

TEMA: CRUD SQLITE EN ANDROID.

INTRODUCCIÓN:

La gestión de los datos es un proceso fundamental para un técnico o ingeniero en sistemas informáticos en el desarrollo de aplicaciones informáticas, es por ello, que se vuelve de vital importancia el hecho de conocer y aplicar en la construcción de App las base de datos para el almacenamiento y manipulación de la información. Se da a conocer la base de datos SQLite incorporada en Android para la construcción de App creando un CRUD básico.

OBJETIVOS:

- Describir que es SQLite y su aplicación en la construcción de App.
- Identificar herramientas para administración de base de datos SQLite.
- Crear una App con base de datos SQLite que implemente un CRUD.
- Utilizar modal o ventajas emergentes para realizar búsquedas en BD.
- Mostrar resultados de búsquedas en ListView.
- Colocar un buscador en tiempo real en combinación con un ListView.
- Mostrar resultados de búsquedas en Spinner.
- Hacer uso de bibliotecas del Framework materialize design para personalizar controles y diseños.
- Hacer uso de la herramienta gráfica para ver el diseño e información de la base de datos SQLite.

DESARROLLO:

El desarrollo comprende la presentación de los diferentes ficheros (Diseños & clase java) para la construcción de la App. Crear el proyecto en el IDE IntelliJ IDEA o Android Studio y proceda a crear los ficheros en el orden en que se lista en este documento.

Diseño de la base de datos.

Nombre base de datos: articulos.db

Campos: codigo integer not null primary key, descripcion text, precio real.

Fichero: AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.mjgl.sis21ademo">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".DetalleArticulos"></activity>
        <activity
            android:configChanges=" screenSize|orientation"
            android:windowSoftInputMode="stateHidden"
            android:name=".ListViewArticulos" />
        <activity android:name=".ConsultaSpinner" />
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name"
            android:configChanges=" screenSize|orientation"
            android:windowSoftInputMode="stateHidden"
            android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

A continuación se presentan los diseños de la App.

Fichero: activity_main.xml



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/AppTheme.AppBarOverlay">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="#A00505"
            app:popupTheme="@style/AppTheme.PopupOverlay" />

    </com.google.android.material.appbar.AppBarLayout>
```

```

<!-- android:background="?attr/colorPrimary"-->
<include layout="@layout/content_main" />

<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/fab"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom|end"
    android:layout_margin="@dimen/fab_margin"
    android:src="@drawable/ic_search" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

Fichero: content_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#696A6B"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context=".MainActivity"
    tools:showIn="@layout/activity_main">

    <TextView
        android:id="@+id/tv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="DATOS DEL ARTÍCULO"
        android:textAlignment="center"
        android:layout_marginTop="20dp"
        android:textSize="26dp"
        android:textColor="#FFFFFF"
        android:textStyle="bold"
        android:fontFamily="serif-monospace"
        />

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout

```

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"

<ImageView
    android:layout_width="wrap_content"
    android:layout_height="167dp"
    android:src="@drawable/computadora1" />

<EditText
    android:id="@+id/et_codigo"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Ingrese el código"
    android:inputType="number"
    android:textColor="#fff"
    android:background="@drawable/estiloet2"
    android:shadowColor="#A82A2A"
    android:shadowDx="-8"
    android:shadowDy="-9"
    android:shadowRadius="5"
    android:paddingLeft="25dp"
    android:drawableStart="@drawable/ic_hola"
    android:drawablePadding="5dp"
    android:layout_marginRight="15dp"
    android:layout_marginLeft="15dp"
    />

<!--
<EditText
    android:id="@+id/et_codigo"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="12dp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="12dp"
    android:background="@drawable/estiloedittext"
    android:drawableStart="@drawable/ic_input1"
    android:drawablePadding="5dp"
    android:hint="Ingrese el código del artículo"
    android:inputType="number"
    android:paddingLeft="25dp" />
-->

<EditText
    android:id="@+id/et_descripcion"
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
    android:hint="Ingrese la descripción del artículo"
    android:layout_marginRight="12dp"
    android:layout_marginLeft="12dp"
    android:layout_marginTop="20dp"
    android:background="@drawable/estiloedittext"
    android:paddingLeft="25dp"
    android:drawableStart="@drawable/ic_descripcion"
    android:drawablePadding="5dp"
  />

<EditText
    android:id="@+id/et_precio"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Ingrese el precio del artículo"
    android:layout_marginRight="12dp"
    android:layout_marginLeft="12dp"
    android:layout_marginTop="20dp"
    android:inputType="numberDecimal"
    android:background="@drawable/estiloedittext"
    android:paddingLeft="25dp"
    android:drawableStart="@drawable/ic_precio"
    android:drawablePadding="5dp"
  />

<!--<TextView
    android:id="@+id/tv_resultado"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="resultado"
    android:layout_marginTop="25dp"
    android:textColor="#fff"
    android:textAlignment="center"
    android:textSize="20dp"
  />-->

<Button
    android:id="@+id/btn_guardar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="30dp"
    android:text="Guardar"
    android:textColor="#fff"
  />
```

```
    android:onClick="alta"
    android:background="@drawable/estiloboton"
    android:shadowColor="#2314A8"
    android:shadowDx="-3"
    android:shadowDy="0"
    android:shadowRadius="8"
    android:drawableRight="@drawable/ic_save"
    android:paddingRight="100dp"
    android:drawablePadding="-140dp"
  />

<Button
    android:id="@+id/btn_consultar1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="5dp"
    android:text="Consultar por código"
    android:textColor="#fff"
    android:onClick="consultaporcodigo"
    android:background="@drawable/estiloboton"
    android:shadowColor="#2314A8"
    android:shadowDx="-3"
    android:shadowDy="0"
    android:shadowRadius="8"
    android:drawableStart="@drawable/ic_search"
    android:paddingLeft="80dp"
    android:drawablePadding="-40dp"
  />

<Button
    android:id="@+id/btn_consultar2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="5dp"
    android:text="Consultar por descripción"
    android:textColor="#fff"
    android:onClick="consultapordescripcion"
    android:background="@drawable/estiloboton"
    android:shadowColor="#2314A8"
    android:shadowDx="-3"
    android:shadowDy="0"
    android:shadowRadius="8"
  />
```

```
<Button
    android:id="@+id(btn_eliminar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="5dp"
    android:text="Borrar"
    android:textColor="#fff"
    android:onClick="bajaporcodigo"
    android:background="@drawable/estiloboton"
    android:shadowColor="#2314A8"
    android:shadowDx="-3"
    android:shadowDy="0"
    android:shadowRadius="8"
/>

<Button
    android:id="@+id	btn_actualizar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="5dp"
    android:layout_marginBottom="20dp"
    android:text="Editar"
    android:textColor="#fff"
    android:onClick="modificacion"
    android:background="@drawable/estiloboton"
    android:shadowColor="#A82A2A"
    android:shadowDx="20"
    android:shadowDy="-9"
    android:shadowRadius="5"
/>

</LinearLayout>
</ScrollView>

</LinearLayout>
```

Fichero: ventana1.xml



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="30dp"
    android:background="#ffffff"
    >

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Busqueda por Codigo"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="25dp"
        android:textSize="20dp"
        />

    <EditText
        android:id="@+id/et_cod"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="12dp"
        android:layout_marginRight="12dp"
        android:layout_marginTop="55dp"
        android:background="@drawable/estiloedittext"
        android:drawableStart="@drawable/ic_llave"
        android:drawablePadding="5dp"
        android:hint="Ingrese el código"
        android:inputType="number"
        android:paddingLeft="25dp" />

    <Button
        android:id="@+id/btn_buscar"
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="62dp"
        android:layout_marginRight="100dp"
        android:layout_marginTop="140dp"
        android:text="Buscar"
        android:textColor="#fff"
        android:background="@drawable/estiloboton"
        android:shadowColor="#2314A8"
        android:shadowDx="-3"
        android:shadowDy="0"
        android:shadowRadius="8"
        android:drawableStart="@drawable/ic_search"
        android:paddingLeft="10dp"
        android:drawablePadding="0dp"
    />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="right"
    android:orientation="vertical">
    <TextView
        android:id="@+id/tv_close"
        android:layout_width="30dp"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="25dp"
        android:textColor="#FF0000"
        android:text="X"
        android:textAlignment="center"
        android:textSize="20dp"
        android:background="@drawable/circulo1"
        android:layout_marginRight="10dp"
    />
</LinearLayout>

</RelativeLayout>

```

Fichero: circulo1.xml

```

<?xml version="1.0" encoding="utf-8"?>
<shape android:shape="oval"
xmlns:android="http://schemas.android.com/apk/res/android">

```

```
<stroke android:color="#fa0000" android:width="2dp" />
</shape>
```

Fichero: *activity_consulta_spinner.xml*



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#C5C5C5"
    tools:context=".ConsultaSpinner">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Seleccione una opción"
        android:layout_marginTop="25dp"
        android:textAlignment="gravity"
        android:textSize="18dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:textColor="#AD0505"
        android:fontFamily="cursive"
    />

    <Spinner
        android:id="@+id/sp_options"
        android:layout_width="fill_parent"
        android:layout_height="40dp"
        android:layout_marginTop="5dp"
        android:drawSelectorOnTop="true"
        android:spinnerMode="dropdown"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:background="@drawable/estilos3spinner"
    />
```

```
    android:paddingLeft="15dp"
    />
<!--android:spinnerMode="dialog"-->

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginTop="25dp">

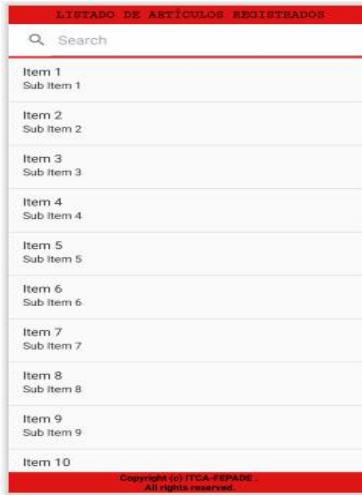
    <TextView
        android:id="@+id/tv_cod"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Código: "
        android:textColor="#AD0505"
        android:textSize="18dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
    />

    <TextView
        android:id="@+id/tv_descripcion"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Descripción: "
        android:textColor="#AD0505"
        android:textSize="18dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:layout_marginTop="25dp"
    />

    <TextView
        android:id="@+id/tv_precio"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Precio: "
        android:textColor="#AD0505"
        android:textSize="18dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:layout_marginTop="50dp"
    />

    </RelativeLayout>
</LinearLayout>
```

Fichero: activity_list_view_articulos.xml



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".ListViewArticulos">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:gravity="center"
        android:background="#E01515"
        android:orientation="vertical">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="LISTADO DE ARTÍCULOS REGISTRADOS"
            android:fontFamily="serif-monospace"
            android:textAlignment="center"
            android:textSize="16dp"
            android:textStyle="bold"
            android:textColor="#000"
            android:paddingTop="5dp"
            android:paddingBottom="5dp"
            android:textAppearance="@style/Base.TextAppearance.AppCompat.Large" />

        <SearchView
```

```

        android:id="@+id/searchView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#fff"
        android:queryHint="Search"
        android:iconifiedByDefault="false"
    />

<Space
    android:layout_width="match_parent"
    android:layout_height="3dp"/>

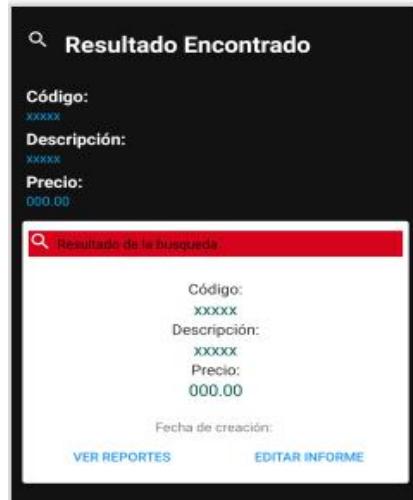
</LinearLayout>

<ListView
    android:id="@+id/listViewPersonas"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="80dp"
    android:layout_marginBottom="35dp"
/>

<TextView
    android:id="@+id/tv_destino"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_alignParentEnd="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentStart="true"
    android:layout_gravity="center"
    android:text="Copyright (c) ITCA-FEPADE . \nAll rights reserved."
    android:textColor="#000"
    android:textSize="12sp"
    android:textStyle="bold"
    android:elevation="4dp"
    android:background="#E01515"
    android:textAlignment="center"
    android:layout_marginTop="10dp"
/>
</RelativeLayout>

```

Fichero: activity_detalles_articulos.xml



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#131212"
    tools:context=".DetalleArticulos">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_marginTop="25dp"
        android:layout_marginLeft="15dp"
        >

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/ic_search"
            />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Resultado Encontrado"
            android:textSize="25sp"
            android:textStyle="bold"
            />
    
```

```
        android:layout_marginBottom="30dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:textColor="#fff"
    />

</LinearLayout>

<ScrollView
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
        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:text="Código: "
            android:layout_marginLeft="15dp"
            android:layout_marginRight="15dp"
            android:textStyle="bold"
            android:textSize="18dp"
            android:textColor="#fff"
        />

        <TextView
            android:id="@+id/tv_codigo"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="xxxxx"
            android:layout_marginBottom="5dp"
            android:layout_marginLeft="15dp"
            android:layout_marginRight="15dp"
            android:textColor="#03A9F4"
        />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Descripción: "
            android:layout_marginLeft="15dp"
            android:layout_marginRight="15dp"

```

```
    android:textStyle="bold"
    android:textSize="18dp"
    android:textColor="#fff"
  />

<TextView
    android:id="@+id/tv_descripcion"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="xxxxx"
    android:layout_marginBottom="5dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:textColor="#03A9F4"
  />

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Precio: "
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:textStyle="bold"
    android:textSize="18dp"
    android:textColor="#fff"
  />

<TextView
    android:id="@+id/tv_precio"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="000.00"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:textColor="#03A9F4"
  />

<!--<android.support.v7.widget.CardView-->
<androidx.cardview.widget.CardView
    xmlns:card_view="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    card_view:cardUseCompatPadding="true"
    card_view:cardElevation="4dp"
    card_view:cardCornerRadius="3dp"
    android:layout_margin="6dp">
```

```
<LinearLayout
    android:id="@+id/linearLayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="6dp"
    android:gravity="center">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_marginTop="5dp"
        android:layout_marginLeft="0dp"
        android:background="#D5041C"
    >

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/ic_search"/>

        <TextView
            android:id="@+id/tvInformId"
            android:textColor="#000"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Resultado de la búsqueda"
            android:padding="5dp"
        />

    </LinearLayout>

    <TextView
        android:id="@+id/tvUserName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="@android:style/TextAppearance"
        android:text="Código:"
        android:layout_marginTop="25dp"
    />

    <TextView
        android:id="@+id/tv_codigo1"
```

```
    android:textColor="@color/colorPrimaryDark"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="@style/Base.TextAppearance.AppCompat.Medium"
    android:text="xxxxx" />

<TextView
    android:id="@+id/tvUserName1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="@android:style/TextAppearance"
    android:text="Descripción:" />

<TextView
    android:id="@+id/tv_descripcion1"
    android:textColor="@color/colorPrimaryDark"
    android:layout_width="wrap_content"
    android:textAppearance="@style/Base.TextAppearance.AppCompat.Medium"
    android:layout_height="wrap_content"
    android:text="xxxxx" />

<TextView
    android:id="@+id/tvUserName2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="@android:style/TextAppearance"
    android:text="Precio:" />

<TextView
    android:id="@+id/tv_precio1"
    android:textColor="@color/colorPrimaryDark"
    android:layout_width="wrap_content"
    android:textAppearance="@style/Base.TextAppearance.AppCompat.Medium"
    android:layout_height="wrap_content"
    android:text="000.00" />

<TextView
    android:id="@+id/tv_fecha"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Fecha de creación: "
    android:layout_marginTop="20dp"
    />
```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <Button
        android:id="@+id	btnGoToReports"
        style="@style/Widget.AppCompat.Button.Borderless.Colored"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Ver reportes" />

    <Button
        style="@style/Widget.AppCompat.Button.Borderless.Colored"
        android:layout_width="0dp"
        android:layout_weight="1"
        android:layout_height="wrap_content"
        android:text="Editar informe"
        android:id="@+id	btnEditInform" />
</LinearLayout>

</LinearLayout>

</androidx.cardview.widget.CardView>

<!-- Tarjeta -->
<!--
<androidx.cardview.widget.CardView
    xmlns:card_view="http://schemas.android.com/apk/res-auto"
    card_view:cardCornerRadius="8dp"
    card_view:cardElevation="5dp"
    card_view:cardBackgroundColor="#ff7c58"
    android:layout_margin="10dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hola"
        android:padding="10dp"
        />

</androidx.cardview.widget.CardView>-->

```

```
</LinearLayout>
</ScrollView>

</LinearLayout>
```

BIBLIOTECAS UTILIZADAS:

Colocar en la siguiente ruta: Gradle Script/build.gradle (Modele: App). Dentro de la sección dependencies{ ... }

```
implementation 'com.google.android.material:material:1.0.0'
implementation 'androidx.cardview:cardview:1.0.0'
implementation 'androidx.recyclerview:recyclerview'
implementation 'com.android.support:recyclerview-v7:29.0.0'
```

A continuación se presentan las clases Java del proyecto.

Clase: Dto.java

```
package com.mjgl.sis21ademo;

import android.content.Context;
import java.io.Serializable;

//Para enviar objetos entre actividades (Como parámetro) se coloca el "implements
Serializable"
public class Dto implements Serializable {
    int codigo;
    String descripcion;
    double precio;

    public Dto() {
    }

    public Dto(int codigo, String descripcion, double precio) {
        this.codigo = codigo;
        this.descripcion = descripcion;
        this.precio = precio;
    }

    public int getCodigo() {
```

```

        return codigo;
    }

    public void setCodigo(int codigo) {
        this.codigo = codigo;
    }

    public String getDescripcion() {
        return descripcion;
    }

    public void setDescripcion(String descripcion) {
        this.descripcion = descripcion;
    }

    public double getPrecio() {
        return precio;
    }

    public void setPrecio(double precio) {
        this.precio = precio;
    }
}

```

Clase: ConexionSQLite.java

```

package com.mjgl.sis21ademo;

import android.content.ContentValues;
import android.content.Context;
import android.content.DialogInterface;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import android.widget.Toast;

import androidx.appcompat.app.AlertDialog;
//import android.app.AlertDialog;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;

public class ConexionSQLite extends SQLiteOpenHelper {

```

```

boolean estadoDelete = true;

ArrayList<String> listaArticulos; //Va a representar la información que se va a mostrar en el combo
ArrayList<Dto> articulosList; //Entidad que representa a los datos de la tabla, en esta caso la tabla Articulos
/*
public ConexionSQLite(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) {
    super(context, name, factory, version);
}
*/
public ConexionSQLite(Context context){
    super(context, "administracion.db",null, 1);
}

@Override
public void onCreate(SQLiteDatabase db) {
    db.execSQL("create table articulos(codigo integer not null primary key, descripcion text, precio real)");
}

@Override
public void onUpgrade(SQLiteDatabase db, int i, int i1) {
    db.execSQL("drop table if exists articulos");
    onCreate(db);
}

public SQLiteDatabase bd(){
    SQLiteDatabase bd = this.getWritableDatabase();
    return bd;
}

public boolean InserTradicional(Dto datos){
    boolean estado = true;
    int resultado;
    //SQLiteDatabase bd = this.getWritableDatabase();
    try {
        int codigo = datos.getCodigo();
        String descripcion = datos.getDescripcion();
        double precio = datos.getPrecio();

        //Cursor fila = this.getWritableDatabase().rawQuery("select codigo from articulos where codigo='"+codigo+"'", null);
        Cursor fila = bd.rawQuery("select codigo from articulos where codigo='"+codigo+"'", null);
    }
}

```

```

if(fila.moveToFirst()==true){
    estado = false;
} else {
    //estado = (boolean)this.getWritableDatabase().insert("datos","nombre, correo,
telefono",registro);
    //resultado = (int) this.getWritableDatabase().insert("usuarios",
"nombres,apellidos,usuario,clave,pregunta,respuesta", registro);
    String SQL = "INSERT INTO articulos \n" +
        "(codigo,descripcion,precio)\n" +
        "VALUES \n" +
        "(""+ String.valueOf(codigo) +", "+ descripcion + ", " +
String.valueOf(precio) + ");";
String.valueOf(precio) + ");";
}

bd().execSQL(SQL);
bd().close();
/*
this.getWritableDatabase().execSQL(SQL);
this.getWritableDatabase().close();
*/
estado = true;
}

catch (Exception e){
    estado = false;
    Log.e("error.",e.toString());
}

return estado;
}

public boolean insertardatos(Dto datos){
    boolean estado = true;
    int resultado;
    ContentValues registro = new ContentValues();
    try{
        //registro.put("codigo",datos.getCodigo());
        registro.put("codigo",datos.getCodigo());
        registro.put("descripcion",datos.getDescripcion());
        registro.put("precio", datos.getPrecio());
        /*adicone estas lineas para verificar sino existe un usuario con el mismo correo
        INICIO.*/
        //Cursor fila = this.getWritableDatabase().rawQuery("select codigo from articulos
where codigo='"+datos.getCodigo()+"'", null);
        Cursor fila = bd().rawQuery("select codigo from articulos where
codigo=" +datos.getCodigo()+"", null);
        if(fila.moveToFirst()==true){
            estado = false;
        }
    }
}

```

```

}else {
    /*FIN*/
    //estado = (boolean)this.getWritableDatabase().insert("datos","nombre, correo,
telefono",registro);
    //resultado = (int) this.getWritableDatabase().insert("articulos",
"codigo,descripcion,precio", registro);
    resultado = (int) bd().insert("articulos", null, registro);
    //resultado = (int) this.getWritableDatabase().insert("articulos", "codigo", registro);
    if (resultado > 0) estado = true;
    else estado = false;
}
}catch (Exception e){
    estado = false;
    Log.e("error.",e.toString());
}
return estado;
}

```

```

public boolean InsertRegister(Dto datos){
    boolean estado = true;
    int resultado;
    try{
        int codigo = datos.getCodigo();
        String descripcion = datos.getDescripcion();
        double precio = datos.getPrecio();

        //getting the current time for joining date
        Calendar cal = Calendar.getInstance();
        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        String fecha1 = sdf.format(cal.getTime());

        Cursor fila = bd().rawQuery("select codigo from articulos where
codigo='"+datos.getCodigo()+"'", null);
        if(fila.moveToFirst()==true){
            estado = false;
        }else {
            //estado = (boolean)this.getWritableDatabase().insert("datos","nombre, correo,
telefono",registro);
            //resultado = (int) this.getWritableDatabase().insert("usuarios",
"nombres,apellidos,usuario,clave,pregunta,respuesta", registro);
            String SQL = "INSERT INTO articulos \n" +
                "(codigo,descripcion,precio)\n" +
                "VALUES \n" +
                "(?, ?, ?);";
            //resultado = (int) this.getWritableDatabase().insert("usuarios",

```

```

"nombres,apellidos,usuario,clave,pregunta,respuesta", registro);
        bd().execSQL(SQL, new String[]{String.valueOf(codigo),
descripcion,String.valueOf(precio)});
        //if (resultado > 0) estado = true;
        //else estado = false;
        estado = true;
    }
}catch (Exception e){
    estado = false;
    Log.e("error.",e.toString());
}
return estado;
}

public boolean consultaCodigo(Dto datos){
    boolean estado = true;
    int resultado;
    SQLiteDatabase bd = this.getWritableDatabase();
    try {
        int codigo = datos.getCodigo();
        //Cursor fila = bd.rawQuery("select descripcion, precio from articulos where codigo=" +
+ codigo + "", null);
        //Cursor fila = bd.rawQuery("select descripcion, precio from articulos where codigo=" +
codigo, null);
        Cursor fila = bd.rawQuery("select codigo, descripcion, precio from articulos
where codigo=" + codigo, null);
        if (fila.moveToFirst()) {
            datos.setCodigo(Integer.parseInt(fila.getString(0)));
            datos.setDescripcion(fila.getString(1));
            datos.setPrecio(Double.parseDouble(fila.getString(2)));
            estado = true;
        }else {
            estado = false;
        }
        bd.close();
    }
}catch (Exception e){
    estado = false;
    Log.e("error.",e.toString());
}
return estado;
}

//Otra forma de realizar una consulta: método opcional de realizar.
public boolean consultaArticulos(Dto datos){
    boolean estado = true;
    int resultado;

```

```

//SQLiteDatabase bd = this.getWritableDatabase();
SQLiteDatabase bd = this.getReadableDatabase();
try {
    String[] parametros = {String.valueOf(datos.getCodigo())};
    String[] campos = {"codigo", "descripcion", "precio"};
    Cursor fila = bd.query("articulos",campos,"codigo=?",parametros,null,null,null);
    //fila.moveToFirst();
    if (fila.moveToFirst()) {
        datos.setCodigo(Integer.parseInt(fila.getString(0)));
        datos.setDescripcion(fila.getString(1));
        datos.setPrecio(Double.parseDouble(fila.getString(2)));
        estado = true;
    }else {
        estado = false;
    }
    fila.close();
    bd.close();
} catch (Exception e){
    estado = false;
    Log.e("error.",e.toString());
}
return estado;
}
//Fin del método opcional

```

```

public boolean consultarDescripcion(Dto datos){
    boolean estado = true;
    int resultado;
    SQLiteDatabase bd = this.getWritableDatabase();
    try {
        String descripcion = datos.getDescripcion();
        Cursor fila = bd.rawQuery("select codigo, descripcion, precio from articulos
where descripcion= '" + descripcion + "'", null);
        if (fila.moveToFirst()) {
            datos.setCodigo(Integer.parseInt(fila.getString(0)));
            datos.setDescripcion(fila.getString(1));
            datos.setPrecio(Double.parseDouble(fila.getString(2)));
            estado = true;
        }else {
            estado = false;
        }
        bd.close();
    } catch (Exception e){
        estado = false;
        Log.e("error.",e.toString());
    }
}

```

```

    }
    return estado;
}

public boolean bajaCodigo(final Context context, final Dto datos){
    //SQLiteDatabase bd = this.getWritableDatabase();
    estadoDelete = true;
    try{

        int codigo = datos.getCodigo();
        Cursor fila = bd().rawQuery("select * from articulos where codigo=" + codigo, null);
        if (fila.moveToFirst()) {
            datos.setCodigo(Integer.parseInt(fila.getString(0)));
            datos.setDescripcion(fila.getString(1));
            datos.setPrecio(Double.parseDouble(fila.getString(2)));

            AlertDialog.Builder builder = new AlertDialog.Builder(context);
            builder.setIcon(R.drawable.ic_delete);
            builder.setTitle("Warning");
            builder.setMessage("¿Está seguro de borrar el registro? \nCódigo: " +
datos.getCodigo()+"\nDescripción: "+datos.getDescripcion());
            builder.setCancelable(false);
            builder.setPositiveButton("SI", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialogInterface, int i) {
                    //String[] parametros = {String.valueOf(datos.getCodigo())};
                    int codigo = datos.getCodigo();
                    int cant = bd().delete("articulos", "codigo=" + codigo, null);
                    //bd().delete("articulos", "codigo=?",parametros);
                    if (cant > 0) {
                        estadoDelete = true;
                        Toast.makeText(context, "Registro eliminado satisfactoriamente.",
Toast.LENGTH_SHORT).show();
                    } else {
                        estadoDelete = false;
                    }
                    bd().close();
                }
            });
            builder.setNegativeButton("Cancelar", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialogInterface, int i) {

                });
            });
            AlertDialog dialog = builder.create();
            dialog.show();
    }
}

```

```

}else{
    Toast.makeText(context, "No hay resultados encontrados para la búsqueda
especificada.", Toast.LENGTH_SHORT).show();
}

}catch (Exception e){
    estadoDelete = false;
    Log.e("Error.", e.toString());
}
return estadoDelete;
}

public boolean modificar(Dto datos){
    boolean estado = true;
    int resultado;
    SQLiteDatabase bd = this.getWritableDatabase();
    try{
        int codigo = datos.getCodigo();
        String descripcion = datos.getDescripcion();
        double precio = datos.getPrecio();

        //String[] parametros = {String.valueOf(datos.getCodigo())};

        ContentValues registro = new ContentValues();
        registro.put("codigo", codigo);
        registro.put("descripcion", descripcion);
        registro.put("precio", precio);
        //int cant = (int) this.getWritableDatabase().update("articulos", registro, "codigo=" +
        codigo, null);
        int cant = (int) bd.update("articulos", registro, "codigo=" + codigo, null);
        //bd.update("articulos",registro,"codigo=?",parametros);

        bd.close();
        if(cant>0) estado = true;
        else estado = false;
    }catch (Exception e){
        estado = false;
        Log.e("error.",e.toString());
    }
    return estado;
}

```

```

//Inicio del Método para crear lista de datos de la BD en el Spinner.
//public void consultaListaArticulos(){
public ArrayList<Dto> consultaListaArticulos(){
    boolean estado = false;
    //SQLiteDatabase bd = this.getWritableDatabase();
    SQLiteDatabase bd = this.getReadableDatabase();

    Dto articulos = null;           //Creamos la instancia vacia.
    articulosList = new ArrayList<Dto>();

    try{
        Cursor fila = bd.rawQuery("select * from articulos",null);
        while (fila.moveToFirst()){
            articulos = new Dto();
            articulos.setCodigo(fila.getInt(0));
            articulos.setDescripcion(fila.getString(1));
            articulos.setPrecio(fila.getDouble(2));

            articulosList.add(articulos);

            Log.i("codigo", String.valueOf(articulos.getCodigo()));
            Log.i("descripción", articulos.getDescripcion().toString());
            Log.i("precio", String.valueOf(articulos.getPrecio()));
        }
    }

    obtenerListaArticulos();

}catch (Exception e){

}

return articulosList;
}

public ArrayList<String> obtenerListaArticulos() {
    listaArticulos = new ArrayList<String>();
    //listaArticulos = new ArrayList<>();
    listaArticulos.add("Seleccione");

    for(int i=0;i<articulosList.size();i++){
        //listaArticulos.add(String.valueOf(articulosList.get(i).getCodigo()));
        listaArticulos.add(articulosList.get(i).getCodigo()+" ~
"+articulosList.get(i).getDescripcion());
    }
    //bd().close();
    return listaArticulos;
}

```

```

//Fin del Spinner.

//Inicio del Método para crear lista de datos de la BD en el ListView.

public ArrayList<String> consultaListaArticulos1(){
    boolean estado = false;
    //SQLiteDatabase bd = this.getWritableDatabase();
    SQLiteDatabase bd = this.getReadableDatabase();

    Dto articulos = null;           //Creamos la instancia vacia.
    articulosList = new ArrayList<Dto>();

    try{
        Cursor fila = bd.rawQuery("select * from articulos",null);
        while (fila.moveToNext()){
            articulos = new Dto();
            articulos.setCodigo(fila.getInt(0));
            articulos.setDescripcion(fila.getString(1));
            articulos.setPrecio(fila.getDouble(2));

            articulosList.add(articulos);
        }
    }

    listaArticulos = new ArrayList<String>();
    //listaArticulos = new ArrayList<>();
    //listaArticulos.add("Seleccione");

    for(int i=0;i<=articulosList.size();i++){
        //listaArticulos.add(String.valueOf(articulosList.get(i).getCodigo()));
        listaArticulos.add(articulosList.get(i).getCodigo()+" ~
"+articulosList.get(i).getDescripcion());
    }
    //bd().close();
    //return listaArticulos;

} catch (Exception e){

}

//return articulosList;
return listaArticulos;
}
}

```

Clase: MainActivity.java

```
package com.mjgl.sis21ademo;

import android.content.ContentValues;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

import android.view.KeyEvent;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    private EditText et_codigo, et_descripcion, et_precio;
    private Button btn_guardar, btn_consultar1, btn_consultar2, btn_eliminar,
btn_actualizar;
    private TextView tv_resultado;

    boolean inputEt=false;
    boolean inputEd=false;
    boolean input1=false;
    int resultadoInsert=0;

    Modal ventanas = new Modal();
    ConexionSQLite conexion = new ConexionSQLite(this);
    Dto datos = new Dto();
    AlertDialog.Builder dialogo;

    @Override
    public boolean onKeyDown(int keyCode, KeyEvent event) {
```

```

if (keyCode == KeyEvent.KEYCODE_BACK) {
    new android.app.AlertDialog.Builder(this)
        .setIcon(R.drawable.ic_close)
        .setTitle("Warning")
        .setMessage("¿Realmente desea salir?")
        .setNegativeButton(android.R.string.cancel, null)
        .setPositiveButton(android.R.string.ok, new DialogInterface.OnClickListener()
    //un listener que al pulsar, cierre la aplicacion
    @Override
    public void onClick(DialogInterface dialog, int which) {
        /*Intent intent = new Intent(DashboardLuces.this, luces_control_sms.class);
        startActivity(intent);*/
        //MainActivity.this.finishAffinity();
        //finish();
        finishAffinity();
    }
})
.show();

// Si el listener devuelve true, significa que el evento esta procesado, y nadie debe hacer nada mas
return true;
}
//para las demas cosas, se reenvia el evento al listener habitual
return super.onKeyDown(keyCode, event);
}

```

```

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

    Toolbar toolbar = findViewById(R.id.toolbar);
    toolbar.setNavigationIcon(getResources().getDrawable(R.drawable.ic_back));
    toolbar.setTitleTextColor(getResources().getColor(R.color.mycolor1));
    toolbar.setTitleMargin(0, 0, 0, 0);
    toolbar.setSubtitle("CRUD SQLite-2019");
    toolbar.setSubtitleTextColor(getResources().getColor(R.color.mycolor));
    toolbar.setTitle("Prof. Gámez");
    setSupportActionBar(toolbar);

    //y esto para pantalla completa (oculta incluso la barra de estado)
    getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
    WindowManager.LayoutParams.FLAG_FULLSCREEN);

    toolbar.setNavigationOnClickListener(new View.OnClickListener() {
        @Override

```

```

public void onClick(View view) {
    confirmacion();
}
});

FloatingActionButton fab = findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        // Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH_LONG)
        //       .setAction("Action", null).show();
        ventanas.Search(MainActivity.this);
    }
});

et_codigo = (EditText) findViewById(R.id.et_codigo);
et_descripcion = (EditText) findViewById(R.id.et_descripcion);
et_precio = (EditText) findViewById(R.id.et_precio);
btn_guardar = (Button) findViewById(R.id.btn_guardar);
btn_consultar1 = (Button) findViewById(R.id.btn_consultar1);
btn_consultar2 = (Button) findViewById(R.id.btn_consultar2);
btn_eliminar = (Button) findViewById(R.id.btn_eliminar);
btn_actualizar = (Button) findViewById(R.id.btn_actualizar);
//tv_resultado = (TextView) findViewById(R.id.tv_resultado);

String senal = "";
String codigo = "";
String descripcion = "";
String precio = "";

try {
    Intent intent = getIntent();
    Bundle bundle = intent.getExtras();
    if (bundle != null) {
        codigo = bundle.getString("codigo");
        senal = bundle.getString("senal");
        descripcion = bundle.getString("descripcion");
        precio = bundle.getString("precio");
        if (senal.equals("1")) {
            et_codigo.setText(codigo);
            et_descripcion.setText(descripcion);
            et_precio.setText(precio);
            //finish();
        }
    }
} catch (Exception e){

```

```

        }

    }

private void confirmacion(){
    String mensaje = "¿Realmente desea salir?";
    dialogo = new AlertDialog.Builder(MainActivity.this);
    dialogo.setIcon(R.drawable.ic_close);
    dialogo.setTitle("Warning");
    dialogo.setMessage(mensaje);
    dialogo.setCancelable(false);
    dialogo.setPositiveButton("Aceptar", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialogo, int id) {
            /*Intent intent = new Intent(DashboardLuces.this, luces_control_sms.class);
            startActivity(intent);*/
            //DashboardLuces.this.finishAffinity();
            MainActivity.this.finish();
        }
    });
    dialogo.setNegativeButton("Cancelar", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialogo, int id) {
            //Toast.makeText(getApplicationContext(), "Operación Cancelada.",
            Toast.LENGTH_LONG).show();
        }
    });
    dialogo.show();
}

}

```

```

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}

```

```

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_limpiar) {

```

```

et_codigo.setText(null);
et_descripcion.setText(null);
et_precio.setText(null);
return true;
}else if(id == R.id.action_listaArticulos){
Intent spinnerActivity = new Intent(MainActivity.this, ConsultaSpinner.class);
startActivity(spinnerActivity);
return true;
}else if(id == R.id.action_listaArticulos1){
Intent listViewActivity = new Intent(MainActivity.this, ListViewArticulos.class);
startActivity(listViewActivity);
return true;
}
}

return super.onOptionsItemSelected(item);
}

```

```

public void alta(View v) {
if(et_codigo.getText().toString().length()==0){
et_codigo.setError("Campo obligatorio");
inputEt = false;
}else {
inputEt=true;
}
if(et_descripcion.getText().toString().length()==0){
et_descripcion.setError("Campo obligatorio");
inputEd = false;
}else {
inputEd=true;
}
if(et_precio.getText().toString().length()==0){
et_precio.setError("Campo obligatorio");
input1 = false;
}else {
input1=true;
}
if (inputEt && inputEd && input1){

try {
datos.setCodigo(Integer.parseInt(et_codigo.getText().toString()));
datos.setDescripcion(et_descripcion.getText().toString());
datos.setPrecio(Double.parseDouble(et_precio.getText().toString()));
//if(conexion.insertardatos(datos)){
//if(conexion.InsertRegister(datos)){
if(conexion.InserTradicional(datos)){

```

```

        Toast.makeText(this, "Registro agregado satisfactoriamente!",
Toast.LENGTH_SHORT).show();
        limpiarDatos();
    }else{
        Toast.makeText(getApplicationContext(), "Error. Ya existe un registro\n" +
            " Código: "+et_codigo.getText().toString(),Toast.LENGTH_LONG).show();
        limpiarDatos();
    }

}catch (Exception e){
    Toast.makeText(this, "ERROR. Ya existe.", Toast.LENGTH_SHORT).show();
}
}

public void mensaje (String mensaje){
    Toast.makeText(this, ""+mensaje, Toast.LENGTH_SHORT).show();
}

public void limpiarDatos(){
    et_codigo.setText(null);
    et_descripcion.setText(null);
    et_precio.setText(null);
    et_codigo.requestFocus();
}

public void consultaporcodigo(View v) {
    if(et_codigo.getText().toString().length()==0){
        et_codigo.setError("Campo obligatorio");
        inputEt = false;
    }else {
        inputEt=true;
    }
    if(inputEt){
        String codigo = et_codigo.getText().toString();
        datos.setCodigo(Integer.parseInt(codigo));
        //if(conexion.consultaCodigo(datos)){
        if(conexion.consultaArticulos(datos)){
            et_descripcion.setText(datos.getDescripcion());
            et_precio.setText(""+datos.getPrecio());
            //Toast.makeText(this, "Se encontro uno", Toast.LENGTH_SHORT).show();
        }else{
            Toast.makeText(this, "No existe un artículo con dicho código",
Toast.LENGTH_SHORT).show();
            limpiarDatos();
        }
    }
}

```

```

}else{
    Toast.makeText(this, "Ingrese el código del artículo a buscar.",
Toast.LENGTH_SHORT).show();
}
}

public void consultapordescripcion(View v) {
    if(et_descripcion.getText().toString().length()==0){
        et_descripcion.setError("Campo obligatorio");
        inputEd = false;
    }else {
        inputEd=true;
    }
    if(inputEd){

        String descripcion = et_descripcion.getText().toString();
        datos.setDescripcion(descripcion);
        if(conexion.consultarDescripcion(datos)){
            et_codigo.setText(""+datos.getCodigo());
            et_descripcion.setText(datos.getDescripcion());
            et_precio.setText(""+datos.getPrecio());
            //Toast.makeText(this, "Se encontró uno", Toast.LENGTH_SHORT).show();
        }else{
            Toast.makeText(this, "No existe un artículo con dicha descripción",
Toast.LENGTH_SHORT).show();
            limpiarDatos();
        }
    }else{
        Toast.makeText(this, "Ingrese la descripción del artículo a buscar.",
Toast.LENGTH_SHORT).show();
    }
}

public void bajaporcodigo(View v) {

    if(et_codigo.getText().toString().length()==0){
        et_codigo.setError("campo obligatorio");
        inputEt = false;
    }else {
        inputEt=true;
    }
}

```

```

if(inputEt){
    String cod = et_codigo.getText().toString();
    datos.setCodigo(Integer.parseInt(cod));
    if(conexion.bajaCodigo(MainActivity.this,datos)){
        //Toast.makeText(this, "Registro eliminado satisfactoriamente.", Toast.LENGTH_SHORT).show();
        limpiarDatos();
    }else{
        Toast.makeText(this, "No existe un artículo con dicho código.", Toast.LENGTH_SHORT).show();
        limpiarDatos();
    }
}
}

```

```

public void modificacion(View v) {

    if(et_codigo.getText().toString().length()==0){
        et_codigo.setError("campo obligatorio");
        inputEt = false;
    }else {
        inputEt=true;
    }

    if(inputEt) {

        String cod = et_codigo.getText().toString();
        String descripcion = et_descripcion.getText().toString();
        double precio = Double.parseDouble(et_precio.getText().toString());

        datos.setCodigo(Integer.parseInt(cod));
        datos.setDescripcion(descripcion);
        datos.setPrecio(precio);
        if(conexion.modificar(datos)){
            Toast.makeText(this, "Registro Modificado Correctamente.", Toast.LENGTH_SHORT).show();
        }else{
            Toast.makeText(this, "No se han encontrado resultados para la búsqueda especificada.", Toast.LENGTH_SHORT).show();
        }
    }
}
}

```

Clase: ConsultaSpinner.java

```
package com.mjgl.sis21ademo;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.TextView;

import java.util.ArrayList;

public class ConsultaSpinner extends AppCompatActivity {
    private Spinner sp_options;
    private TextView tv_cod, tv_descripcion, tv_precio;

    //ArrayList<String> listaArticulos;
    //ArrayList<Dto> articulosList;

    ConexionSQLite conexion = new ConexionSQLite(this);
    Dto datos = new Dto();

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

        sp_options = (Spinner)findViewById(R.id.sp_options);
        tv_cod = (TextView)findViewById(R.id.tv_cod);
        tv_descripcion = (TextView)findViewById(R.id.tv_descripcion);
        tv_precio = (TextView)findViewById(R.id.tv_precio);

        //conexion.obtenerListaArticulos();
        conexion.consultaListaArticulos();

        //ArrayAdapter<CharSequence> adaptador = new ArrayAdapter(this,
        android.R.layout.simple_spinner_item, listaArticulos);
        ArrayAdapter<CharSequence> adaptador = new ArrayAdapter(this,
        android.R.layout.simple_spinner_item, conexion.obtenerListaArticulos());
        sp_options.setAdapter(adaptador);

        sp_options.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> adapterView, View view, int position,
```

```
long l) {  
    if (position!=0){  
        tv_cod.setText("Código: "+conexion.consultaListaArticulos().get(position-1).getCodigo());  
        tv_descripcion.setText("Descripción:  
"+conexion.consultaListaArticulos().get(position-1).getDescripcion());  
        tv_precio.setText("Precio:  
"+String.valueOf(conexion.consultaListaArticulos().get(position-1).getPrecio()));  
    }else{  
        tv_cod.setText("Código: ");  
        tv_descripcion.setText("Descripción: ");  
        tv_precio.setText("Precio: ");  
    }  
}  
  
@Override  
public void onNothingSelected(AdapterView<?> adapterView) {  
}  
}  
});  
}  
}
```

Clase: ListViewArticulos.java

```
package com.mjgl.sis21ademo;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.SearchView;
import android.widget.Toast;

import java.util.ArrayList;

public class ListViewArticulos extends AppCompatActivity {

    ListView listViewPersonas;
    ArrayAdapter adaptador;
    SearchView searchView;
    ListView listView;
    ArrayList<String> list;
    ArrayAdapter adapter;
```

```

String[] version =
{"Aestro","Blender","CupCake","Donut","Eclair","Froyo","GingerBread","HoneyComb",
"IceCream Sandwich",
 "Jelly Bean","Kitkat","Lollipop","Marshmallow","Nought","Oreo"};

ConexionSQLite conexion = new ConexionSQLite(this);
Dto datos = new Dto();

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

    listViewPersonas= (ListView) findViewById(R.id.listViewPersonas);
    searchView = (SearchView) findViewById(R.id.searchView);
    //searchView = findViewById(R.id.searchView);

    //conexion.consultaListaArticulos();

    //ArrayAdapter<CharSequence> adaptador = new ArrayAdapter(this,
    android.R.layout.simple_spinner_item, listaArticulos);
    adaptador = new ArrayAdapter(this, android.R.layout.simple_list_item_1,
    conexion.consultaListaArticulos1());
    listViewPersonas.setAdapter(adaptador);

    /*
    list = new ArrayList<>();
    for (int i = 0;i<version.length;i++){
        list.add(version[i]);
    }

    adapter = new
    ArrayAdapter(ListViewArticulos.this,android.R.layout.simple_list_item_1,list);
    listViewPersonas.setAdapter(adapter);
    */

    searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String s) {

            return false;
        }

        @Override
        public boolean onQueryTextChange(String s) {
            //if(conexion.consultaListaArticulos1().contains(s)){

```

```

/*
if(list.contains(s)){
    adapter.getFilter().filter(s);
}
return true;
}
*/
String text = s;
adaptador.getFilter().filter(text);
return false;
}

});

listViewPersonas.setOnItemClickListener(new AdapterView.OnItemClickListener() {
@Override
public void onItemClick(AdapterView<?> parent, View view, int pos, long l) {

    String informacion="Código:
"+conexion.consultaListaArticulos().get(pos).getDescripcion()+"\n";
informacion+="Descripción:
"+conexion.consultaListaArticulos().get(pos).getDescripcion()+"\n";
informacion+="Precio: "+conexion.consultaListaArticulos().get(pos).getPrecio();

    Dto articulos = conexion.consultaListaArticulos().get(pos);
Intent intent = new Intent(ListViewArticulos.this, DetalleArticulos.class);
Bundle bundle = new Bundle();
bundle.putSerializable("articulo", articulos);
intent.putExtras(bundle);
startActivity(intent);

}
});

}
}

```

Clase: DetalleArticulos.java

```

package com.mjgl.sis21ademo;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;
import org.w3c.dom.Text;

```

```

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

public class DetalleArticulos extends AppCompatActivity {

    private TextView tv_codigo, tv_descripcion, tv_precio;
    private TextView tv_codigo1, tv_descripcion1, tv_precio1, tv_fecha;

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

        tv_codigo = (TextView)findViewById(R.id.tv_codigo);
        tv_descripcion = (TextView)findViewById(R.id.tv_descripcion);
        tv_precio = (TextView)findViewById(R.id.tv_precio);

        tv_codigo1 = (TextView)findViewById(R.id.tv_codigo1);
        tv_descripcion1 = (TextView)findViewById(R.id.tv_descripcion1);
        tv_precio1 = (TextView)findViewById(R.id.tv_precio1);
        tv_fecha = (TextView)findViewById(R.id.tv_fecha);

        Bundle objeto = getIntent().getExtras();
        Dto dto = null;
        if(objeto != null){
            dto = (Dto)objeto.getSerializable("articulo");
            tv_codigo.setText(""+dto.getCodigo());
            tv_descripcion.setText(dto.getDescripcion());
            tv_precio.setText(String.valueOf(dto.getPrecio()));

            tv_codigo1.setText(""+dto.getCodigo());
            tv_descripcion1.setText(dto.getDescripcion());
            tv_precio1.setText(String.valueOf(dto.getPrecio()));
            tv_fecha.setText("Fecha de creación: "+getTime());
        }
    }

    private String getTime() {
        SimpleDateFormat dateFormat = new SimpleDateFormat(
                "yyyy-MM-dd HH:mm:ss a", Locale.getDefault());
        Date date = new Date();
        return dateFormat.format(date);
    }
}

```

Clase: Modal.java

```
package com.mjgl.sis21ademo;
import android.app.AlertDialog;
import android.app.Dialog;
import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class Modal {
    Dialog myDialog;
    AlertDialog.Builder dialogo;
    boolean validalInput = false;
    String codigo;
    String descripcion;
    String precio;

    SQLiteDatabase bd = null;
    Dto datos = new Dto();

    public void Search(final Context context){
        myDialog = new Dialog(context);
        myDialog.setContentView(R.layout.ventana1);
        myDialog.setTitle("Search");
        myDialog.setCancelable(false);
        final ConexionSQLite conexion = new ConexionSQLite(context);
        final EditText et_cod = (EditText)myDialog.findViewById(R.id.et_cod);
        Button btn_buscar = (Button)myDialog.findViewById(R.id.btn_buscar);
        TextView tv_close = (TextView)myDialog.findViewById(R.id.tv_close);
        tv_close.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                myDialog.dismiss();
            }
        });
        btn_buscar.setOnClickListener(new View.OnClickListener() {
            @Override
```

```

public void onClick(View view) {
    //ConexionSQLite conexionSQLite = new ConexionSQLite(context,
    "administracion.db", null, 1);
    //SQLiteDatabase bd = conexionSQLite.getWritableDatabase();

    if(et_cod.getText().toString().length()==0){
        validalnput = false;
        et_cod.setError("Campo obligatorio");
    }else{
        validalnput = true;
    }

    if(validalnput) {
        //int cod = Integer.parseInt(et_cod.getText().toString());
        //datos.setCodigo(cod);
        String cod = et_cod.getText().toString();
        datos.setCodigo(Integer.parseInt(cod));
        if(conexion.consultaCodigo(datos)){

            codigo = String.valueOf(datos.getCodigo());
            descripcion = datos.getDescripcion();
            precio = String.valueOf(datos.getPrecio());
            String action;
            Intent intent = new Intent(context, MainActivity.class);
            intent.putExtra("senal", "1");
            intent.putExtra("codigo", codigo);
            intent.putExtra("descripcion", descripcion);
            intent.putExtra("precio", precio);
            context.startActivity(intent);

            myDialog.dismiss();
        }else{
            Toast.makeText(context, "No se han encontrado resultados para la
busqueda especificada.", Toast.LENGTH_SHORT).show();
        }
    }else{
        Toast.makeText(context, "No ha especificado lo que desea buscar.",
Toast.LENGTH_SHORT).show();
    }
};

myDialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT));
myDialog.show();
}
}

```

CONCLUSIONES:

- La creación de App con acceso a base de datos permite tener un mejor control de la información.
- La ventaja de la manipulación de la información con base de datos es que permite la creación de Backup/copias de seguridad en caso de perdidas de información.

Investigar:

- **Como mostrar una consulta general de la base de datos en un control recyclerview.**
- **Como crear una copia de seguridad de su base de datos SQLite desde su proyecto, colocando una opción que diga: “Crear copia bd”. Esta opción puede estar en la pantalla principal del proyecto por medio de Button o una opción en el menú.**
- **greenDAO: ORM (mapeo relacional / objeto) de Android para su base de datos SQLite**

Referencia bibliográfica:

<http://angrytools.com/android/>

<https://developer.android.com/reference/android/database/sqlite/SQLiteDatabase>

<https://greenrobot.org/greendao/>

<https://www.omgubuntu.co.uk/2019/07/scrcpy-mirror-android-to-ubuntu-linux>

<https://github.com/Genymobile/scrcpy>