Доповідь на тему: “Основи роботи з базами даних, SQLite”

Підготував Меленко Олександр

**Слайд 1**

Бази даних є невід'ємною складовою сучасних мобільних додатків, оскільки вони дозволяють зберігати, обробляти та швидко отримувати доступ до даних, необхідних для роботи додатку.

У мобільних додатках бази даних забезпечують зберігання даних, що може включати користувацькі дані, налаштування, контент та іншу інформацію, яка повинна зберігатись локально або синхронізуватись із сервером.  
  
**Слайд 2**

Бази даних – це організовані сховища інформації, які дозволяють зберігати, керувати й швидко знаходити великі обсяги даних. Вони використовуються для структурованого зберігання даних, таких як текст, числа, зображення, відео та інші типи інформації. Бази даних забезпечують можливість ефективно здійснювати пошук, оновлення, видалення та аналіз даних, що робить їх важливими для роботи програмного забезпечення, аналітичних систем, вебсайтів, бізнес-додатків та багато іншого.

Бази даних відіграють важливу роль у мобільних додатках, забезпечуючи зберігання й організацію даних, необхідних для роботи додатка. Вони дозволяють зберігати інформацію на пристрої, що забезпечує доступ до даних навіть без підключення до Інтернету, і допомагають економити трафік та зменшувати час завантаження. Завдяки базам даних мобільні додатки можуть ефективно обробляти інформацію користувача, як-от облікові записи, налаштування, історію дій та персоналізовані дані.  
  
**Слайд 3**  
  
SQLite – це легка, вбудована система управління базами даних (СУБД), яка зберігає дані в одному файлі на диску, без потреби в окремому сервері. Її часто використовують у мобільних додатках, вбудованих системах і для невеликих веб-проектів, де важлива простота й автономність. SQLite забезпечує базові можливості реляційної бази даних, як-от зберігання, запити та індексація даних, ідеально підходить для програм, де не потрібне одночасне використання великою кількістю користувачів. Вона також підтримує транзакції й забезпечує надійне зберігання даних завдяки механізму журналювання.

**Слайд 4**

Для включення SQLite у проект Android і управління базою даних часто використовують клас SQLiteOpenHelper. Він полегшує створення та оновлення бази даних, а також надає методи для підключення до неї. Давайте розглянемо етапи включення SQLite в Android-проект, використання SQLiteOpenHelper та визначення схеми бази даних.

SQLite вже вбудована в Android SDK, тому додаткових залежностей не потрібно. Просто почніть створення бази даних за допомогою класу SQLiteOpenHelper.

Загальний процес роботи з SQLite у проекті Android

1. Створення класу, що успадковує SQLiteOpenHelper.
2. Визначення схеми бази даних у методі onCreate.
3. Реалізація логіки оновлення в методі onUpgrade.
4. Використання об’єкта SQLiteDatabase для виконання CRUD-операцій (створення, читання, оновлення, видалення).

**Слайд 5**  
  
Щоб полегшити роботу з базою даних у проєкті Android, можна створити інтерфейс для взаємодії з нею. Такий інтерфейс абстрагує основні операції, що виконуються над базою даних: створення, читання, оновлення та видалення (CRUD-операції). Для отримання даних з бази використовується Cursor – об'єкт, який дозволяє переміщатися по рядках результату запиту  
  
Використання курсорів спрощує отримання й обробку даних з бази:

1. **Інтерфейс** визначає основні CRUD-операції для таблиці.
2. **Реалізація інтерфейсу** обробляє з’єднання з базою даних і виконує SQL-запити.
3. **Курсори** забезпечують доступ до результатів запитів і зручну обробку даних.

**Слайд 6**

Для підвищення продуктивності можна використовувати кешування, особливо для запитів, результати яких часто не змінюються. Можна застосувати підходи до кешування результатів запитів, як-от збереження даних у пам’яті або використання SharedPreferences для простих структур даних.  
  
SQLite є потокобезпечною, але операції з базою можуть блокувати основний потік, що погіршує користувацький досвід. Виконання цих операцій в асинхронному режимі запобігає заморожуванню інтерфейсу.  
  
Щоб уникнути витоків пам'яті, важливо належним чином закривати об'єкти Cursor та з'єднання з базою даних. SQLite не є автоматично керованою ресурсами базою, тому курсори слід закривати після використання, а базу даних – закривати у належний момент, як-от під час завершення роботи додатку.  
  
**Слайд 7**

Тестування й налагодження бази даних в Android допомагає знайти та виправити помилки при взаємодії з базою даних SQLite. Android Studio надає кілька інструментів, зокрема **Database Inspector**, який дозволяє відслідковувати стан бази даних в режимі реального часу, виконувати запити та аналізувати структуру бази.

**Слайд 8**

База даних SQLite часто використовується для зберігання профілю користувача, а також для обробки аутентифікації чи персоналізованих налаштувань. Можна зберігати такі дані, як ім'я, електронна пошта, телефонний номер, а також збереження історії дій користувача.

SQLite може бути використана для зберігання налаштувань додатку, таких як теми, мова, або інші параметри, що змінюються в процесі використання. Такі налаштування зберігаються в окремих таблицях або базах даних для швидкого доступу.

SQLite ідеально підходить для зберігання списків даних, таких як список завдань, контактів, товарів чи будь-яких інших елементів, які мають структуру списку з кількома атрибутами.