|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**Отчет по практической работе №12**

по дисциплине «Разработка мобильных приложений»

|  |  |
| --- | --- |
| **Выполнил:**  студент группыИКБО-28-22 | Некрасов Г.А. |
| **Проверил:**  к.э.н доцент | Петросян Л.Э. |

Москва 2024 г.

**СОДЕРЖАНИЕ**

[Задание 3](#_Toc167880686)

[Выполнение первого задания 3](#_Toc167880687)

[Создание 1 проекта для передачи данных 3](#_Toc167880688)

[Создание 2 проекта для приёма данных 10](#_Toc167880689)

[Тестирование первого задания 14](#_Toc167880690)

[Выполнение второго задания 15](#_Toc167880691)

[Тестирование второго задания 20](#_Toc167880692)

# Задание

* Реализовать передачу данных в другое приложение через провайдер контента
* Выполнить преобразование данных в JSON формат и сохранение в отдельный файл. Выполнить преобразование данных из файла JSON в различные поля.

# Выполнение первого задания

## Создание 1 проекта для передачи данных

Листинг 1. MyContentProvider.java

|  |
| --- |
| package com.example.nekrasovglebandreevich\_12pract;  import android.content.ContentProvider; import android.content.ContentUris; import android.content.ContentValues; import android.content.Context; import android.content.UriMatcher; import android.database.Cursor; import android.database.sqlite.SQLiteDatabase; import android.database.sqlite.SQLiteException; import android.database.sqlite.SQLiteOpenHelper; import android.database.sqlite.SQLiteQueryBuilder; import android.net.Uri; import java.util.HashMap;  public class MyContentProvider extends ContentProvider {  public MyContentProvider() {  }   // defining authority so that other application can access it  static final String PROVIDER\_NAME = "com.demo.user.provider";   // defining content URI  static final String URL = "content://" + PROVIDER\_NAME + "/users";   // parsing the content URI  static final Uri CONTENT\_URI = Uri.parse(URL);   static final String id = "id";  static final String name = "name";  static final int uriCode = 1;  static final UriMatcher uriMatcher;  private static HashMap<String, String> values;   static {   // to match the content URI  // every time user access table under content provider  uriMatcher = new UriMatcher(UriMatcher.NO\_MATCH);   // to access whole table  uriMatcher.addURI(PROVIDER\_NAME, "users", uriCode);   // to access a particular row  // of the table  uriMatcher.addURI(PROVIDER\_NAME, "users/\*", uriCode);  }  @Override  public String getType(Uri uri) {  switch (uriMatcher.match(uri)) {  case uriCode:  return "vnd.android.cursor.dir/users";  default:  throw new IllegalArgumentException("Unsupported URI: " + uri);  }  }  // creating the database  @Override  public boolean onCreate() {  Context context = getContext();  DatabaseHelper dbHelper = new DatabaseHelper(context);  db = dbHelper.getWritableDatabase();  if (db != null) {  return true;  }  return false;  }  @Override  public Cursor query(Uri uri, String[] projection, String selection,  String[] selectionArgs, String sortOrder) {  SQLiteQueryBuilder qb = new SQLiteQueryBuilder();  qb.setTables(TABLE\_NAME);  switch (uriMatcher.match(uri)) {  case uriCode:  qb.setProjectionMap(values);  break;  default:  throw new IllegalArgumentException("Unknown URI " + uri);  }  if (sortOrder == null || sortOrder == "") {  sortOrder = id;  }  Cursor c = qb.query(db, projection, selection, selectionArgs, null,  null, sortOrder);  c.setNotificationUri(getContext().getContentResolver(), uri);  return c;  }   // adding data to the database  @Override  public Uri insert(Uri uri, ContentValues values) {  long rowID = db.insert(TABLE\_NAME, "", values);  if (rowID > 0) {  Uri \_uri = ContentUris.withAppendedId(CONTENT\_URI, rowID);  getContext().getContentResolver().notifyChange(\_uri, null);  return \_uri;  }  throw new SQLiteException("Failed to add a record into " + uri);  }   @Override  public int update(Uri uri, ContentValues values, String selection,  String[] selectionArgs) {  int count = 0;  switch (uriMatcher.match(uri)) {  case uriCode:  count = db.update(TABLE\_NAME, values, selection, selectionArgs);  break;  default:  throw new IllegalArgumentException("Unknown URI " + uri);  }  getContext().getContentResolver().notifyChange(uri, null);  return count;  }   @Override  public int delete(Uri uri, String selection, String[] selectionArgs) {  int count = 0;  switch (uriMatcher.match(uri)) {  case uriCode:  count = db.delete(TABLE\_NAME, selection, selectionArgs);  break;  default:  throw new IllegalArgumentException("Unknown URI " + uri);  }  getContext().getContentResolver().notifyChange(uri, null);  return count;  }   // creating object of database  // to perform query  private SQLiteDatabase db;   // declaring name of the database  static final String DATABASE\_NAME = "UserDB";   // declaring table name of the database  static final String TABLE\_NAME = "Users";   // declaring version of the database  static final int DATABASE\_VERSION = 1;   // sql query to create the table  static final String CREATE\_DB\_TABLE = " CREATE TABLE " + TABLE\_NAME  + " (id INTEGER PRIMARY KEY AUTOINCREMENT, "  + " name TEXT NOT NULL);";   // creating a database  private static class DatabaseHelper extends SQLiteOpenHelper {   // defining a constructor  DatabaseHelper(Context context) {  super(context, DATABASE\_NAME, null, DATABASE\_VERSION);  }   // creating a table in the database  @Override  public void onCreate(SQLiteDatabase db) {   db.execSQL(CREATE\_DB\_TABLE);  }   @Override  public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {   // sql query to drop a table  // having similar name  db.execSQL("DROP TABLE IF EXISTS " + TABLE\_NAME);  onCreate(db);  }  } } |

Листинг 2. MainActivity.java

|  |
| --- |
| **package com.example.nekrasovglebandreevich\_12pract;  import androidx.appcompat.app.AppCompatActivity;  import android.content.ContentValues; import android.content.Context; import android.database.Cursor; import android.net.Uri; import android.os.Bundle; import android.view.MotionEvent; import android.view.View; import android.view.inputmethod.InputMethodManager; import android.widget.EditText; import android.widget.TextView; import android.widget.Toast;  public class MainActivity extends AppCompatActivity {   Uri CONTENT\_URI = Uri.parse("content://com.demo.user.provider/users");   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  }   @Override  public boolean onTouchEvent(MotionEvent event) {  InputMethodManager imm = (InputMethodManager)getSystemService(Context.INPUT\_METHOD\_SERVICE);  imm.hideSoftInputFromWindow(getCurrentFocus().getWindowToken(), 0);  return true;  }   public void onClickAddDetails(View view) {  // class to add values in the database  ContentValues values = new ContentValues();   // fetching text from user  values.put("name", ((EditText) findViewById(R.id.textName)).getText().toString());   // inserting into database through content URI  getContentResolver().insert(MyContentProvider.CONTENT\_URI, values);   // displaying a toast message  Toast.makeText(getBaseContext(), "New Record Inserted", Toast.LENGTH\_LONG).show();  }   public void onClickShowDetails(View view) {  // inserting complete table details in this text field  TextView resultView = (TextView) findViewById(R.id.res);   // creating a cursor object of the content URI  Cursor cursor = getContentResolver().query(CONTENT\_URI, null, null, null, null);   // iteration of the cursor to print the whole table  if (cursor != null && cursor.moveToFirst()) {  StringBuilder strBuild = new StringBuilder();  int idIndex = cursor.getColumnIndex("id");  int nameIndex = cursor.getColumnIndex("name");   if (idIndex != -1 && nameIndex != -1) {  while (!cursor.isAfterLast()) {  strBuild.append("\n")  .append(cursor.getString(idIndex))  .append(" - ")  .append(cursor.getString(nameIndex));  cursor.moveToNext();  }  resultView.setText(strBuild);  } else {  resultView.setText("Columns not found");  }  cursor.close();  } else {  resultView.setText("No Records Found");  }  } }** |

Листинг 3. Разметка activity\_main.xml

|  |
| --- |
| **<?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout  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:background="#168BC34A"  tools:context=".MainActivity">   <LinearLayout  android:id="@+id/linearLayout"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical"  app:layout\_constraintBottom\_toTopOf="@+id/imageView"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintVertical\_bias="0.13">   <TextView  android:id="@+id/textView1"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="40dp"  android:layout\_marginBottom="70dp"  android:text="@string/heading"  android:textAlignment="center"  android:textAppearance="@style/TextAppearance.AppCompat.Large"  android:textColor="@android:color/holo\_green\_dark"  android:textSize="36sp"  android:textStyle="bold" />   <EditText  android:id="@+id/textName"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginEnd="20dp"  android:layout\_marginBottom="40dp"  android:hint="@string/hintText"  android:minHeight="48dp" />   <Button  android:id="@+id/insertButton"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginTop="10dp"  android:layout\_marginEnd="20dp"  android:layout\_marginBottom="20dp"  android:background="#4CAF50"  android:onClick="onClickAddDetails"  android:text="@string/insertButtontext"  android:textAlignment="center"  android:textAppearance="@style/TextAppearance.AppCompat.Display1"  android:textColor="#000000"  android:textStyle="bold" />   <Button  android:id="@+id/loadButton"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginTop="10dp"  android:layout\_marginEnd="20dp"  android:layout\_marginBottom="20dp"  android:background="#4CAF50"  android:onClick="onClickShowDetails"  android:text="@string/loadButtonText"  android:textAlignment="center"  android:textAppearance="@style/TextAppearance.AppCompat.Display1"  android:textColor="#000000"  android:textStyle="bold" />   <TextView  android:id="@+id/res"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginEnd="20dp"  android:clickable="false"  android:ems="10"  android:textColor="@android:color/holo\_green\_dark"  android:textSize="18sp"  android:textStyle="bold" />   </LinearLayout>   <ImageView  android:id="@+id/imageView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@id/linearLayout"  app:srcCompat="@drawable/ic\_launcher\_foreground" />  </androidx.constraintlayout.widget.ConstraintLayout>** |

## Создание 2 проекта для приёма данных

Листинг 4. MainActivity.java

|  |
| --- |
| **package com.example.nekrasovglebandreevich\_12practproverka;  import androidx.appcompat.app.AppCompatActivity; import android.database.Cursor; import android.net.Uri; import android.os.Bundle; import android.view.View; import android.widget.TextView;   public class MainActivity extends AppCompatActivity {   Uri CONTENT\_URI = Uri.parse("content://com.demo.user.provider/users");   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  }   public void onClickShowDetails(View view) {  // inserting complete table details in this text field  TextView resultView= findViewById(R.id.res);   // creating a cursor object of the  // content URI  Cursor cursor = getContentResolver().query(CONTENT\_URI, null, null, null, null);   // Check if the cursor is not null and contains at least one row  if(cursor != null && cursor.moveToFirst()) {  StringBuilder strBuild = new StringBuilder();  int idIndex = cursor.getColumnIndex("id");  int nameIndex = cursor.getColumnIndex("name");   // Check if the columns exist in the cursor  if (idIndex != -1 && nameIndex != -1) {  while (!cursor.isAfterLast()) {  strBuild.append("\n").append(cursor.getString(idIndex)).append("-").append(cursor.getString(nameIndex));  cursor.moveToNext();  }  resultView.setText(strBuild);  } else {  resultView.setText("Required columns not found in the cursor");  }   cursor.close(); // Close the cursor when done  } else {  resultView.setText("No Records Found");  }  } }** |

Листинг 5. Разметка activity\_main.xml

|  |
| --- |
| **<?xml version="1.0" encoding="utf-8"?> <androidx.constraintlayout.widget.ConstraintLayout  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:background="#168BC34A"  tools:context=".MainActivity">   <LinearLayout  android:id="@+id/linearLayout"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_centerVertical="true"  android:orientation="vertical"  app:layout\_constraintBottom\_toTopOf="@+id/imageView"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"  app:layout\_constraintVertical\_bias="0.13"  tools:ignore="MissingConstraints">   <TextView  android:id="@+id/textView1"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginTop="40dp"  android:layout\_marginBottom="70dp"  android:text="@string/heading"  android:textAlignment="center"  android:textAppearance="@style/TextAppearance.AppCompat.Large"  android:textColor="@android:color/holo\_green\_dark"  android:textSize="36sp"  android:textStyle="bold" />    <Button  android:id="@+id/loadButton"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginStart="20dp"  android:layout\_marginTop="10dp"  android:layout\_marginEnd="20dp"  android:layout\_marginBottom="20dp"  android:background="#4CAF50"  android:onClick="onClickShowDetails"  android:text="@string/loadButtonText"  android:textAlignment="center"  android:textAppearance="@style/TextAppearance.AppCompat.Display1"  android:textColor="#FFFFFF"  android:textStyle="bold" />   <TextView  android:id="@+id/res"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginEnd="20dp"  android:clickable="false"  android:ems="10"  android:textColor="@android:color/holo\_green\_dark"  android:textSize="18sp"  android:textStyle="bold" />   </LinearLayout>   <ImageView  android:id="@+id/imageView"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  app:layout\_constraintBottom\_toBottomOf="parent"  app:layout\_constraintEnd\_toEndOf="parent"  app:layout\_constraintStart\_toStartOf="parent"  app:srcCompat="@drawable/ic\_launcher\_foreground" />  </androidx.constraintlayout.widget.ConstraintLayout>** |

# Тестирование первого задания



Рисунок 1. Запуск 1 проекта (в котором передаём какой-то текст и проверяем как он будет передаваться в другом приложении)

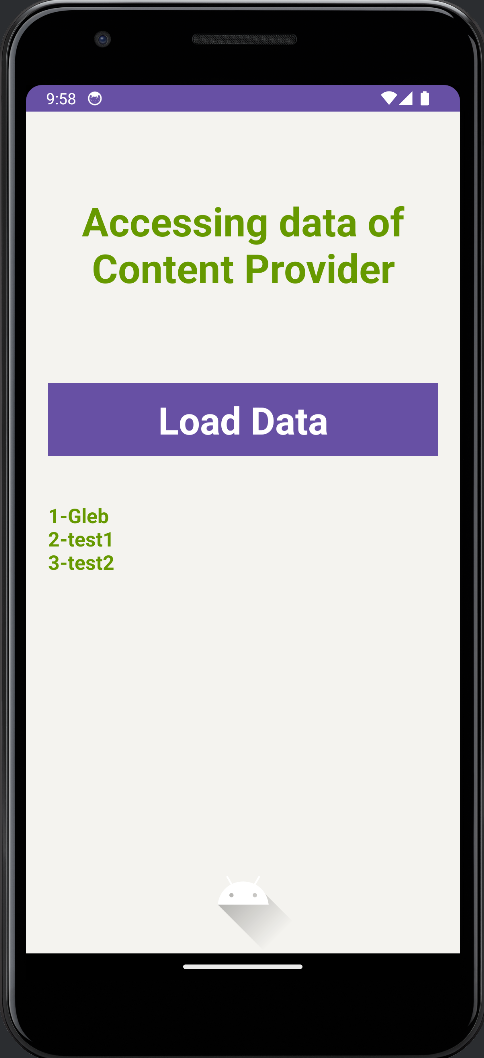


Рисунок 2. Запуск 2 проекта (в котором проверяем передаваемую информацию из 1 проекта)

# Выполнение второго задания

Листинг 6. User.java

|  |
| --- |
| **package com.example.nekrasovglebandreevich\_12practzadanie2;  public class User {  private String name;  private int age;  private String email;   public User(String name, int age, String email) {  this.name = name;  this.age = age;  this.email = email;  }   // Getters and setters   public String getName() {  return name;  }   public void setName(String name) {  this.name = name;  }   public int getAge() {  return age;  }   public void setAge(int age) {  this.age = age;  }   public String getEmail() {  return email;  }   public void setEmail(String email) {  this.email = email;  } }** |

Листинг 7. MainActivity.java

|  |
| --- |
| **package com.example.nekrasovglebandreevich\_12practzadanie2;  import androidx.appcompat.app.AppCompatActivity;  import android.os.Bundle; import android.util.Log; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.Toast;  import com.google.gson.Gson;  import java.io.BufferedReader; import java.io.FileInputStream; import java.io.FileOutputStream; import java.io.InputStreamReader;  public class MainActivity extends AppCompatActivity {   private EditText editTextName, editTextAge, editTextEmail;  private Button buttonSave, buttonLoad;  private static final String FILE\_NAME = "user.json";   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);   editTextName = findViewById(R.id.editTextName);  editTextAge = findViewById(R.id.editTextAge);  editTextEmail = findViewById(R.id.editTextEmail);  buttonSave = findViewById(R.id.buttonSave);  buttonLoad = findViewById(R.id.buttonLoad);   buttonSave.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  saveData();  }  });   buttonLoad.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  loadData();  }  });  }   private void saveData() {  String name = editTextName.getText().toString();  int age = Integer.parseInt(editTextAge.getText().toString());  String email = editTextEmail.getText().toString();   User user = new User(name, age, email);   Gson gson = new Gson();  String jsonString = gson.toJson(user);   try (FileOutputStream fos = openFileOutput(FILE\_NAME, MODE\_PRIVATE)) {  fos.write(jsonString.getBytes());  Toast.makeText(this, "Data saved to " + FILE\_NAME, Toast.LENGTH\_SHORT).show();  } catch (Exception e) {  e.printStackTrace();  Toast.makeText(this, "Failed to save data", Toast.LENGTH\_SHORT).show();  }  }   private void loadData() {  try (FileInputStream fis = openFileInput(FILE\_NAME);  InputStreamReader isr = new InputStreamReader(fis);  BufferedReader reader = new BufferedReader(isr)) {  StringBuilder stringBuilder = new StringBuilder();  String line;  while ((line = reader.readLine()) != null) {  stringBuilder.append(line);  }  String jsonString = stringBuilder.toString();   Gson gson = new Gson();  User user = gson.fromJson(jsonString, User.class);   editTextName.setText(user.getName());  editTextAge.setText(String.valueOf(user.getAge()));  editTextEmail.setText(user.getEmail());   Toast.makeText(this, "Data loaded from " + FILE\_NAME, Toast.LENGTH\_SHORT).show();  } catch (Exception e) {  e.printStackTrace();  Toast.makeText(this, "Failed to load data", Toast.LENGTH\_SHORT).show();  }  } }** |

Листинг 8. Разметка 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=".MainActivity">   <EditText  android:id="@+id/editTextName"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Name" />   <EditText  android:id="@+id/editTextAge"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Age"  android:inputType="number"  android:layout\_below="@id/editTextName"  android:layout\_marginTop="16dp"/>   <EditText  android:id="@+id/editTextEmail"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Email"  android:inputType="textEmailAddress"  android:layout\_below="@id/editTextAge"  android:layout\_marginTop="16dp"/>   <Button  android:id="@+id/buttonSave"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Save Data"  android:layout\_below="@id/editTextEmail"  android:layout\_marginTop="16dp"  android:layout\_centerHorizontal="true"/>   <Button  android:id="@+id/buttonLoad"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Load Data"  android:layout\_below="@id/buttonSave"  android:layout\_marginTop="16dp"  android:layout\_centerHorizontal="true"/> </RelativeLayout>** |

# Тестирование второго задания

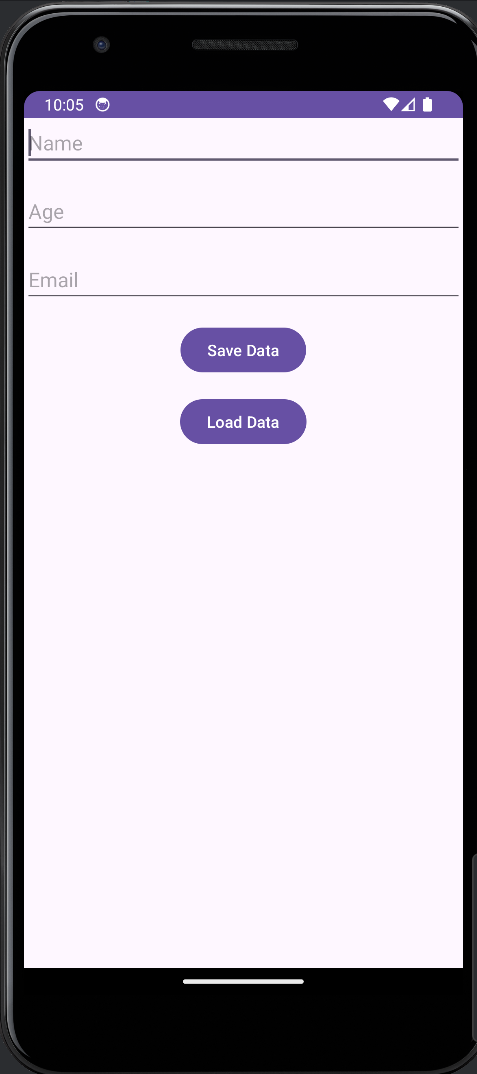


Рисунок 3. Запуск 3 проекта для тестирования JSON-формата

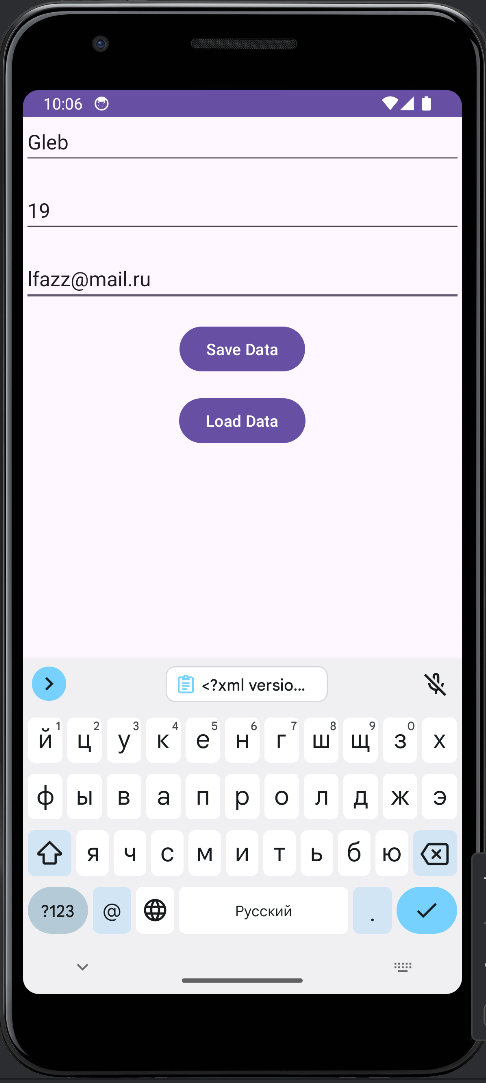


Рисунок 4. Загрузка данных в JSON-файл (Save Data)

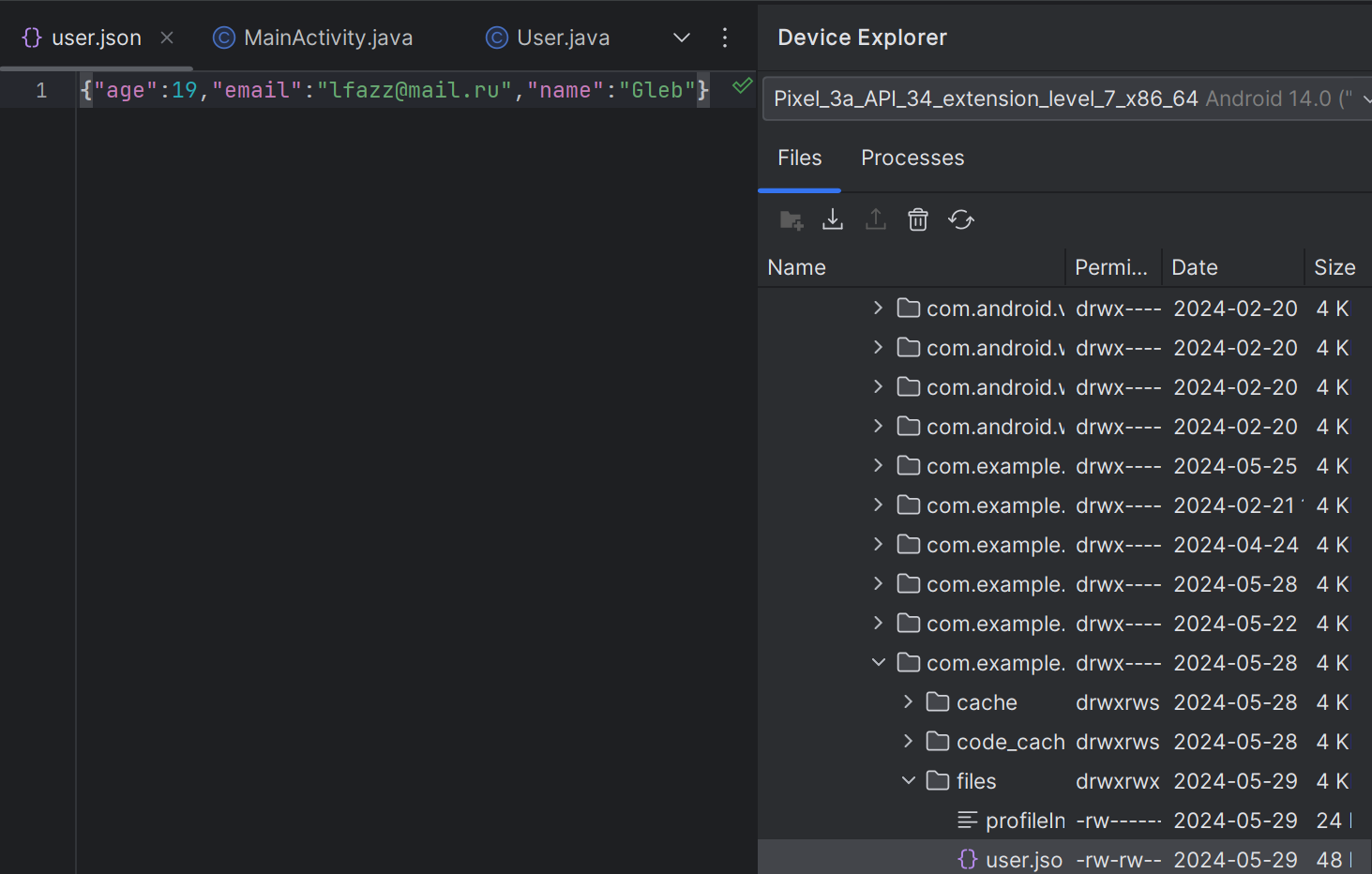


Рисунок 5. Проверка созданного JSON-файла

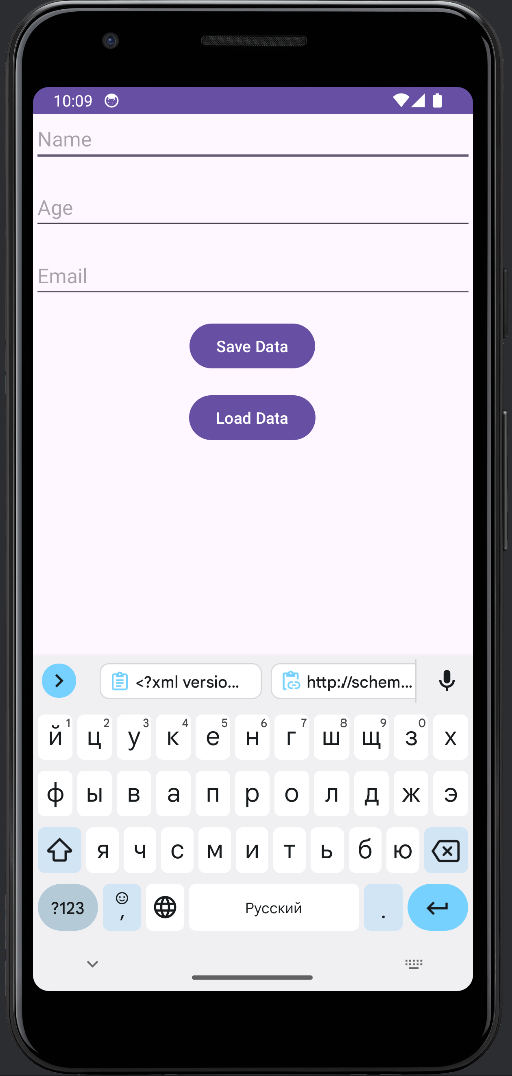


Рисунок 6. Убрали данные из полей для ввода

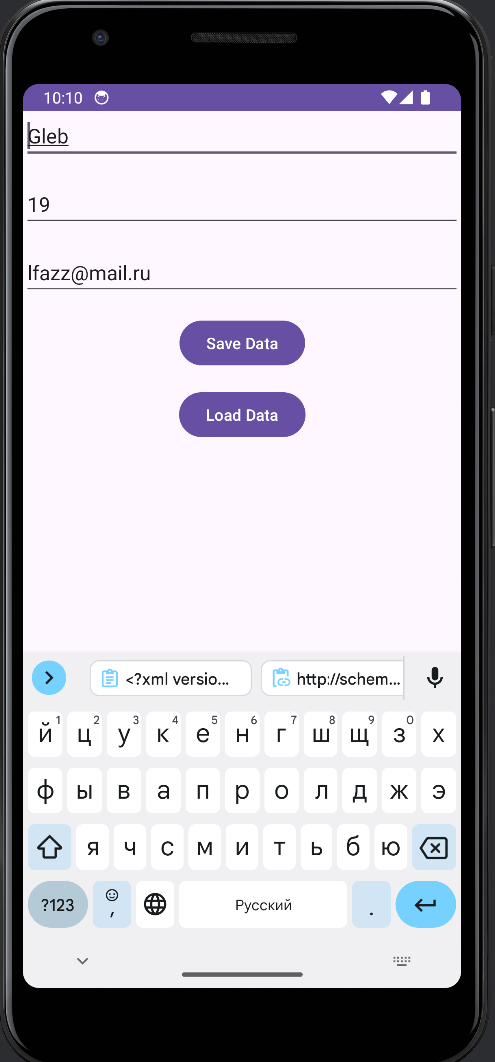


Рисунок 7. Нажали кнопку Load Data и все данные заполнились по полям