# Додавання та видалення рядків у базі даних SQLite до Android

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:orientation="vertical">

<EditText

android:id="@+id/etId"

android:hint="Id"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

</EditText>

<EditText

android:id="@+id/etName"

android:hint="Name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

</EditText>

<EditText

android:id="@+id/etEmail"

android:hint="Email"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

</EditText>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<Button

android:id="@+id/btnUpd"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Update">

</Button>

<Button

android:id="@+id/btnDel"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Delete">

</Button>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<Button

android:id="@+id/btnAdd"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Add">

</Button>

<Button

android:id="@+id/btnRead"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Read">

</Button>

<Button

android:id="@+id/btnClear"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Clear">

</Button>

</LinearLayout>

</LinearLayout>

package ...

import android.content.Context;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

public class DBHelper extends SQLiteOpenHelper{

public static final int DATABASE\_VERSION = 1;

public static final String DATABASE\_NAME = "contactDb";

public static final String TABLE\_CONTACTS = "contacts";

public static final String KEY\_ID = "\_id";

public static final String KEY\_NAME = "name";

public static final String KEY\_MAIL = "mail";

public DBHelper(Context context) {

super(context, DATABASE\_NAME, null, DATABASE\_VERSION);

}

@Override

public void onCreate(SQLiteDatabase db) {

db.execSQL("create table" + TABLE\_CONTACTS + "(" + KEY\_ID

+ "integer primary key," + KEY\_NAME + "text," + KEY\_MAIL + "text" + ")");

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

db.execSQL("drop table if exists" + TABLE\_CONTACTS);

onCreate(db);

}

}

package ...

import android.content.ContentValues;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

public class MainActivity extends AppCompatActivity implements View.OnClickListener{

Button btnAdd, btnRead, btnClear, btnUpd, btnDel;

EditText etName, etEmail, etId;

DBHelper dbHelper;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btnAdd = (Button) findViewById(R.id.btnAdd);

btnAdd.setOnClickListener(this);

btnRead = (Button) findViewById(R.id.btnRead);

btnRead.setOnClickListener(this);

btnUpd = (Button) findViewById(R.id.btnUpd);

btnUpd.setOnClickListener(this);

btnDel = (Button) findViewById(R.id.btnDel);

btnDel.setOnClickListener(this);

btnClear = (Button) findViewById(R.id.btnClear);

btnClear.setOnClickListener(this);

etName = (EditText) findViewById(R.id.etName);

etEmail = (EditText) findViewById(R.id.etEmail);

etId = (EditText) findViewById(R.id.etId);

dbHelper = new DBHelper(this);

}

@Override

public void onClick(View v) {

String name = etName.getText().toString();

String email = etEmail.getText().toString();

String id = etId.getText().toString();

SQLiteDatabase database = dbHelper.getWritableDatabase();

ContentValues contentValues = new ContentValues();

switch (v.getId()) {

case R.id.btnAdd:

contentValues.put(DBHelper.KEY\_NAME, name);

contentValues.put(DBHelper.KEY\_MAIL, email);

database.insert(DBHelper.TABLE\_CONTACTS, null, contentValues);

break;

case R.id.btnRead:

Cursor cursor = database.query(DBHelper.TABLE\_CONTACTS, null, null, null, null, null, null);

if (cursor.moveToFirst()) {

int idIndex = cursor.getColumnIndex(DBHelper.KEY\_ID);

int nameIndex = cursor.getColumnIndex(DBHelper.KEY\_NAME);

int emailIndex = cursor.getColumnIndex(DBHelper.KEY\_MAIL);

do {

Log.d("mLog", "ID = " + cursor.getInt(idIndex) +

", name = " + cursor.getString(nameIndex) +

", email = " + cursor.getString(emailIndex));

} while (cursor.moveToNext());

} else

Log.d("mLog","0 rows");

cursor.close();

break;

case R.id.btnClear:

database.delete(DBHelper.TABLE\_CONTACTS, null, null);

break;

case R.id.btnUpd:

if (id.equalsIgnoreCase("")){

break;

}

contentValues.put(DBHelper.KEY\_MAIL, email);

contentValues.put(DBHelper.KEY\_NAME, name);

int updCount = database.update(DBHelper.TABLE\_CONTACTS, contentValues, DBHelper.KEY\_ID + "=?", new String[] {id});

Log.d("mLog", "updates rows count="+updCount);

case R.id.btnDel:

if (id.equalsIgnoreCase("")){

break;

}

int delCount = database.delete(DBHelper.TABLE\_CONTACTS, DBHelper.KEY\_ID + "=" + id, null);

Log.d("mLog", "deleted rows count="+delCount);

}

dbHelper.close();

}

}

# Робота з базою даних SQLite в Android, групування, сортування, вибірка за умовою

Будемо працювати з базою даних SQLite в Android. Створимо додаток, на прикладі якого покажемо, як робити групування, сортування та вибірку даних за умовою в базі даних SQLite за допомогою методу query з параметрами columns, selection, selectionArgs, groupBy, having, orderBy.

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:orientation="vertical">

<TextView

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:text="Довідник країн"

android:textSize="14sp"

android:gravity="center\_horizontal"

android:layout\_marginBottom="5dp"

android:layout\_marginTop="5dp">

</TextView>

<Button

android:id="@+id/btnAll"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Всі записи"

android:layout\_marginTop="5dp">

</Button>

<LinearLayout

android:id="@+id/linearLayout1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5dp">

<Button

android:id="@+id/btnFunc"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Функція">

</Button>

<EditText

android:id="@+id/etFunc"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1">

<requestFocus>

</requestFocus>

</EditText>

</LinearLayout>

<LinearLayout

android:id="@+id/linearLayout2"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5dp">

<Button

android:id="@+id/btnPeople"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Населення >">

</Button>

<EditText

android:id="@+id/etPeople"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:inputType="number">

</EditText>

</LinearLayout>

<Button

android:id="@+id/btnGroup"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Населення по регіону"

android:layout\_marginTop="5dp">

</Button>

<LinearLayout

android:id="@+id/linearLayout4"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5dp">

<Button

android:id="@+id/btnHaving"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Населення по регіону >">

</Button>

<EditText

android:id="@+id/etRegionPeople"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:inputType="number">

</EditText>

</LinearLayout>

<LinearLayout

android:id="@+id/linearLayout3"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5dp">

<Button

android:id="@+id/btnSort"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Сортування">

</Button>

<RadioGroup

android:id="@+id/rgSort"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content">

<RadioButton

android:id="@+id/rName"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:checked="true"

android:text="Найменування">

</RadioButton>

<RadioButton

android:id="@+id/rPeople"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Населення">

</RadioButton>

<RadioButton

android:id="@+id/rRegion"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Регіон">

</RadioButton>

</RadioGroup>

</LinearLayout>

</LinearLayout>

package ...

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.RadioGroup;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

final String LOG\_TAG = "myLogs";

String name[] = { "Китай", "США", "Бразилія", "Україна", "Японія", "Німеччина", "Єгипет", "Італія", "Франція", "Канада"};

int people[] = { 1400, 311, 195, 142, 128, 82, 80, 60, 66, 35};

String region[] = { "Азія", "Америка", "Америка", "Європа", "Азія", "Європа", "Африка", "Європа", "Європа", "Америка"};

Button btnAll, btnFunc, btnPeople, btnSort, btnGroup, btnHaving;

EditText etFunc, etPeople, etRegionPeople;

RadioGroup rgSort;

DBHelper dbHelper;

SQLiteDatabase db;

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btnAll = (Button) findViewById(R.id.btnAll);

btnAll.setOnClickListener(this);

btnFunc = (Button) findViewById(R.id.btnFunc);

btnFunc.setOnClickListener(this);

btnPeople = (Button) findViewById(R.id.btnPeople);

btnPeople.setOnClickListener(this);

btnSort = (Button) findViewById(R.id.btnSort);

btnSort.setOnClickListener(this);

btnGroup = (Button) findViewById(R.id.btnGroup);

btnGroup.setOnClickListener(this);

btnHaving = (Button) findViewById(R.id.btnHaving);

btnHaving.setOnClickListener(this);

etFunc = (EditText) findViewById(R.id.etFunc);

etPeople = (EditText) findViewById(R.id.etPeople);

etRegionPeople = (EditText) findViewById(R.id.etRegionPeople);

rgSort = (RadioGroup) findViewById(R.id.rgSort);

dbHelper = новий DBHelper(this);

// Підключаємося до бази

db = dbHelper.getWritableDatabase();

// Перевірка існування записів

Cursor c = db.query("mytable", null, null, null, null, null, null);

if (c.getCount() == 0) {

ContentValues ​​cv = New ContentValues();

// заповнимо таблицю

for (int i = 0; i < 10; i++) {

cv.put("name", name[i]);

cv.put("people", people[i]);

cv.put("region", region[i]);

Log.d(LOG\_TAG, "id="+db.insert("mytable", null, cv));

}

}

c.close();

dbHelper.close();

// Емулюємо натискання кнопки btnAll

onClick(btnAll);

}

public void onClick(View v) {

// Підключаємося до бази

db = dbHelper.getWritableDatabase();

// Дані з екрану

String sFunc = etFunc.getText().toString();

String sPeople = etPeople.getText().toString();

String sRegionPeople = etRegionPeople.getText().toString();

// змінні для query

String[] columns = null;

String selection = null;

String[] selectionArgs = null;

String groupBy = null;

String having = null;

String orderBy = null;

// курсор

Cursor c=null;

// Визначаємо натиснуту кнопку

switch (v.getId()) {

// Усі записи

case R.id.btnAll:

Log.d(LOG\_TAG, "--- Всі записи ---");

c = db.query("mytable", null, null, null, null, null, null);

break;

// Функція

case R.id.btnFunc:

Log.d(LOG\_TAG, "--- Функція " + sFunc + " ---");

columns = new String[] { sFunc };

c = db.query("mytable", columns, null, null, null, null, null);

break;

// Населення більше, ніж

case R.id.btnPeople:

Log.d(LOG\_TAG, "--- Населення більше " + sPeople + " ---");

selection = "people > ?";

selectionArgs = new String[] { sPeople };

c = db.query("mytable", null, selection, selectionArgs, null, null,

null);

break;

// Населення по регіону

case R.id.btnGroup:

Log.d(LOG\_TAG, "--- Населення по регіону ---");

columns = new String[] { "region", "sum(people) as people" };

groupBy="region";

c = db.query("mytable", columns, null, null, groupBy, null, null);

break;

// Населення по регіону більше, ніж

case R.id.btnHaving:

Log.d(LOG\_TAG, "--- Регіони з населенням більше " + sRegionPeople

+ "---");

columns = new String[] { "region", "sum(people) as people" };

groupBy="region";

having = "sum(people) > " + sRegionPeople;

c = db.query("mytable", columns, null, null, groupBy, having, null);

break;

// Сортування

case R.id.btnSort:

// сортування за

switch (rgSort.getCheckedRadioButtonId()) {

// найменування

case R.id.rName:

Log.d(LOG\_TAG, "--- Сортування за назвою ---");

orderBy = "name";

break;

// населення

case R.id.rPeople:

Log.d(LOG\_TAG, "--- Сортування за населенням ---");

orderBy = "люди";

break;

// регіон

case R.id.rRegion:

Log.d(LOG\_TAG, "--- Сортування по регіону ---");

orderBy = "Регіон";

break;

}

c = db.query("mytable", null, null, null, null, null, orderBy);

break;

}

if (c! = null) {

if (c.moveToFirst()) {

String str;

do {

str = "";

for (String cn : c.getColumnNames()) {

str = str.concat(cn + " = "

+ c.getString(c.getColumnIndex(cn)) + "; ");

}

Log.d(LOG\_TAG, str);

} while (c.moveToNext());

}

c.close();

} else

Log.d(LOG\_TAG, "Cursor is null");

dbHelper.close();

}

class DBHelper extends SQLiteOpenHelper {

public DBHelper(Context context) {

// конструктор суперкласу

super(context, "myDB", null, 1);

}

public void onCreate(SQLiteDatabase db) {

Log.d(LOG\_TAG, "--- onCreate database ---");

// Створюємо таблицю з полями

db.execSQL("create table mytable ("

+ "id integer primary key autoincrement," + "name text,"

+ "people integer," + "region text" + ");");

}

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

}

}

}