SQLite

Mobilne aplikacije

SQLite

- Android aplikacije mogu da koriste ugrađen sistem za upravljanje bazama podataka (SQLite)
- Za razliku od većine sistema za upravljanje bazama podataka, SQLite se izvršava u istom procesu kao i aplikacija koja koristi njegove usluge
- Obezbeđuje referencijalni integritet i omogućava rad u transakcijama

sqlite3

Naredba	Opis
`.databases`	Lists names and files of attached databases.
`.tables ?TABLE?`	Lists names of tables (if TABLE is specified, only dumps tables matching LIKE pattern TABLE).
`.dump ?TABLE? ...`	Dumps the database in an SQL text format (if TABLE is specified, only dumps tables matching LIKE pattern TABLE).
`.schema ?TABLE?`	Shows the CREATE statements (if TABLE is specified, only dumps tables matching LIKE pattern TABLE).
`.backup ?DB? FILE`	Backups database (default "main") to FILE.
`.restore ?DB? FILE`	Restores content of the database (default "main") from FILE.

Table 1: Sqlite3 naredbe.

sqlite3

Naredba	Opis
`.read FILENAME`	Executes SQL in FILENAME.
`.import FILE TABLE`	Imports data from FILE into TABLE.
`.headers on|off`	Turns display of headers on or off.
`.mode MODE ?TABLE?`	Set output mode where MODE is one of: csv (comma-separated values), column (left-aligned columns), html (HTML <table> code), insert (SQL insert statements for TABLE), line (one value per line), list (values delimited by .separator string), tabs (tab-separated values) or tcl (TCL list elements)
`.nullvalue STRING`	Use STRING in place of NULL values.
`<sql statement>`	Može se izvršiti i proizvoljna SQL naredba.

Table 2: Sqlite3 naredbe.

sqlite3

```
C:\Users\user\AppData\Local\Android\Sdk\platform-tools>adb devices
List of devices attached
emulator-5554   device

> adb -s emulator-5554 shell

> run-as package_name

> cd databases

> sqlite3 db_name
SQLite version 3.9.2
Enter ".help" for usage hints.
.... enter commands, then quit...

sqlite>.exit
> _

```

SQLite

- Za pravljenje, izmenu i otvaranje baze podataka koristi se `SQLiteOpenHelper` klasa
- Potrebno je implementirati neke od sledećih metoda:
  - `void onCreate(SQLiteDatabase database)`
  - `void onOpen(SQLiteDatabase database)`
  - `void onUpgrade(SQLiteDatabase database, int old_ver, int new_ver)`
  - `void onDowngrade(SQLiteDatabase database, int old_ver, int new_ver)`

SQLiteOpenHelper.java

```java
public class ExampleOpenHelper extends SQLiteOpenHelper {

  private static final String CREATE_DATABASE =
    "create table NOTES ( " +
    "  _id integer primary key autoincrement, " +
    "  naslov text not null, " +
    "  vreme  text not null, " +
    "  tekst  text not null);";

  public ExampleOpenHelper(Context context) {
    super(context, DATABASE_NAME, null, DATABASE_VERSION);
  }

  @Override
  public void onCreate(SQLiteDatabase db) {
    db.execSQL(CREATE_DATABASE);
  }

  @Override
  public void onUpgrade(SQLiteDatabase db, int old_ver, int new_ver) {
    db.execSQL("DROP TABLE IF EXISTS " + DATABASE_TABLE);
    onCreate(db);
  }
}
```

SQLite

- Baza podataka predstavljena je klasom `SQLiteDatabase`.
- CRUD operacije nad bazom podataka izvršavaju se pozivom insert, query, update i delete metoda
  - `long insert(String table, String null_hack, ContentValues entry)`
  - `Cursor query(String table, String[] columns, String whereClause, String[] whereArgs, String groupBy, String having, String orderBy, String limit)`
  - `int update(String table, ContentValues values, String whereClause, String[] whereArgs)`
  - `int delete(String table, String whereClause, String[] whereArgs)`

SQLiteDatabase

```java
// Connects to the database in write mode
SQLiteOpenHelper helper = new ExampleOpenHelper(this.context);
SQLiteDatabase db = helper.getWritableDatabase();

```

SQLiteDatabase

```
// Demonstrates the usage of insert method
ContentValues entry = new ContentValues();
entry.put("naslov", "Namirnice");
entry.put("vreme", "00:53");
entry.put("tekst", "Kupiti hleb i mleko.");
long id = db.insert(DATABASE_TABLE, null, entry);
```

SQLiteDatabase

```java
// Demonstrates the usage of query method
Cursor c = db.query(
  DATABASE_TABLE,
  new String[] {_ID, TITLE, TIMESTAMP, TEXT},
  "_ID = ?",
  {id},
  groupBy,
  having,
  orderBy,
  limit);
```

SQLiteDatabase

```
// Demonstrates the usage of update method
ContentValues entry = new ContentValues();
entry.put("naslov", "Namirnice");
entry.put("vreme", "00:53");
entry.put("tekst", "Kupiti hleb i mleko.");
long id = db.update(DATABASE_TABLE, entry, whereClause, whereArgs);
```

SQLiteDatabase

```
// Demonstrates the usage of delete method
long id = db.delete(DATABASE_TABLE, "_ID = ?", {id});
```

Kursori

- Relacija koja je rezultat SQL upita predstavljena je kursorom (`Cursor`)
- Kursori se koriste za navigaciju kroz rezultat upita:
  - `boolean move(int offset)`
  - `boolean moveToFirst()`
  - `boolean moveToLast()`
  - `boolean moveToNext()`
  - `boolean moveToPrevious()`
- kao i za čitanje rezultata upita:
  - `int getCount()`
  - `int getColumnIndex(String column_name)`
  - `String getColumnName(int column_index)`
  - `String getString(int column_index)`
  - `int getInt(int column_index)`
  - `long getLong(int column_index)`
  - `float getFloat(int column_index)`
  - `double getDouble(int column_index)`