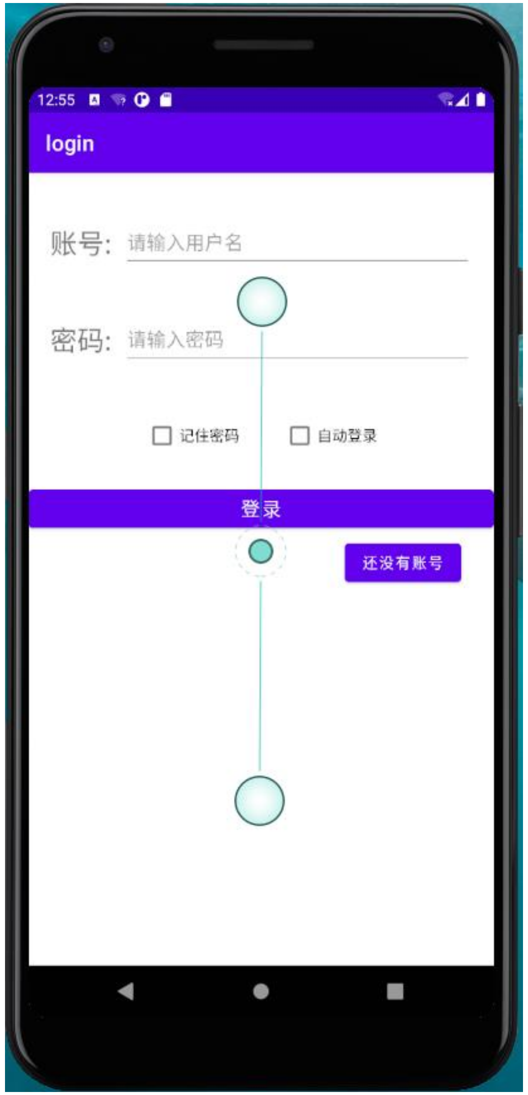
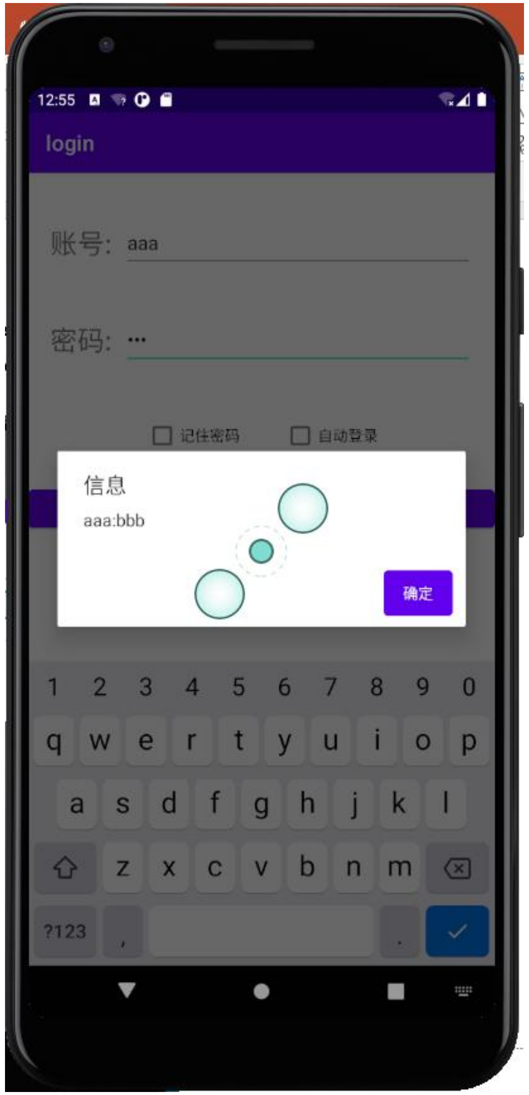
Android实战之登录和注册

案例效果





账号： aaa

密码： •••

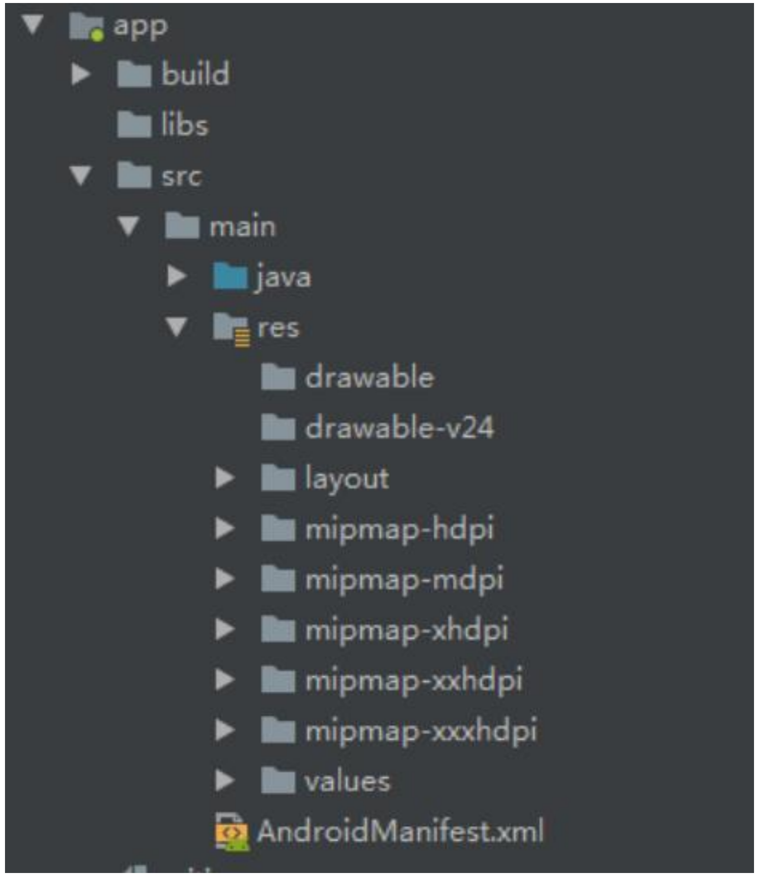
信息

aaa:bbb

本案例能学到的知识

- 1. android基本工程结构
- 2. android布局及基本控件
- 3. android主题和颜色设置
- 4. android代码获得控件
- 5. android基本事件
- 6. android基本对话框和Toast消息
- 7. android页面之间跳转

项目结构说明



```
▼ app
    ▶ build
      libs
    ▼ src
        ▼ main
            ▶ java
            ▼ res
                  drawable
                  drawable-v24
                ▶ layout
                ▶ mipmap-hdpi
                ▶ mipmap-mdpi
                ▶ mipmap-xhdpi
                ▶ mipmap-xxhdpi
                ▶ mipmap-xxxhdpi
                ▶ values
              AndroidManifest.xml
```

app/src/main/assets：放置原生文件，里面的文件会保留原有格式，文件的读取需要通
流
app/src/main/java：项目的源代码
app/src/main/res：项目的资源
　app/src/main/res/anim：存放动画的XML文件
　app/src/main/res/drawable：存放各种位图文件(.png，.jpg，.9png，.gif等)和drawabl
类型的XML文件
　app/src/main/res/drawable-v24：存放自定义Drawables类（Android API 24开始，可
在XML中使用）
　app/src/main/res/layout：存放布局文件
　app/src/main/res/menu：存放菜单文件
　app/src/main/res/mipmap-hdpi：存放高分辨率图片资源
　app/src/main/res/mipmap-mdpi：存放中等分辨率图片资源
　app/src/main/res/mipmap-xdpi：存放超高分辨率图片资源
　app/src/main/res/mipmap-xxdpi：存放超超分辨率图片资源
　app/src/main/res/mipmap-xxxdpi：存放超超超高分辨率图片资源

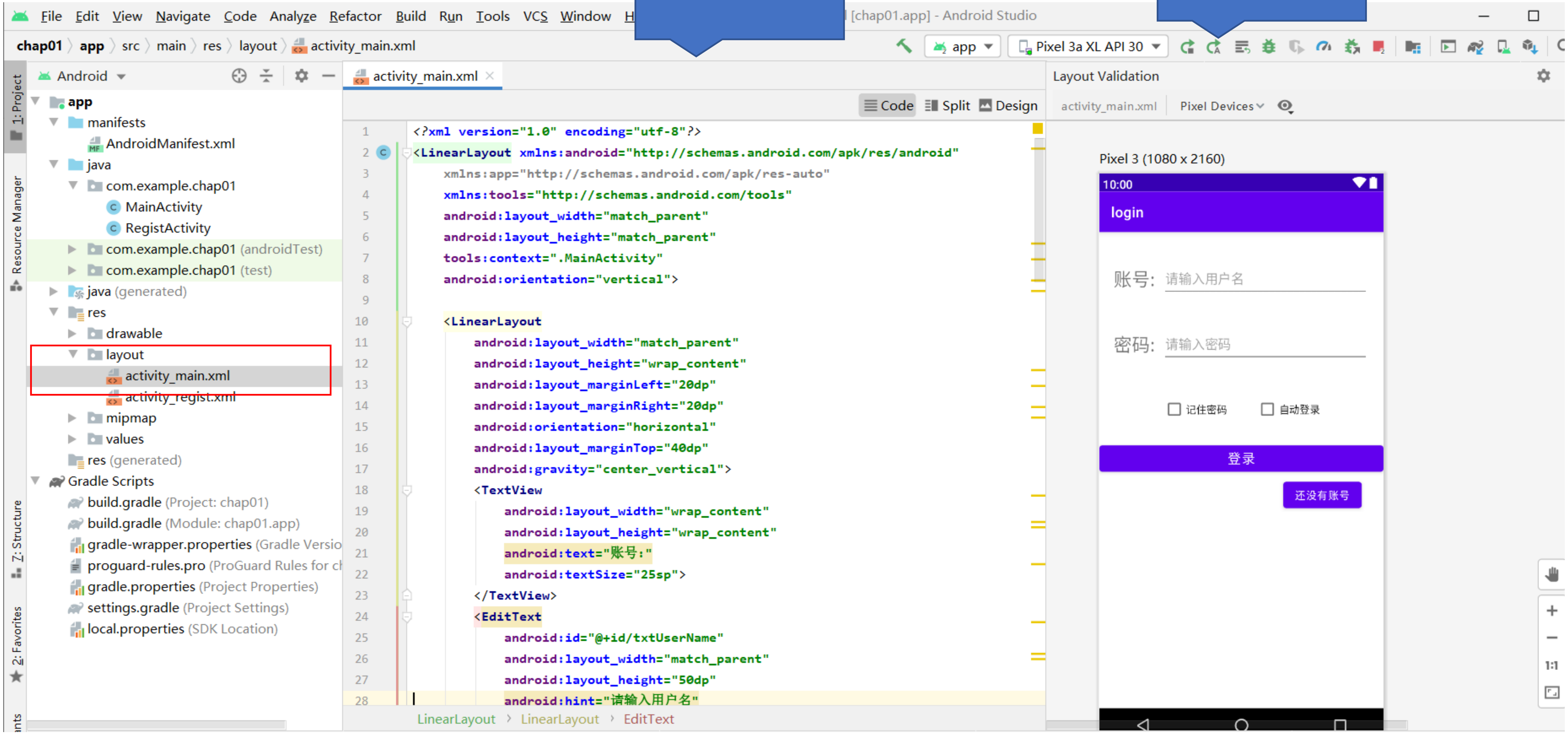
项目结构说明

app/src/main/res/raw：存放各种原生资源(音频，视频，一些XML文件等)
app/src/main/res/values： 存放各种配置资源（颜色，尺寸，样式，字符串等）
app/src/main/res/values/attrs.xml：自定义控件时用的较多，自定义控件的属性
app/src/main/res/values/arrays.xml：定义数组资源
app/src/main/res/values/colors.xml：定义颜色资源
app/src/main/res/values/dimens.xml：定义尺寸资源
app/src/main/res/values/string.xml：定义字符串资源
app/src/main/res/values/styles.xml：定义样式资源
app/src/main/res/values-v11：在API 11+的设备上调用
app/src/main/res/values-v14：在API 14+的设备上调用
app/src/main/res/values-v21：在API 21+的设备上调用
app/src/main/res/AndroidManifest.xml：项目的清单文件（名称、版本、SDK、权限等配置信息）

UI文件

UI代码

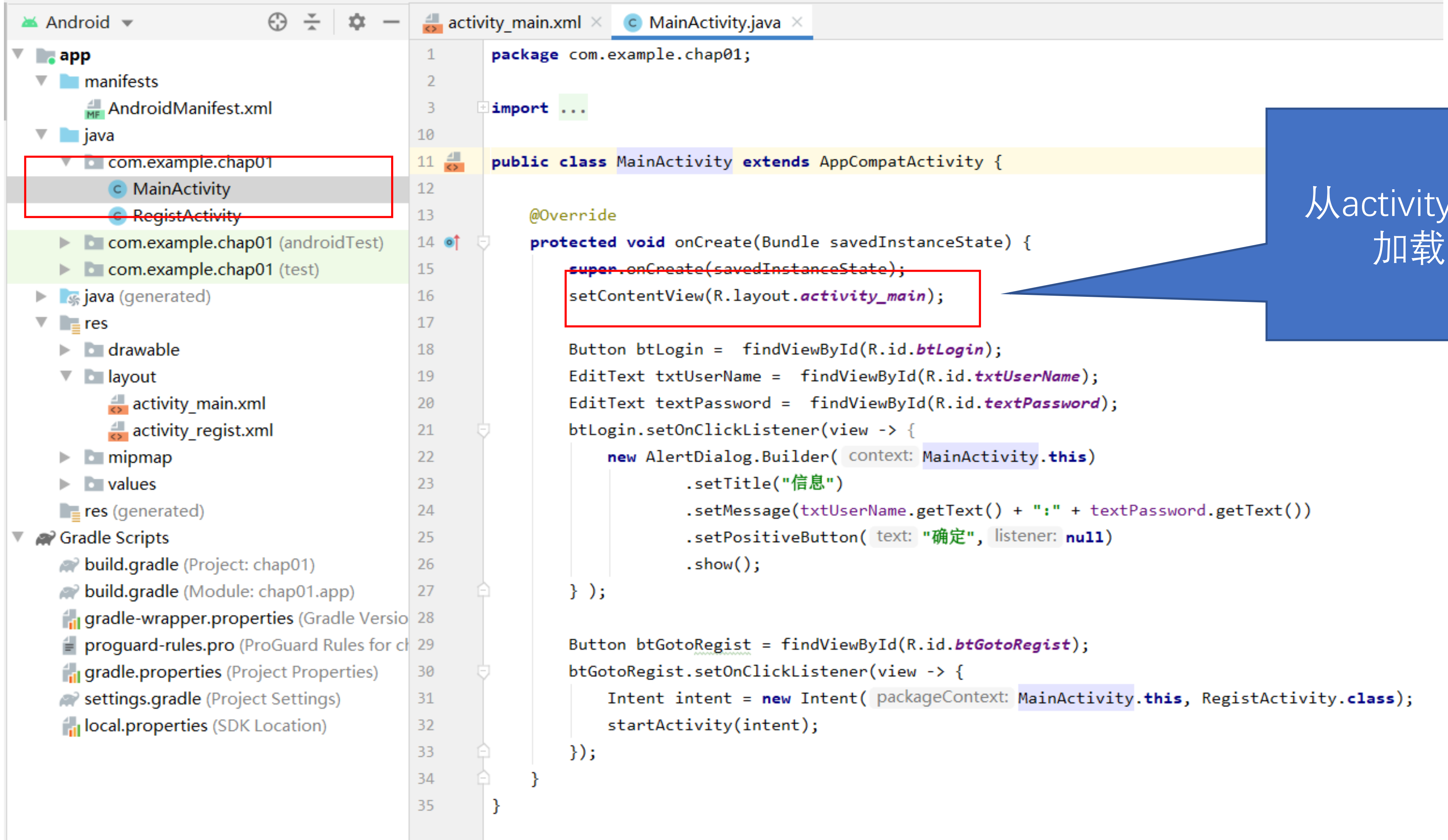
UI预览



File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window H
[chap01.app] - Android Studio
chap01 › app › src › main › res › layout › activity_main.xml
app
Pixel 3a XL API 30
Android
app
manifests
AndroidManifest.xml
java
com.example.chap01
MainActivity
RegistActivity
com.example.chap01 (androidTest)
com.example.chap01 (test)
java (generated)
res
drawable
layout
activity_main.xml
activity_regist.xml
mipmap
values
res (generated)
Gradle Scripts
build.gradle (Project: chap01)
build.gradle (Module: chap01.app)
gradle-wrapper.properties (Gradle Versio
proguard-rules.pro (ProGuard Rules for ch
gradle.properties (Project Properties)
settings.gradle (Project Settings)
local.properties (SDK Location)
1: Project
Resource Manager
7: Structure
2: Favorites
activity_main.xml
Code
Split
Design
Layout Validation
activity_main.xml
Pixel Devices
Pixel 3 (1080 x 2160)
10:00
login
账号:
请输入用户名
密码:
请输入密码
☐ 记住密码
☐ 自动登录
登录
还没有账号
LinearLayout › LinearLayout › EditText

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

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:orientation="horizontal"
        android:layout_marginTop="40dp"
        android:gravity="center_vertical">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="账号:"
            android:textSize="25sp">
        </TextView>
        <EditText
            android:id="@+id/txtUserName"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:hint="请输入用户名"
```

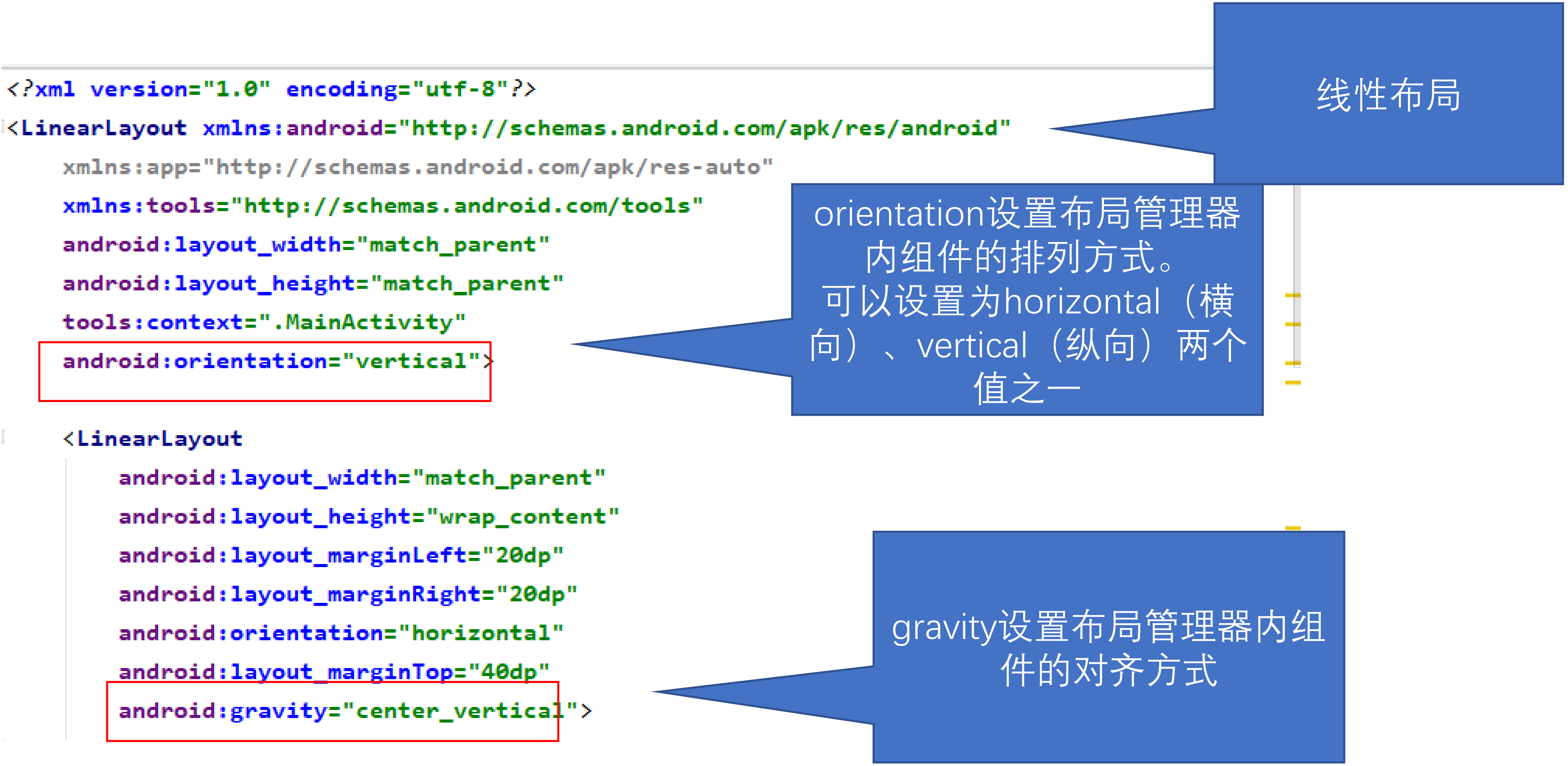
UI加载方法


Android
app
manifests
AndroidManifest.xml
java
com.example.chap01
MainActivity
RegistActivity
com.example.chap01 (androidTest)
com.example.chap01 (test)
java (generated)
res
drawable
layout
activity_main.xml
activity_regist.xml
mipmap
values
res (generated)
Gradle Scripts
build.gradle (Project: chap01)
build.gradle (Module: chap01.app)
gradle-wrapper.properties (Gradle Versio
proguard-rules.pro (ProGuard Rules for ch
gradle.properties (Project Properties)
settings.gradle (Project Settings)
local.properties (SDK Location)
activity_main.xml
MainActivity.java

```java
package com.example.chap01;

import ...

public class MainActivity extends AppCompatActivity {

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

        Button btLogin =  findViewById(R.id.btLogin);
        EditText txtUserName =  findViewById(R.id.txtUserName);
        EditText textPassword =  findViewById(R.id.textPassword);
        btLogin.setOnClickListener(view -> {
            new AlertDialog.Builder( context: MainActivity.this)
                    .setTitle("信息")
                    .setMessage(txtUserName.getText() + ":" + textPassword.getText())
                    .setPositiveButton( text: "确定", listener: null)
                    .show();
        } );

        Button btGotoRegist = findViewById(R.id.btGotoRegist);
        btGotoRegist.setOnClickListener(view -> {
            Intent intent = new Intent( packageContext: MainActivity.this, RegistActivity.class);
            startActivity(intent);
        });
    }
}
```

从activity_main
加载UI

案例说明-UI设计



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

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:orientation="horizontal"
        android:layout_marginTop="40dp"
        android:gravity="center_vertical">
```

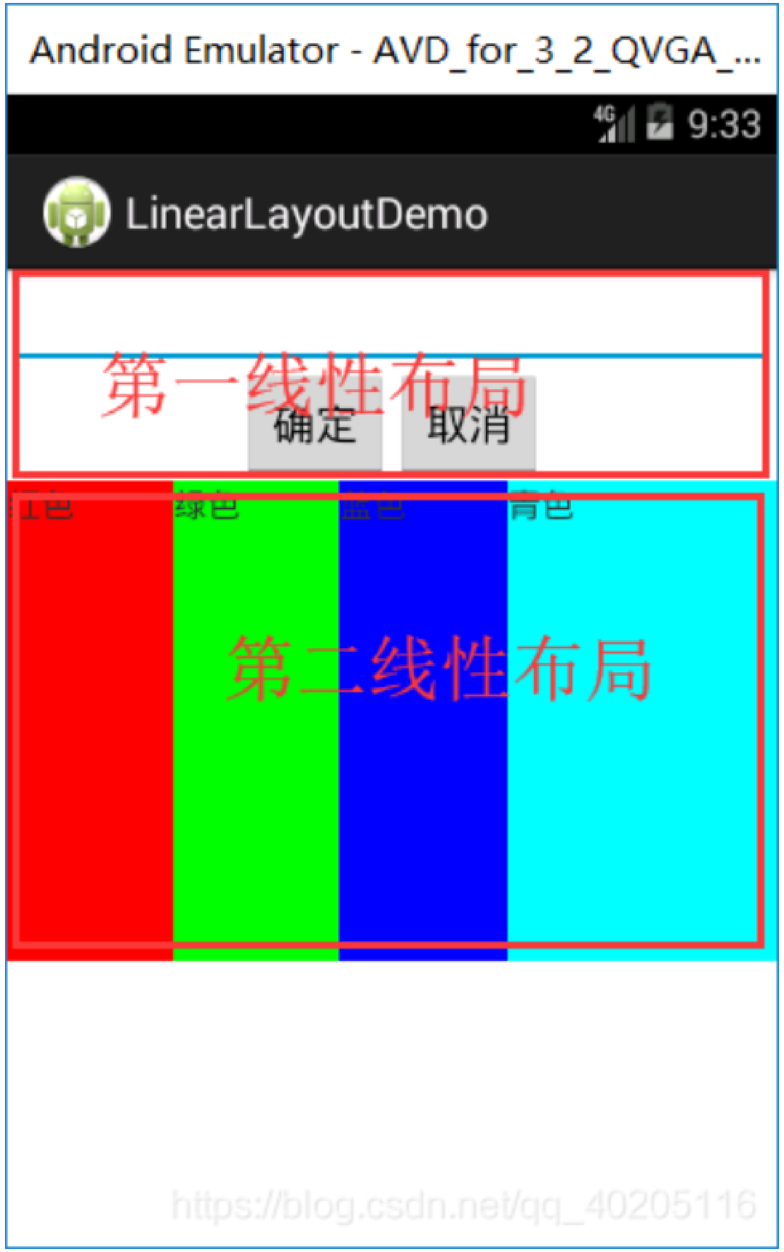
线性布局

orientation设置布局管理器内组件的排列方式。
可以设置为horizontal（横向）、vertical（纵向）两个值之一

gravity设置布局管理器内组件的对齐方式

Android布局

Android六大基本布局分别是：
线性布局LinearLayout
表格布局TableLayout
相对布局RelativeLayout
层布局FrameLayout
绝对布局AbsoluteLayout

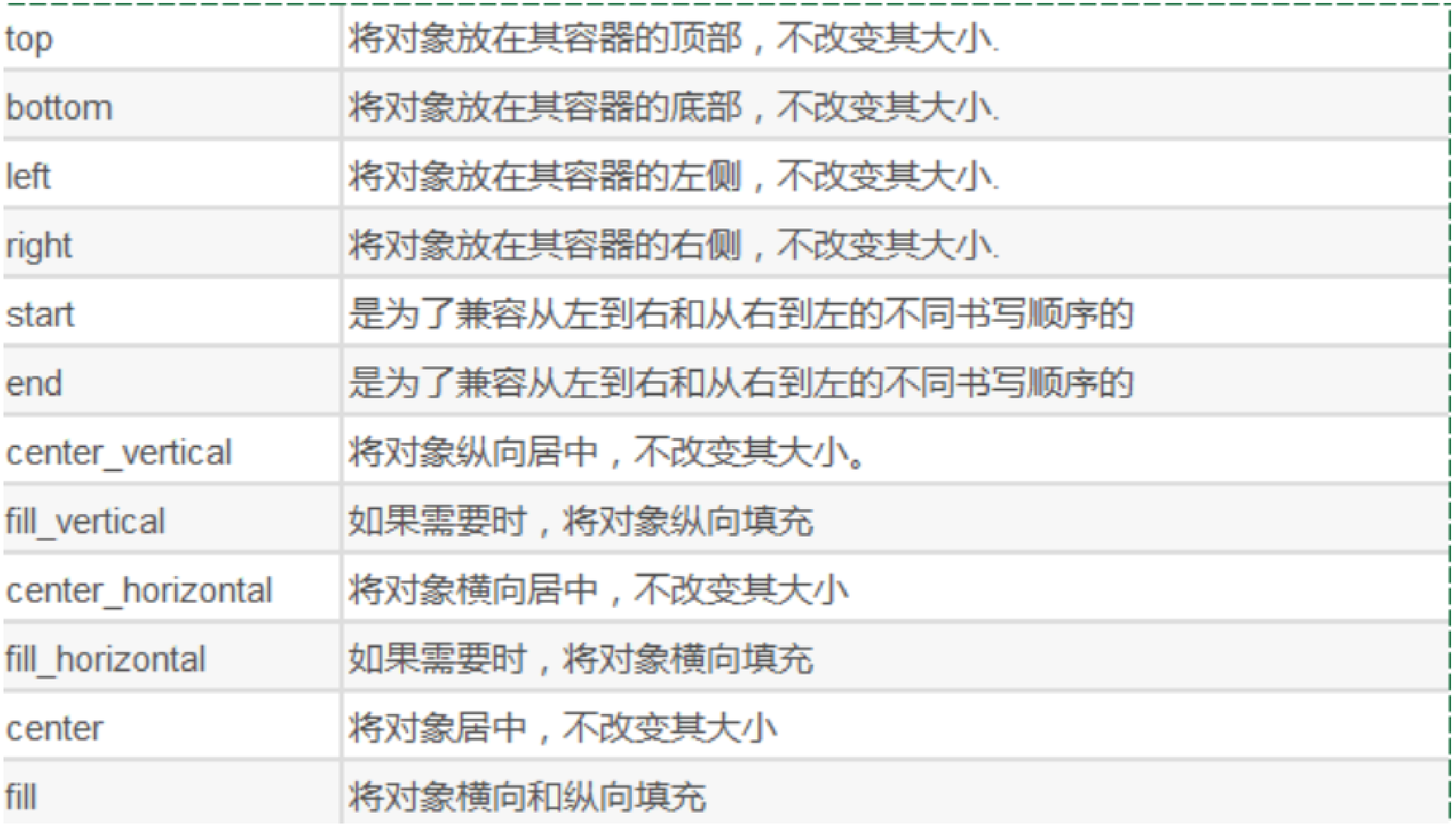


wrap_content,match_parent

- wrap_content:是layout_width和layout_height的属性值之一，表示和自身内容一样的长度。
- match_parent:是layout_width和layout_height的属性值之一，表示和父组件一样的长度。

Gravity属性

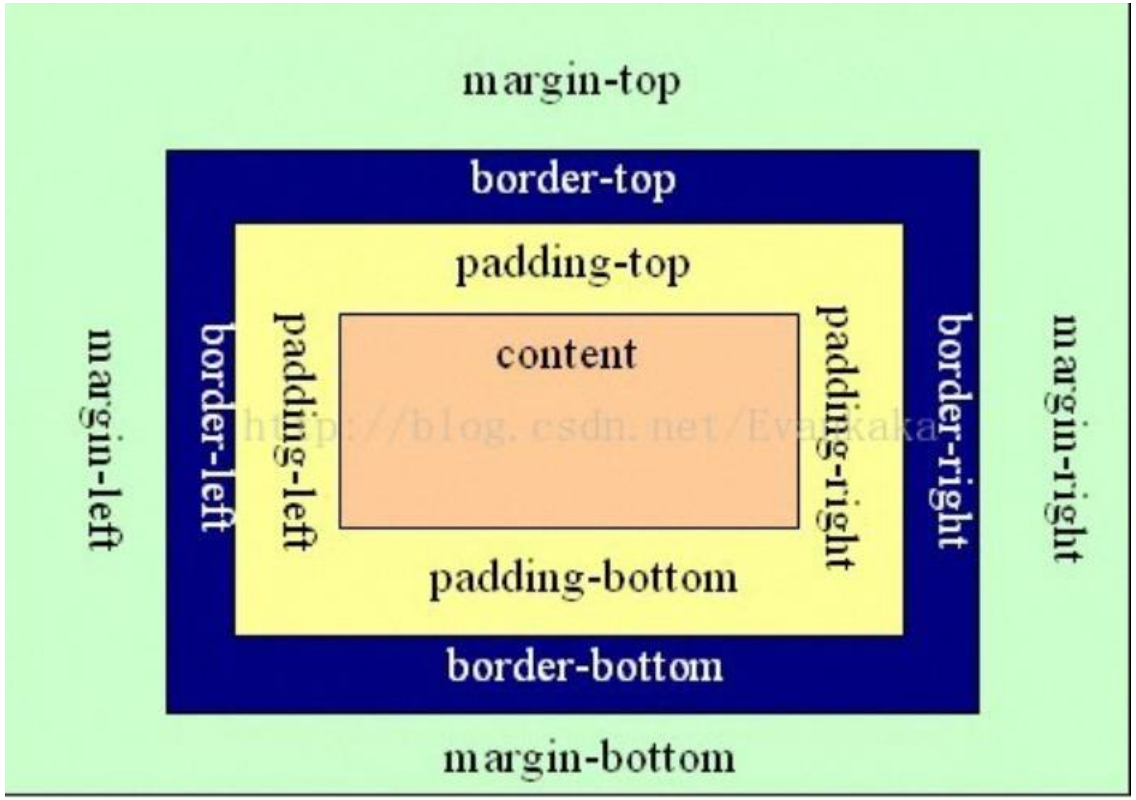
- gravity设置布局管理器内组件的对齐方式，layout_gravity控制自己在父元素的位置



top	将对象放在其容器的顶部，不改变其大小.
bottom	将对象放在其容器的底部，不改变其大小.
left	将对象放在其容器的左侧，不改变其大小.
right	将对象放在其容器的右侧，不改变其大小.
start	是为了兼容从左到右和从右到左的不同书写顺序的
end	是为了兼容从左到右和从右到左的不同书写顺序的
center_vertical	将对象纵向居中，不改变其大小。
fill_vertical	如果需要时，将对象纵向填充
center_horizontal	将对象横向居中，不改变其大小
fill_horizontal	如果需要时，将对象横向填充
center	将对象居中，不改变其大小
fill	将对象横向和纵向填充

padding与layout_margin

- android:layout_margin就是设置view的上下左右边框的额外空间
- android:padding是设置内容相对view的边框的距离

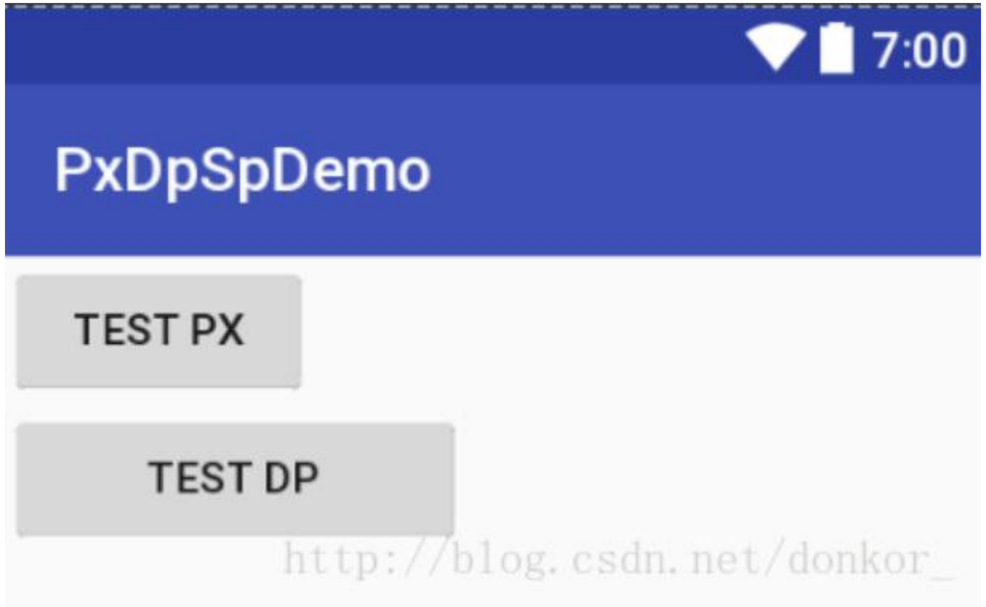


Android中PX、DP、SP

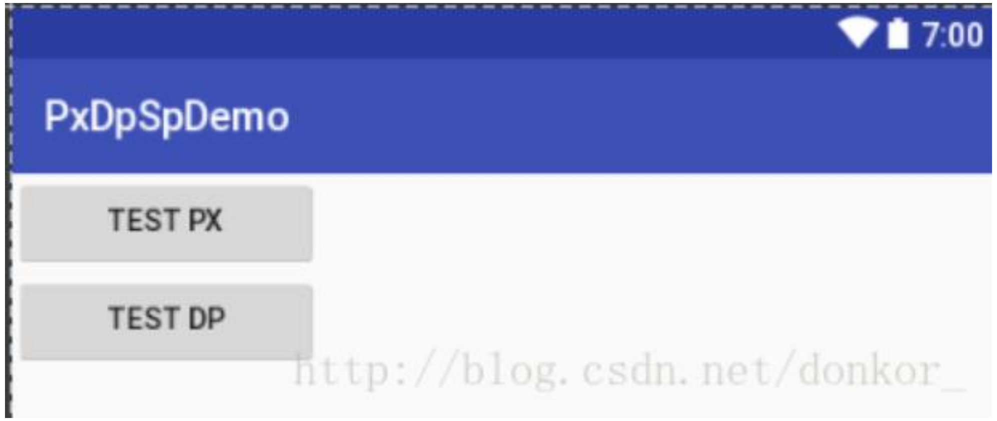
- px： 其实就是像素单位，比如我们通常说的手机分辨列表800*400都是px的单位
- sp： 同dp相似，还会根据用户的字体大小偏好来缩放
- dp： 虚拟像素，在不同的像素密度的设备上会自动适配
- dip： 同dp

Android中PX、DP、SP

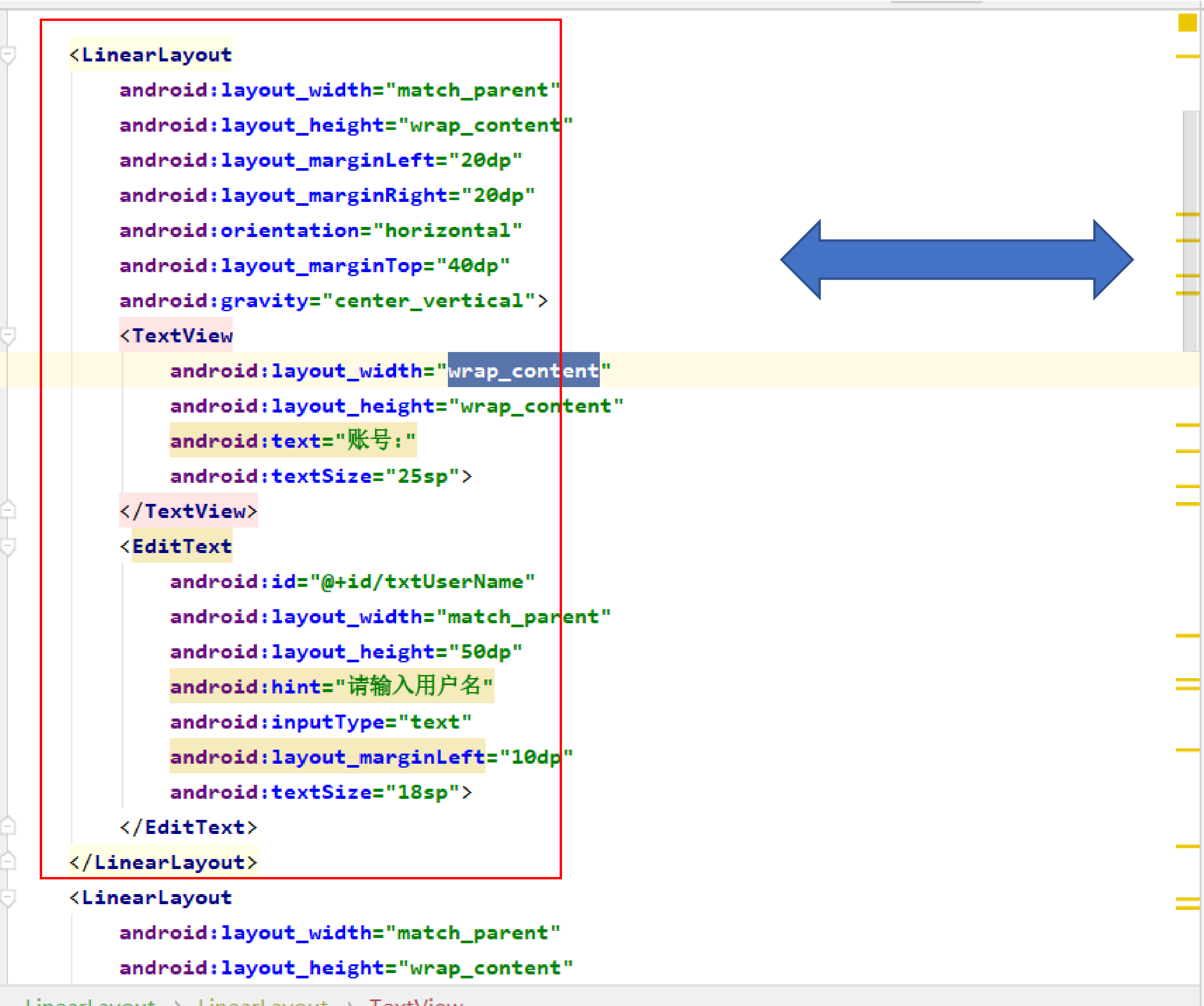
在480*800分辨率中，3.7屏幕对角线英寸数的设备效果图如下



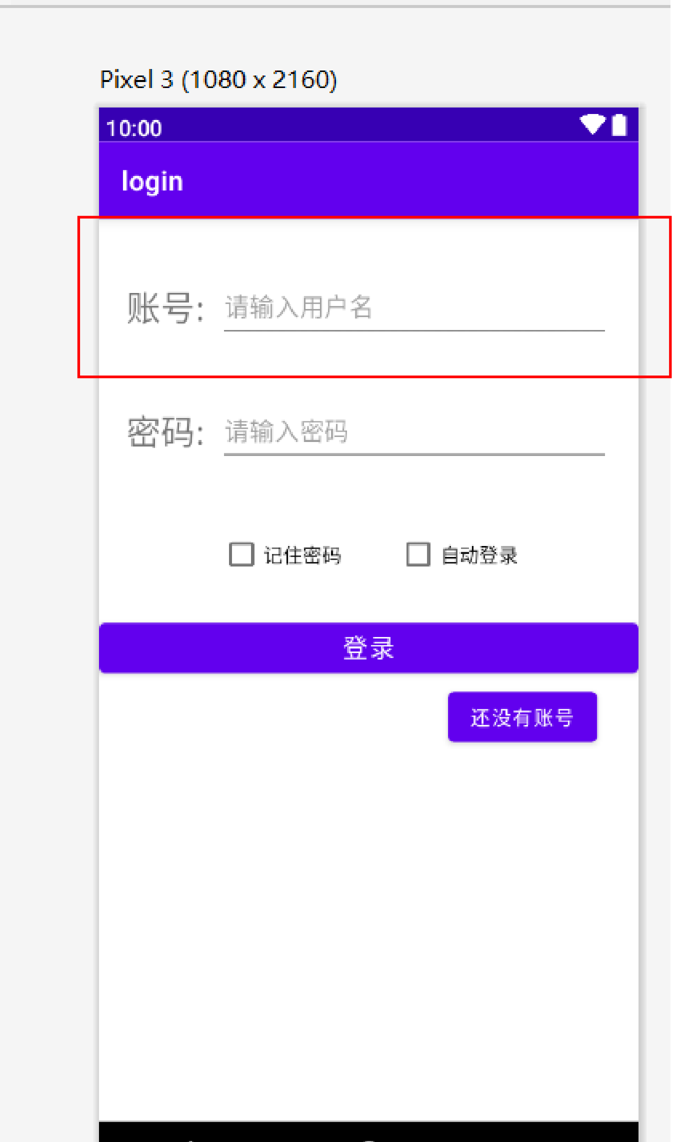
在480*800分辨率中，5.1屏幕对角线英寸数的设备效果图如下



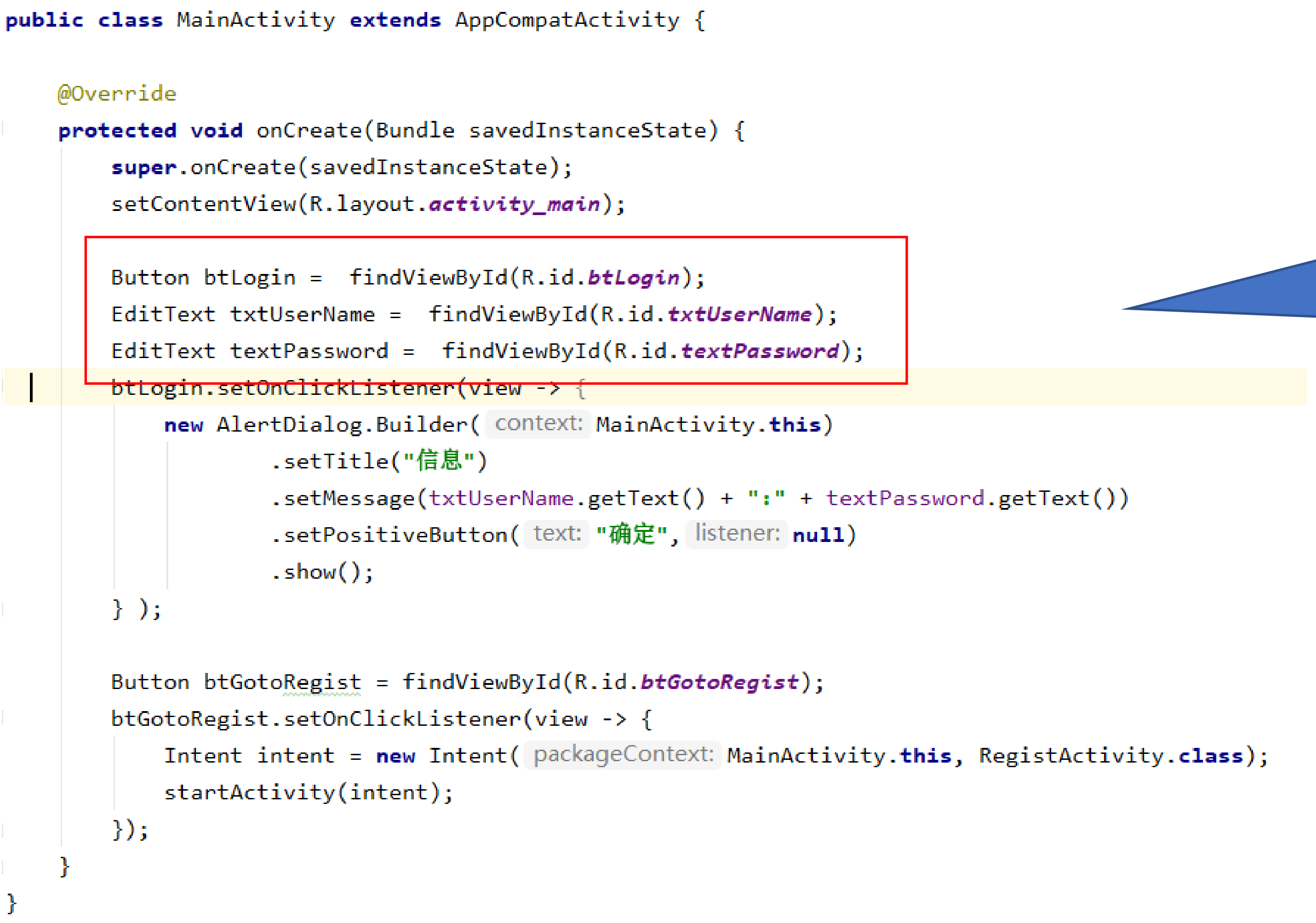
案例说明



```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:orientation="horizontal"
    android:layout_marginTop="40dp"
    android:gravity="center_vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="账号:"
        android:textSize="25sp">
    </TextView>
    <EditText
        android:id="@+id/txtUserName"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:hint="请输入用户名"
        android:inputType="text"
        android:layout_marginLeft="10dp"
        android:textSize="18sp">
    </EditText>
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
```



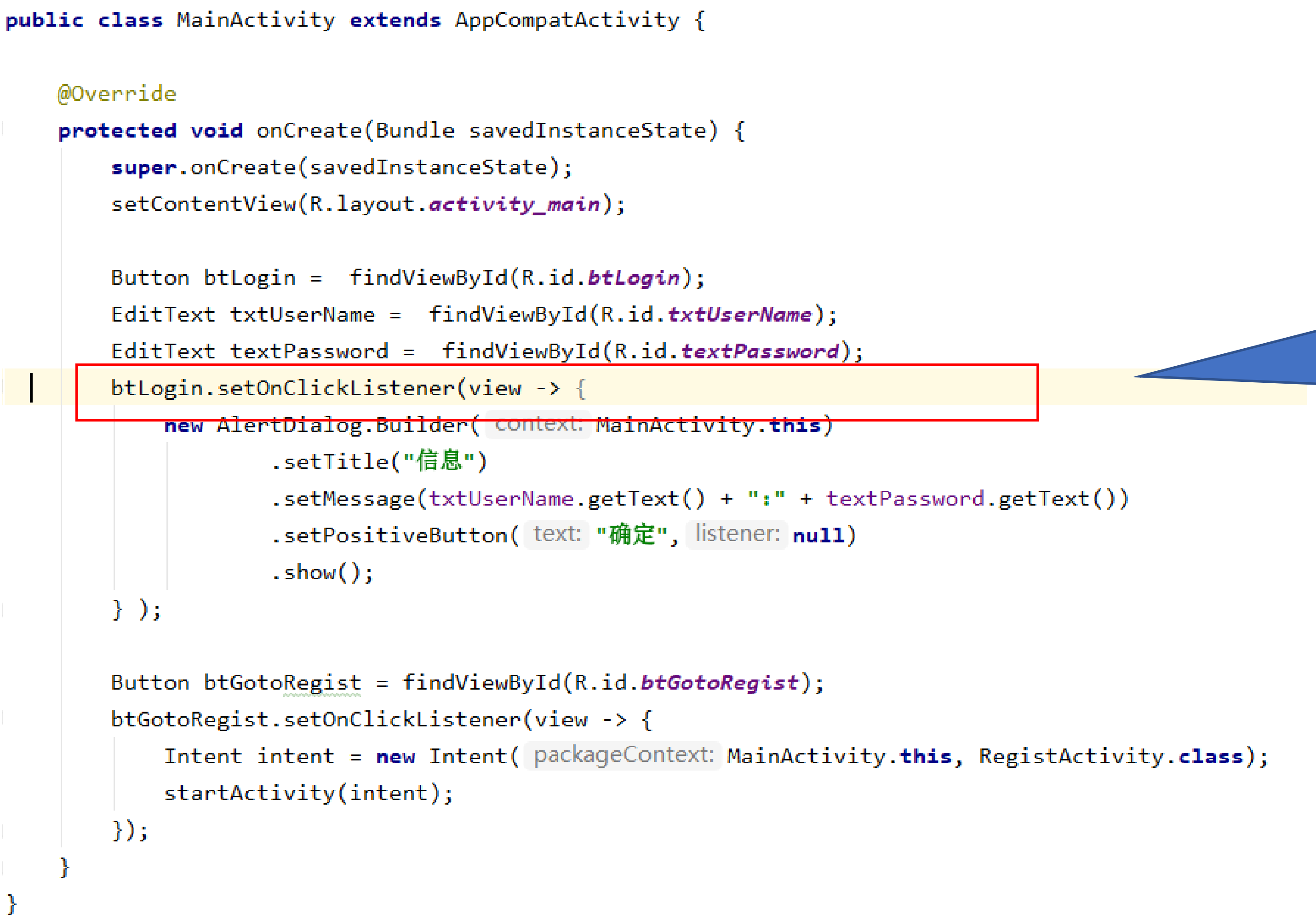
findViewById



```
public class MainActivity extends AppCompatActivity {

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

        Button btLogin =  findViewById(R.id.btLogin);
        EditText txtUserName =  findViewById(R.id.txtUserName);
        EditText textPassword =  findViewById(R.id.textPassword);
        btLogin.setOnClickListener(view -> {
            new AlertDialog.Builder( context: MainActivity.this)
                    .setTitle("信息")
                    .setMessage(txtUserName.getText() + ":" + textPassword.getText())
                    .setPositiveButton( text: "确定", listener: null)
                    .show();
        } );

        Button btGotoRegist = findViewById(R.id.btGotoRegist);
        btGotoRegist.setOnClickListener(view -> {
            Intent intent = new Intent( packageContext: MainActivity.this, RegistActivity.class);
            startActivity(intent);
        });
    }
}
```

findViewById通过控件的id获得控件对象

Android事件

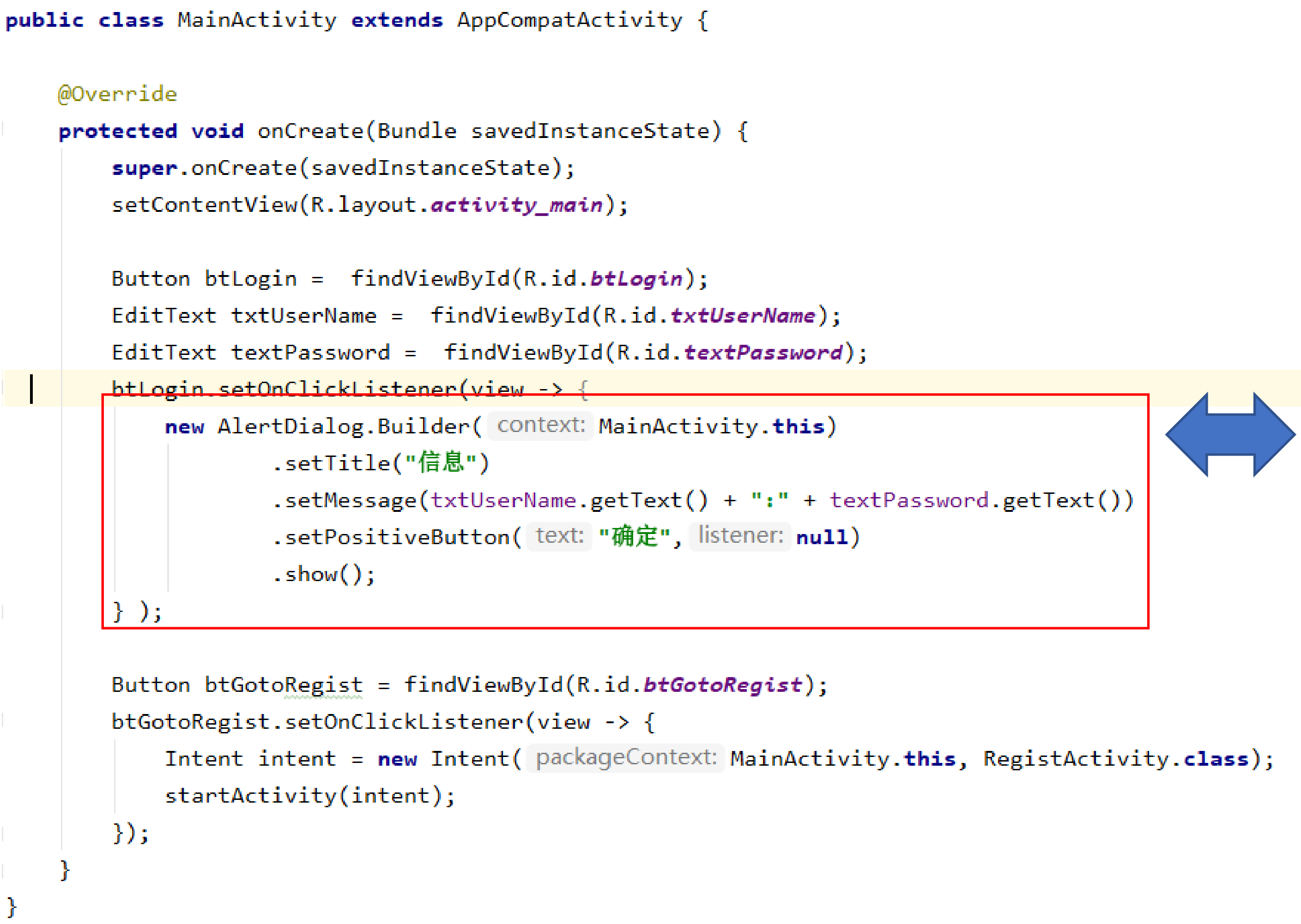


```
public class MainActivity extends AppCompatActivity {

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

        Button btLogin =  findViewById(R.id.btLogin);
        EditText txtUserName =  findViewById(R.id.txtUserName);
        EditText textPassword =  findViewById(R.id.textPassword);
        btLogin.setOnClickListener(view -> {
            new AlertDialog.Builder( context: MainActivity.this)
                    .setTitle("信息")
                    .setMessage(txtUserName.getText() + ":" + textPassword.getText())
                    .setPositiveButton( text: "确定", listener: null)
                    .show();
        } );

        Button btGotoRegist = findViewById(R.id.btGotoRegist);
        btGotoRegist.setOnClickListener(view -> {
            Intent intent = new Intent( packageContext: MainActivity.this, RegistActivity.class);
            startActivity(intent);
        });
    }
}
```

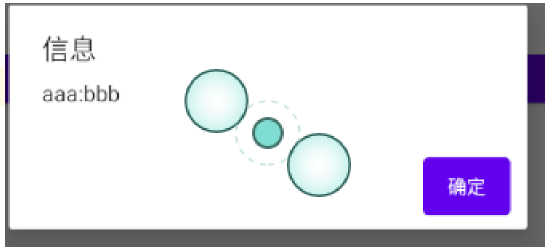
Android通过xxxListener
设置事件

Click为点击事件

Android Alter弹窗



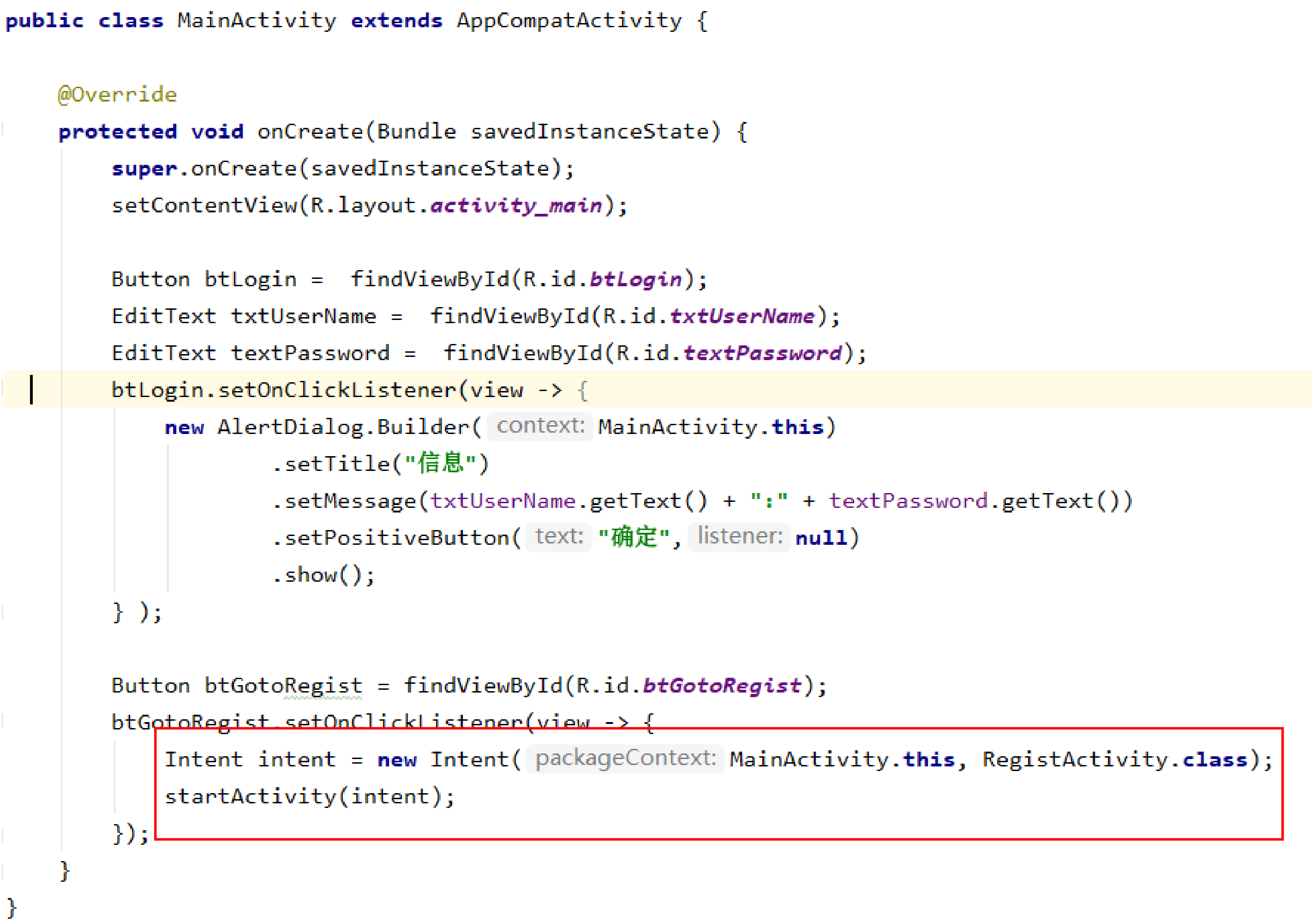
```
public class MainActivity extends AppCompatActivity {

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

        Button btLogin =  findViewById(R.id.btLogin);
        EditText txtUserName =  findViewById(R.id.txtUserName);
        EditText textPassword =  findViewById(R.id.textPassword);
        btLogin.setOnClickListener(view -> {
            new AlertDialog.Builder( context: MainActivity.this)
                    .setTitle("信息")
                    .setMessage(txtUserName.getText() + ":" + textPassword.getText())
                    .setPositiveButton( text: "确定", listener: null)
                    .show();
        } );

        Button btGotoRegist = findViewById(R.id.btGotoRegist);
        btGotoRegist.setOnClickListener(view -> {
            Intent intent = new Intent( packageContext: MainActivity.this, RegistActivity.class);
            startActivity(intent);
        });
    }
}
```



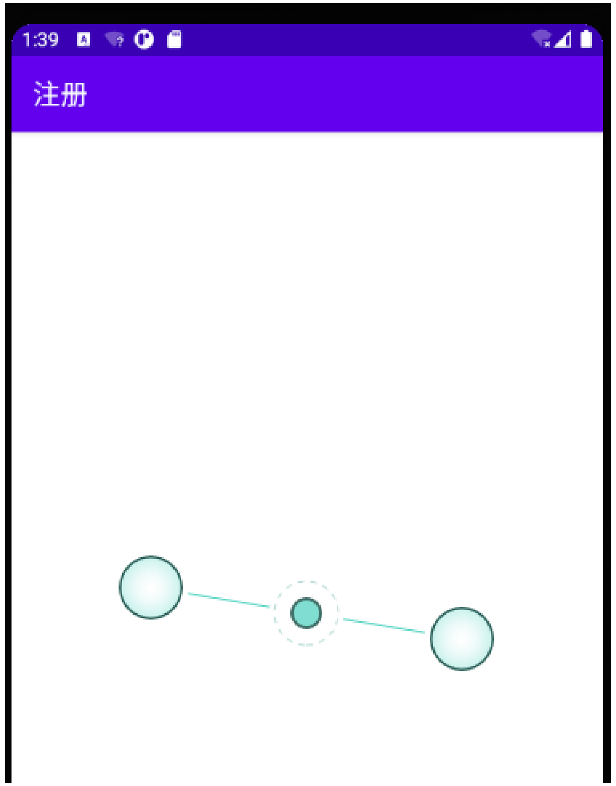
信息

aaa:bbb

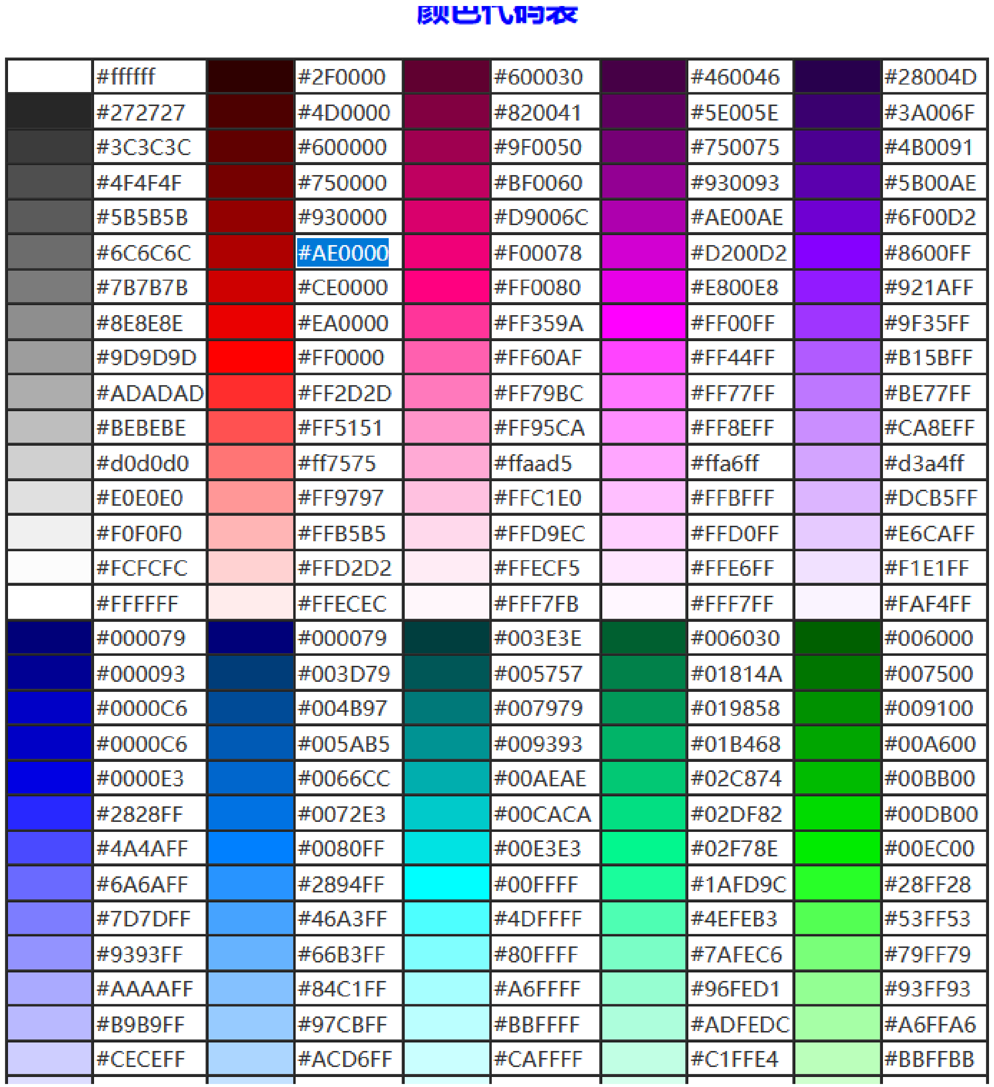
Intent 页面跳转



```
public class MainActivity extends AppCompatActivity {

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

        Button btLogin =  findViewById(R.id.btLogin);
        EditText txtUserName =  findViewById(R.id.txtUserName);
        EditText textPassword =  findViewById(R.id.textPassword);
        btLogin.setOnClickListener(view -> {
            new AlertDialog.Builder( context: MainActivity.this)
                    .setTitle("信息")
                    .setMessage(txtUserName.getText() + ":" + textPassword.getText())
                    .setPositiveButton( text: "确定", listener: null)
                    .show();
        } );

        Button btGotoRegist = findViewById(R.id.btGotoRegist);
        btGotoRegist.setOnClickListener(view -> {
            Intent intent = new Intent( packageContext: MainActivity.this, RegistActivity.class);
            startActivity(intent);
        });
    }
}
```

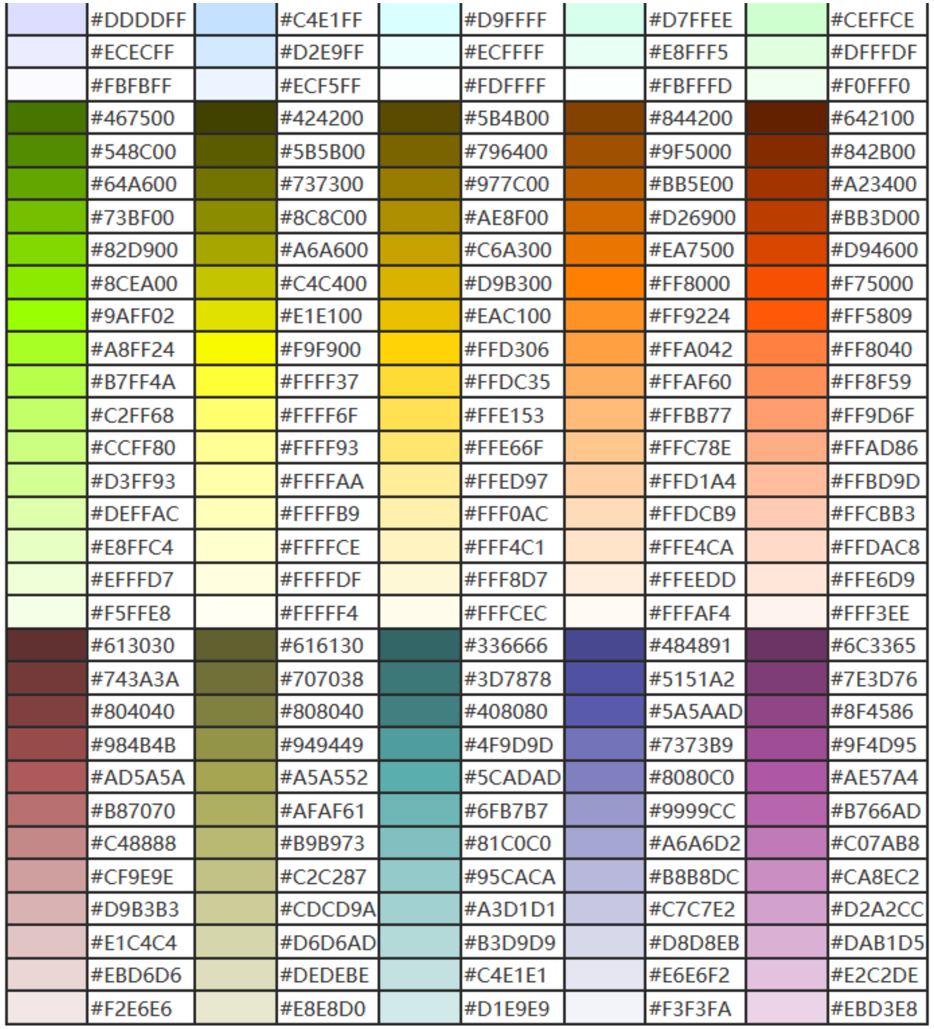


颜色数值表



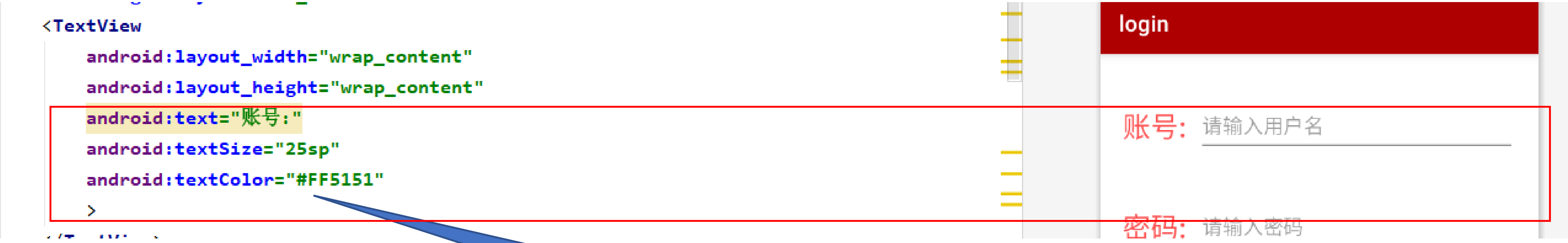
颜色代码表

#ffffff	#2F0000	#600030	#460046	#28004D
#272727	#4D0000	#820041	#5E005E	#3A006F
#3C3C3C	#600000	#9F0050	#750075	#4B0091
#4F4F4F	#750000	#BF0060	#930093	#5B00AE
#5B5B5B	#930000	#D9006C	#AE00AE	#6F00D2
#6C6C6C	#AE0000	#F00078	#D200D2	#8600FF
#7B7B7B	#CE0000	#FF0080	#E800E8	#921AFF
#8E8E8E	#EA0000	#FF359A	#FF00FF	#9F35FF
#9D9D9D	#FF0000	#FF60AF	#FF44FF	#B15BFF
#ADADAD	#FF2D2D	#FF79BC	#FF77FF	#BE77FF
#BEBEBE	#FF5151	#FF95CA	#FF8EFF	#CA8EFF
#d0d0d0	#ff7575	#ffaad5	#ffa6ff	#d3a4ff
#E0E0E0	#FF9797	#FFC1E0	#FFBFFF	#DCB5FF
#F0F0F0	#FFB5B5	#FFD9EC	#FFD0FF	#E6CAFF
#FCFCFC	#FFD2D2	#FFECF5	#FFE6FF	#F1E1FF
#FFFFFF	#FFECEC	#FFF7FB	#FFF7FF	#FAF4FF
#000079	#000079	#003E3E	#006030	#006000
#000093	#003D79	#005757	#01814A	#007500
#0000C6	#004B97	#007979	#019858	#009100
#0000C6	#005AB5	#009393	#01B468	#00A600
#0000E3	#0066CC	#00AEAE	#02C874	#00BB00
#2828FF	#0072E3	#00CACA	#02DF82	#00DB00
#4A4AFF	#0080FF	#00E3E3	#02F78E	#00EC00
#6A6AFF	#2894FF	#00FFFF	#1AFD9C	#28FF28
#7D7DFF	#46A3FF	#4DFFFF	#4EFEB3	#53FF53
#9393FF	#66B3FF	#80FFFF	#7AFEC6	#79FF79
#AAAAFF	#84C1FF	#A6FFFF	#96FED1	#93FF93
#B9B9FF	#97CBFF	#BBFFFF	#ADFEDC	#A6FFA6
#CECEFF	#ACD6FF	#CAFFFF	#C1FFE4	#BBFFBB



#DDDDFF	#C4E1FF	#D9FFFF	#D7FFEE	#CEFFCE
#ECECFF	#D2E9FF	#ECFFFF	#E8FFF5	#DFFFDF
#FBFBFF	#ECF5FF	#FDFFFF	#FBFFFD	#F0FFF0
#467500	#424200	#5B4B00	#844200	#642100
#548C00	#5B5B00	#796400	#9F5000	#842B00
#64A600	#737300	#977C00	#BB5E00	#A23400
#73BF00	#8C8C00	#AE8F00	#D26900	#BB3D00
#82D900	#A6A600	#C6A300	#EA7500	#D94600
#8CEA00	#C4C400	#D9B300	#FF8000	#F75000
#9AFF02	#E1E100	#EAC100	#FF9224	#FF5809
#A8FF24	#F9F900	#FFD306	#FFA042	#FF8040
#B7FF4A	#FFFF37	#FFDC35	#FFAF60	#FF8F59
#C2FF68	#FFFF6F	#FFE153	#FFBB77	#FF9D6F
#CCFF80	#FFFF93	#FFE66F	#FFC78E	#FFAD86
#D3FF93	#FFFFAA	#FFED97	#FFD1A4	#FFBD9D
#DEFFAC	#FFFFB9	#FFF0AC	#FFDCB9	#FFCBB3
#E8FFC4	#FFFFCE	#FFF4C1	#FFE4CA	#FFDAC8
#EFFFD7	#FFFFDF	#FFF8D7	#FFEEDD	#FFE6D9
#F5FFE8	#FFFFF4	#FFFCEC	#FFFAF4	#FFF3EE
#613030	#616130	#336666	#484891	#6C3365
#743A3A	#707038	#3D7878	#5151A2	#7E3D76
#804040	#808040	#408080	#5A5AAD	#8F4586
#984B4B	#949449	#4F9D9D	#7373B9	#9F4D95
#AD5A5A	#A5A552	#5CADAD	#8080C0	#AE57A4
#B87070	#AFAF61	#6FB7B7	#9999CC	#B766AD
#C48888	#B9B973	#81C0C0	#A6A6D2	#C07AB8
#CF9E9E	#C2C287	#95CACA	#B8B8DC	#CA8EC2
#D9B3B3	#CDCD9A	#A3D1D1	#C7C7E2	#D2A2CC
#E1C4C4	#D6D6AD	#B3D9D9	#D8D8EB	#DAB1D5
#EBD6D6	#DEDEBE	#C4E1E1	#E6E6F2	#E2C2DE
#F2E6E6	#E8E8D0	#D1E9E9	#F3F3FA	#EBD3E8

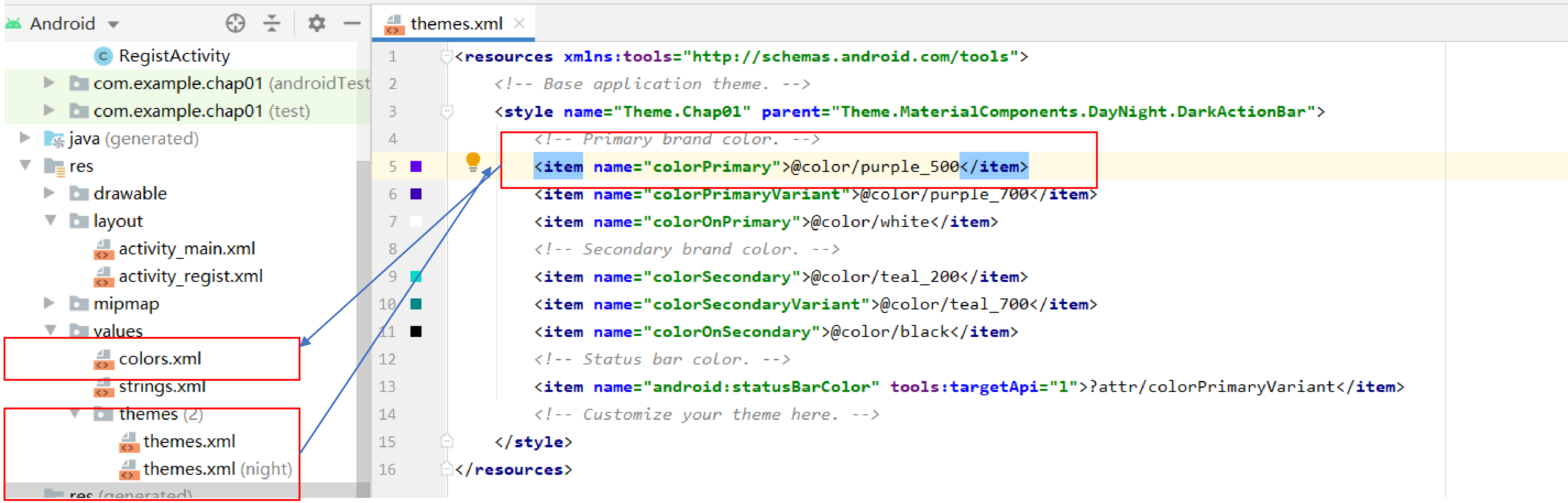
修改字体颜色



```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="账号:"
    android:textSize="25sp"
    android:textColor="#FF5151"
    >
</TextView>
```

修改字体颜色通过颜色数值即可修改

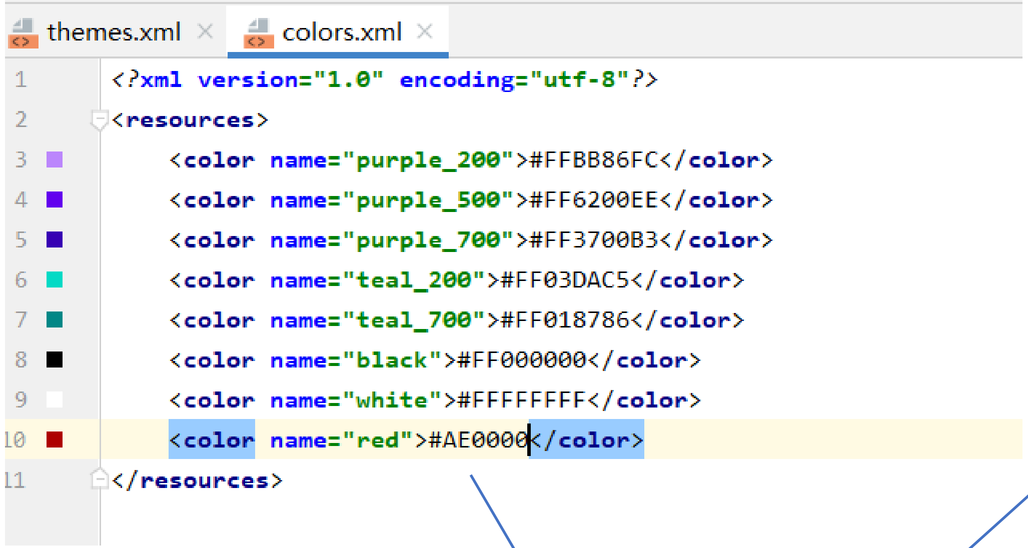
修改主题皮肤



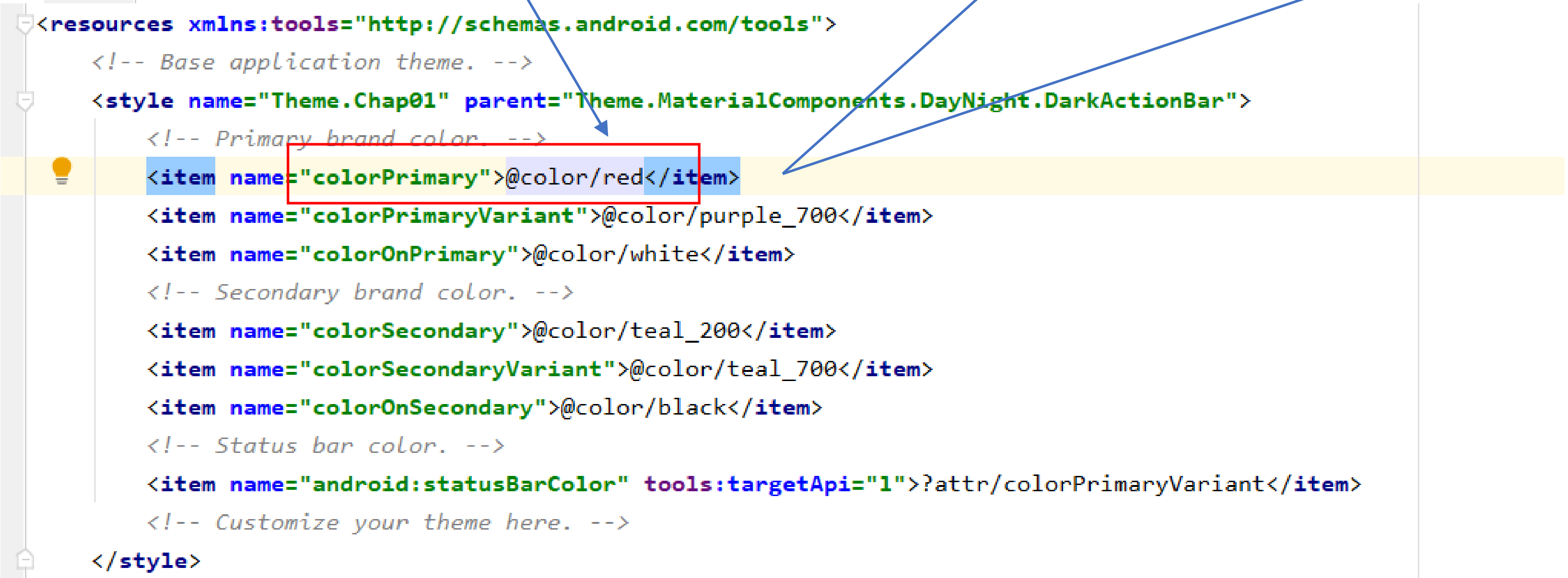
```xml
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Theme.Chap01" parent="Theme.MaterialComponents.DayNight.DarkActionBar">
        <!-- Primary brand color. -->
        <item name="colorPrimary">@color/purple_500</item>
        <item name="colorPrimaryVariant">@color/purple_700</item>
        <item name="colorOnPrimary">@color/white</item>
        <!-- Secondary brand color. -->
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/teal_700</item>
        <item name="colorOnSecondary">@color/black</item>
        <!-- Status bar color. -->
        <item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</item>
        <!-- Customize your theme here. -->
    </style>
</resources>
```

Themes是android的主体，注意内定义的颜色使用的是colors中的颜色定义

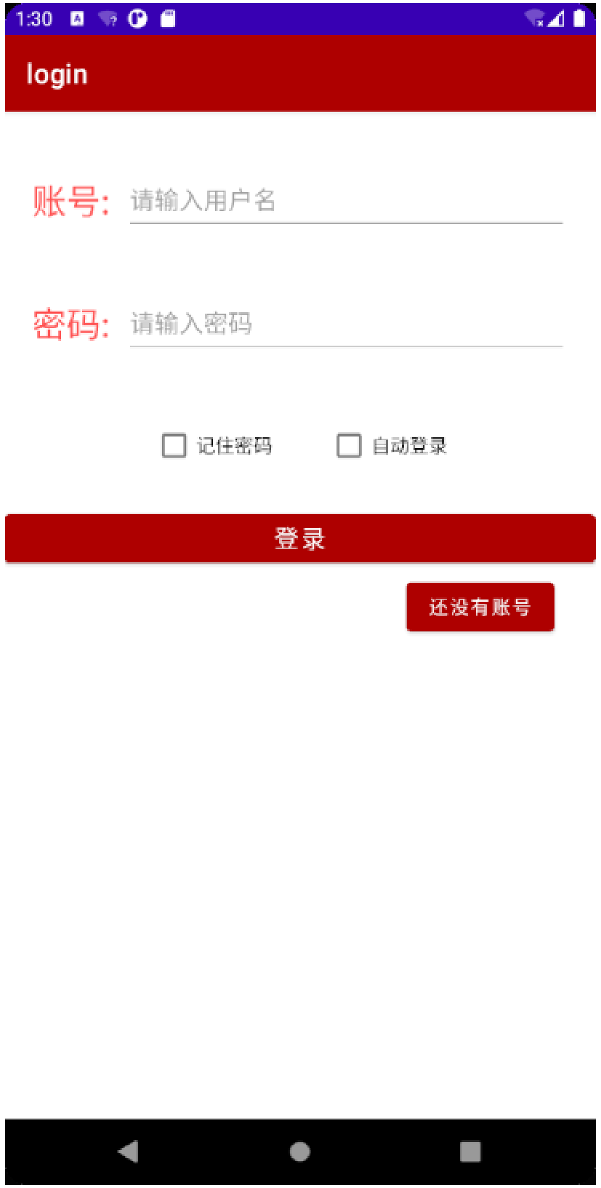
颜色定义



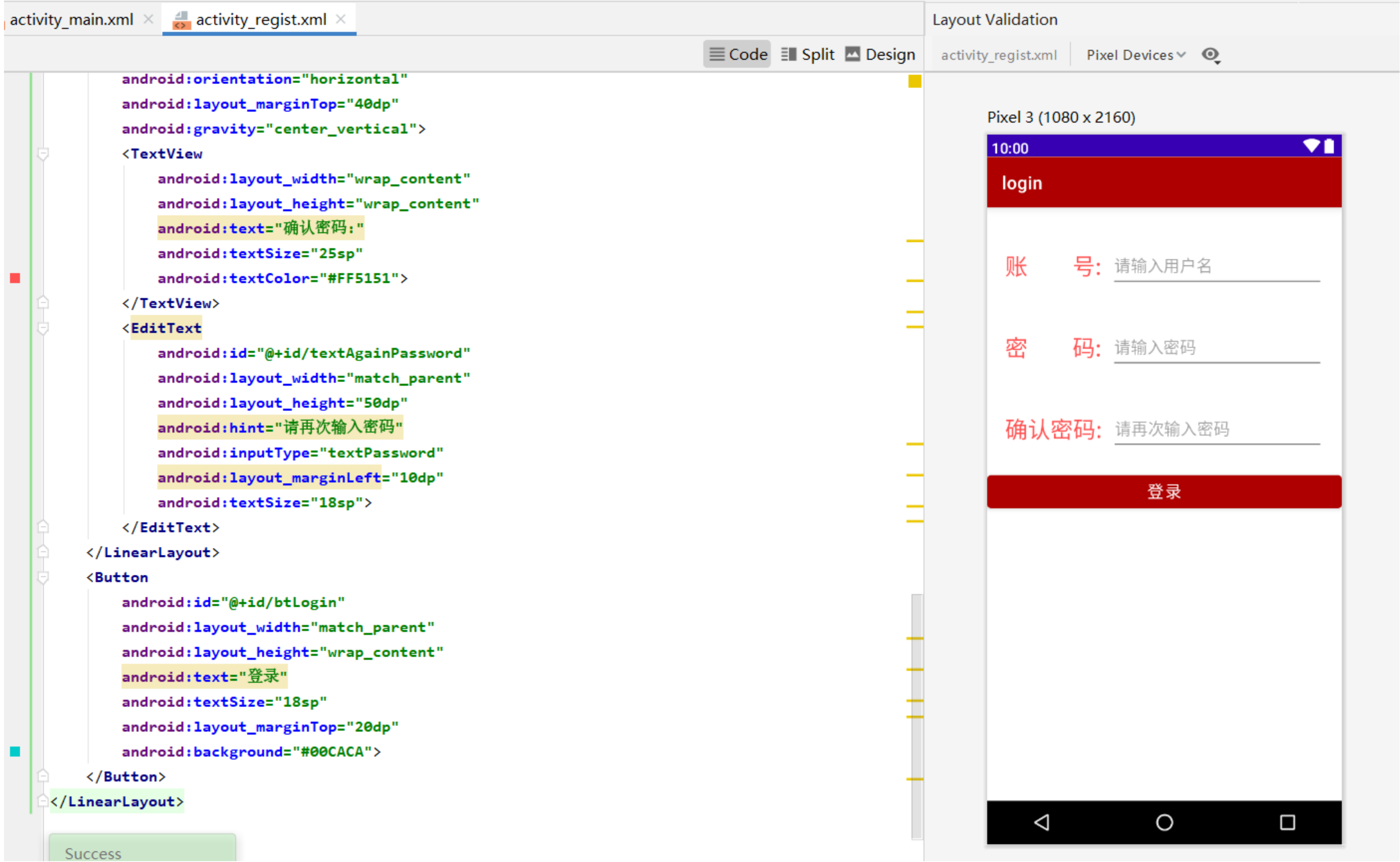
```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="red">#AE0000</color>
</resources>
```



```
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Theme.Chap01" parent="Theme.MaterialComponents.DayNight.DarkActionBar">
        <!-- Primary brand color. -->
        <item name="colorPrimary">@color/red</item>
        <item name="colorPrimaryVariant">@color/purple_700</item>
        <item name="colorOnPrimary">@color/white</item>
        <!-- Secondary brand color. -->
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/teal_700</item>
        <item name="colorOnSecondary">@color/black</item>
        <!-- Status bar color. -->
        <item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</item>
        <!-- Customize your theme here. -->
    </style>
```



注册页面



```
        android:orientation="horizontal"
        android:layout_marginTop="40dp"
        android:gravity="center_vertical">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="确认密码:"
            android:textSize="25sp"
            android:textColor="#FF5151">
        </TextView>
        <EditText
            android:id="@+id/textAgainPassword"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:hint="请再次输入密码"
            android:inputType="textPassword"
            android:layout_marginLeft="10dp"
            android:textSize="18sp">
        </EditText>
    </LinearLayout>
    <Button
        android:id="@+id/btLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="登录"
        android:textSize="18sp"
        android:layout_marginTop="20dp"
        android:background="#00CACA">
    </Button>
</LinearLayout>
```

Pixel 3 (1080 x 2160)

10:00

login

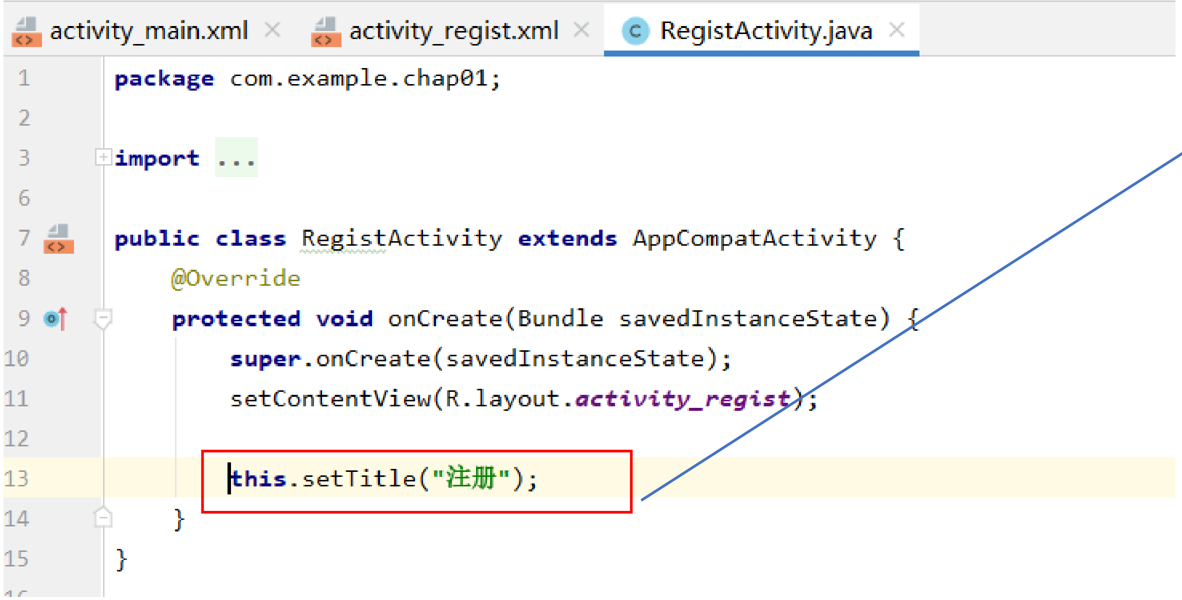
账　　号:　请输入用户名

密　　码:　请输入密码

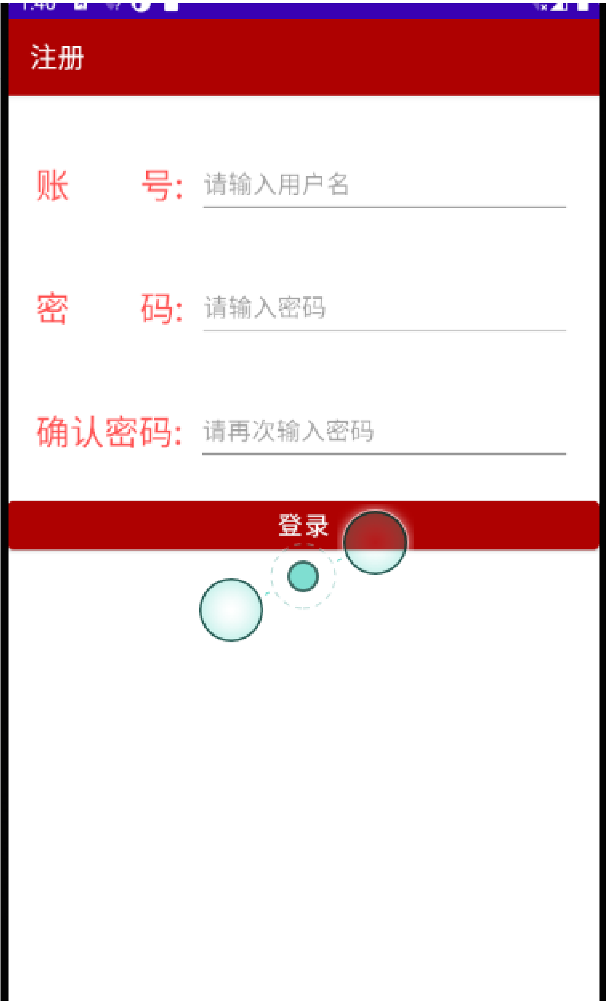
确认密码:　请再次输入密码

登录

更改页面标题



```java
package com.example.chap01;

import ...

public class RegistActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_regist);

        this.setTitle("注册");
    }
}
```



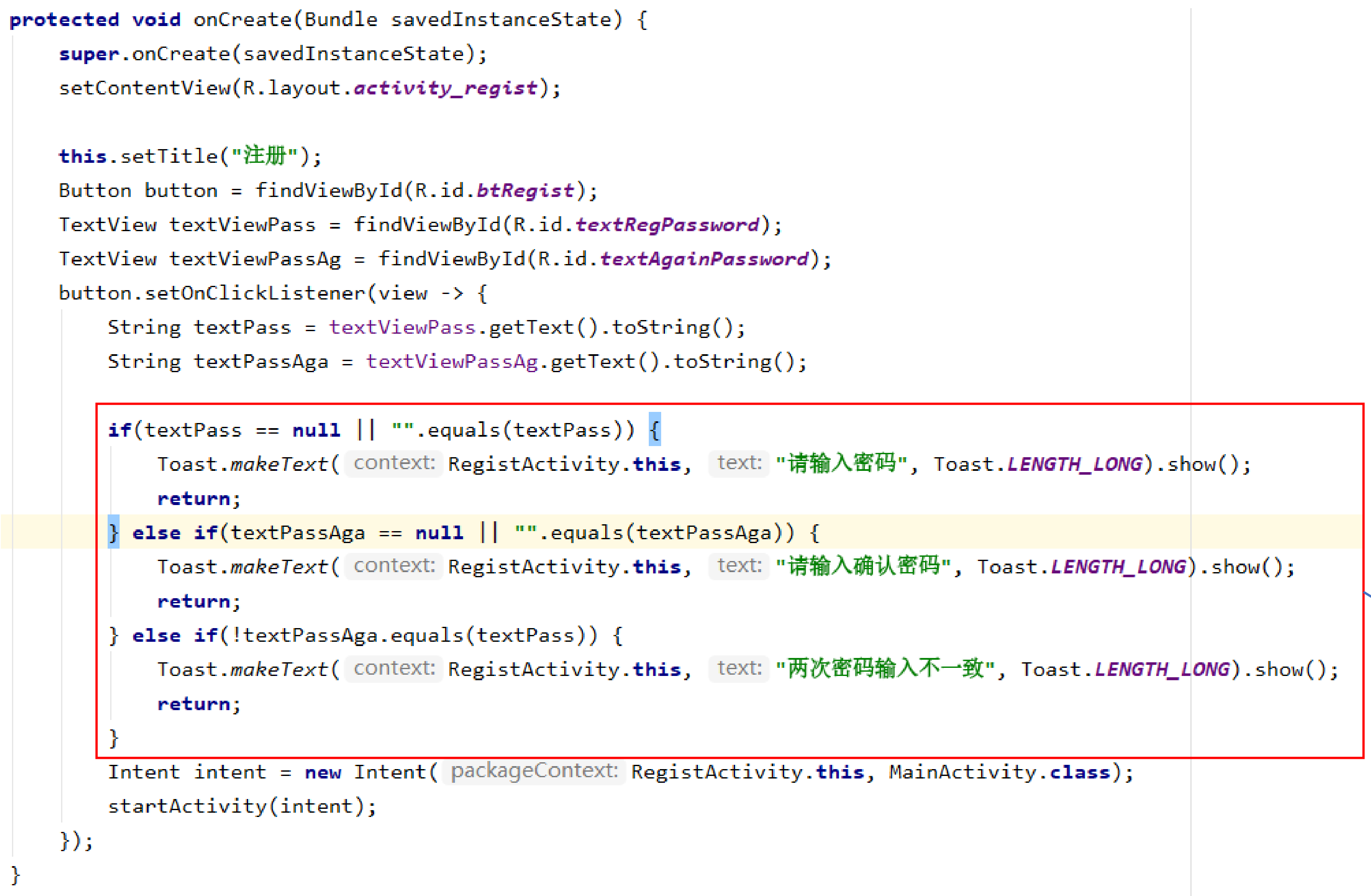
注册

账　　号: 请输入用户名

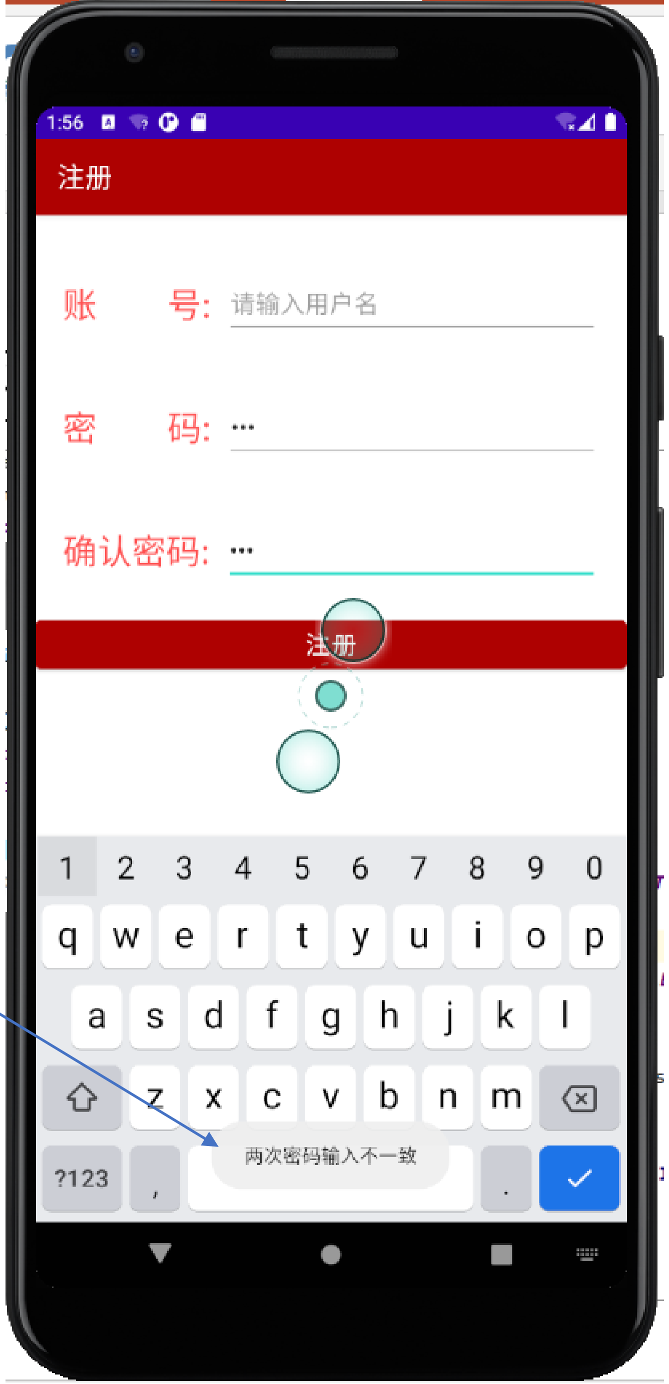
密　　码: 请输入密码

确认密码: 请再次输入密码

确认密码是否一致



```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_regist);

    this.setTitle("注册");
    Button button = findViewById(R.id.btRegist);
    TextView textViewPass = findViewById(R.id.textRegPassword);
    TextView textViewPassAg = findViewById(R.id.textAgainPassword);
    button.setOnClickListener(view -> {
        String textPass = textViewPass.getText().toString();
        String textPassAga = textViewPassAg.getText().toString();

        if(textPass == null || "".equals(textPass)) {
            Toast.makeText( context: RegistActivity.this,  text: "请输入密码", Toast.LENGTH_LONG).show();
            return;
        } else if(textPassAga == null || "".equals(textPassAga)) {
            Toast.makeText( context: RegistActivity.this,  text: "请输入确认密码", Toast.LENGTH_LONG).show();
            return;
        } else if(!textPassAga.equals(textPass)) {
            Toast.makeText( context: RegistActivity.this,  text: "两次密码输入不一致", Toast.LENGTH_LONG).show();
            return;
        }
        Intent intent = new Intent( packageContext: RegistActivity.this, MainActivity.class);
        startActivity(intent);
    });
}
```



1:56

注册

账 号: 请输入用户名

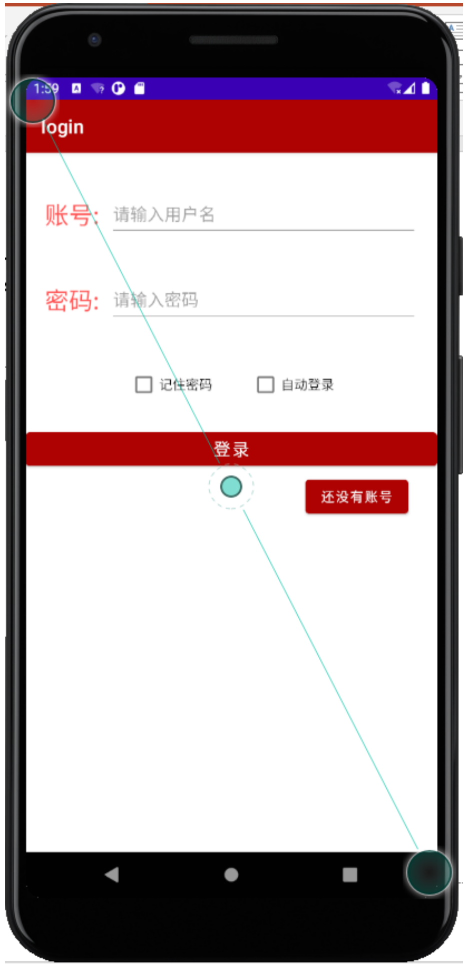
密 码: •••

确认密码: •••

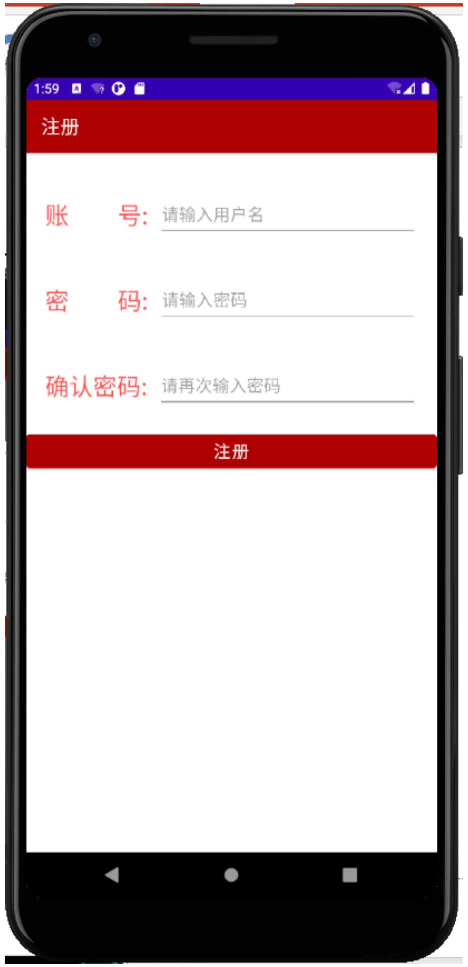
注册

两次密码输入不一致

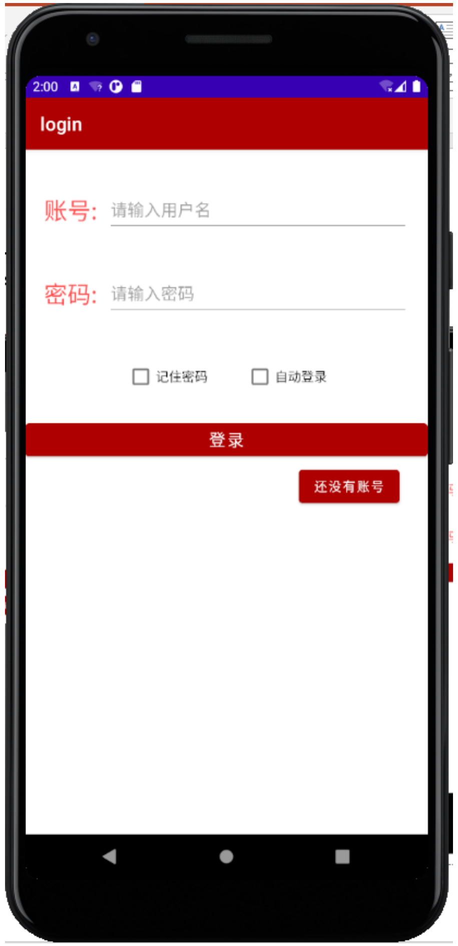
页面之间的跳转



```
Intent intent = new Intent(MainActivity.this, RegistActivity.class);
startActivity(intent);
```



```
Intent intent = new Intent(RegistActivity.this, MainActivity.class);
startActivity(intent);
```


2:00
login
账号: 请输入用户名
密码: 请输入密码
□ 记住密码
□ 自动登录
登录
还没有账号