Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №2

«Отображение списка элементов на Android загруженного с использованием json»

Выполнил:

Студент 4 курса

Группы АС-56

Волк Н.А.

Проверил:

Козинский А.А.

Брест 2022

Лабораторная работа №2

**Цель работы:** Разработка приложения, предоставляющего возможность загрузки и отображения json файла с удаленного сервера.

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»

Приложение в минимальном исполнении должно:

* отображать список элементов внутри фрагмента
* список занимает более одного экрана (прокрутка)
* список можно пролистать
* отдельный элемент списка с пользовательским стилем/дизайном
* 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 (то, что способствует оценке выше 4)

* – Возможность настраивать приложение (выбирать или вписывать путь к серверу, выбирать размер выводимой информации – число строк, вывод полей и т.п.)
* – Присутствие на экране кнопок управления страницами, строками, видом отображаемой информации)
* – Возможность выбирать один из нескольких возможных запросов на получение информации
* – Преобразование и сохранение информации запроса (например, в текстовый файл или другой формат CSV, локально или в сеть…)
* – Передача результатов запроса (электронная почта, месенджер и т.п.)
* Обработка исключений с выводом сообщений.

**Код программы:**

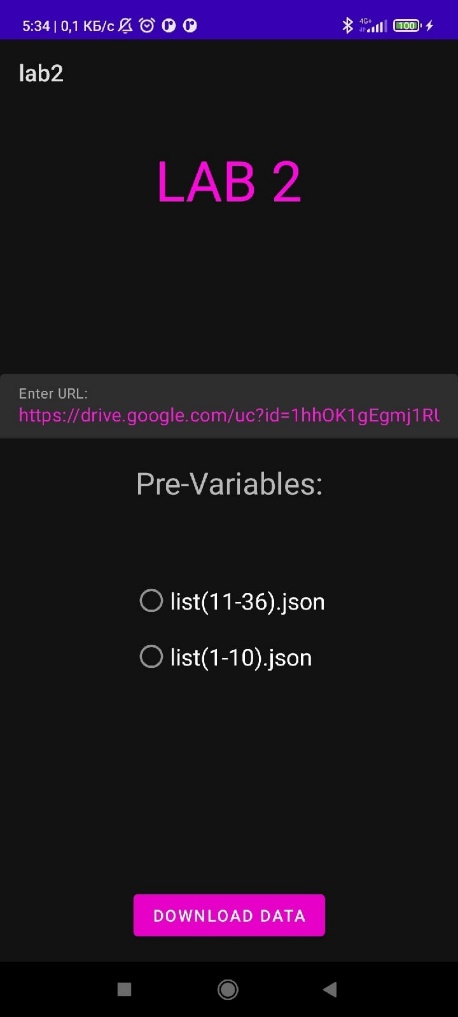
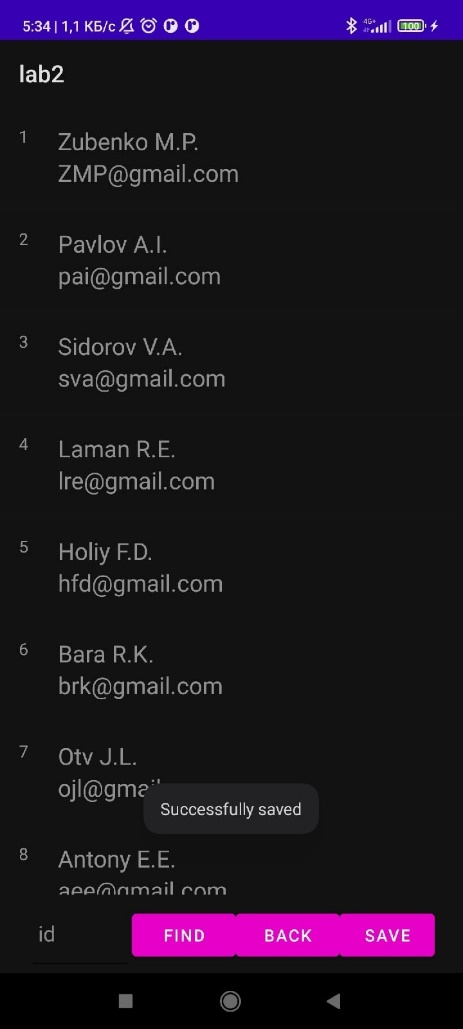
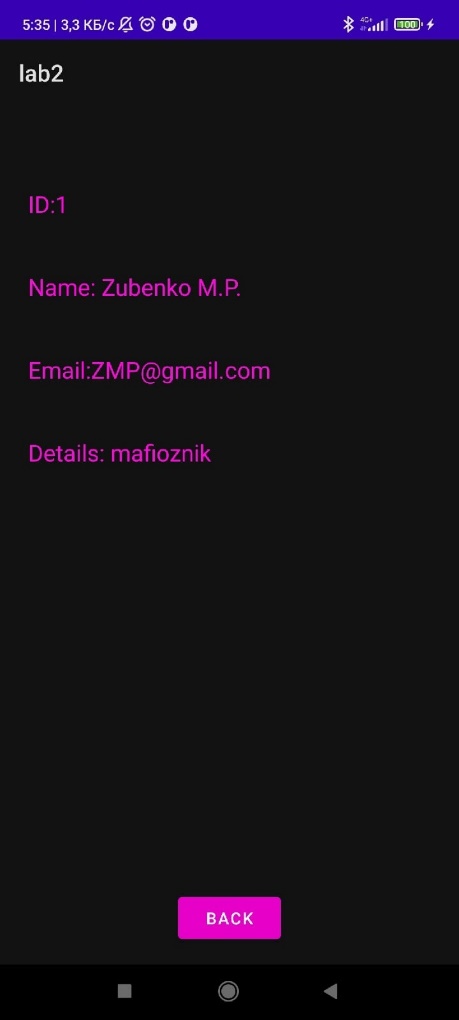
MainActivity.java

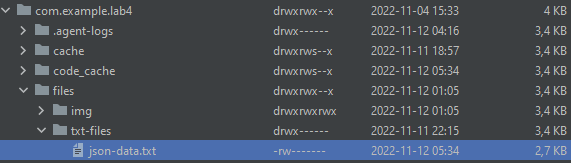
package com.example.lab4;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
  
import android.os.Environment;  
import android.util.Log;  
import android.view.KeyEvent;  
import android.view.View;  
import android.view.inputmethod.EditorInfo;  
import android.widget.AdapterView;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ListAdapter;  
import android.widget.ListView;  
import android.widget.ProgressBar;  
import android.widget.RadioGroup;  
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.File;  
import java.io.FileWriter;  
import java.io.UnsupportedEncodingException;  
import java.lang.ref.WeakReference;  
import java.util.ArrayList;  
import java.lang.String;  
  
public class MainActivity extends AppCompatActivity {  
 private static String *JSON\_URL*;  
 ArrayList<JSONObject> listItems;  
 ListView listView;  
 private EditText mEditURL;  
 private TextInputLayout mTextInputLayout;  
 private RadioGroup radioGroup;  
  
 public enum eLastClicked {*ratio*, *text*};  
 public eLastClicked lastClicked = eLastClicked.*text*;  
  
 private ArrayList<String> sourceList = new ArrayList<String>();  
 private static final String *EMPTY\_STRING* = "";  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*start\_view*);  
 radioGroup = findViewById(R.id.*radioGroup*);  
  
 sourceList.add("https://drive.google.com/uc?id=1bObeMc5iz87-w5weMIB2l6UozGPxf9YZ");  
 sourceList.add("https://drive.google.com/uc?id=1Doerwaw6BASzntYoWf\_czGLEexxyWlhy");  
  
 radioGroup.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {  
 @Override  
 public void onCheckedChanged(RadioGroup group, int checkedId) {  
 setLastClicked(eLastClicked.*ratio*);  
 }  
 });  
  
 Button downloadButton = findViewById(R.id.*DwnBtn*);  
 mEditURL= findViewById(R.id.*editTextURL*);  
 mTextInputLayout = findViewById(R.id.*textInputLayout*);  
 mEditURL.setOnEditorActionListener(ActionListener.*newInstance*(this));  
  
 downloadButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 if (shouldShowError() == true && lastClicked.equals(eLastClicked.*text*)){  
 showError();  
 }  
 else {  
 hideError();  
 if (lastClicked == eLastClicked.*text*)  
 *JSON\_URL* = mEditURL.getText().toString();  
 else{  
 int checkedRadioButtonId = radioGroup.getCheckedRadioButtonId();  
  
 if (checkedRadioButtonId == findViewById(R.id.*radioButton1*).getId()){  
 *JSON\_URL* = sourceList.get(0);  
 }else if (checkedRadioButtonId == findViewById(R.id.*radioButton2*).getId()){  
 *JSON\_URL* = sourceList.get(1);  
 }else *JSON\_URL* = sourceList.get(0);  
 }  
 renderMainView();  
 }  
 }  
 });  
 }  
  
 public void renderMainView(){  
 setContentView(R.layout.*activity\_main*);  
 listView = findViewById(R.id.*listView*);  
 loadJSONFromURL(*JSON\_URL*);  
  
 Button returnBtn = findViewById(R.id.*BackBtn*);  
 returnBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 setContentView(R.layout.*start\_view*);  
 radioGroup = findViewById(R.id.*radioGroup*);  
  
 sourceList.add("https://drive.google.com/uc?id=1bObeMc5iz87-w5weMIB2l6UozGPxf9YZ");  
 sourceList.add("https://drive.google.com/uc?id=1Doerwaw6BASzntYoWf\_czGLEexxyWlhy");  
  
 radioGroup.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {  
 @Override  
 public void onCheckedChanged(RadioGroup group, int checkedId) {  
 setLastClicked(eLastClicked.*ratio*);  
 }  
 });  
  
 Button downloadButton = findViewById(R.id.*DwnBtn*);  
 mEditURL= findViewById(R.id.*editTextURL*);  
 mTextInputLayout = findViewById(R.id.*textInputLayout*);  
  
 downloadButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 if (shouldShowError() == true && lastClicked.equals(eLastClicked.*text*)){  
 showError();  
 }  
 else {  
 hideError();  
 if (lastClicked == eLastClicked.*text*)  
 *JSON\_URL* = mEditURL.getText().toString();  
 else{  
 int checkedRadioButtonId = radioGroup.getCheckedRadioButtonId();  
  
 if (checkedRadioButtonId == findViewById(R.id.*radioButton1*).getId()){  
 *JSON\_URL* = sourceList.get(0);  
 }else if (checkedRadioButtonId == findViewById(R.id.*radioButton2*).getId()){  
 *JSON\_URL* = sourceList.get(1);  
 }else *JSON\_URL* = sourceList.get(0);  
 }  
 renderMainView();  
 }  
 }  
 });  
 }  
 });  
  
 Button goButton = findViewById(R.id.*goBtn*);  
  
 goButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 EditText number = findViewById(R.id.*editTextNumber*);  
 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 == true) {  
 listView.smoothScrollToPosition(neededId);  
 }  
 else {  
 Toast.*makeText*(getApplicationContext(), "Not found", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 }  
 });  
  
 Button saveBtn = findViewById(R.id.*SaveBtn*);  
 saveBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 System.*out*.println(listItems);  
 //File dir = new File(Environment.getExternalStorageDirectory().getAbsolutePath());  
 File dir = new File(MainActivity.this.getFilesDir(), "/txt-files/");  
 //File dir = new File("Android/data/com.example.lab4/files/");  
 if(!dir.exists()) dir.mkdir();  
 try{  
 File file = new File(dir, "json-data.txt");  
 if (!file.exists()) file.createNewFile();  
 FileWriter writer = new FileWriter(file);  
 //writer.write(listItems.toString());  
 writer.append(listItems.toString());  
 writer.flush();  
 writer.close();  
 System.*out*.println("Success");  
 System.*out*.println(file.getAbsoluteFile());  
  
 } catch (Exception e) {System.*out*.println(":(");}  
 }  
 });  
  
 listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
  
 setContentView(R.layout.*concrete\_item*);  
  
 TextView textViewID = findViewById(R.id.*IDConcrete*);  
 TextView textViewName = findViewById(R.id.*NameConcrete*);  
 TextView textViewEmail = findViewById(R.id.*EmailConcrete*);  
 TextView textViewAbout = findViewById(R.id.*AboutConcrete*);  
  
 JSONObject temp = listItems.get(position);  
  
 try {  
 textViewID.setText(textViewID.getText() + temp.getString("id"));  
 textViewName.setText(textViewName.getText() + temp.getString("name"));  
 textViewEmail.setText(textViewEmail.getText() + temp.getString("email"));  
 textViewAbout.setText(textViewAbout.getText() + temp.getString("about"));  
 } catch (JSONException e) {  
 e.printStackTrace();  
 }  
  
  
 Button returnBtn = findViewById(R.id.*BackBtn*);  
  
 returnBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 setContentView(R.layout.*activity\_main*);  
 renderMainView();  
 }  
 });  
 }  
 });  
 }  
  
 private void loadJSONFromURL(String url){  
 ProgressBar progressBar = (ProgressBar) findViewById(R.id.*progressBar*);  
 progressBar.setVisibility(ListView.*VISIBLE*);  
 StringRequest stringRequest = new StringRequest(Request.Method.*GET*, url,  
 new Response.Listener<String>() {  
 @Override  
 public void onResponse(String response) {  
 progressBar.setVisibility(View.*INVISIBLE*);  
 try {  
 JSONObject object = new JSONObject(*EncodingToUTF8*(response));  
 //getJSONArray - извлекает массив  
 JSONArray jsonArray = object.getJSONArray("users");  
 //по ключам получаем значения  
 listItems = getArrayListFromJSONArray(jsonArray);  
 //передаем список в адаптер, а он уже занимается его выводом  
 ListAdapter adapter = new ListViewAdapter(getApplicationContext(),R.layout.*row*,R.id.*textViewName*,listItems);  
 listView.setAdapter(adapter);  
 } catch (JSONException e){  
 e.printStackTrace();  
 }  
 }  
 },  
 new Response.ErrorListener(){  
 @Override  
 public void onErrorResponse(VolleyError error) {  
 Log.*d*("error", error.toString());  
 }  
 });  
 RequestQueue requestQueue = Volley.*newRequestQueue*(this);  
 requestQueue.add(stringRequest);  
 }  
  
 //по ключам получаем значения  
 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;  
 }  
  
 //чтобы имена не выводились набором загогулек  
 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<MainActivity> mainActivityWeakReference;  
  
  
 public static ActionListener newInstance(MainActivity mainActivity) {  
 WeakReference<MainActivity> mainActivityWeakReference = new WeakReference<>(mainActivity);  
 return new ActionListener(mainActivityWeakReference);  
 }  
  
 private ActionListener(WeakReference<MainActivity> mainActivityWeakReference) {  
 this.mainActivityWeakReference = mainActivityWeakReference;  
 }  
  
 @Override  
 public boolean onEditorAction(TextView v, int actionId, KeyEvent event) {  
 MainActivity mainActivity = mainActivityWeakReference.get();  
 mainActivity.setLastClicked(eLastClicked.*text*);  
 if (mainActivity != null) {  
 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.substring(0,4).equals("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;  
 }  
}

ListViewAdapter.java:

package com.example.lab4;  
  
import android.app.Activity;  
import android.content.Context;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.ArrayAdapter;  
import android.widget.TextView;  
import org.json.JSONException;  
import org.json.JSONObject;  
  
import java.util.ArrayList;  
  
public class ListViewAdapter extends ArrayAdapter<JSONObject> {  
 int listLayout;  
 ArrayList<JSONObject> usersList;  
 Context context;  
  
 //конструктор класса  
 public ListViewAdapter(Context context, int listLayout , int field, ArrayList<JSONObject> usersList) {  
 super(context, listLayout, field, usersList);  
 this.context = context;  
 this.listLayout = listLayout;  
 this.usersList = usersList;  
 }  
  
 @Override  
 public View getView(int position, View convertView, ViewGroup parent) {  
 //Создаем новый экземпляр LayoutInflater, связанный с определенным контекстом  
 LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.*LAYOUT\_INFLATER\_SERVICE*);  
 //View inflate (int resource, ViewGroup root)  
 View listViewItem = inflater.inflate(listLayout, null);  
 TextView name = listViewItem.findViewById(R.id.*textViewName*);  
 TextView email = listViewItem.findViewById(R.id.*textViewEmail*);  
 TextView id = listViewItem.findViewById(R.id.*textViewId*);  
 try {  
 name.setText(usersList.get(position).getString("name"));  
 email.setText(usersList.get(position).getString("email"));  
 id.setText(usersList.get(position).getString("id"));  
 } catch (JSONException je){  
 je.printStackTrace();  
 }  
 return listViewItem;  
 }  
}

Пример выполнения программы:



***Вывод:*** разработал приложение, предоставляющее возможность загрузки и отображения json файла с удаленного сервера.