3.4 My Network

mainActivity.java

|  |
| --- |
| package com.example.mynetwork;  import android.app.ProgressDialog; import android.graphics.Bitmap; import android.graphics.BitmapFactory; import android.net.ConnectivityManager; import android.net.NetworkInfo; import android.os.Bundle;  import com.google.android.material.floatingactionbutton.FloatingActionButton; import com.google.android.material.snackbar.Snackbar;  import androidx.annotation.NonNull; import androidx.appcompat.app.AppCompatActivity; import androidx.appcompat.widget.Toolbar;  import android.os.Handler; import android.os.Message; import android.view.View; import android.view.Menu; import android.view.MenuItem; import android.widget.Button; import android.widget.ImageView; import android.widget.Toast;  import java.io.IOException; import java.io.InputStream; import java.net.HttpURLConnection; import java.net.MalformedURLException; import java.net.URL; import java.net.URLConnection;  import javax.net.ssl.HttpsURLConnection;  public class MainActivity extends AppCompatActivity {   private ProgressDialog progressDialog;  private Bitmap bitmap=null;  Button b1;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate( savedInstanceState );  setContentView( R.layout.*activity\_main* );   b1 = (Button) findViewById(R.id.*button*);  b1.setOnClickListener( new View.OnClickListener(){  @Override  public void onClick(View view) {  checkInternetConnection();  downloadImage("http://wwww.tracdisk.com/gamejam.jpg");  }  });  }  private void downloadImage(String urlStr){  progressDialog = ProgressDialog.*show*( this," ", "Downloading Image from "+urlStr );  final String url = urlStr;  new Thread( ){  public void run(){  InputStream in =null;   Message msg=Message.*obtain*();  msg.what=1;   try{  in=openHttpConnection(url);  bitmap = BitmapFactory.*decodeStream*( in );  Bundle b = new Bundle();  b.putParcelable("bitmap", bitmap);  msg.setData(b);  in.close();  } catch (IOException e1){  e1.printStackTrace();  }  messageHandler.sendMessage(msg);  }  }.start();  }   private InputStream openHttpConnection(String urlStr){  InputStream in= null;  int resCode=-1;   try{  URL url=new URL( urlStr );  URLConnection urlConn=url.openConnection();   if (!(urlConn instanceof HttpsURLConnection)){  throw new IOException( "URL is not an HTTP URL" );  }  HttpURLConnection httpConn=(HttpURLConnection) urlConn;  httpConn.setAllowUserInteraction( false );  httpConn.setInstanceFollowRedirects( true );  httpConn.setRequestMethod( "GET" );  httpConn.connect();  resCode=httpConn.getResponseCode();   if (resCode==HttpURLConnection.*HTTP\_OK*){  in=httpConn.getInputStream();  }  } catch (MalformedURLException e){  e.printStackTrace();  }  catch (IOException e){  e.printStackTrace();  }  return in;  }  private Handler messageHandler = new Handler(){  public void handleMessage(Message msg){  super.handleMessage( msg );  ImageView img=(ImageView) findViewById( R.id.*imageView*);  img.setImageBitmap((Bitmap ) (msg.getData().getParcelable( "bitmap" )));  progressDialog.dismiss();  }  };   private boolean checkInternetConnection(){  ConnectivityManager connec=(ConnectivityManager)getSystemService( getBaseContext().*CONNECTIVITY\_SERVICE*);   if(connec.getNetworkInfo(0).getState()== NetworkInfo.State.*CONNECTED*||  connec.getNetworkInfo( 0 ).getState()== NetworkInfo.State.*CONNECTING*||  connec.getNetworkInfo( 1 ).getState()== NetworkInfo.State.*CONNECTING*||  connec.getNetworkInfo( 1 ).getState()== NetworkInfo.State.*CONNECTED*){  Toast.*makeText*( this,"Connected",Toast.*LENGTH\_LONG* ).show();  return true;  }else if(connec.getNetworkInfo( 0 ).getState()== NetworkInfo.State.*DISCONNECTED* ||  connec.getNetworkInfo( 1 ).getState()== NetworkInfo.State.*DISCONNECTED*){  Toast.*makeText*( this, "Not Connected",Toast.*LENGTH\_LONG*).show();  return false;  }  return false;  }   @Override  public boolean onCreateOptionsMenu(Menu menu) {  getMenuInflater().inflate( R.menu.*menu\_main*,menu);  return true;  }   @Override  public boolean onOptionsItemSelected(@NonNull MenuItem item) {  int id=item.getItemId();  if (id==R.id.*action\_settings*){  return true;  }  return super.onOptionsItemSelected( item );  } } |

activityMain.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <RelativeLayout  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:paddingLeft="@dimen/activity\_horizontal\_margin"  android:paddingRight="@dimen/activity\_horizontal\_margin"  android:paddingTop="@dimen/activity\_vertical\_margin"  android:paddingBottom="@dimen/activity\_vertical\_margin">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Animator Viewer"  android:id="@+id/textView"  android:textSize="25sp"  android:layout\_centerHorizontal="true"/>  <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Network Connection"  android:id="@+id/textView2"  android:textColor="#ff36ff15"  android:textIsSelectable="false"  android:textSize="35dp"  android:layout\_below="@+id/textView"  android:layout\_centerHorizontal="true"/>  <ImageView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:id="@+id/imageView"  android:layout\_below="@+id/textView2"  android:layout\_centerHorizontal="true"/>  <Button  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Download Image"  android:id="@+id/button"  android:layout\_below="@+id/imageView"  android:layout\_centerHorizontal="true"  android:layout\_marginTop="76dp"/> </RelativeLayout> |

AndroidManifest.xml +

<uses-permission android:name="android.permission.INTERNET"/>  
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE"/>

