时也要在主题设置状态栏颜色为透明,

即可实现隐藏状态栏.

```
<android.support.design.widget.CoordinatorLayout
    ......
    android:fitsSystemWindows="true">

    <!--图片详情页标题栏-->
    <android.support.design.widget.AppBarLayout
        ......
        android:fitsSystemWindows="true">

        <android.support.design.widget.CollapsingToolbarLayout
            ......
            android:fitsSystemWindows="true">

            <ImageView
                ......
                android:fitsSystemWindows="true" />

            <android.support.v7.widget.Toolbar
                ...... />

        </android.support.design.widget.CollapsingToolbarLayout>

    </android.support.design.widget.AppBarLayout>

    < ...... />

</android.support.design.widget.CoordinatorLayout>
```

```
<!----------------------values-v21/styles.xml---------------------->
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <style name="GirlActivityTheme" parent="AppTheme">
        <item name="android:statusBarColor">@android:color/transparent</item>
    </style>
</resources>

<!----------------------values/styles.xml---------------------->
<resources>

    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.NoActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>
```

```xml
    <style name="GirlActivityTheme" parent="AppTheme"></style>

</resources>

<!----------------------AndroidManifest.xml---------------------->
<manifest
    ......>

    <application
        ......
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity
            android:name=".GirlActivity"
            android:theme="@style/GirlActivityTheme" />
    </application>

</manifest>
```



XiaoYa

XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi
aoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYa
XiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiao
YaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXiaoYaXi