

En esta clase comenzamos a integrar el uso de base de datos a las app utilizando sqlite

Main activity

```
public class MainActivity extends AppCompatActivity {

    private EditText etCod,etNom,etPre;

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

        etCod=findViewById(R.id.et1);
        etNom=findViewById(R.id.et2);
        etPre=findViewById(R.id.et3);
    }

    public void crear(View v){
        Admindb admin=new Admindb(this,"Productos", null, 1);
        SQLiteDatabase base=admin.getWritableDatabase();

        String codigo=etCod.getText().toString();
        String nombre=etNom.getText().toString();
        String precio=etPre.getText().toString();

        if(!codigo.isEmpty()&&!nombre.isEmpty()&&!precio.isEmpty()){
            ContentValues crear = new ContentValues();
            crear.put("codigo", codigo);
            crear.put("nombre", nombre);
            crear.put("precio", precio);
            base.insert("producto", null,crear);
            base.close();
            etCod.setText("");
            etNom.setText("");
            etPre.setText("");
            Toast.makeText(this, "Producto creado", Toast.LENGTH_LONG).show();
        }
        else{
            Toast.makeText(this, "Debes llenar todos los campos",
            Toast.LENGTH_LONG).show();
        }
    }

    public void buscar(View V){
        Admindb admin=new Admindb(this,"Productos", null, 1);
        SQLiteDatabase base=admin.getWritableDatabase();
```

```

String codigo=etCod.getText().toString();

if(!codigo.isEmpty()){
    Cursor fila= base.rawQuery("select nombre, precio from producto where
codigo="+codigo, null);
    if(fila.moveToFirst()){
        etNom.setText(fila.getString(0));
        etPre.setText(fila.getString(1));
        base.close();
    }
    else{
        Toast.makeText(this, "No existe el producto", Toast.LENGTH_LONG).show();
    }
}

public void modificar(View V){
    Adminadb admin=new Adminadb(this,"Productos", null, 1);
    SQLiteDatabase base=admin.getWritableDatabase();

    String codigo=etCod.getText().toString();
    String nombre=etNom.getText().toString();
    String precio=etPre.getText().toString();

    if(!codigo.isEmpty()&&!nombre.isEmpty()&&!precio.isEmpty()){
        ContentValues modificar = new ContentValues();
        modificar.put("codigo", codigo);
        modificar.put("nombre", nombre);
        modificar.put("precio", precio);
        base.update("producto", modificar, "codigo="+codigo, null);
        base.close();
        Toast.makeText(this, "Actualizacion exitosa", Toast.LENGTH_LONG).show();
    }
    else{
        Toast.makeText(this, "Debes completar todos los campos",
Toast.LENGTH_LONG).show();
    }
}

public void eliminar(View V){
    Adminadb admin=new Adminadb(this,"Productos", null, 1);
    SQLiteDatabase base=admin.getWritableDatabase();

    String codigo=etCod.getText().toString();

    if(!codigo.isEmpty()){
        base.delete("producto", "codigo="+codigo, null);
        base.close();
    }
}

```

```
        Toast.makeText(this, "Producto eliminado", Toast.LENGTH_LONG).show();
    }
    else{
        Toast.makeText(this, "Debes completar todos los campos",
Toast.LENGTH_LONG).show();
    }
}
}
```

pruebaBdd

Ingrese codigo

Ingrese nombre

Ingrese valor

CREAR

BUSCAR

ACTUALIZAR

ELIMINAR

Base de datos

```
public class AdminDb extends SQLiteOpenHelper {
    public AdminDb(@Nullable Context context, @Nullable String name, @Nullable
        SQLiteDatabase.CursorFactory factory, int version) {
        super(context, name, factory, version);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("create table producto(codigo int primary key, nombre varchar, precio
            int)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    }
}
```