МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Звіт

з практичної роботи № 4 з дисципліни

Програмування під Android

Виконала: Перевірив:

ст. гр. ПЗПІ-23-5 ст. викладач кафедри ПІ

Висоцький Ігор Сокорчук Ігор Петрович

2024

ЗАВДАННЯ

Ознайомитися із різними способами збереження даних у Android, зокрема з використанням SharedPreferences, бази даних SQLite та роботи з файлами. Навчитися застосовувати ці методи у прикладних проєктах.

ХІД РОБОТИ

Cтворюємо новий проєкт та обираємо Empty Views Activity.

Робота з SharedPreferences:

SharedPreferences — це простий спосіб зберігати пари "ключ-значення", які зазвичай використовуються для збереження налаштувань застосунку або невеликих даних користувача. Файл налаштувань зберігається у вигляді XML у внутрішній пам'яті застосунку.

Створимо два EditText:  
1. <EditText  
2. android:id="@+id/editTextName"  
3. android:layout\_width="wrap\_content"  
4. android:layout\_height="wrap\_content"  
5. android:layout\_alignParentTop="true"  
6. android:layout\_marginTop="40dp"  
7. android:hint="Enter Name"  
8. android:minHeight="48dp" />  
9. <EditText  
10. android:id="@+id/editTextAge"  
11. android:layout\_width="wrap\_content"  
12. android:layout\_height="wrap\_content"  
13. android:layout\_below="@id/editTextName"  
14. android:layout\_marginTop="20dp"  
15. android:hint="Enter Age"  
16. android:minHeight="48dp" />

Далі додаємо кнопки для взаємодії з цими полями, а саме для збереження з використанням SharedPreferences. Повний файл розмітки можна подивитися у [ДОДАТОК А]. Код з використання SharedPreferences:  
1. SharedPreferences sharedPreferences = getSharedPreferences("MyPrefs",

2. *MODE\_PRIVATE*);  
3. String name = sharedPreferences.getString("name", "Default Name");  
4. String age = sharedPreferences.getString("age", "0");  
5. displayText.setText("Name: " + name + "\nAge: " + age);

Далі наведено частину коду для простого збереження даних:  
1. saveToFileButton.setOnClickListener(new View.OnClickListener() {  
2. @Override  
3. public void onClick(View v) {  
4. String name = editTextName.getText().toString();  
5. String age = editTextAge.getText().toString();  
 6. String data = "Name: " + name + "\nAge: " + age;  
7. writeToFile(FILE\_NAME, data);  
8. }  
9. });

Повний код використання sharedPreferences наведено у [ДОДАТОК Б].

SQLite є вбудованою реляційною базою даних, яка дозволяє працювати з великими обсягами структурованих даних. SQLite не вимагає налаштування сервера, і дані зберігаються локально у файлі. - Для роботи з SQLite у Android зазвичай створюється клас, який успадковується від SQLiteOpenHelper.

Отже, створємо клас SQLiteOpenHelper:  
1. public class DBHelper extends SQLiteOpenHelper {  
2. public DBHelper(Context context) {  
3. super(context, "MyDB", null, 1);  
4. }  
5 . @Override  
6. public void onCreate(SQLiteDatabase db) {  
7. db.execSQL("CREATE TABLE users (id INTEGER PRIMARY KEY, name 8. TEXT, age INTEGER)");  
9. }  
10. @Override  
11. public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {  
12. }  
13. public void addUser(String name, int age) {  
14. SQLiteDatabase db = this.getWritableDatabase();  
15. ContentValues values = new ContentValues();  
16. values.put("name", name);  
17. values.put("age", age);  
18. db.insert("users", null, values);  
19. db.close();  
20. }  
21. public Cursor getUsers() {  
22. SQLiteDatabase db = this.getReadableDatabase();  
23. return db.query("users", null, null, null, null, null, null);  
24. }  
25. public void clearDatabase() {  
26. SQLiteDatabase db = this.getWritableDatabase();  
27. db.delete("users", null, null); // Видаляємо всі записи  
28. db.close();  
29 }  
30}

Для взаємодії з базою даних я додав ще кнопок попередньо закоментувавши вже не потрібні, також я закоментував попередню частину коду для взаємодії з sharedPreferences. Новий код з використанням SQLite:   
1. dbHelper = new DBHelper(this);  
2. saveButton.setOnClickListener(new View.OnClickListener() {  
3. @Override  
4. public void onClick(View v) {  
5. String name = editTextName.getText().toString();  
6. int age = Integer.*parseInt*(editTextAge.getText().toString());  
7. dbHelper.addUser(name, age);  
8. }  
9. });

10. writeFileButton.setOnClickListener(new View.OnClickListener() {  
11. @Override  
12. public void onClick(View v) {  
13. writeToFile("users\_data.txt");  
14. }  
15. });

Повний код використання SQLite наведено у [ДОДАТОК Б].

ВИСНОВКИ

Під час практичної роботи було: ознайомлено з основними способами збереження даних у Android: Використання SharedPreferences для збереження простих налаштувань користувача у вигляді пар ключ-значення. Застосування SQLite для зберігання великих структурованих даних у реляційній базі. Робота з файлами для запису та зчитування даних із локального сховища додатка.

Відеозвіт на YouTube. URL: <https://www.youtube.com/watch?v=Y_O2wflg7cI>

ВИКОРИСТАНІ ДЖЕРЕЛА

1. Документація AndroidDevelopers SharedPreferences. URL: <https://developer.android.com/training/data-storage/shared-preferences> (дата звернення: 15.11.2024)

2. Документація AndroidDevelopers Save data using SQLite. URL: [https://developer.android.com/training/data-storage/sqlite (дата звернення:](https://developer.android.com/training/data-storage/sqlite%20(дата%20звернення:%20) 15.11.2024)

3. Документація AndroidDevelopers SQLiteDatabase. URL: [https://developer.android.com/reference/android/database/sqlite/SQLiteDatabase](https://developer.android.com/reference/android/database/sqlite/SQLiteDatabase%20) (дата звернення: 15.11.2024)

ДОДАТОК А

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:id="@+id/main"  
 android:layout\_height="match\_parent"  
 android:gravity="center">  
  
 <EditText  
 android:id="@+id/editTextName"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_marginTop="40dp"  
 android:hint="Enter Name"  
 android:minHeight="48dp" />  
  
 <EditText  
 android:id="@+id/editTextAge"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/editTextName"  
 android:layout\_marginTop="20dp"  
 android:hint="Enter Age"  
 android:minHeight="48dp" />  
  
 <Button  
 android:id="@+id/saveButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Save"  
 android:layout\_below="@id/editTextAge"  
 android:layout\_marginTop="20dp" />  
  
<!-- <TextView  
 android:id="@+id/displayText"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Your settings will appear here"  
 android:layout\_below="@id/saveButton"  
 android:layout\_marginTop="20dp"  
 android:layout\_marginBottom="20dp"/>  
  
 <Button  
 android:id="@+id/saveToFileButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/displayText"  
 android:text="Save to File" />  
  
 <Button  
 android:id="@+id/loadFromFileButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/saveToFileButton"  
 android:text="Load from File" />-->  
  
 <!-- Кнопка для завантаження списку користувачів -->  
 <Button  
 android:id="@+id/loadButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Load Users"  
 android:layout\_below="@id/saveButton"  
 android:layout\_marginBottom="16dp"  
 android:layout\_alignBaseline="@id/saveButton"  
 android:layout\_toRightOf="@id/saveButton"/>  
 <Button  
 android:id="@+id/clearButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Clear Database"  
 android:layout\_marginBottom="16dp"  
 android:layout\_below="@id/saveButton"  
 />  
 <!-- Список для відображення користувачів -->  
  
 <Button  
 android:id="@+id/writeFileButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Write to File"  
 android:layout\_below="@id/clearButton"  
 android:layout\_marginTop="16dp" />  
  
 <Button  
 android:id="@+id/readFileButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Read from File"  
 android:layout\_below="@id/writeFileButton"  
 android:layout\_marginTop="16dp" />  
  
  
 <TextView  
 android:id="@+id/textViewFileContent"  
 android:layout\_width="150dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/readFileButton"  
 android:layout\_marginTop="16dp"  
 android:textSize="16sp"  
 android:padding="8dp"  
 android:background="#f0f0f0"  
 android:text="File content will appear here." />  
 <ListView  
 android:id="@+id/listView"  
 android:layout\_width="100dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/textViewFileContent"  
 />  
  
</RelativeLayout>

ДОДАТОК Б

package ihor.vysotskyi.nure;  
  
import android.content.SharedPreferences;  
import android.database.Cursor;  
import android.os.Bundle;  
  
import androidx.activity.EdgeToEdge;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
import android.view.View;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ListView;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import java.io.\*;  
import java.util.ArrayList;  
  
public class MainActivity extends AppCompatActivity {  
  
 private EditText editTextName, editTextAge;  
 private TextView displayText;  
 private Button saveButton, loadButton, clearButton, writeFileButton, readFileButton, saveToFileButton, loadFromFileButton;  
 private ListView listView;  
 private DBHelper dbHelper;  
 private TextView textViewFileContent;  
 private final String FILE\_NAME = "user\_data.txt";  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.*enable*(this);  
 setContentView(R.layout.*activity\_main*);  
 ViewCompat.*setOnApplyWindowInsetsListener*(findViewById(R.id.*main*), (v, insets) -> {  
 Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.*systemBars*());  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  
 return insets;  
 });  
  
 /\*editTextName = findViewById(R.id.editTextName);  
 editTextAge = findViewById(R.id.editTextAge);  
 displayText = findViewById(R.id.displayText);  
 saveButton = findViewById(R.id.saveButton);  
 saveToFileButton = findViewById(R.id.saveToFileButton);  
 loadFromFileButton = findViewById(R.id.loadFromFileButton);  
  
 // Load shared preferences  
 SharedPreferences sharedPreferences = getSharedPreferences("MyPrefs", MODE\_PRIVATE);  
 String name = sharedPreferences.getString("name", "Default Name");  
 String age = sharedPreferences.getString("age", "0");  
 displayText.setText("Name: " + name + "\nAge: " + age);  
  
 // Save data when button is clicked  
 saveButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String name = editTextName.getText().toString();  
 String age = editTextAge.getText().toString();  
  
 // Save to SharedPreferences  
 SharedPreferences sharedPreferences = getSharedPreferences("MyPrefs", MODE\_PRIVATE);  
 SharedPreferences.Editor editor = sharedPreferences.edit();  
 editor.putString("name", name);  
 editor.putString("age", age);  
 editor.apply();  
  
 // Update displayed values  
 displayText.setText("Name: " + name + "\nAge: " + age);  
 }  
 });  
  
 // Save data to a file  
 saveToFileButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String name = editTextName.getText().toString();  
 String age = editTextAge.getText().toString();  
 String data = "Name: " + name + "\nAge: " + age;  
  
 writeToFile(FILE\_NAME, data);  
 }  
 });  
  
 // Load data from a file  
 loadFromFileButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String fileData = readFromFile(FILE\_NAME, true);  
 if (fileData != null) {  
 displayText.setText(fileData);  
 }  
 }  
 });\*/  
 editTextName = findViewById(R.id.*editTextName*);  
 editTextAge = findViewById(R.id.*editTextAge*);  
 saveButton = findViewById(R.id.*saveButton*);  
 loadButton = findViewById(R.id.*loadButton*);  
 listView = findViewById(R.id.*listView*);  
 clearButton = findViewById(R.id.*clearButton*);  
 writeFileButton = findViewById(R.id.*writeFileButton*);  
 readFileButton = findViewById(R.id.*readFileButton*);  
 textViewFileContent = findViewById(R.id.*textViewFileContent*);  
  
 dbHelper = new DBHelper(this);  
  
 saveButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String name = editTextName.getText().toString();  
 int age = Integer.*parseInt*(editTextAge.getText().toString());  
 dbHelper.addUser(name, age);  
 }  
 });  
  
 loadButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Cursor cursor = dbHelper.getUsers();  
 ArrayList<String> userList = new ArrayList<>();  
  
 if (cursor != null && cursor.moveToFirst()) {  
 // Отримуємо індекси колонок  
 int nameIndex = cursor.getColumnIndex("name");  
 int ageIndex = cursor.getColumnIndex("age");  
  
 // Перевірка індексів  
 if (nameIndex != -1 && ageIndex != -1) {  
 do {  
 String name = cursor.getString(nameIndex);  
 int age = cursor.getInt(ageIndex);  
 userList.add("Name: " + name + ", Age: " + age);  
 } while (cursor.moveToNext());  
 }  
 cursor.close(); // Закриваємо курсор  
 } else {  
 userList.add("No data available");  
 }  
  
 // Встановлюємо адаптер для ListView  
 ArrayAdapter<String> adapter = new ArrayAdapter<>(  
 MainActivity.this,  
 android.R.layout.*simple\_list\_item\_1*,  
 userList  
 );  
  
 listView.setAdapter(adapter);  
 }  
 });  
 clearButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 dbHelper.clearDatabase();  
 ArrayAdapter<String> adapter = new ArrayAdapter<>(  
 MainActivity.this,  
 android.R.layout.*simple\_list\_item\_1*,  
 new ArrayList<>()  
 );  
 listView.setAdapter(adapter); // Очищаємо список  
 }  
 });  
  
 writeFileButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 writeToFile("users\_data.txt");  
 }  
 });  
  
 readFileButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String content = readFromFile("users\_data.txt");  
 textViewFileContent.setText(content);  
 }  
 });  
 }  
 private void writeToFile(String fileName, String content) {  
 try (FileOutputStream fos = openFileOutput(fileName, *MODE\_PRIVATE*)) {  
 fos.write(content.getBytes());  
 Toast.*makeText*(this, "Data written to file", Toast.*LENGTH\_SHORT*).show();  
 } catch (IOException e) {  
 Toast.*makeText*(this, "Error writing to file: " + e.getMessage(), Toast.*LENGTH\_LONG*).show();  
 }  
 }  
  
 // Method to read data from a file  
 private String readFromFile(String fileName, boolean overload) {  
 try (FileInputStream fis = openFileInput(fileName)) {  
 StringBuilder data = new StringBuilder();  
 int c;  
 while ((c = fis.read()) != -1) {  
 data.append((char) c);  
 }  
 Toast.*makeText*(this, "Data loaded from file", Toast.*LENGTH\_SHORT*).show();  
 return data.toString();  
 } catch (IOException e) {  
 Toast.*makeText*(this, "Error reading from file: " + e.getMessage(), Toast.*LENGTH\_LONG*).show();  
 return null;  
 }  
 }  
  
 private void writeToFile(String fileName) {  
 try (FileOutputStream fos = openFileOutput(fileName, *MODE\_PRIVATE*)) {  
 Cursor cursor = dbHelper.getUsers();  
 StringBuilder data = new StringBuilder();  
  
 int nameIndex = cursor.getColumnIndex("name");  
 int ageIndex = cursor.getColumnIndex("age");  
  
 if (nameIndex == -1 || ageIndex == -1) {  
 Toast.*makeText*(this, "Error: Invalid column name", Toast.*LENGTH\_LONG*).show();  
 cursor.close();  
 return;  
 }  
  
 while (cursor.moveToNext()) {  
 String name = cursor.getString(nameIndex);  
 int age = cursor.getInt(ageIndex);  
 data.append("Name: ").append(name).append(", Age: ").append(age).append("\n");  
 }  
 cursor.close();  
  
 fos.write(data.toString().getBytes());  
 Toast.*makeText*(this, "Data written to file", Toast.*LENGTH\_SHORT*).show();  
 } catch (IOException e) {  
 Toast.*makeText*(this, "Error writing to file: " + e.getMessage(), Toast.*LENGTH\_LONG*).show();  
 }  
 }  
  
 private String readFromFile(String fileName) {  
 StringBuilder content = new StringBuilder();  
 try (FileInputStream fis = openFileInput(fileName)) {  
 int c;  
 while ((c = fis.read()) != -1) {  
 content.append((char) c);  
 }  
 } catch (IOException e) {  
 Toast.*makeText*(this, "Error reading from file: " + e.getMessage(), Toast.*LENGTH\_LONG*).show();  
 }  
 return content.toString();  
 }  
}