**Лабораторная работа № 8**

**Тема:** Работа с базой данных SQLite.

**Цель:** Научиться работать с основными командами SQL.

**Задание:**

1. Разработать мобильное приложение на основе базы данных для реализации команд добавить, удалить, обновить данные;

**Порядок выполнения:**

1. Создадим новый проект;
2. Добавим на форму 3 компонента «EditText\*», а также 5 компонентов «Button» с соответствующими именами «ID», «Name», «E-mail», «Добавить», «Удалить», «Очистить», «Считать», «Обновить».
3. Добавим новый класс для работы с базой данных и назовём его «DBHelper»;
4. Расширим этот класс методами «onCreate» и «onUpdate»;
5. Выберем конструктор суперкласса «SQLLiteOpenHelper» с параметрами – «Context», «Name», «Factory», «Version» и удалим лишние параметры:

**public** DBHelper(@Nullable Context context)  
{  
 **super**(context, ***DATABASE\_NAME***, **null**, ***DATABASE\_VERSION***);  
}

1. Запишем код:

**public class** DBHelper **extends** SQLiteOpenHelper  
{  
 **public static final int *DATABASE\_VERSION*** = 1;  
 **public static final** String ***DATABASE\_NAME*** = **"myBase"**;  
 **public static final** String ***TABLE\_PERSONS*** = **"persons"**;  
 **public static final** String ***KEY\_ID*** = **"\_id"**;  
 **public static final** String ***KEY\_NAME*** = **"name"**;  
 **public static final** String ***KEY\_MAIL*** = **"mail"**;  
  
 **public** DBHelper(@Nullable Context context)  
 {  
 **super**(context, ***DATABASE\_NAME***, **null**, ***DATABASE\_VERSION***);  
 }  
  
 @Override  
 **public void** onCreate(SQLiteDatabase db)  
 {  
 db.execSQL(**"create table "** + ***TABLE\_PERSONS*** + **"("** + ***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\_PERSONS***);  
 onCreate(db);  
 }  
}

1. Расширим класс «MainActivity» и добавим наши переменные:
2. **public class** MainActivity **extends** AppCompatActivity **implements** View.OnClickListener  
   {  
    Button **buttonAdd**, **buttonDelete**, **buttonClear**, **buttonRead**, **buttonUpdate**;  
    EditText **etName**, **etEmail**, **etID**;  
    DBHelper **dbHelper**;  
     
    @Override  
    **protected void** onCreate(Bundle savedInstanceState)  
    {  
    **super**.onCreate(savedInstanceState);  
    setContentView(R.layout.***activity\_main***);  
     
    **buttonAdd** = (Button) findViewById(R.id.***buttonAdd***);  
    **buttonAdd**.setOnClickListener(**this**);  
     
    **buttonRead** = (Button) findViewById(R.id.***buttonRead***);  
    **buttonRead**.setOnClickListener(**this**);  
     
    **buttonClear** = (Button) findViewById(R.id.***buttonClear***);  
    **buttonClear**.setOnClickListener(**this**);  
     
    **buttonUpdate** = (Button) findViewById(R.id.***buttonUpdate***);  
    **buttonUpdate**.setOnClickListener(**this**);  
     
    **buttonDelete** = (Button) findViewById(R.id.***buttonDelete***);  
    **buttonDelete**.setOnClickListener(**this**);  
     
    **etID** = (EditText) findViewById(R.id.***etID***);  
    **etName** = (EditText) findViewById(R.id.***etName***);  
    **etEmail** = (EditText) findViewById(R.id.***etEmail***);  
     
    **dbHelper** = **new** DBHelper(**this**);  
    }
3. Создадим 1 свой обработчик на все кнопки:

@Override  
**public void** onClick(View v)  
{  
 String ID = **etID**.getText().toString();  
 String name = **etName**.getText().toString();  
 String email = **etEmail**.getText().toString();  
  
 SQLiteDatabase database = **dbHelper**.getWritableDatabase();  
 ContentValues contentValues = **new** ContentValues(); *// класс для добавления новых строк в таблицу* **switch** (v.getId())  
 {  
 **case** R.id.***buttonAdd***:  
 contentValues.put(DBHelper.***KEY\_NAME***, name);  
 contentValues.put(DBHelper.***KEY\_MAIL***, email);  
 database.insert(DBHelper.***TABLE\_PERSONS***, **null**, contentValues);  
 **break**;  
  
 **case** R.id.***buttonRead***:  
 Cursor cursor = database.query(DBHelper.***TABLE\_PERSONS***, **null**, **null**, **null**,  
 **null**, **null**, **null**); *// все поля без сортировки и группировки* **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(); *// освобождение памяти* **break**;  
  
 **case** R.id.***buttonClear***:  
 database.delete(DBHelper.***TABLE\_PERSONS***, **null**, **null**);  
 **break**;  
  
 **case** R.id.***buttonDelete***:  
 **if** (ID.equalsIgnoreCase(**""**))  
 {  
 **break**;  
 }  
 **int** delCount = database.delete(DBHelper.***TABLE\_PERSONS***, DBHelper.***KEY\_ID*** + **"= "** + ID, **null**);  
 Log.*d*(**"mLog"**, **"Удалено строк = "** + delCount);  
  
 **case** R.id.***buttonUpdate***:  
 **if** (ID.equalsIgnoreCase(**""**))  
 {  
 **break**;  
 }  
 contentValues.put(DBHelper.***KEY\_MAIL***, email);  
 contentValues.put(DBHelper.***KEY\_NAME***, name);  
 **int** updCount = database.update(DBHelper.***TABLE\_PERSONS***, contentValues, DBHelper.***KEY\_ID*** + **"= ?"**, **new** String[] {ID});  
 Log.*d*(**"mLog"**, **"Обновлено строк = "** + updCount);  
 }  
 **dbHelper**.close(); *// закрываем соединение с БД*}

1. Найти созданную вами базу данных через путь: View–>Tool Windows–>Device File Explorer–>…..databases–>имя\_БД;
2. Проверить работоспособность всех кнопок через окно «Logcat»;
3. Составить отчёт.

\* *поля EditText (Name, E-mail) можно изменить на свои.*