МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Звіт

з практичної роботи № 4 з дисципліни

Програмування під Android

Виконав: Перевірив:

ст. гр. ПЗПІ-23-4 ст. викладач кафедри ПІ

Юзков Олександр Сокорчук Ігор Петрович

Харків 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.

Почнемо з того, що створимо об’єкт, який зберігається:

public class User {

Integer userId;

String userName;

Integer userAge;

public User(Integer userId, String userName, Integer userAge) {

this.userId = userId;

this.userName = userName;

this.userAge = userAge;

}

public Integer getUserId() {

return userId;

}

public String getUserName() {

return userName;

}

public Integer getUserAge() {

return userAge;

}

}

Маємо користувача з ім’ям, віком і унікальним ID, який згодом буде використовуватися для роботи з базою даних SQLite. Тепер створимо інтерфейс для взаємодії з користувачем, використовуючи ConstraintLayout і вкладений RecyclerView. Додамо поля для введення імені та віку нового користувача, а також кнопки для збереження та отримання інформації.

SQLite — це вбудована реляційна база даних, яка дозволяє працювати з великими обсягами структурованих даних без необхідності налаштовувати сервер. Дані зберігаються локально у файлі.

Для роботи з SQLite в Android зазвичай створюється клас, який успадковується від SQLiteOpenHelper. Створимо клас DatabaseHelper, який міститиме логіку роботи з даними.

DatabaseHelper — це клас, що забезпечує взаємодію з базою даних SQLite і дозволяє зберігати налаштування користувача за допомогою SharedPreferences. Він успадковується від SQLiteOpenHelper та забезпечує виконання операцій із базою, таких як додавання нових користувачів, зчитування даних і збереження їх у локальних налаштуваннях.

Основні функції класу:

1. Збереження користувача в базі даних:  
   Метод insertUser() додає нового користувача до таблиці user.
2. Зчитування даних користувачів:  
   Метод getUsersFromDatabase() отримує всі записи з таблиці user та повертає їх як список об'єктів User.
3. Робота з налаштуваннями:  
   Ім’я та вік користувача зберігаються у SharedPreferences через методи saveUserToPreferences(), getNameFromPreferences() та getAgeFromPreferences().

Цей клас забезпечує зручний інструмент для роботи як із базою даних, так і з локальними налаштуваннями користувачів.

Далі створимо клас UserAdapter, який буде адаптером для RecyclerView. Він використовує DataBinding для відображення списку користувачів. UserAdapter працює зі списком об'єктів User, які містять ім’я, вік і ID.

Ключові методи адаптера:

* updateUserList: оновлює дані в списку користувачів для подальшого відображення в інтерфейсі.

Цей підхід забезпечує інтерактивність і зручність роботи з даними користувачів у застосунку.

відповідає за оновлення списку користувачів у адаптері. Якщо переданий список не порожній, він очищує поточний список і додає нові елементи, після чого викликається notifyDataSetChanged() для оновлення відображення в RecyclerView. Метод onCreateViewHolder створює нові елементи інтерфейсу, інфлейтячи розмітку з допомогою DataBinding. Він повертає новий об'єкт UserViewHolder, який тримає посилання на ListUserBinding, що відповідає за прив'язку даних до елементів інтерфейсу. Метод onBindViewHolder прив'язує конкретні дані користувача (ім'я та вік) до відповідних полів в інтерфейсі.

Клас також містить метод getItemCount, який повертає кількість елементів у списку, необхідну для відображення. Весь процес забезпечує зручну інтеграцію між даними та UI за допомогою DataBinding, що значно спрощує роботу з інтерфейсом користувача.

Переходимо до MainActivity. MainActivity є головною активністю додатка, що дозволяє користувачам додавати нових користувачів, зберігати та відображати налаштування, а також працювати з базою даних та локальними файлами.

Основний функціонал включає:

1. Додавання користувачів: Користувач може ввести ім'я та вік, і після натискання кнопки ці дані додаються до бази даних SQLite, а також зберігаються в налаштуваннях.
2. Відображення користувачів: За допомогою RecyclerView та адаптера UserAdapter відображається список користувачів, що зберігаються в базі даних.
3. Збереження та зчитування налаштувань: Є можливість зберігати налаштування (у вигляді тексту) у файл та зчитувати їх для відображення.
4. Локальні файли: Дані з налаштувань записуються у файл config.txt, який згодом можна прочитати.

ВИСНОВКИ

Під час практичної роботи було вивчено основні методи збереження даних в Android: використання SharedPreferences для зберігання простих налаштувань користувача у вигляді пар ключ-значення, застосування SQLite для роботи з великими структурованими даними в реляційній базі, а також робота з файлами для запису та зчитування даних з локального сховища додатка.

Посилання на відеозвіт завантажений на YouTube. URL: https://youtu.be/EN0k3Kdj2qA

ВИКОРИСТАНІ ДЖЕРЕЛА

1. Документація AndroidDevelopers SharedPreferences. URL: https://developer.android.com/training/data-storage/shared-preferences (дата звернення: 11.01.2025)
2. Документація AndroidDevelopers Save data using SQLite. URL: https://developer.android.com/training/data-storage/sqlite (дата звернення: 11.01.2025)
3. Документація AndroidDevelopers SQLiteDatabase. URL: https://developer.android.com/reference/android/database/sqlite/SQLiteDatabase (дата звернення: 11.01.2025)