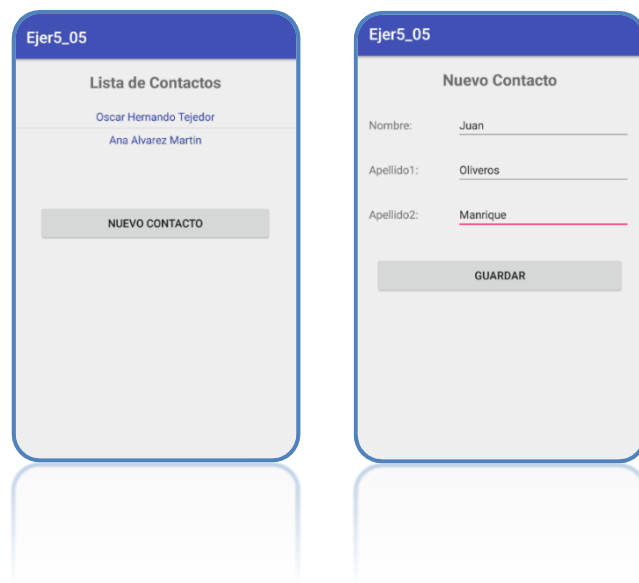


UNIDAD 5: Base de datos (Método 1)

Para ver como funciona el primer método de utilización de bases de datos SQLite, vamos a crear una aplicación que nos permita mantener una lista de contactos muy básica en la que sólo almacenaremos el nombre y los dos apellidos de cada contacto.

La aplicación estará compuesta de dos actividades:

- **MainActivity**: mostrará una lista con los contactos actuales y un botón para añadir un nuevo contacto que lanzará la actividad **AddActivity**.
- **AddActivity**: mostrará un formulario para introducir los datos del contacto y un botón "Guardar" que recogerá los datos del formulario y los almacenará como nuevo contacto en la BD.



CREACIÓN DE LA INTERFAZ

La interfaz de nuestra aplicación estará formada por un *layout* para cada actividad (activity_main.xml y activity_add.xml), y un tercer *layout* (list_item.xml) para los elementos de la lista:

activity_main.xml

Esta pantalla mostrará un título "Lista de contactos", un *ListView* (listaContactos) que es donde se mostrarán los contactos existentes en la BD, y un botón (botNuevo) que lanzará la actividad *AddActivity* para añadir un nuevo contacto a la BD.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context="com.example.oscar.exbasedatos01.MainActivity"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="Lista de Contactos"
        android:textSize="20sp"
        android:textStyle="bold"
        android:layout_margin="16dp"/>

    <ListView
        android:id="@+id/listaContactos"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="60dp" />

    <Button
        android:id="@+id/botNuevo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Nuevo Contacto"
        android:layout_gravity="center"
        android:layout_marginVertical="8dp"
        android:layout_marginHorizontal="32dp"/>

</LinearLayout>
```

activity_add.xml

Esta pantalla mostrará un formulario para recoger los datos del contacto y un botón (botGuardar) para guardar los datos en la BD.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="Nuevo Contacto"
        android:textSize="20sp"
        android:textStyle="bold"
        android:layout_margin="16dp"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_marginHorizontal="16dp"
        android:layout_marginVertical="8dp">
        <TextView
            android:id="@+id/txtNombre"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Nombre: "
            android:textSize="14sp"
            android:layout_weight="2"/>
        <EditText
            android:id="@+id/editNombre"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Nombre"
            android:textSize="14sp"
            android:layout_weight="1"/>
    </LinearLayout>
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginHorizontal="16dp"
    android:layout_marginVertical="8dp">
    <TextView
        android:id="@+id/txtApellido1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Apellido1: "
        android:textSize="14sp"
        android:layout_weight="2"/>
    <EditText
        android:id="@+id/editApellido1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Primer apellido"
        android:textSize="14sp"
        android:layout_weight="1"/>
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginHorizontal="16dp"
    android:layout_marginVertical="8dp">
    <TextView
        android:id="@+id/txtApellido2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Apellido2: "
        android:textSize="14sp"
        android:layout_weight="2"/>
    <EditText
        android:id="@+id/editApellido2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Segundo apellido"
        android:textSize="14sp"
        android:layout_weight="1"/>
</LinearLayout>

<Button
    android:id="@+id/botGuardar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Guardar"
    android:layout_gravity="center"
    android:layout_margin="24dp" />

</LinearLayout>
```

list_item.xml

Para mostrar los elementos de la lista vamos a utilizar el adaptador más simple posible, para ello cada elemento estará formado por un único *String* en el que concatenaremos el nombre y los dos apellidos del contacto, así que este *layout* estará formado por un único *TextView* para mostrar el *String* con la información del contacto.

```
<?xml version="1.0" encoding="utf-8"?>
<TextView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/tv"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:padding="5sp"
    android:singleLine="true"
    android:textColor="@color/colorPrimaryDark" />
```

IMPLEMENTACIÓN DE LA CLASE MAINACTIVITY

Nuestra clase *MainActivity* va a contener los siguientes métodos:

- **onCreate**: este método es invocado cuando se crea la actividad.

Tras generar la vista de la actividad llamará al método *listar* que se encargará de obtener la información de la BD y de mostrarla en el *ListView* *listaContactos*.

Además, se añadirá un *listener* al botón *botNuevo* que lanzará la actividad *AddActivity* para añadir un nuevo contacto a la BD.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    // Mostramos el contenido de la BD en el ListView
    listar();

    // Añadir Contacto
    Button botNuevo = (Button)findViewById(R.id.botNuevo);
    botNuevo.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intento = new Intent(MainActivity.this, AddActivity.class);
            startActivity(intento);
        }
    });
}
```

- **onResume**: este método es invocado cuando la actividad es reiniciada tras haber sido pausada. En el caso de nuestra actividad se llamará a este método cuando volvamos de la actividad *AddActivity*, así que aquí llamaremos de nuevo al método *listar* para que se actualicen los cambios realizados en la BD.

```
@Override
protected void onRestart() {
    super.onRestart();
    listar();
}
```

- **listar**: este método se encarga de obtener la información de los contactos de nuestra BD y de mostrar esa información en el *ListView* listaContactos.

En primer lugar creamos un *ArrayList* de *String* para almacenar los datos obtenidos de la BD, un adaptador para suministrar los datos del *ArrayList* al *ListView*, y obtenemos la referencia a nuestro *ListView* listaContactos:

```
ArrayList<String> lista = new ArrayList<String>();  
ArrayAdapter<String> adaptador = null;  
ListView listaContactos = (ListView) findViewById(R.id.listaContactos);
```

A continuación, obtenemos un objeto *SQLiteDatabase* mediante una llamada al método *openOrCreateDatabase* y creamos la tabla contactos si no existe:

```
SQLiteDatabase db;  
db = openOrCreateDatabase("MisContactos", Context.MODE_PRIVATE, null);  
db.execSQL("CREATE TABLE IF NOT EXISTS misContactos  
           (Nombre VARCHAR, Apellido1 VARCHAR, Apellido2 VARCHAR);");
```

Una vez que tenemos el objeto *SQLiteDatabase* y nos hemos asegurado de que existe la tabla, ya podemos ejecutar la consulta, obteniendo un cursor para recorrer los resultados:

```
Cursor cur = db.rawQuery("SELECT * FROM MisContactos", null);
```

Si el número de elementos del cursor es 0, no tenemos ningún contacto en la BD y por tanto no hay nada que mostrar en la lista, simplemente mostramos una alerta:

```
if (cur.getCount() == 0) {  
    String mensa = "Lista vacía";  
    int duracion = Toast.LENGTH_SHORT;  
    Toast.makeText(getApplicationContext(), mensa, duracion).show();  
}
```

En caso contrario, recorreremos los registros obtenidos utilizando el cursor y los vamos almacenando en el *ArrayList*,

```
else {  
    while (cur.moveToNext()) {  
        lista.add(cur.getString(0) + " " + cur.getString(1) + " " +  
                  cur.getString(2));  
    }  
    Context context = getApplicationContext();  
    adaptador = new ArrayAdapter<String>(context, R.layout.list_item, lista);  
    listaContactos.setAdapter(adaptador);  
}
```

Por último, cerramos tanto el cursor como la conexión con la BD:

```
cur.close();  
db.close();
```

El método completo quedaría así:

```
private void listar() {  
    ArrayList<String> lista = new ArrayList<String>();  
    ArrayAdapter<String> adaptador = null;  
    ListView listaContactos = (ListView) findViewById(R.id.listaContactos);  
  
    // Abrimos o creamos la BD  
    SQLiteDatabase db;  
    db = openOrCreateDatabase("MisContactos", Context.MODE_PRIVATE, null);  
    db.execSQL("CREATE TABLE IF NOT EXISTS misContactos  
                (Nombre VARCHAR, Apellido1 VARCHAR, Apellido2 VARCHAR);");  
  
    // Obtenemos los datos de la BD  
    Cursor cur = db.rawQuery("SELECT * FROM MisContactos", null);  
    if (cur.getCount() == 0) {  
        String mensa = "Lista vacía";  
        int duracion = Toast.LENGTH_SHORT;  
        Toast.makeText(getApplicationContext(), mensa, duracion).show();  
    } else {  
        while (cur.moveToNext()) {  
            lista.add(cur.getString(0) + " " + cur.getString(1) + " "  
                    + cur.getString(2));  
        }  
        Context c = getApplicationContext();  
        adaptador = new ArrayAdapter<String>(c, R.layout.list_item, lista);  
        listaContactos.setAdapter(adaptador);  
    }  
  
    // Cerramos el cursor y la BD  
    cur.close();  
    db.close();  
}
```


IMPLEMENTACIÓN DE LA CLASE ADDACTIVITY

Nuestra clase *AddActivity* va a contar únicamente con el método *onCreate*, en el que tras generar la vista de la actividad, se añadirá un *listener* al botón botGuardar que se encargará de recoger los datos del formulario y de almacenarlos en la BD.

En primer lugar recogemos los datos de las cajas de texto:

```
EditText editNombre = (EditText) findViewById(R.id.editNombre);
EditText editApe1 = (EditText) findViewById(R.id.editApellido1);
EditText editApe2 = (EditText) findViewById(R.id.editApellido2);
String nombre = editNombre.getText().toString().trim();
String ape1 = editApe1.getText().toString().trim();
String ape2 = editApe2.getText().toString().trim();
String mensa = "";
```

Comprobamos que todas las cajas de texto se hayan rellenado correctamente en cuyo caso almacenamos los datos en la BD.

Para ello obtenemos el objeto *SQLiteDatabase* invocando el método *openOrCreateDatabase*, creamos la tabla contactos si no existe e insertamos los datos recogidos de las cajas de texto teniendo en cuenta de que estos valores deben ir entrecomillados, lo que nos obliga a cerrar el *String* y concatenar los valores.

Tras insertar los datos mostramos una alerta que nos indique que el contacto ha sido añadido y vaciamos las cajas de texto.

```
if (!nombre.equals("") && !ape1.equals("") && !ape2.equals("")) {
    // Abrimos o creamos la BD
    SQLiteDatabase db;
    db = openOrCreateDatabase("MisContactos", Context.MODE_PRIVATE, null);
    db.execSQL("CREATE TABLE IF NOT EXISTS misContactos
        (Nombre VARCHAR, Apellido1 VARCHAR, Apellido2 VARCHAR);");
    db.execSQL("INSERT INTO misContactos VALUES
        ('" + nombre + "', '" + ape1 + "', '" + ape2 + "')");
    // Alerta: contacto añadido
    mensa = "Contacto añadido";
    Toast.makeText(getApplicationContext(), mensa, Toast.LENGTH_SHORT).show();
    // Borramos los datos de pantalla
    editNombre.setText("");
    editApe1.setText("");
    editApe2.setText("");
}
```

En caso de que falten datos, simplemente mostramos una alerta que lo indique:

```
else {
    // Alerta: faltan datos
    mensa = "Faltan datos";
    Toast.makeText(getApplicationContext(), mensa, Toast.LENGTH_SHORT).show();
}
```

La clase completa quedaría así:

```
public class AddActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_add);

        // Guardar datos en la BD
        Button botGuardar = (Button)findViewById(R.id.botGuardar);
        botGuardar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                // Recogemos los datos
                EditText editNombre = (EditText)findViewById(R.id.editNombre);
                EditText editApel1 = (EditText)findViewById(R.id.editApellido1);
                EditText editApe2 = (EditText)findViewById(R.id.editApellido2);
                String nombre = editNombre.getText().toString().trim();
                String apel = editApel1.getText().toString().trim();
                String ape2 = editApe2.getText().toString().trim();
                String mensa = "";

                if (!nombre.equals("") && !apel.equals("") && !ape2.equals("")) {

                    // Abrimos o creamos la BD
                    SQLiteDatabase db;
                    db = openOrCreateDatabase("MisContactos", Context.MODE_PRIVATE, null);
                    db.execSQL("CREATE TABLE IF NOT EXISTS misContactos
                                (Nombre VARCHAR, Apellido1 VARCHAR, Apellido2 VARCHAR);");
                    db.execSQL("INSERT INTO misContactos VALUES
                                ('" + nombre + "','" + apel + "','" + ape2 + "')");

                    // Alerta: contacto añadido
                    mensa = "Contacto añadido";
                    Toast.makeText(getApplicationContext(), mensa, Toast.LENGTH_SHORT).show();

                    // Borramos los datos de pantalla
                    editNombre.setText("");
                    editApel1.setText("");
                    editApe2.setText("");

                } else {
                    // Alerta: faltan datos
                    mensa = "Faltan datos";
                    Toast.makeText(getApplicationContext(), mensa, Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}
```