МИНЕСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧЕРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«Брестский государственный технический университет»

Кафедра «Интеллектуальные информационные технологии»

Лабораторная работа №2

По дисциплине «Мобильные технологии»

За 7 семестр

Тема: «Отображение списка элементов на Android загруженного с использованием json»

Выполнил:  
студент 4 курса  
группы АС-56

Марук К.В.

Проверил:

Савонюк В.А.

Брест 2022

*Цель работы:* Реализовать интерфейс приложения для отображения списка элементов. В качестве данных для списка использовать файл в формате json, загруженный с удаленного сервера. Загрузка выполняется в ходе работы по команде пользователя, например, «Загрузить данные»

*Задание:*

* отображать список элементов внутри фрагмента;
* список занимает более одного экрана (прокрутка);
* список можно пролистать;
* выполнять запрос на получение данных с удаленного сервера;
* выполнять преобразование json-структуры в коллекцию объектов;
* выделение отдельного элемента списка с отображение детальной информации на отдельном экране;
* отображать детальную информацию об элементе внутри отдельного фрагмента.

*Бонусы:*

* Присутствие на экране кнопок управления страницами;
* Возможность выбирать один из нескольких возможных запросов на получение информации.

*activity\_main.xml:*

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".lab2MainActivity">  
  
 <ProgressBar  
 android:id="@+id/progressBar"  
 android:layout\_width="75dp"  
 android:layout\_height="75dp"  
 android:layout\_centerInParent="true"  
 android:visibility="visible"  
 android:indeterminate="true"  
 android:indeterminateTintMode="src\_atop"  
 android:indeterminateTint="#127335"/>  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <LinearLayout  
 android:id="@+id/linearLayout2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:orientation="horizontal">  
  
 <ListView  
 android:id="@+id/listView"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginStart="6dp"  
 android:layout\_marginEnd="6dp"  
 android:layout\_weight="10"  
 android:orientation="horizontal">  
  
 <Button  
 android:id="@+id/backBtn"  
 android:layout\_width="100dp"  
 android:layout\_height="wrap\_content"  
 android:text="Назад" />  
  
 <EditText  
 android:id="@+id/editTextNumber"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:ems="10"  
 android:textSize="16sp"  
 android:hint="Введите id"  
 android:gravity="center"  
 android:inputType="number" />  
  
 <Button  
 android:id="@+id/goBtn"  
 android:layout\_width="100dp"  
 android:layout\_height="wrap\_content"  
 android:text="Найти" />  
  
 </LinearLayout>  
  
 </LinearLayout>  
  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
</RelativeLayout>

*activity\_main.java:*

package com.example.lab1;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.KeyEvent;  
import android.view.View;  
import android.view.inputmethod.EditorInfo;  
import android.widget.AdapterView;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ListAdapter;  
import android.widget.ListView;  
import android.widget.ProgressBar;  
import android.widget.TextView;  
import android.widget.Toast;  
import com.android.volley.Request;  
import com.android.volley.RequestQueue;  
import com.android.volley.Response;  
import com.android.volley.VolleyError;  
import com.android.volley.toolbox.StringRequest;  
import com.android.volley.toolbox.Volley;  
import com.google.android.material.textfield.TextInputLayout;  
import org.json.JSONArray;  
import org.json.JSONException;  
import org.json.JSONObject;  
import java.io.UnsupportedEncodingException;  
import java.lang.ref.WeakReference;  
import java.util.ArrayList;  
import java.lang.String;  
  
public class lab2MainActivity extends AppCompatActivity {  
 private static String *JSON\_URL*;  
 ArrayList<JSONObject> listItems;  
 ListView listView;  
 private EditText mEditURL;  
 private TextInputLayout mTextInputLayout;  
 public enum eLastClicked {*text*};  
 public eLastClicked lastClicked = eLastClicked.*text*;  
 private static final String *EMPTY\_STRING* = "";  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*lab2\_start\_view*);  
 setTitle("Лабораторная №2");  
  
 Button downloadButton = findViewById(R.id.*DwnBtn*);  
 Button clearButton = findViewById(R.id.*clearBtn*);  
 mEditURL= findViewById(R.id.*editTextURL*);  
 mTextInputLayout = findViewById(R.id.*textInputLayout*);  
 mEditURL.setOnEditorActionListener(ActionListener.*newInstance*(this));  
  
 downloadButton.setOnClickListener(v -> {  
 if (shouldShowError() && lastClicked.equals(eLastClicked.*text*)) showError();  
 else {  
 hideError();  
 if (lastClicked == eLastClicked.*text*)  
 *JSON\_URL* = mEditURL.getText().toString();  
 renderMainView();  
 }  
 });  
 clearButton.setOnClickListener(v -> { mEditURL.getText().clear(); });  
 }  
  
 public void renderMainView(){  
 setContentView(R.layout.*lab2\_activity\_main*);  
 listView = findViewById(R.id.*listView*);  
 loadJSONFromURL(*JSON\_URL*);  
  
 Button goButton = findViewById(R.id.*goBtn*);  
 Button backButton = findViewById(R.id.*backBtn*);  
  
 backButton.setOnClickListener(v -> {  
 Intent intent = new Intent(this, lab2MainActivity.class);  
 startActivity(intent);  
 });  
  
 goButton.setOnClickListener(v -> {  
 EditText number = findViewById(R.id.*editTextNumber*);  
 if (number.getText().toString().equals(""))  
 number.setHint("Вы забыли ввести id.");  
 else {  
 int neededId = Integer.*parseInt*(number.getText().toString());  
 boolean found = false;  
 int tempId= -1;  
 for (int i = 0; i < listItems.size() ; ++i){  
 try {  
 tempId = listItems.get(i).getInt("id");  
 } catch (JSONException e) {  
 e.printStackTrace();  
 }  
 if (tempId == neededId){  
 found = true;  
 neededId = i;  
 break;  
 }  
 }  
 if (found) listView.smoothScrollToPosition(neededId);  
 else Toast.*makeText*(getApplicationContext(), "Не найдено", Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
  
 listView.setOnItemClickListener((parent, view, position, id) -> {  
 setContentView(R.layout.*lab2\_concrete\_item*);  
  
 TextView textViewID = findViewById(R.id.*IDConcrete*);  
 TextView textViewName = findViewById(R.id.*NameConcrete*);  
 TextView textViewSecondName = findViewById(R.id.*SecondNameConcrete*);  
 TextView textViewMobile = findViewById(R.id.*MobileConcrete*);  
 TextView textViewEmail = findViewById(R.id.*EmailConcrete*);  
  
 JSONObject temp = listItems.get(position);  
  
 try {  
 textViewID.setText(textViewID.getText() + temp.getString("id"));  
 textViewName.setText(textViewName.getText() + temp.getString("name"));  
 textViewSecondName.setText(textViewSecondName.getText() + temp.getString("secondName"));  
 textViewMobile.setText(textViewMobile.getText() + temp.getString("mobile"));  
 textViewEmail.setText(textViewEmail.getText() + temp.getString("email"));  
 } catch (JSONException e) {  
 e.printStackTrace();  
 }  
  
 Button returnBtn = findViewById(R.id.*BackBtn*);  
 returnBtn.setOnClickListener(v -> {  
 setContentView(R.layout.*lab2\_activity\_main*);  
 renderMainView();  
 }); }); }  
  
 private void loadJSONFromURL(String url){  
 ProgressBar progressBar = findViewById(R.id.*progressBar*);  
 progressBar.setVisibility(ListView.*VISIBLE*);  
 StringRequest stringRequest = new StringRequest(Request.Method.*GET*, url,  
 response -> {  
 progressBar.setVisibility(View.*INVISIBLE*);  
 try {  
 JSONObject object = new JSONObject(*EncodingToUTF8*(response));  
 JSONArray jsonArray = object.getJSONArray("users");  
 listItems = getArrayListFromJSONArray(jsonArray);  
 ListAdapter adapter = new lab2ListViewAdapter(getApplicationContext(),R.layout.*lab2\_row*,R.id.*textViewName*,listItems);  
 listView.setAdapter(adapter);  
 } catch (JSONException e){  
 e.printStackTrace();  
 }  
 },  
 error -> Log.*d*("error", error.toString()));  
 RequestQueue requestQueue = Volley.*newRequestQueue*(this);  
 requestQueue.add(stringRequest);  
 }  
  
 private ArrayList<JSONObject> getArrayListFromJSONArray(JSONArray jsonArray){  
 ArrayList<JSONObject> aList = new ArrayList<JSONObject>();  
 try {  
 if(jsonArray!= null)  
 for(int i = 0; i< jsonArray.length();i++)  
 aList.add(jsonArray.getJSONObject(i));  
 } catch (JSONException js){  
 js.printStackTrace();  
 }  
 return aList; }  
  
 public static String EncodingToUTF8(String response){  
 try {  
 byte[] code = response.toString().getBytes("ISO-8859-1");  
 response = new String(code, "UTF-8");  
 } catch (UnsupportedEncodingException e){  
 e.printStackTrace();  
 return null;  
 }  
 return response;  
 }  
  
 private static final class ActionListener implements TextView.OnEditorActionListener {  
 private final WeakReference<lab2MainActivity> mainActivityWeakReference;  
  
 public static ActionListener newInstance(lab2MainActivity mainActivity) {  
 WeakReference<lab2MainActivity> mainActivityWeakReference = new WeakReference<>(mainActivity);  
 return new ActionListener(mainActivityWeakReference);  
 }  
  
 private ActionListener(WeakReference<lab2MainActivity> mainActivityWeakReference) {  
 this.mainActivityWeakReference = mainActivityWeakReference;  
 }  
  
 @Override  
 public boolean onEditorAction(TextView v, int actionId, KeyEvent event) {  
 lab2MainActivity mainActivity = mainActivityWeakReference.get();  
 mainActivity.setLastClicked(eLastClicked.*text*);  
 if (actionId == EditorInfo.*IME\_ACTION\_GO* && mainActivity.shouldShowError())  
 mainActivity.showError();  
 else mainActivity.hideError();  
 return true;  
 } }  
 private boolean shouldShowError() {  
 int textLength = mEditURL.getText().length();  
 String text = mEditURL.getText().toString();  
 return textLength < 4 || !text.startsWith("http");  
 }  
 private void showError() {  
 mTextInputLayout.setError(getString(R.string.*error*));  
 }  
 private void hideError() {  
 mTextInputLayout.setError(*EMPTY\_STRING*);  
 }  
 public void setLastClicked(eLastClicked lastClicked) {  
 this.lastClicked = lastClicked;  
 }}

*Вывод по лабораторной работе:* реализовано приложение для отображения списка элементов формата json, загруженного с удаленного сервера. Отображается список элементов, реализована прокрутка, выполняется запрос на получение данных с сервера, выполняется преобразование json-структуры, присутствуют кнопки управления страницами, есть возможность выбора одного из нескольких запросов на получение информации.