# 复习

Res

Layout

Activity\_main.xml

加载xml是通过R.layout.activity\_main

<textView android:id=”@+id/textview”>

Class R{

Class layout{

Iny activity\_main

}

Class id

{

int textView=2

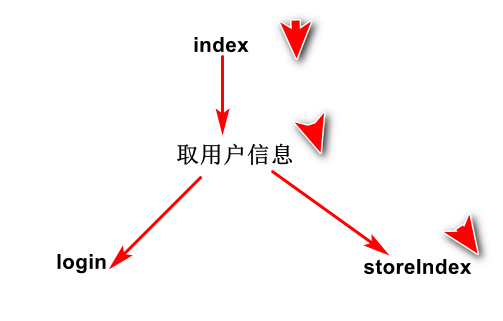
}

}

是通过R.id.textView来找到控件

断点作用

1. 排错



1. 在那加断点

入口，关键方法的第一行

1. 使用技巧

F6,F8

# Android

## Android 布局学习

创建项目project02

### 界面1

创建view1.xml，如图-76所示。

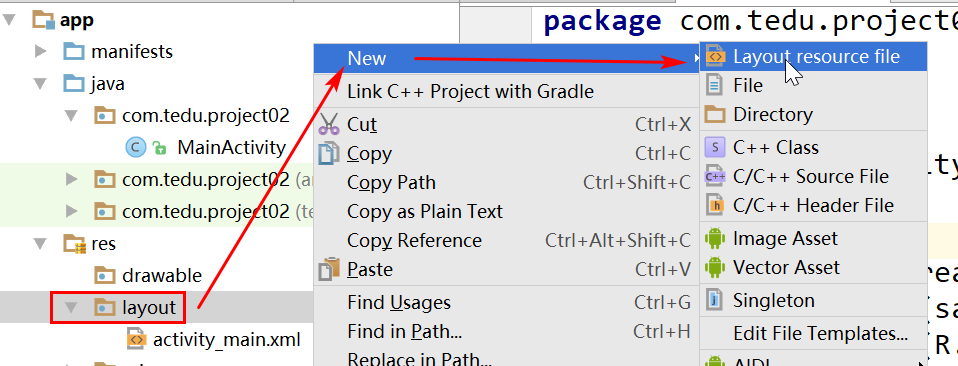


图- 76

在file name中输入view1，如图-77所示。

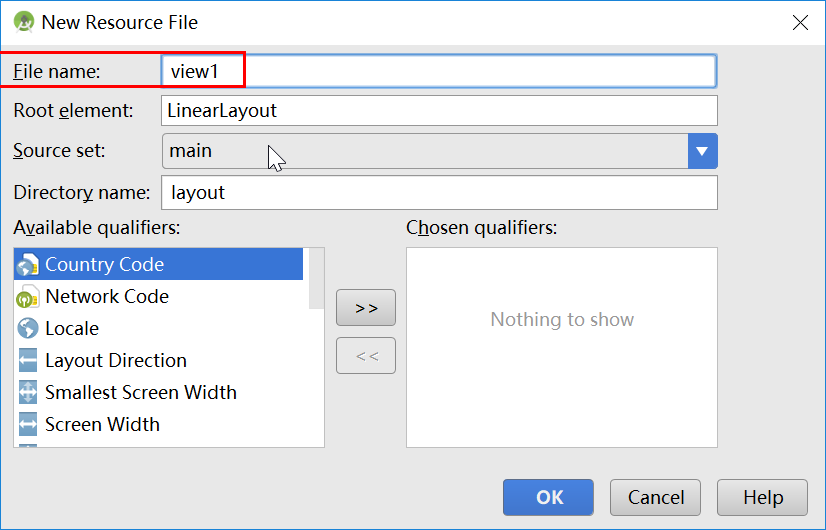


图- 77

View1.xml

*<?*xml version="1.0" encoding="utf-8"*?>  
<!--vertical 从上向下排 horizontal从左向右排-->*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="horizontal" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="button1"  
 />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="button2"  
 />  
</LinearLayout>

### 界面2

创建view2.xml

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="400dp"  
 android:text="button1"  
 android:background="#00FF00"  
 android:layout\_margin="24dp"  
 />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="400dp"  
 android:text="button2"  
 android:background="#00FF00"  
 android:layout\_margin="24dp"  
  
 />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="400dp"  
 android:text="button3"  
 android:background="#00FF00"  
 android:layout\_margin="24dp"  
  
 />  
  
</LinearLayout>

### 界面3

创建view3.xml，把linearLayout改成scrollView,添加linearlayout,拷贝view2.xml中的button

*<?*xml version="1.0" encoding="utf-8"*?>  
<!--把linnearLayout改成scrollView-->*<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="400dp"  
 android:layout\_margin="24dp"  
 android:background="#00FF00"  
 android:text="button1" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="400dp"  
 android:layout\_margin="24dp"  
 android:background="#00FF00"  
 android:text="button2"  
  
 />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="400dp"  
 android:layout\_margin="24dp"  
 android:background="#00FF00"  
 android:text="button3"  
  
 />  
 </LinearLayout>  
</ScrollView>

### 事件处理

View3.xml中给第一个button加id

<Button  
 android:id="@+id/button1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="400dp"  
 android:layout\_margin="24dp"  
 android:background="#00FF00"  
 android:text="button1" />

修改MainActivity.java

**public class** MainActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***view3***);  
 Button btn1 = (Button) findViewById(R.id.***button1***);  
 btn1.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Toast.*makeText*(MainActivity.**this**, **"你单击了btn1"**, Toast.***LENGTH\_LONG***).show();  
 }  
 });  
 }  
}

### R类出错的原因

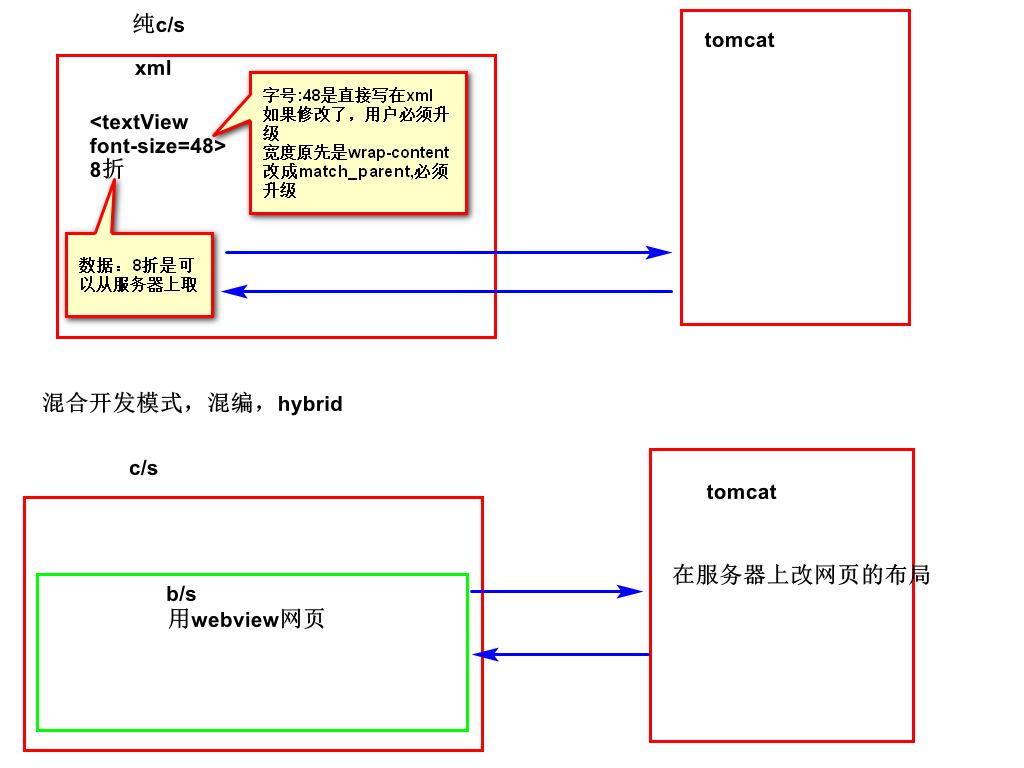
1．Res下的文件名以数字开头或大写，重新build就报错。

2.Xml中属性名写错了。

R.layout.view1或R.id.btn1 常量报错原因是，导入的R类不是本项目的R类，导入的是android.support.v7.appcompat.R

## Android与js交互

### 服务器上一改，客户端就变



#### 在服务器上创建11.html

#### 添加权限

注意uses-permission标签放在manifest标签中

*<?*xml version="1.0" encoding="utf-8"*?>*<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.pd.project3">

<uses-permission android:name="android.permission.INTERNET"/>  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 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>  
  
</manifest>

#### 创建布局webview\_layout.xml

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
<Button  
 android:text="button01"  
 android:layout\_weight="1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp" />  
*<!--相当于浏览器-->* <WebView  
 android:layout\_weight="9"  
 android:id="@+id/webView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp">  
</WebView>  
  
</LinearLayout>

#### Android代码

**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **final** WebView webView= (WebView) findViewById(R.id.***webView***);  
 webView.clearCache(**true**);  
 String url=**"http://10.1.6.102:80/11.html"**;  
 webView.loadUrl(url);  
   
}

加载京东

*//1,找到webView*WebView webView= (WebView)  
 findViewById(R.id.***webView***);  
*//2,加载11.html*String url=**"https://www.jd.com"**;  
*//允许执行js*WebSettings webSettings=webView.getSettings();  
webSettings.setJavaScriptEnabled(**true**);  
  
*//调用手机浏览器打开网页，  
//不用手机浏览器打开网页，用webview打开网页*webView.setWebViewClient  
 (**new** WebViewClient());  
  
webView.loadUrl(url);  
*//清除缓存*webView.clearCache(**true**);

### Android调JavaScript()

需求，如图-78所示。

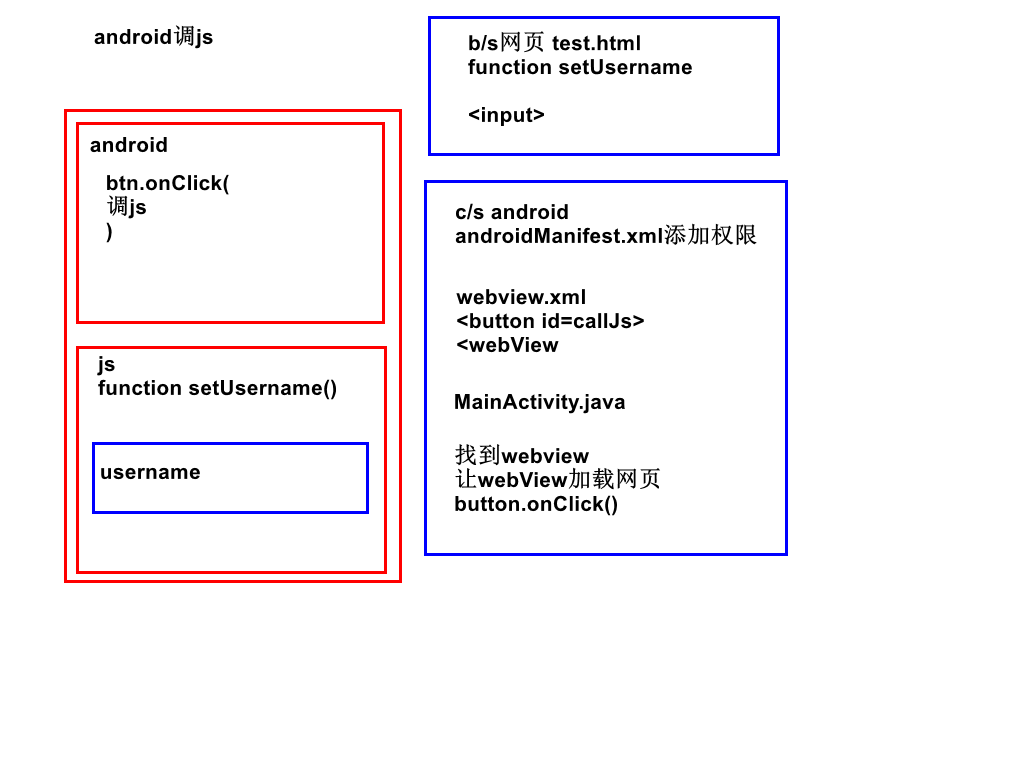


图- 78

test1.html中的内容如下

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<script type=*"text/javascript"*>

//通过java来调用

**function** setUsername() {

**var** input = document.getElementById("username");

input.value = "android";

}

</script>

</head>

<body>

username:

<input id=*"username"* />

</body>

</html>

在电脑上用chrome console测试changeUsername函数是否正确。

MainActivity.onCreate中增加如下代码

*//1,找到webview***final** WebView webView= (WebView) findViewById(R.id.***webView***);  
*//2，加载网页*webView.loadUrl(**"http://10.1.6.10:8080/test1.html"**);  
*//3,设置允许执行js*WebSettings webSettings=webView.getSettings();  
webSettings.setJavaScriptEnabled(**true**);  
  
*//找到button*Button btn= (Button)  
 findViewById(R.id.***btn***);  
*//为button设置事件监听*btn.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 //在java中调用js中的函数  
//webview.loadUrl(http://)默认是加载网页  
//loadUrl("javascript:"),不加载网页  
//去执行当前网页中的函数

webView.loadUrl(**"javascript:setUsername()"**);  
 }  
});

### 通过JavaScript调用Android中的方法。

需求，如图-79所示。

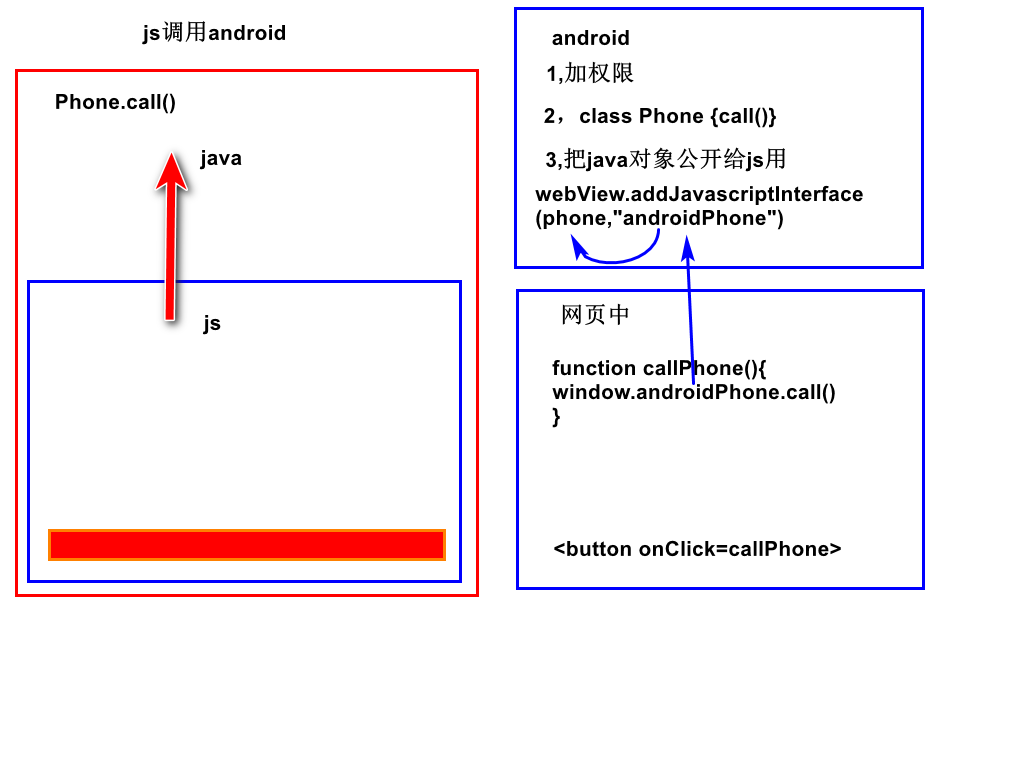


图- 79

androidManifest.xml加call\_phone权限

<uses-permission android:name="android.permission.CALL\_PHONE"/>

#### android代码

**public class** MainActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 *//找到webview* WebView webView= (WebView)  
 findViewById(R.id.***webView***);  
 *//让webview加载网页* webView.loadUrl(**"http://10.1.6.10:8080/test2.html"**);  
 WebSettings webSettings=webView  
 .getSettings();  
 webSettings.setJavaScriptEnabled(**true**);  
 *//把java打电话功能公开出去，给js调* Phone phone=**new** Phone();  
 *//js通过androidPhoneObject调用phone  
 //网页中function call(){window.androidPhoneObject}  
 //js中：window没有andordPhoneObject  
 //webView调phone* webView.addJavascriptInterface(phone  
 ,**"androidPhoneObject"**);  
 }  
 *//内部类，打电话* **final class** Phone{  
 *//给js调的方法，必须加注解* @JavascriptInterface  
 **public void** call()  
 {  
 *//intent意图，目标  
 //url  
 //uri代表的是打电话的资源* Uri uri=Uri.*parse*(**"tel:110"**);  
 Intent intent=  
 **new** Intent  
 (Intent.***ACTION\_CALL***,uri);  
 startActivity(intent);  
 }  
 }

#### js代码

<script type="text/javascript">

**function** call()

{

//调android平台打电话

//androidPhoneObject是android平台

//android平台有phone java对象有call()方法

//android webview.addJavaScriptInterface(phone,androidPhoneObject)

window.androidPhoneObject.call();

}

</script>

</head>

<body>

<button onclick="call()">110</button>

</body>

### Gps

http://lbsyun.baidu.com/index.php?title=android-locsdk

### 拍照上传

https://www.cnblogs.com/plokmju/p/Android\_SystemCamera.html

### 语音转文字

https://blog.csdn.net/qq1271396448/article/details/78818029

### 视频识别

<https://baijiahao.baidu.com/s?id=1569066740553662&wfr=spider&for=pc>

https://download.csdn.net/download/qq\_27524749/10344415

### 二维码

https://www.cnblogs.com/exmyth/p/5142600.html

### 总结

Android与js互相调用，如图-80所示。

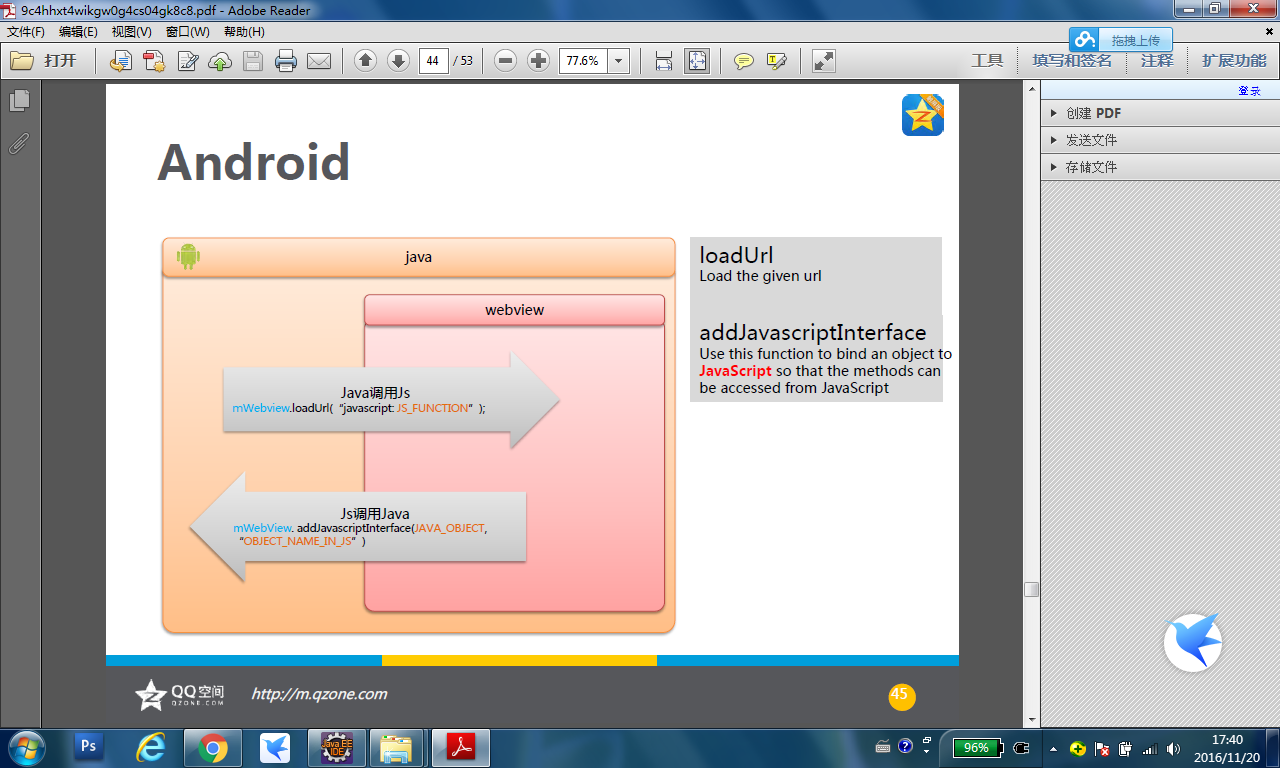
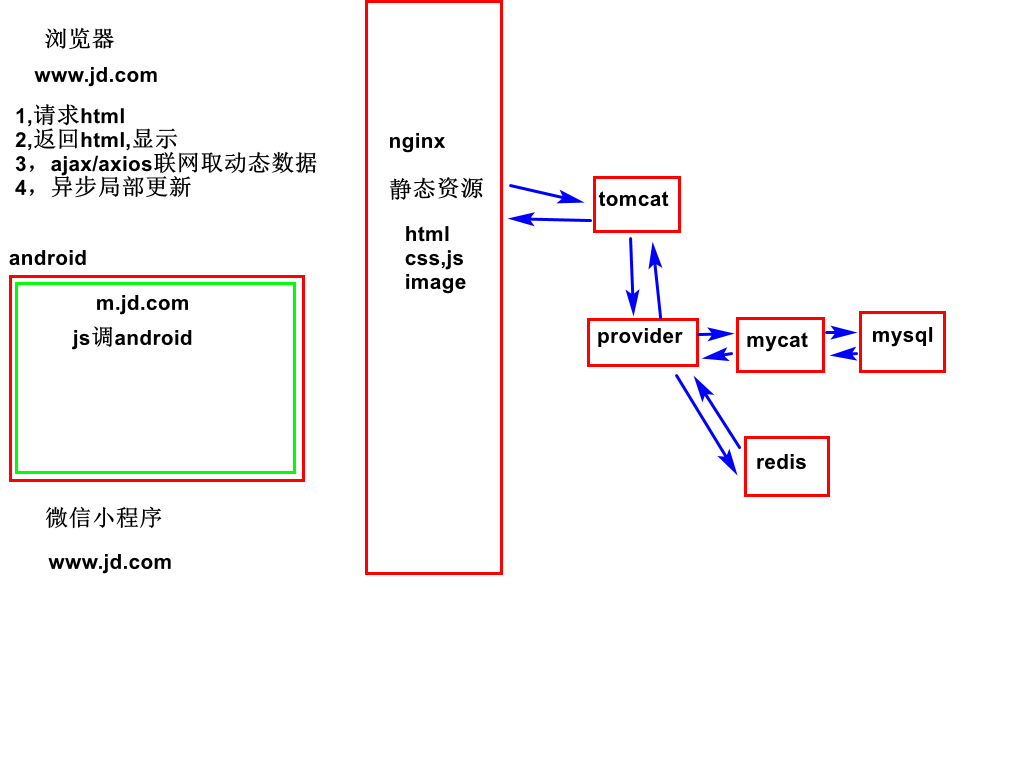


图- 80

## Android与小程序,后端,前端对比

运行环境不一样



# 拼多商城android端

包名是com.tedu.pdApp1809

## 欢迎页

欢迎页，登录页，注册页之间的关系，如图-81所示。

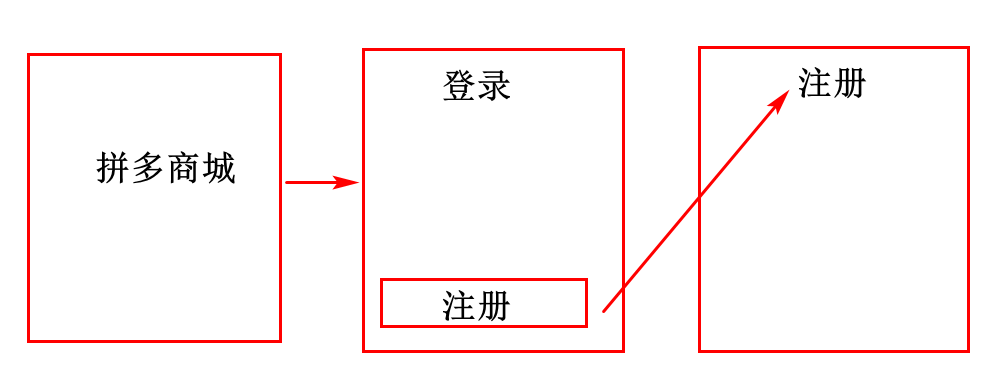


图- 81

删除res,拷贝” \培优课前资料\亿发,移动端课前资料\android\res”

创建activity\_main.xml

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical" android:layout\_width="match\_parent"  
 android:background="@drawable/first"  
 android:layout\_height="match\_parent">  
  
</LinearLayout>

## 登录

### 界面分析

登录页控件嵌套关系，如图-82所示。

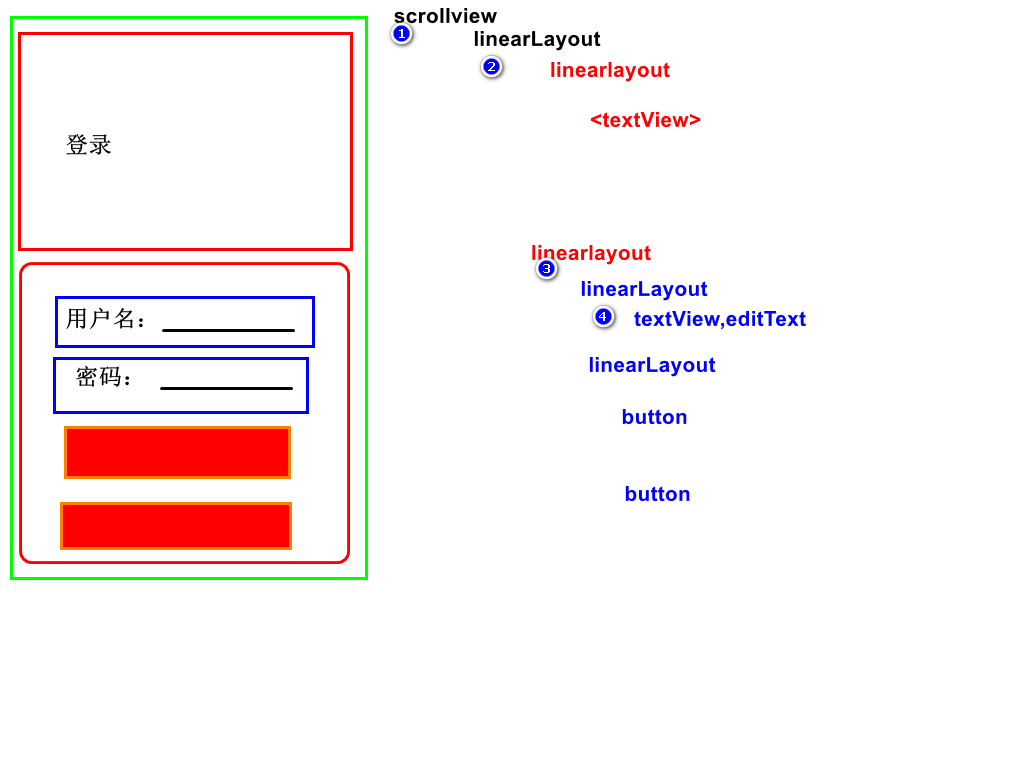


图- 82

### 实现

1.创建LoginActivity，如图-83所示。

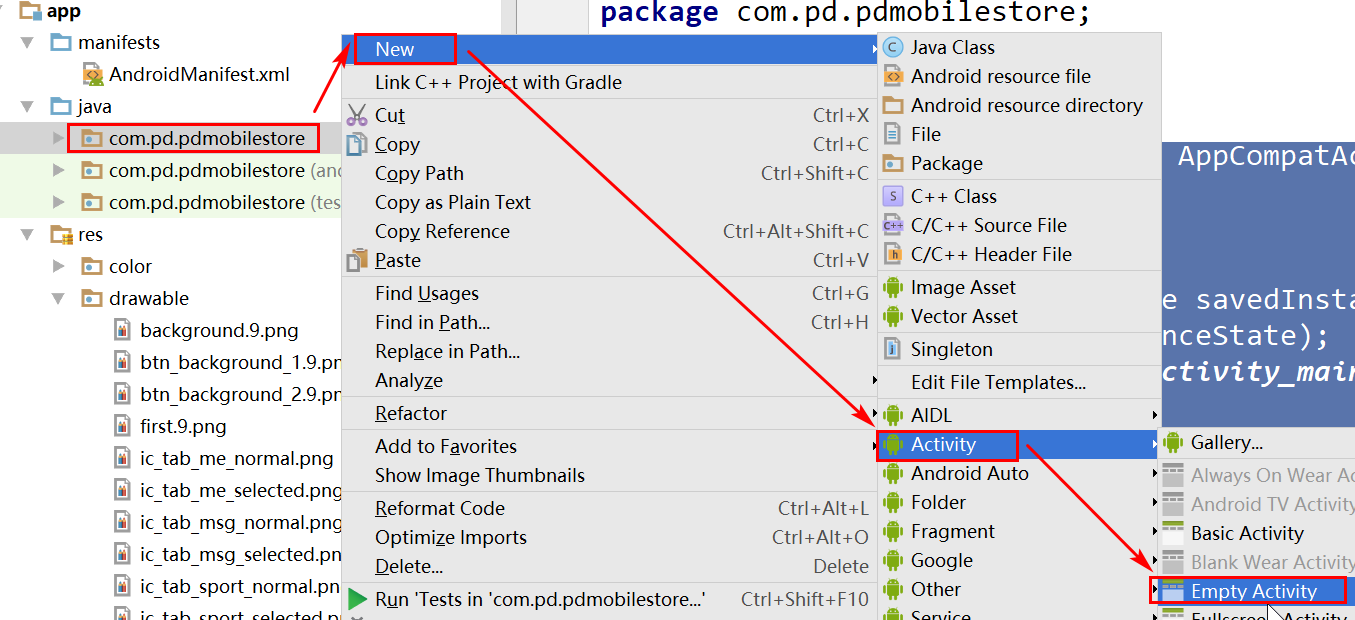


图- 83

*<?*xml version="1.0" encoding="utf-8"*?>*<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/activity\_login"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
  
 tools:context="com.tedu.pdapp.LoginActivity">  
<LinearLayout  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
 *<!--1，标题-->* <LinearLayout  
 android:background="#FF670c"  
 android:paddingTop="76dp"  
 android:paddingLeft="48dp"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="200dp">  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="登录"  
 android:textSize="48dp"  
 />  
 </LinearLayout>  
 *<!--2,2个输入框和2个按钮-->*<LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="@drawable/background"  
 android:layout\_margin="8dp"  
 android:orientation="vertical"  
 >  
 *<!--2.1 用户名-->* <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 >  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="用户名"  
 />  
 <EditText  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/login\_username\_et"  
 />  
 </LinearLayout>  
 *<!--2.2 密码-->* <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 >  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="密码"  
 />  
 <EditText  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/login\_password\_et"  
 />  
 </LinearLayout>  
 *<!--2.3 登录按钮-->* <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="24dp"  
 android:background=  
 "@drawable/btn\_background\_1"  
 android:id="@+id/login\_submit"  
 android:text="登录"  
 />  
 *<!--2.4 注册按钮-->* <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="24dp"  
 android:background=  
 "@drawable/btn\_background\_1"  
 android:id="@+id/login\_toRegister"  
 android:text="注册"  
 />  
</LinearLayout>  
</LinearLayout>  
</ScrollView>

### 自动跳到登录，如图-84所示。

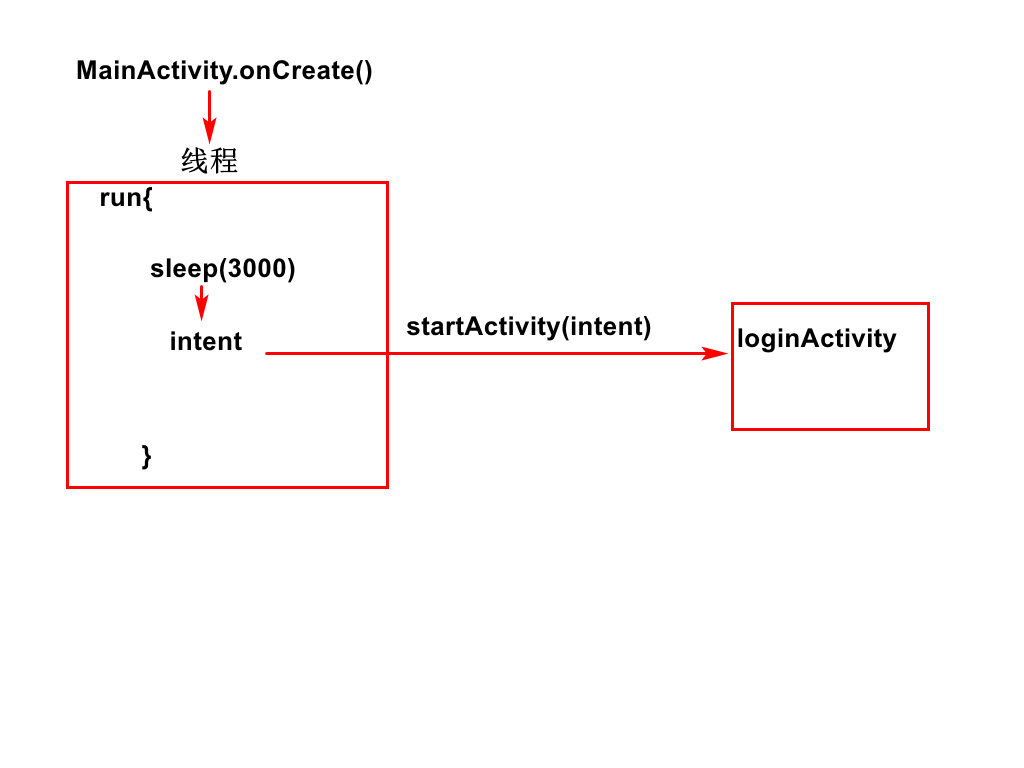


图- 84

**public class** MainActivity **extends** AppCompatActivity {  
  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 ToLoginThread toLoginThread = **new** ToLoginThread();  
 toLoginThread.start();  
 }  
  
 **class** ToLoginThread **extends** Thread {  
 @Override  
 **public void** run() {  
 **try** {  
 *sleep*(10000);  
 Intent intent = **new** Intent(MainActivity.**this**, LoginActivity.**class**);  
 startActivity(intent);  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
}

## 注册

需求，如图-85所示。

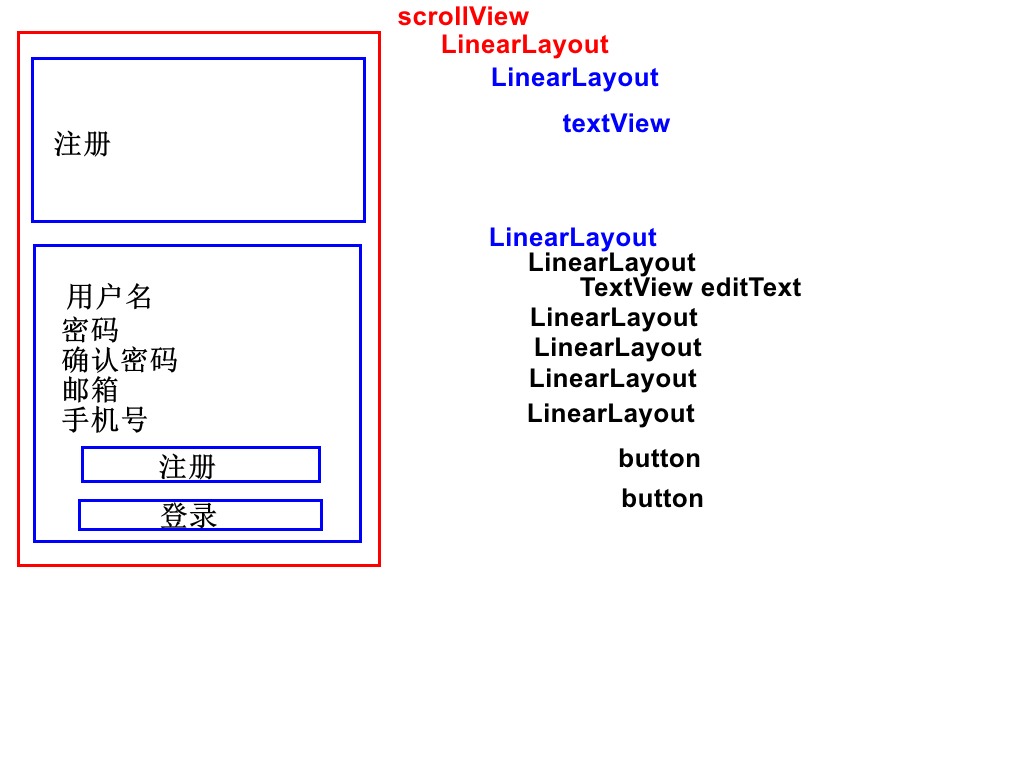


图- 85

1.创建RegisterActivity

2.制作注册界面

3.登录中单击按钮跳到到RegisterActivity

LoginActivity中 id写错，用断点排错

在LoginActivity增加下面代码

*//单击登录界面中的注册按钮跳到注册界面*Button registerButton = (Button)  
 findViewById(R.id.***login\_registerButton***);  
  
registerButton.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent intent = **new** Intent(LoginActivity.**this** , RegisterActivity.**class**);  
 startActivity(intent);  
 }  
});