实验一 基础控件使用——QQ登录界面实现

**——陈洁怡+201906061703**

一、基础实验——基础控件使用

（一）实验目的

1. 熟悉Android工程结构，掌握使用Android Studio开发Android应用的基本步骤；
2. 掌握Android基本控件的使用方法；
3. 初步掌握Android界面约束布局的使用方法；
4. 掌握Android界面优化方法。

（二）基本知识与原理

1. Android应用采用MVC界面设计模式，通过布局文件、资源文件与程序逻辑代码的分离，实现界面设计与逻辑开发的并行进行。
2. TextView、Edit Text、Button、ImageButton、ConstrainLayout 等常用Android控件的显示效果可通过属性设置实现，对于特殊展现形式（如不规则的QQ登录按钮），需要结合shape.xml定义等技术完成。
3. 实验内容及步骤

1、更改创建工程自带的activi\_main.xml，作为应用的主界面前端

|  |
| --- |
| <?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"  android:orientation="vertical"  android:rotationX="2"   tools:context=".MainActivity">   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:layout\_marginTop="80dp"  android:orientation="horizontal">   <ImageView  android:id="@+id/img"  android:layout\_width="80dp"  android:layout\_height="80dp"  android:layout\_gravity="center"  android:padding="4dp"  android:scaleType="centerInside"  android:src="@drawable/qq\_item" />   <TextView  android:id="@+id/qqItem"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:text="@string/qq"  android:textColor="#000000"  android:textSize="60sp"   app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />  </LinearLayout>    <EditText  android:id="@+id/editQQNum"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginTop="40dp"  android:layout\_marginEnd="20dp"  android:background="@drawable/shape\_rounded\_rec"  android:gravity="center"  android:hint="@string/qqNumHint"  android:maxLines="1"  android:minHeight="60dp"  android:textColor="#000000"   android:textSize="20sp"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginTop="10dp"  android:layout\_marginEnd="20dp"  android:background="@drawable/shape\_rounded\_rec"  android:minHeight="60dp"  android:orientation="horizontal">    <ImageButton  android:id="@+id/pwd"  android:layout\_width="25dp"  android:layout\_height="25dp"  android:layout\_gravity="center"  android:layout\_marginLeft="20dp"  android:background="@drawable/transparent"  android:foregroundGravity="center"  android:scaleType="fitCenter"/> <!-- android:src="" />-->   <EditText  android:id="@+id/editQQPassword"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:background="#00FFFFFF"  android:gravity="center"  android:hint="输入密码"  android:inputType="textPassword"  android:layout\_marginRight="40dp"  android:maxLines="1"  android:minHeight="50dp"  android:textColor="#000000"   android:textSize="20sp"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />    </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="10dp">   <CheckBox  android:id="@+id/readed"  android:layout\_width="17dp"  android:layout\_height="17dp"  android:layout\_gravity="center"  android:layout\_marginLeft="50dp"  android:background="@drawable/check\_style"  android:button="@null"  android:gravity="center"   app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <TextView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:layout\_marginLeft="5dp"  android:gravity="left"  android:text="@string/serviceAgreement"  android:textColor="#000000"  android:textSize="15sp" />  </LinearLayout>   <ImageButton  android:id="@+id/ibLogin"  android:layout\_width="80dp"  android:layout\_height="80dp"  android:layout\_gravity="center"  android:layout\_marginTop="30dp"  android:background="@drawable/shape\_circle\_gray"  android:gravity="center"  android:paddingStart="28dp"  android:paddingTop="28dp"  android:paddingEnd="28dp"  android:paddingBottom="28dp"  android:scaleType="fitCenter"  android:src="@drawable/rightarrow\_white" />   <LinearLayout  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:layout\_marginTop="130dp"  android:gravity="bottom"  android:orientation="horizontal">   <TextView  android:id="@+id/sjhdl"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"   android:layout\_gravity="center"  android:text="@string/sjhdl"  android:textColor="#000000"  android:textSize="15sp"   app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <TextView  android:id="@+id/G\_1"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:paddingStart="20dp"  android:paddingEnd="20dp"  android:text=" | "  android:textColor="#737171"  android:textSize="15sp" />   <TextView  android:id="@+id/zhmm"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:text="@string/zhmm"  android:textColor="#000000"  android:textSize="15sp"   app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />   <TextView  android:id="@+id/G\_2"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:paddingStart="20dp"  android:paddingEnd="20dp"  android:text=" | "  android:textColor="#737171"  android:textSize="15sp" />   <TextView  android:id="@+id/xyhzc"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:text="@string/xyhzc"  android:textColor="#000000"  android:textSize="15sp"   app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintLeft\_toLeftOf="parent"  app:layout\_constraintRight\_toRightOf="parent"  app:layout\_constraintTop\_toTopOf="parent" />  </LinearLayout>  </LinearLayout> |

2、更改创建工程自带的MainActivity.java，作为应用的主界面后端，主要实现逻辑功能有：1.当账号密码文本框均不为空时，将登录按钮由灰色转为蓝色，2.当密码文本框不为空时，显示文本框左边的隐藏按钮（密码是否可见按钮），同时根据点击该按钮，显示不同的按钮样式，并且显示或不显示密码；当密码文本框为空时，隐藏该按钮，3.判断当账号为“juraws”，密码为“123456”，同时勾选协议时，允许跳转至登录成功页面的请求，其代码如下：

|  |
| --- |
| <package com.example.androidtest1;  import androidx.appcompat.app.AppCompatActivity;  import android.annotation.SuppressLint; import android.app.Activity; import android.content.Intent; import android.graphics.Color; import android.graphics.drawable.Drawable; import android.os.Bundle; import android.text.Editable; import android.text.TextUtils; import android.text.TextWatcher; import android.text.method.HideReturnsTransformationMethod; import android.text.method.PasswordTransformationMethod; import android.view.View; import android.widget.Button; import android.widget.CheckBox; import android.widget.EditText; import android.widget.ImageButton; import android.widget.TextView;  public class MainActivity extends Activity {   private EditText usernameET;  private EditText passwordET;  private CheckBox readedCB;  private ImageButton loginButton;  private ImageButton pwd;  private boolean pwdCanSee;   @SuppressLint("WrongViewCast")  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);   loginButton = (ImageButton) findViewById(R.id.ibLogin);  usernameET = findViewById(R.id.editQQNum);  passwordET = findViewById(R.id.editQQPassword);  pwd = findViewById(R.id.pwd);  pwdCanSee = false;   loginButton.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View view) {  Intent intent = null;  String username = usernameET.getText().toString().trim();  String password = passwordET.getText().toString().trim();  readedCB = findViewById(R.id.readed);  if (username.equals("juraws") && password.equals("123456") && readedCB.isChecked()) {  intent = new Intent(MainActivity.this, LoginSucceeded.class);  startActivity(intent);  } else {   }  }  });   pwd.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View view) {  if(pwdCanSee) {  pwdCanSee = false;  pwd.setEnabled(true);  pwd.setBackgroundResource(R.drawable.eye\_close);  passwordET.setTransformationMethod(PasswordTransformationMethod.getInstance());  }  else{  pwdCanSee = true;  pwd.setEnabled(true);  pwd.setBackgroundResource(R.drawable.eye\_on);  passwordET.setTransformationMethod(HideReturnsTransformationMethod.getInstance());  }  }  });   usernameET.addTextChangedListener(new TextWatcher() {  @Override  public void beforeTextChanged(CharSequence charSequence, int i, int i1, int i2) {   }   @Override  public void onTextChanged(CharSequence charSequence, int i, int i1, int i2) {   }   @Override  public void afterTextChanged(Editable editable) {  String username = usernameET.getText().toString().trim();  String password = passwordET.getText().toString().trim();  if(!TextUtils.isEmpty(username) && !TextUtils.isEmpty(password)) {  loginButton.setEnabled(true);  loginButton.setBackgroundResource(R.drawable.shape\_circle);  }  else {  loginButton.setEnabled(true);  loginButton.setBackgroundResource(R.drawable.shape\_circle\_gray);  }  }  });   passwordET.addTextChangedListener(new TextWatcher() {  @Override  public void beforeTextChanged(CharSequence charSequence, int i, int i1, int i2) {   }   @Override  public void onTextChanged(CharSequence charSequence, int i, int i1, int i2) {  if(!pwdCanSee) {  pwd.setEnabled(true);  pwd.setBackgroundResource(R.drawable.eye\_close);  passwordET.setTransformationMethod(PasswordTransformationMethod.getInstance());  }  else {  pwd.setEnabled(true);  pwd.setBackgroundResource(R.drawable.eye\_on);  passwordET.setTransformationMethod(HideReturnsTransformationMethod.getInstance());  }  }   @Override  public void afterTextChanged(Editable editable) {  String username = usernameET.getText().toString().trim();  String password = passwordET.getText().toString().trim();  if(!TextUtils.isEmpty(username) && !TextUtils.isEmpty(password)) {  loginButton.setEnabled(true);  loginButton.setBackgroundResource(R.drawable.shape\_circle);  }  else {  loginButton.setEnabled(true);  loginButton.setBackgroundResource(R.drawable.shape\_circle\_gray);  }   if(TextUtils.isEmpty(password)) {  pwd.setEnabled(true);  pwd.setBackgroundResource(R.drawable.transparent);  }   }  });    } } |

3、res->drawable右键单击，点击new，选择Drawable Resource File，新建一个名为shape\_circle.xml的文件，作为qq登录界面登录按钮的样式，其代码如下：

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android"  android:shape="oval">   <solid android:color="#30CCF3" />  <corners android:radius="360dip" />   <padding  android:bottom="10dp"  android:left="10dp"  android:right="10dp"  android:top="10dp" />  </shape> |

4、按步骤1 的方式新建一个新的文件，命名为shape\_circle\_gray.xml,作为当没有填写账号密码时的登录按钮的样式，这里是背景颜色作了更改，其代码如下：

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android"  android:shape="oval">   <solid android:color="#E9DDE1E1" />  <corners android:radius="360dip" />   <padding  android:bottom="10dp"  android:left="10dp"  android:right="10dp"  android:top="10dp" />  </shape> |

5、按步骤1 的方式新建一个新的文件，命名为shape\_rounded\_rec.xml,作为账号密码文本框的样式，其代码如下

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android">   <solid android:color="#E9DDE1E1" />   <corners  android:bottomLeftRadius="500px"  android:bottomRightRadius="500px"  android:topLeftRadius="500px"  android:topRightRadius="500px" />  <padding  android:bottom="5dp"  android:left="5dp"  android:right="5dp"  android:top="5dp" />  </shape> |

1. 6、按步骤1 的方式新建一个新的文件，命名为check\_style.xml,作为协议框（checkbox）的样式，其代码如下

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <selector xmlns:android="http://schemas.android.com/apk/res/android">  <!-- 选中状态-->  <item android:drawable="@drawable/checked\_blue" android:state\_checked="true"/>  <!-- 不选中状态-->  <item android:drawable="@drawable/selected\_false" android:state\_checked="false"/>  <item android:drawable="@drawable/checked\_blue" android:state\_pressed="true"/>  <!-- 默认状态-->  <item android:drawable="@drawable/selected\_false"/> </selector> |

7、按步骤1 的方式新建一个新的文件，命名为select.xml,作为协议框（checkbox）勾选时的样式，其代码如下

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android"  android:shape="oval">   <solid android:color="#03A9F4" /> <!-- <gradient android:="@drawable/check" />-->  <corners android:radius="360dip" />   <padding  android:bottom="10dp"  android:left="10dp"  android:right="10dp"  android:top="10dp" />  </shape> |

8、按步骤1 的方式新建一个新的文件，命名为select.xml,作为协议框（checkbox）未勾选时的样式，其代码如下

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android"  android:shape="oval">   <solid android:color="#FFFFFF" />  <corners android:radius="360dip" />   <stroke android:width="1dp"  android:color="#514F51"/>   <padding  android:bottom="10dp"  android:left="10dp"  android:right="10dp"  android:top="10dp" />  </shape> |

9、layout->new-> layout resource file 新建一个名为activity\_login\_succeeded.xml 的文件，作为登录成功页面的前端，其代码如下

|  |
| --- |
| <?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"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  android:padding="10dp"  tools:context=".LoginSucceeded">   <TextView  android:id="@+id/login\_succeed\_tv1"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_gravity="center"  android:gravity="center"  android:text="@string/loginsucceed"  android:textSize="25sp" />  </FrameLayout> |

10、在MainActivity同目录下新建一个loginSucceeded.java 的文件作为登录成功界面的后端

|  |
| --- |
| package com.example.androidtest1;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle;  public class LoginSucceeded extends AppCompatActivity {   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_login\_succeeded);  } } |

（四）运行结果截图

1、主页面布局情况



图4-1 主页面布局情况

2、app运行的登录界面，此时qq账号和密码均为空，登录按钮为灰色



图4-2 登录界面1

3、输入任意qq账号和密码使两个对话框均不为空，此时登录按钮为蓝色，同时由于密码不为空，左边出现密码是否可见的选择按钮



图4-3 登录界面2

4、点击密码是否可见的选择按钮，显示密码，同时该按钮样式改变



图4-4 登录界面3

5、当密码框为空时，隐藏该按钮同时由于密码框为空，登录按钮变回灰色



图4-5 登录界面4

6、勾选协议框时，改变checkbox样式



图4-6 登录界面6

7、当账号为“juraws”，密码为“123456”，同时勾选协议框时，点击登录按钮，跳转至成功登录页面

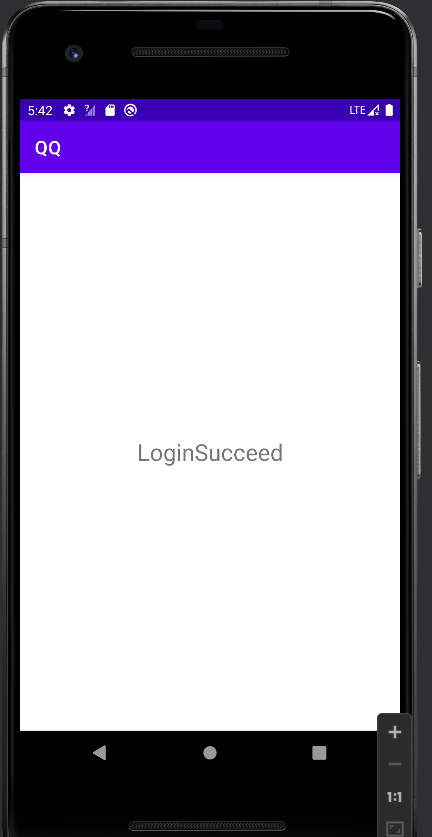


图4-7 登录成功界面

（五）实验收获及总结

在对qq登录界面分析后，我努力照着它的外观功能对自己的项目反复调整，虽然没有完全一模一样，在这个过程中锻炼了我发现问题，思考实现方式，解决问题（实现）的能力。我在实验中遇到了不少问题，包括一直提示 “waiting all target to come online” ，虚拟机运行不起来的问题、xx keeps stopping 的问题 等，一一通过搜索以及log cat的提示自行解决， 使我对安卓虚拟机的理解和debug能力都有所提升，同时在更改各种button、textview、checkbox等控件的样式时遇到一些问题，解决后，使我对相关控件属性，局面布局等的理解加深了，也顺利完成了我的第一个android stdio 项目。