

# Programación Orientada a Objetos

CRUD en modo texto usando listas  
dinámicas



Sesión 04

Ing. David Mamani Pari



## Misión

“Somos una comunidad universitaria de la Iglesia Adventista del Séptimo Día que **modela** personas a fin de que sean Ingenieros de Sistemas **íntegros**, **misioneros**, e **innovadores** basados en la cosmovisión bíblica-cristiana para servir a Dios y a la humanidad”

## Visión



“Ser referentes en el mundo por el **modelamiento** de Ingenieros de Sistemas **íntegros**, **misionero** e **innovadores** con un estilo de vida saludable”

Programa. EP Ingeniería de Sistemas Filial Juliaca      Semestre. 2022-2

**Leyenda.**

76 - 100 %   50 - 75 %   0 - 49 %   Sin Asistencia

Mod. resumen   Mod. detallado   Mod. acta   Buscar

Nro.	Código Alumno	Apellidos y Nombres	Sesión. 1	Sesión. 2	Sesión. 3	Sesión. 4	Sesión. 5	Sesión. 6	Sesión. 7	Sesión. 8	Sesión. 9	Sesión. 10
1	201811296	Avalos Humpiri Max Alexander	●	●	●	●	●	●	●	●	●	●
2	202010809	Burgos Pari Diego Armando	●	●	●	●	●	●	●	●	●	●
3	201911332	Condori Cotrado Gabriel Matias	●	●	●	●	●	●	●	●	●	●
4	201911747	Larico Cusi Laydy Maryori	●	●	●	●	●	●	●	●	●	●
5	202012385	Mamani Chura Juan Lenin	●	●	●	●	●	●	●	●	●	●
6	202012387	Morocco Mamani Yohel Aldahyr	●	●	●	●	●	●	●	●	●	●
7	202010800	Paricela Pongo Jack Steve	●	●	●	●	●	●	●	●	●	●
8	202014019	Puma Mestas Brandon Everick	●	●	●	●	●	●	●	●	●	●
9	202012377	Quispe Valero Jhoselyn Noemi	●	●	●	●	●	●	●	●	●	●
10	202013998	Sucasaire Mamani Jhon Erick	●	●	●	●	●	●	●	●	●	●
11	202012384	Ticona Gonzalo Jhobany Elduard	●	●	●	●	●	●	●	●	●	●
12	202012380	Velasquez Rosas Henry Jhonatan	●	●	●	●	●	●	●	●	●	●
13	202012378	Vilca Mamani Jheyson Willy	●	●	●	●	●	●	●	●	●	●

<https://youtu.be/mCE1z0TEQ2A>

Donde almacena  
los datos  
académico la  
UPeU?

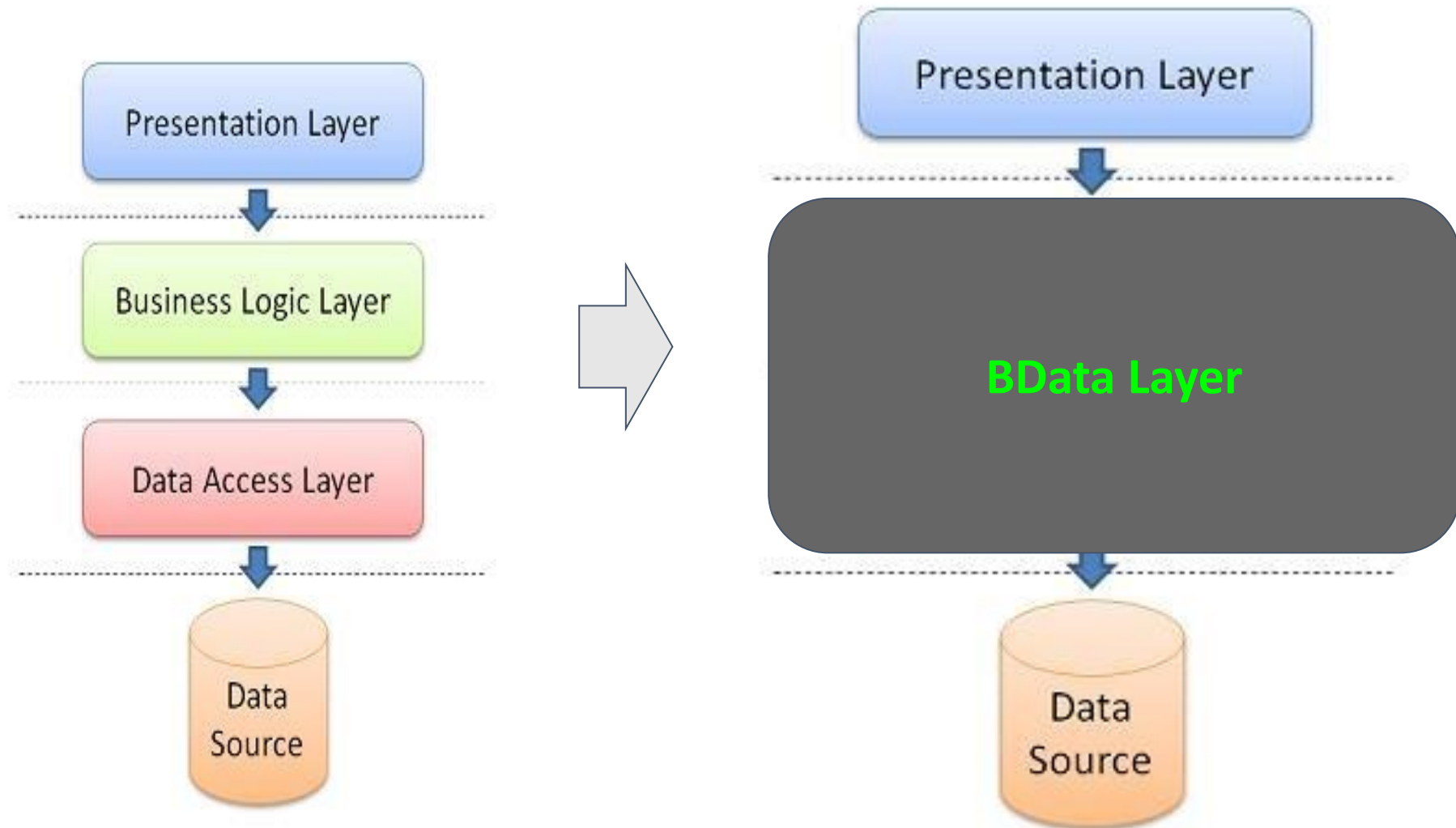
Donde almacena el  
Whatsapp mis  
conversaciones?



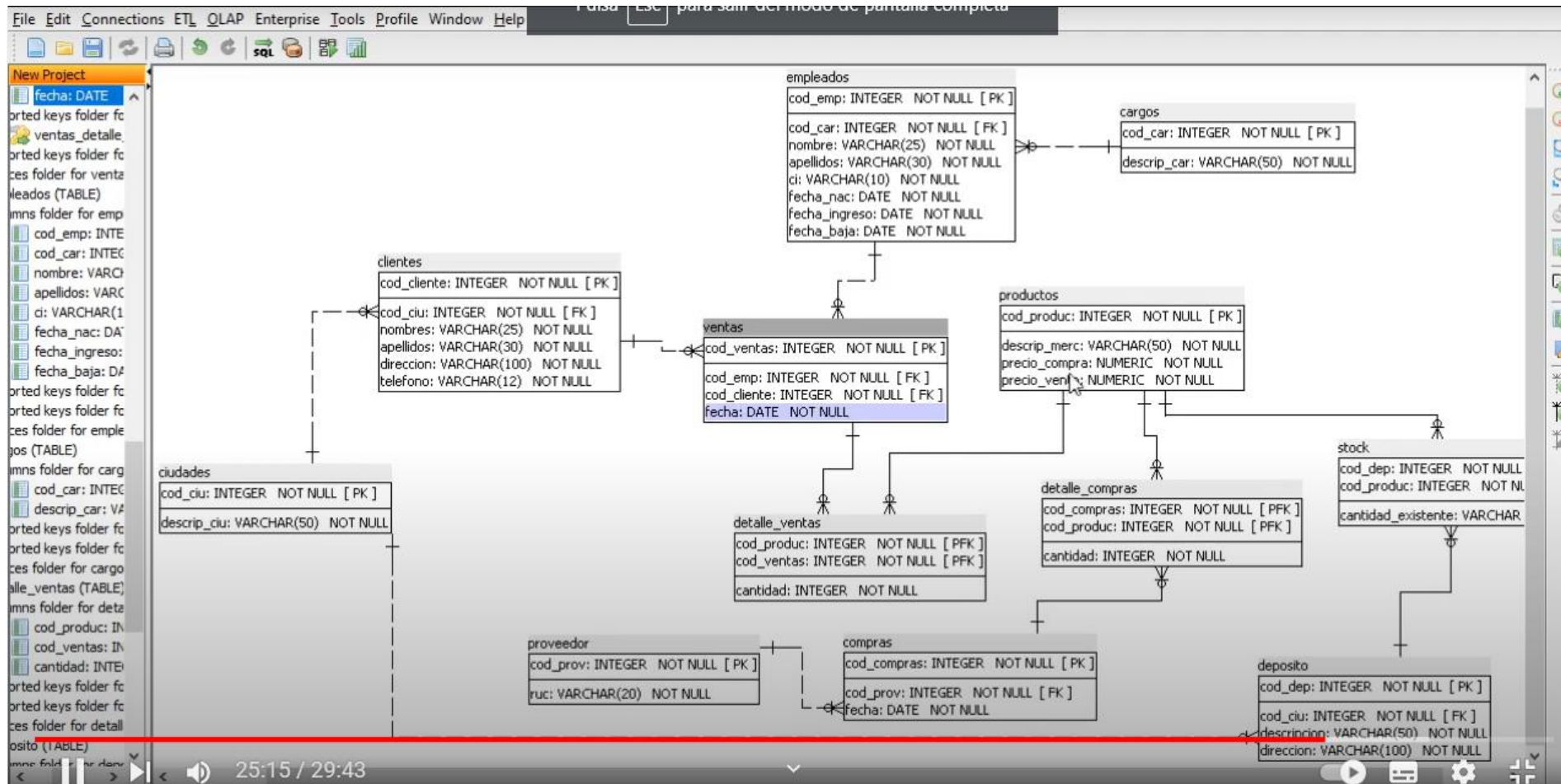
Íntegro

Sé Misionero

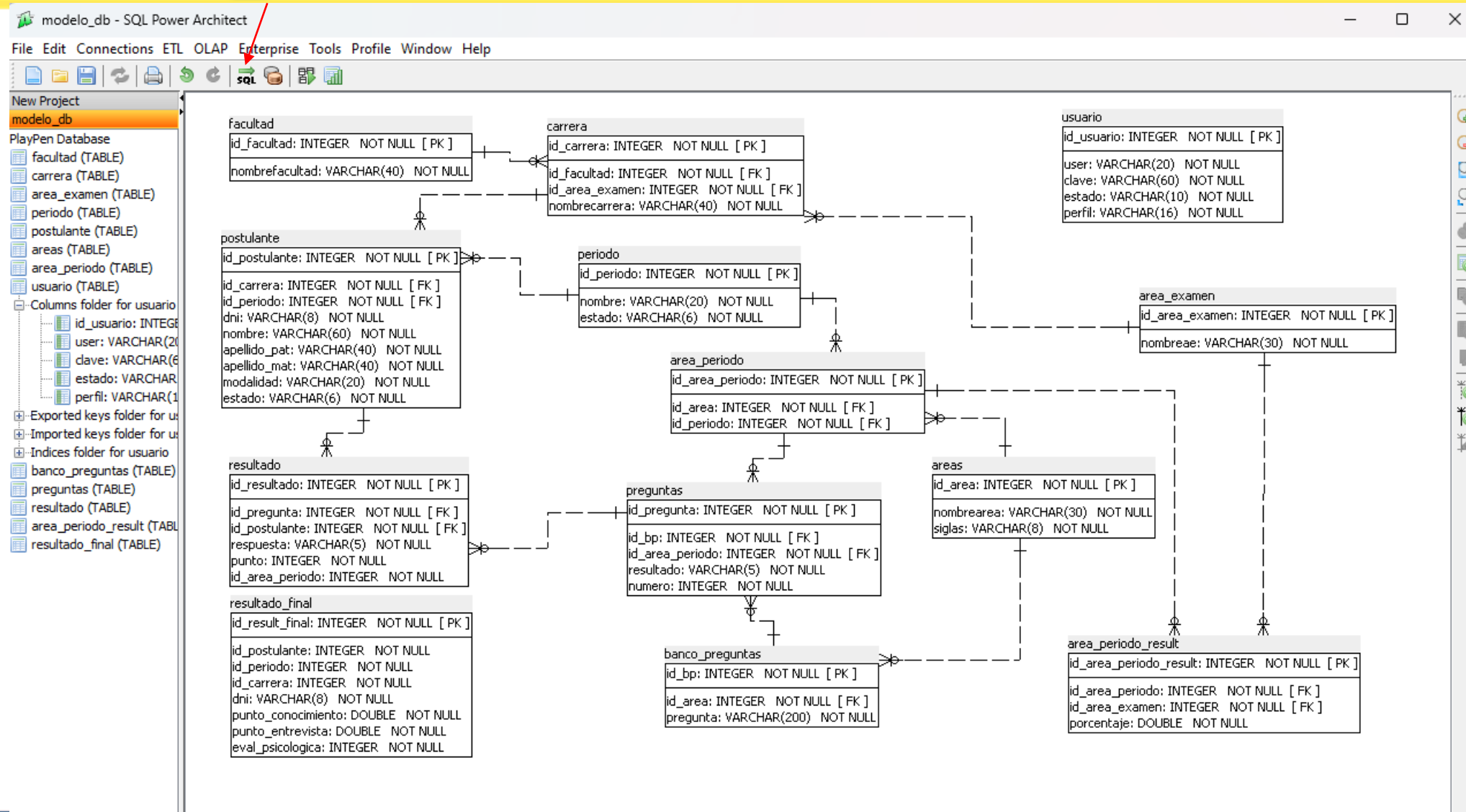
Sé Innovador

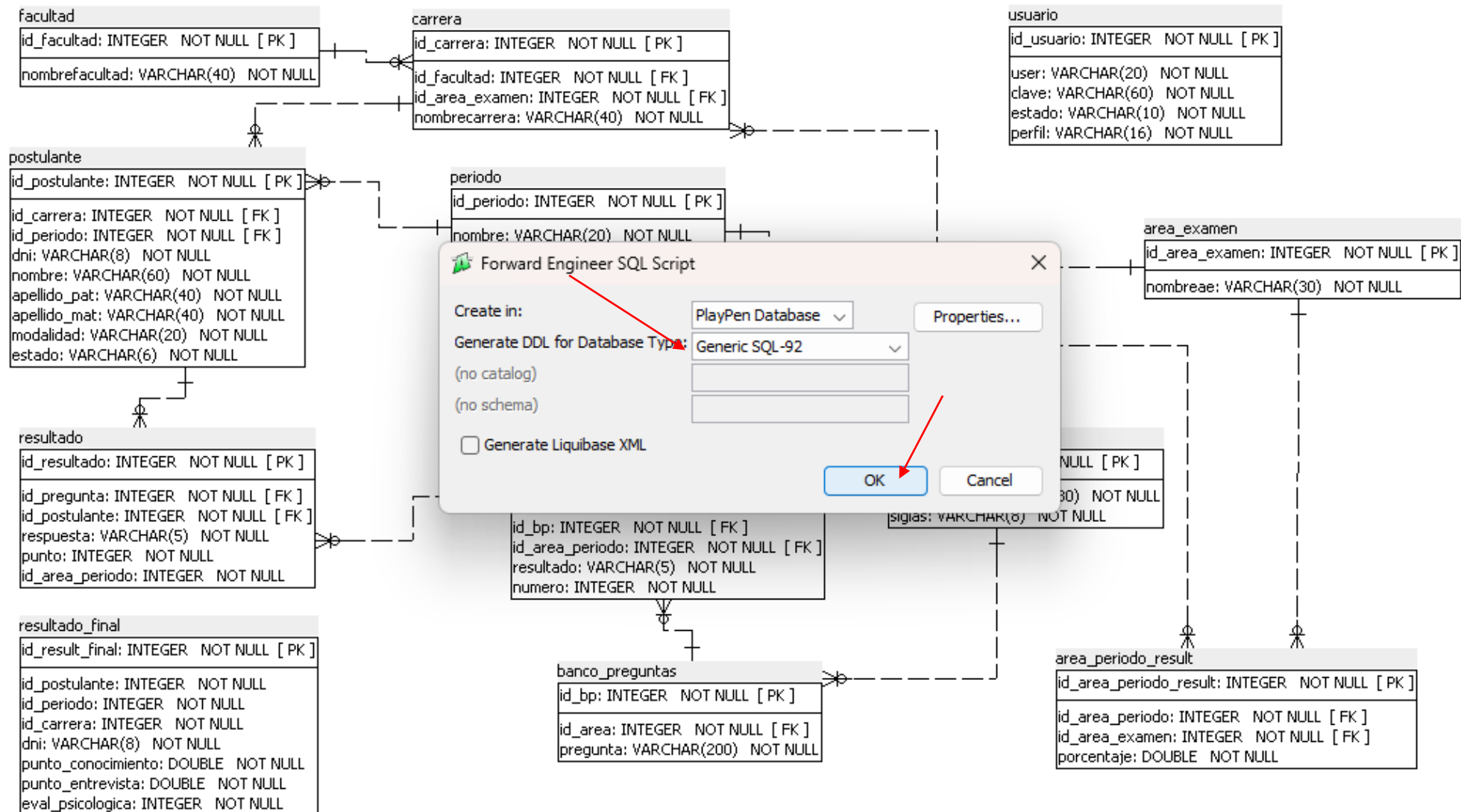


<https://www.youtube.com/watch?v=UDHVVH2tVOWE>









```

CREATE TABLE resultado (
    id_resultado INTEGER NOT NULL,
    id_pregunta INTEGER NOT NULL,
    id_postulante INTEGER NOT NULL,
    respuesta VARCHAR(5) NOT NULL,
    punto INTEGER NOT NULL,
    id_area_periodo INTEGER NOT NULL,
    CONSTRAINT resultado_pk PRIMARY KEY (id_resultado AUTOINCREMENT)
);

CREATE TABLE area_periodo (
    id_area_periodo INTEGER NOT NULL,
    id_area INTEGER NOT NULL,
    id_periodo INTEGER NOT NULL,
    CONSTRAINT area periodo pk PRIMARY KEY (id area periodo AUTOINCREMENT),
    FOREIGN KEY (id_area) REFERENCES areas (id_area) ON DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE
);

ALTER TABLE area periodo ADD CONSTRAINT areas_area_periodo_fk
FOREIGN KEY (id_area) REFERENCES areas (id_area) ON DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE;

ALTER TABLE banco_preguntas ADD CONSTRAINT areas_banco_preguntas_fk
FOREIGN KEY (id_area)
REFERENCES areas (id_area)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;
    
```

Agregar en todas las tablas



```
CREATE TABLE resultado_final (  
    id_result_final INTEGER NOT NULL,  
    id_postulante INTEGER NOT NULL,  
    id_periodo INTEGER NOT NULL,  
    id_carrera INTEGER NOT NULL,  
    dni VARCHAR(8) NOT NULL,  
    punto_conocimiento DOUBLE NOT NULL,  
    punto_entrevista DOUBLE NOT NULL,  
    eval_psicologica INTEGER NOT NULL,  
    CONSTRAINT resultado_final_pk PRIMARY KEY  
    (id_result_final AUTOINCREMENT)  
);  
  
CREATE TABLE usuario (  
    id_usuario INTEGER NOT NULL,  
    user VARCHAR(20) NOT NULL,  
    clave VARCHAR(60) NOT NULL,  
    estado VARCHAR(10) NOT NULL,  
    perfil VARCHAR(16) NOT NULL,  
    CONSTRAINT usuario_pk PRIMARY KEY  
    (id_usuario AUTOINCREMENT)  
);
```

```
CREATE TABLE areas (  
    id_area INTEGER NOT NULL,  
    nombrearea VARCHAR(30) NOT NULL,  
    siglas VARCHAR(8) NOT NULL,  
    CONSTRAINT areas_pk PRIMARY KEY (id_area  
    AUTOINCREMENT)  
);  
  
CREATE TABLE banco_preguntas (  
    id_bp INTEGER NOT NULL,  
    id_area INTEGER NOT NULL,  
    pregunta VARCHAR(200) NOT NULL,  
    CONSTRAINT banco_preguntas_pk PRIMARY KEY (id_bp  
    AUTOINCREMENT),  
    FOREIGN KEY (id_area) REFERENCES areas (id_area) ON  
    DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE  
);
```

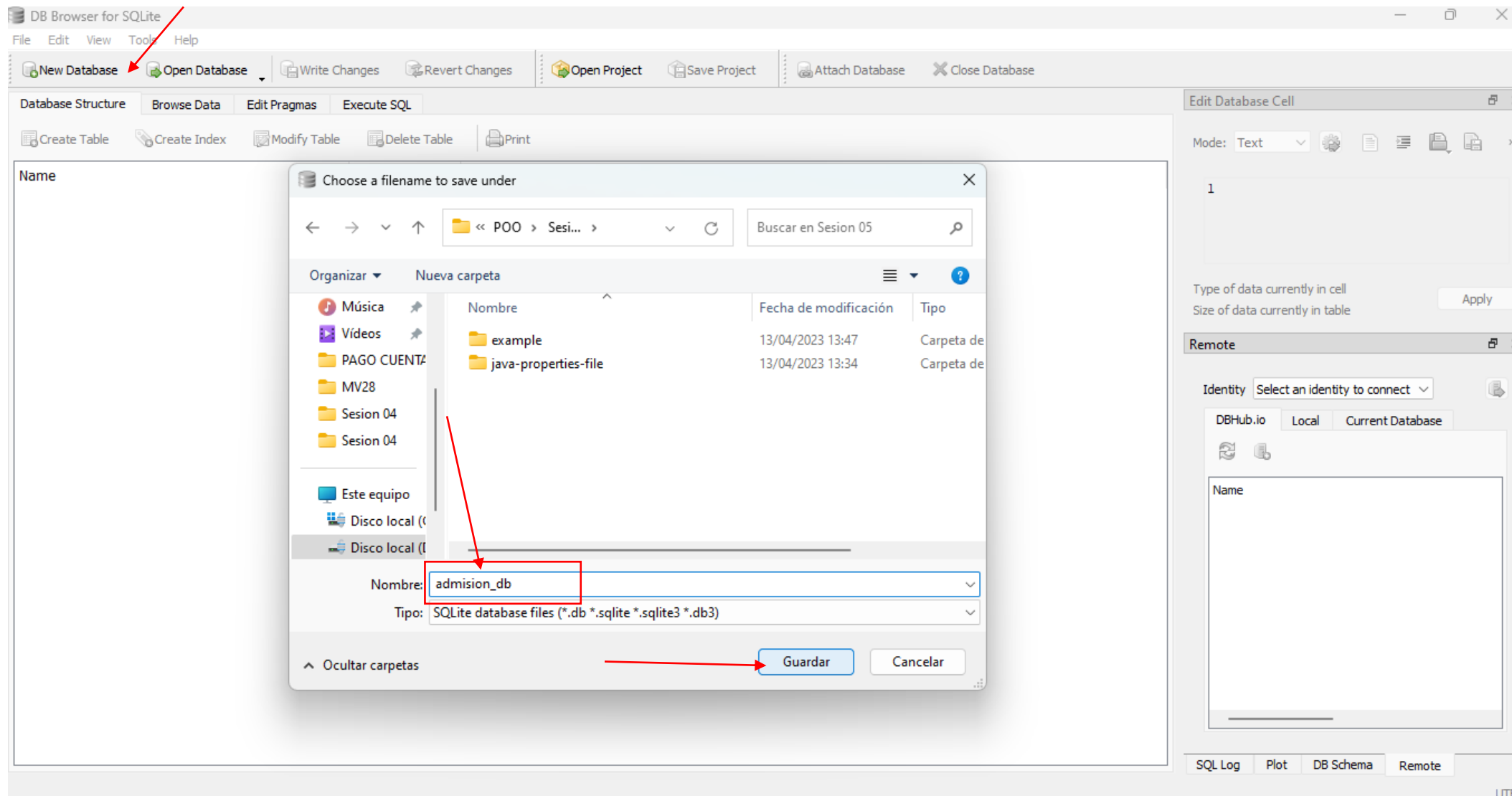
```
CREATE TABLE periodo (  
    id_periodo INTEGER NOT NULL,  
    nombre VARCHAR(20) NOT NULL,  
    estado VARCHAR(6) NOT NULL,  
    CONSTRAINT periodo_pk PRIMARY KEY (id_periodo  
AUTOINCREMENT)  
);  
  
CREATE TABLE preguntas (  
    id_pregunta INTEGER NOT NULL,  
    id_bp INTEGER NOT NULL,  
    id_area_periodo INTEGER NOT NULL,  
    resultado VARCHAR(5) NOT NULL,  
    numero INTEGER NOT NULL,  
    CONSTRAINT preguntas_pk PRIMARY KEY (id_pregunta  
AUTOINCREMENT),  
    FOREIGN KEY (id_bp) REFERENCES banco_preguntas  
(id_bp) ON DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE,  
    FOREIGN KEY (id_area_periodo) REFERENCES  
area_periodo (id_area_periodo) ON DELETE NO ACTION ON UPDATE NO  
ACTION NOT DEFERRABLE  
);
```

```
CREATE TABLE area_examen (  
    id_area_examen INTEGER NOT NULL,  
    nombreae VARCHAR(30) NOT NULL,  
    CONSTRAINT area_examen_pk PRIMARY KEY  
(id_area_examen AUTOINCREMENT)  
);  
  
CREATE TABLE area_periodo_result (  
    id_area_periodo_result INTEGER NOT NULL,  
    id_area_periodo INTEGER NOT NULL,  
    id_area_examen INTEGER NOT NULL,  
    porcentaje DOUBLE NOT NULL,  
    CONSTRAINT area_periodo_result_pk PRIMARY KEY  
(id_area_periodo_result AUTOINCREMENT),  
    FOREIGN KEY (id_area_periodo) REFERENCES  
area_periodo (id_area_periodo) ON DELETE NO ACTION ON UPDATE NO  
ACTION NOT DEFERRABLE,  
    FOREIGN KEY (id_area_examen) REFERENCES  
area_examen (id_area_examen) ON DELETE NO ACTION ON UPDATE NO  
ACTION NOT DEFERRABLE  
);
```

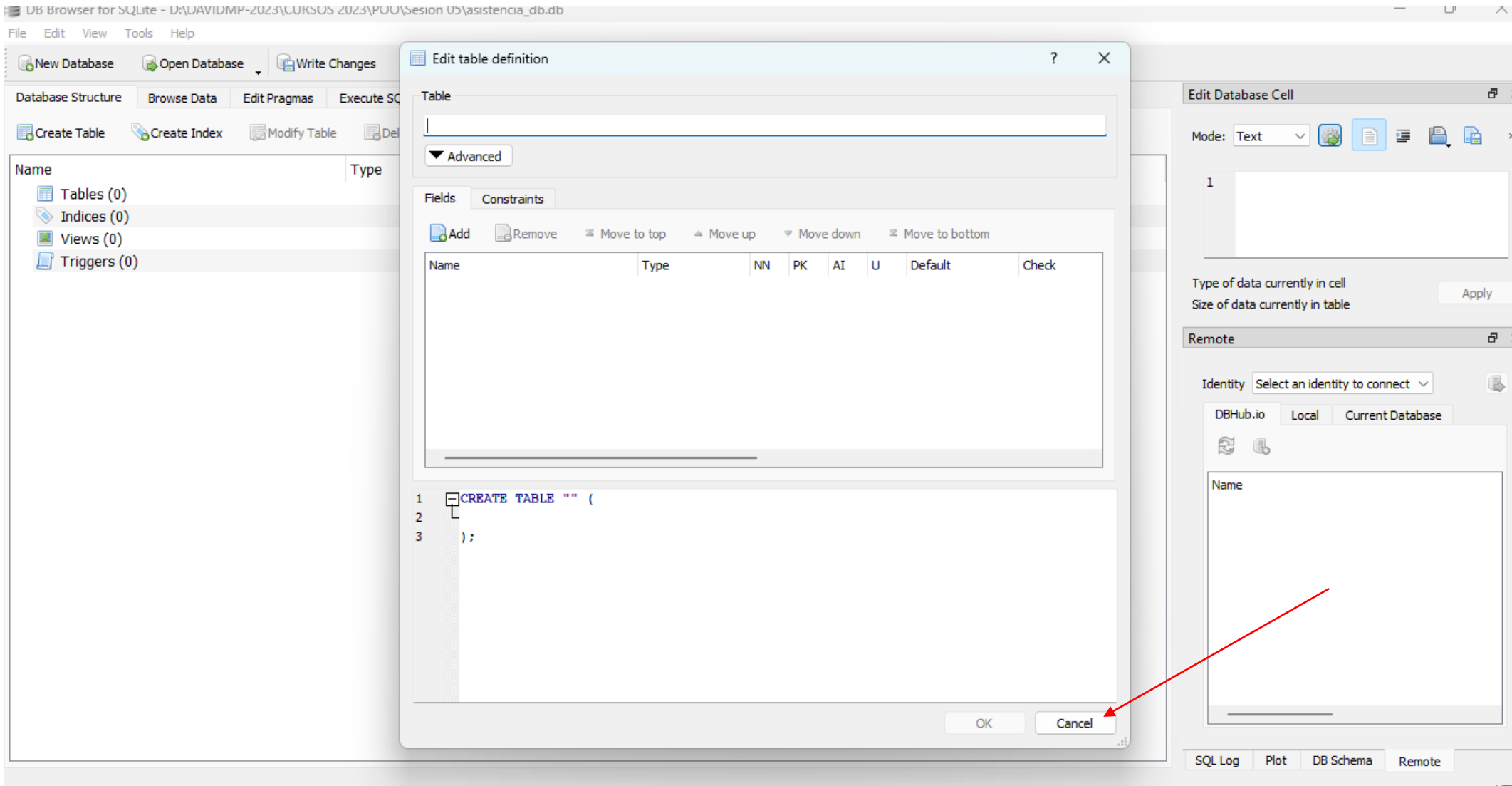
```
CREATE TABLE facultad (  
    id_facultad INTEGER NOT NULL,  
    nombrefacultad VARCHAR(40) NOT NULL,  
    CONSTRAINT facultad_pk PRIMARY KEY  
(id_facultad AUTOINCREMENT)  
);  
  
CREATE TABLE carrera (  
    id_carrera INTEGER NOT NULL,  
    id_facultad INTEGER NOT NULL,  
    id_area_examen INTEGER NOT NULL,  
    nombrecarrera VARCHAR(40) NOT NULL,  
    CONSTRAINT carrera_pk PRIMARY KEY  
(id_carrera AUTOINCREMENT),  
    FOREIGN KEY (id_area_examen) REFERENCES  
area_examen (id_area_examen) ON DELETE NO ACTION ON  
UPDATE NO ACTION NOT DEFERRABLE,  
    FOREIGN KEY (id_facultad) REFERENCES  
facultad (id_facultad) ON DELETE NO ACTION ON UPDATE NO  
ACTION NOT DEFERRABLE  
);
```

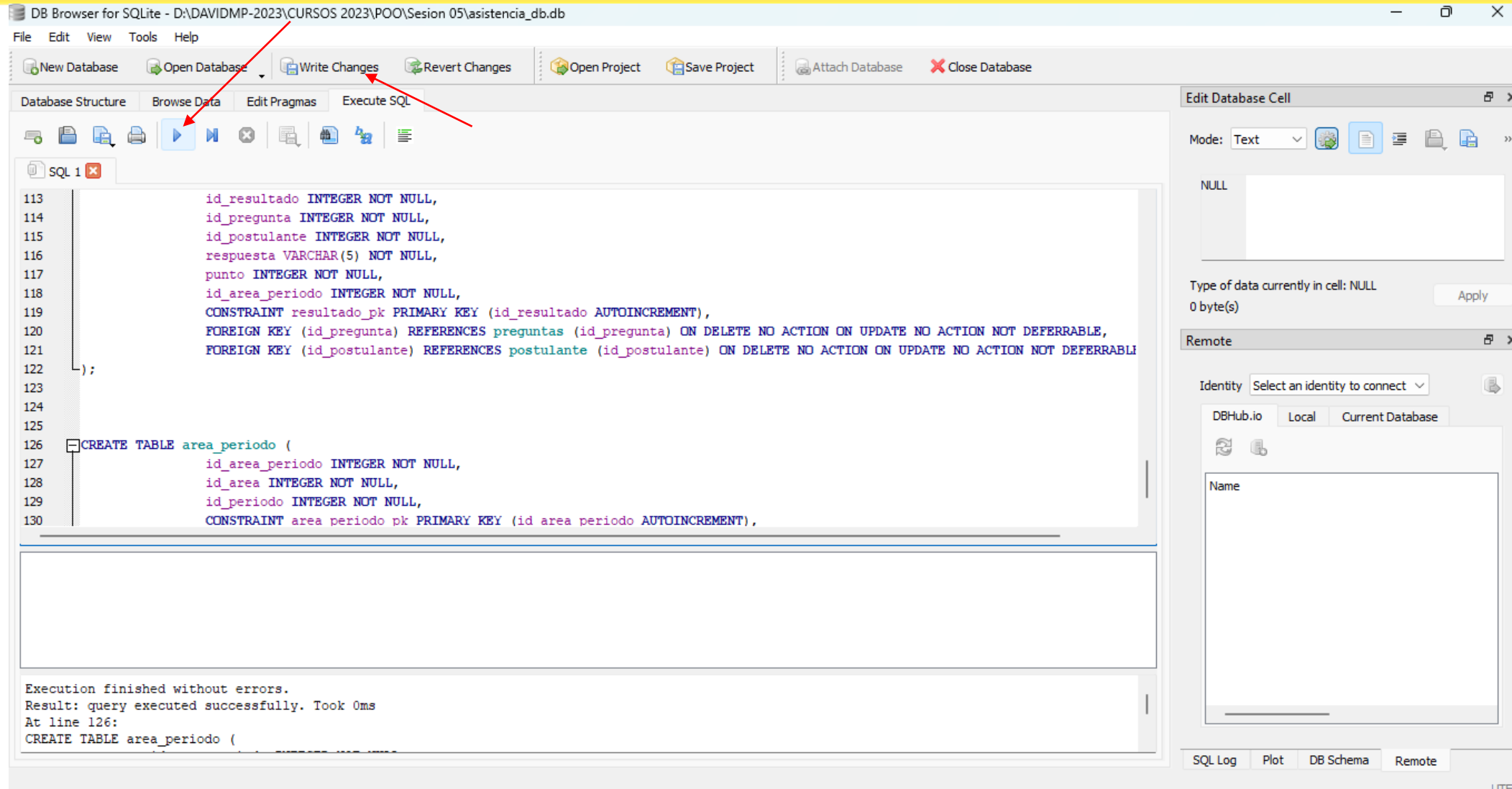
```
CREATE TABLE postulante (  
    id_postulante INTEGER NOT NULL,  
    id_carrera INTEGER NOT NULL,  
    id_periodo INTEGER NOT NULL,  
    dni VARCHAR(8) NOT NULL,  
    nombre VARCHAR(60) NOT NULL,  
    apellido_pat VARCHAR(40) NOT NULL,  
    apellido_mat VARCHAR(40) NOT NULL,  
    modalidad VARCHAR(20) NOT NULL,  
    estado VARCHAR(6) NOT NULL,  
    CONSTRAINT postulante_pk PRIMARY KEY (id_postulante AUTOINCREMENT),  
    FOREIGN KEY (id_periodo) REFERENCES periodo (id_periodo) ON DELETE  
NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE,  
    FOREIGN KEY (id_carrera) REFERENCES carrera (id_carrera) ON DELETE  
NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE  
);  
  
CREATE TABLE resultado (  
    id_resultado INTEGER NOT NULL,  
    id_pregunta INTEGER NOT NULL,  
    id_postulante INTEGER NOT NULL,  
    respuesta VARCHAR(5) NOT NULL,  
    punto INTEGER NOT NULL,  
    id_area_periodo INTEGER NOT NULL,  
    CONSTRAINT resultado_pk PRIMARY KEY (id_resultado AUTOINCREMENT),  
    FOREIGN KEY (id_pregunta) REFERENCES preguntas (id_pregunta) ON  
DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE,  
    FOREIGN KEY (id_postulante) REFERENCES postulante (id_postulante)  
ON DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE  
);
```

```
CREATE TABLE area_periodo (  
    id_area_periodo INTEGER NOT NULL,  
    id_area INTEGER NOT NULL,  
    id_periodo INTEGER NOT NULL,  
    CONSTRAINT area_periodo_pk PRIMARY KEY (id_area_periodo AUTOINCREMENT),  
    FOREIGN KEY (id_area) REFERENCES areas (id_area) ON DELETE NO ACTION ON UPDATE NO ACTION  
NOT DEFERRABLE,  
    FOREIGN KEY (id_periodo) REFERENCES periodo (id_periodo) ON DELETE NO ACTION ON UPDATE NO  
ACTION NOT DEFERRABLE  
);
```









The screenshot shows the DB Browser for SQLite application window. The title bar indicates the file path: D:\DAVIDMP-2023\CURSOS 2023\POO\Sesion 05\asistencia\_db.db. The menu bar includes File, Edit, View, Tools, and Help. The toolbar contains icons for New Database, Open Database, Write Changes, Revert Changes, Open Project, Save Project, Attach Database, and Close Database. The main interface has tabs for Database Structure, Browse Data, Edit Pragma, and Execute SQL. The Execute SQL tab is active, showing a SQL script with two CREATE TABLE statements. The first statement defines a table with columns: id\_resultado (INTEGER NOT NULL), id\_pregunta (INTEGER NOT NULL), id\_postulante (INTEGER NOT NULL), respuesta (VARCHAR(5) NOT NULL), punto (INTEGER NOT NULL), and id\_area\_periodo (INTEGER NOT NULL). It includes a PRIMARY KEY constraint on id\_resultado and two FOREIGN KEY constraints. The second statement defines a table with columns: id\_area\_periodo (INTEGER NOT NULL), id\_area (INTEGER NOT NULL), and id\_periodo (INTEGER NOT NULL), with a PRIMARY KEY constraint on id\_area\_periodo. The status bar at the bottom indicates "Execution finished without errors." and "Result: query executed successfully. Took 0ms".

```
113         id_resultado INTEGER NOT NULL,  
114         id_pregunta INTEGER NOT NULL,  
115         id_postulante INTEGER NOT NULL,  
116         respuesta VARCHAR(5) NOT NULL,  
117         punto INTEGER NOT NULL,  
118         id_area_periodo INTEGER NOT NULL,  
119         CONSTRAINT resultado_pk PRIMARY KEY (id_resultado AUTOINCREMENT),  
120         FOREIGN KEY (id_pregunta) REFERENCES preguntas (id_pregunta) ON DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE,  
121         FOREIGN KEY (id_postulante) REFERENCES postulante (id_postulante) ON DELETE NO ACTION ON UPDATE NO ACTION NOT DEFERRABLE;  
122     );  
123  
124  
125  
126 CREATE TABLE area_periodo (  
127     id_area_periodo INTEGER NOT NULL,  
128     id_area INTEGER NOT NULL,  
129     id_periodo INTEGER NOT NULL,  
130     CONSTRAINT area_periodo_pk PRIMARY KEY (id_area_periodo AUTOINCREMENT),  
131 );
```

Execution finished without errors.  
Result: query executed successfully. Took 0ms  
At line 126:  
CREATE TABLE area\_periodo (

DB Browser for SQLite - D:\DAVIDMP-2023\CURSOS 2023\POO\Sesion 05\asistencia\_db.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

Create Table Create Index Print

**Database Structure**

Name	Type	Schema
Tables (14)		
area_examen		CREATE TABLE area_examen ( id_area_examen INTEGER NOT NULL, nombreae VARCHAR(30) NOT NULL, CO
area_periodo		CREATE TABLE area_periodo ( id_area_periodo INTEGER NOT NULL, id_area INTEGER NOT NULL, id_periodo I
area_periodo_result		CREATE TABLE area_periodo_result ( id_area_periodo_result INTEGER NOT NULL, id_area_periodo INTEGER N
areas		CREATE TABLE areas ( id_area INTEGER NOT NULL, nombrearea VARCHAR(30) NOT NULL, siglas VARCHAR(8
banco_preguntas		CREATE TABLE banco_preguntas ( id_bp INTEGER NOT NULL, id_area INTEGER NOT NULL, pregunta VARCHAR
carrera		CREATE TABLE carrera ( id_carrera INTEGER NOT NULL, id_facultad INTEGER NOT NULL, id_area_examen INT
facultad		CREATE TABLE facultad ( id_facultad INTEGER NOT NULL, nombrefacultad VARCHAR(40) NOT NULL, CONSTRA
periodo		CREATE TABLE periodo ( id_periodo INTEGER NOT NULL, nombre VARCHAR(20) NOT NULL, estado VARCHAR(
postulante		CREATE TABLE postulante ( id_postulante INTEGER NOT NULL, id_carrera INTEGER NOT NULL, id_periodo INT
preguntas		CREATE TABLE preguntas ( id_pregunta INTEGER NOT NULL, id_bp INTEGER NOT NULL, id_area_periodo INTE
resultado		CREATE TABLE resultado ( id_resultado INTEGER NOT NULL, id_pregunta INTEGER NOT NULL, id_postulante IN
resultado_final		CREATE TABLE resultado_final ( id_result_final INTEGER NOT NULL, id_postulante INTEGER NOT NULL, id_peri
sqlite_sequence		CREATE TABLE sqlite_sequence(name,seq)
usuario		CREATE TABLE usuario ( id_usuario INTEGER NOT NULL, user VARCHAR(20) NOT NULL, clave VARCHAR(60) N
Indices (0)		
Views (0)		
Triggers (0)		

**Edit Database Cell**

Mode: Text

NULL

Type of data currently in cell: NULL  
0 byte(s)

Apply

Remote

Identity Select an identity to connect

DBHub.io Local Current Database

Name

Este equipo > Disco local (D:) > DAVIDMP-2023 > CURSOS 2023 > POO > Sesion 05

Nombre	Fecha de modificación	Tipo	Tamaño
example	13/04/2023 13:47	Carpeta de archivos	
java-properties-file	13/04/2023 13:34	Carpeta de archivos	
36960.example.zip	13/04/2023 13:43	Carpeta comprimi...	2 KB
asistencia_db.db	18/04/2023 3:48	Data Base File	68 KB
java-properties-file.zip	13/04/2023 13:33	Carpeta comprimi...	7 KB
Sesion 05-CRUD Base de datos SQLite.pdf	17/03/2023 6:24	Foxit PDF Reader ...	1.664 KB
Sesion 05-CRUD Base de datos SQLite.pptx	17/04/2023 20:39	Presentación de ...	1.346 KB

Inicio

Escritorio

Descargas

Documentos

Imágenes

Música

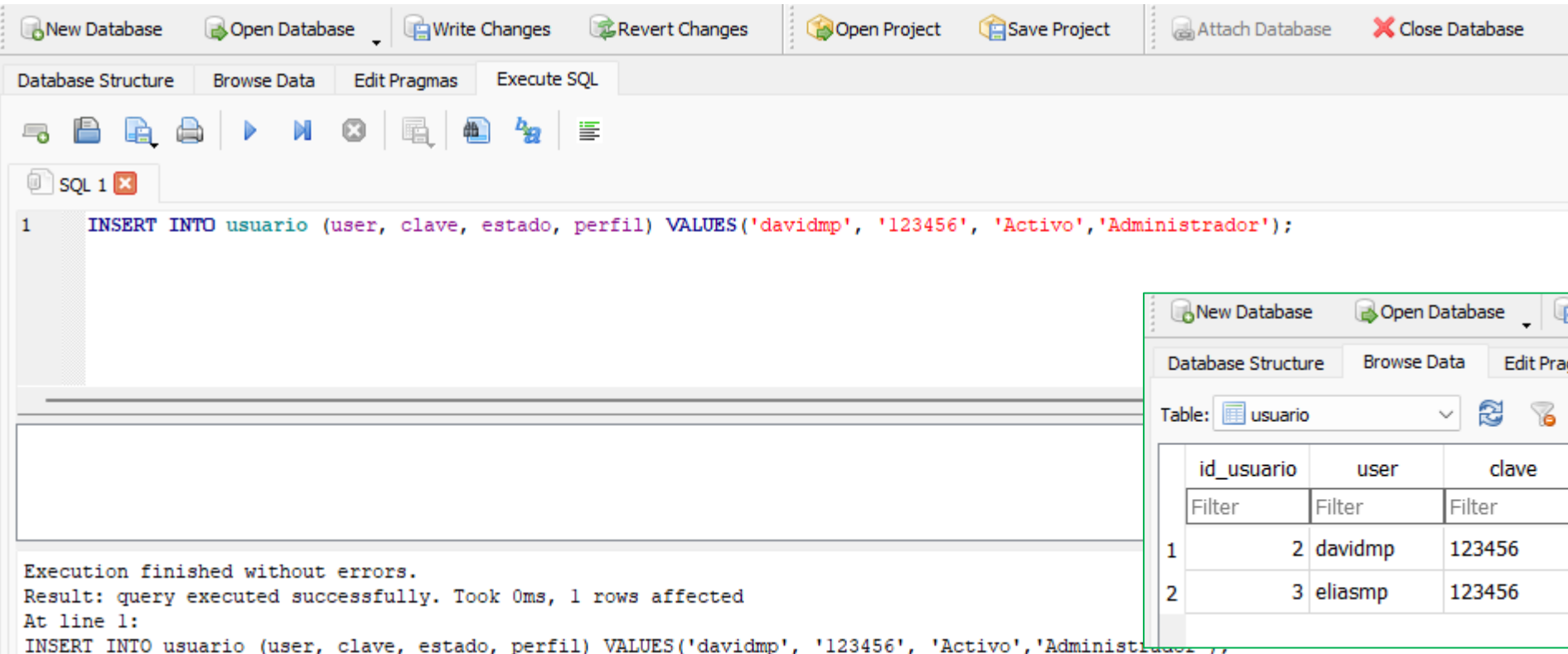
Vídeos

<https://www.sqlitetutorial.net/>

INSERT INTO table (column1,column2 ,..) VALUES( value1, value2 ,...);

INSERT INTO **usuario** (user, clave, estado, perfil) VALUES('davidmp', '123456', 'Activo','Administrador');

INSERT INTO **usuario** (user, clave, estado, perfil) VALUES('eliasmp', '123456', 'Activo','Admsion');

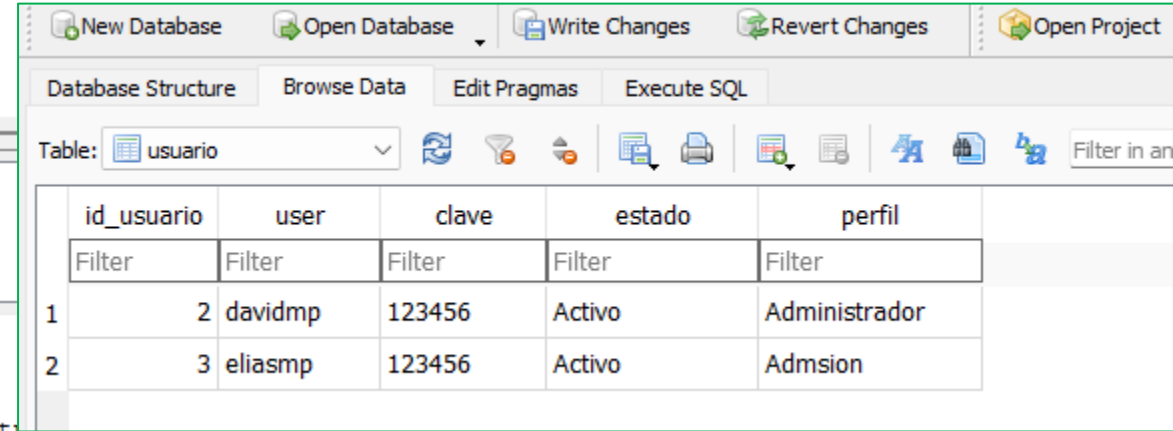


The screenshot shows a database management interface with a menu bar (New Database, Open Database, Write Changes, Revert Changes, Open Project, Save Project, Attach Database, Close Database) and tabs (Database Structure, Browse Data, Edit Pragmas, Execute SQL). The SQL editor contains the following query:

```
1  INSERT INTO usuario (user, clave, estado, perfil) VALUES('davidmp', '123456', 'Activo','Administrador');
```

The execution results pane at the bottom shows:

```
Execution finished without errors.
Result: query executed successfully. Took 0ms, 1 rows affected
At line 1:
INSERT INTO usuario (user, clave, estado, perfil) VALUES('davidmp', '123456', 'Activo','Administrador');
```



The screenshot shows a database management interface with a menu bar (New Database, Open Database, Write Changes, Revert Changes, Open Project) and tabs (Database Structure, Browse Data, Edit Pragmas, Execute SQL). The 'Table: usuario' is selected, and the data is displayed in a table view.

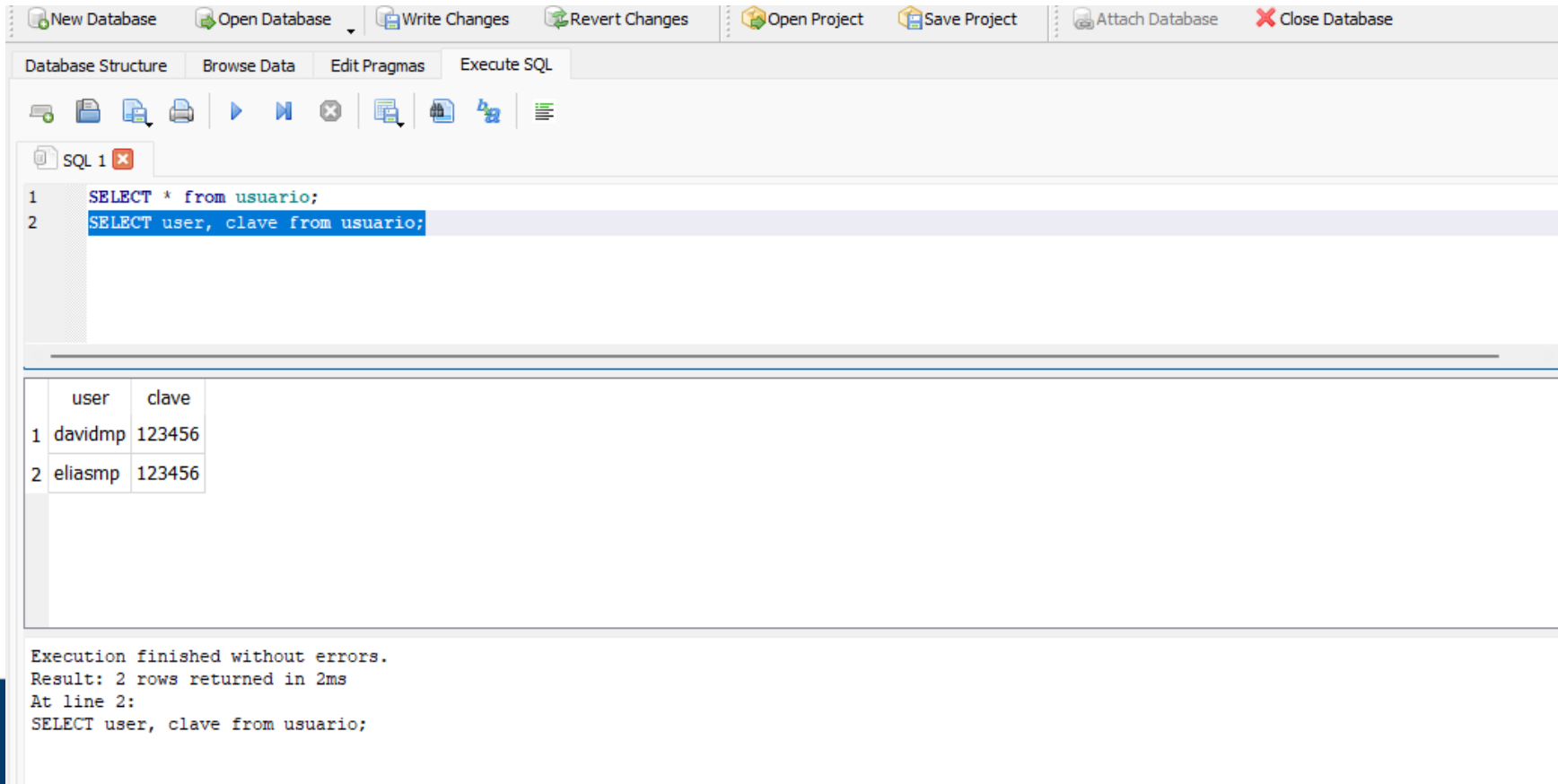
	id_usuario	user	clave	estado	perfil
	Filter	Filter	Filter	Filter	Filter
1	2	davidmp	123456	Activo	Administrador
2	3	eliasmp	123456	Activo	Admsion

<https://www.sqlitetutorial.net/>

SELECT column\_list FROM table;

SELECT \* from **usuario**;

SELECT **user**, **clave** from **usuario**;



The screenshot shows a SQLite IDE window with the following components:

- Toolbar:** New Database, Open Database, Write Changes, Revert Changes, Open Project, Save Project, Attach Database, Close Database.
- Menu Bar:** Database Structure, Browse Data, Edit Pragmas, Execute SQL.
- SQL Editor:** Contains two lines of SQL code:

```
1 SELECT * from usuario;  
2 SELECT user, clave from usuario;
```
- Results Panel:** Displays the results of the second query in a table format:

	user	clave
1	davidmp	123456
2	eliasmp	123456
- Status Bar:** Execution finished without errors. Result: 2 rows returned in 2ms. At line 2: SELECT user, clave from usuario;

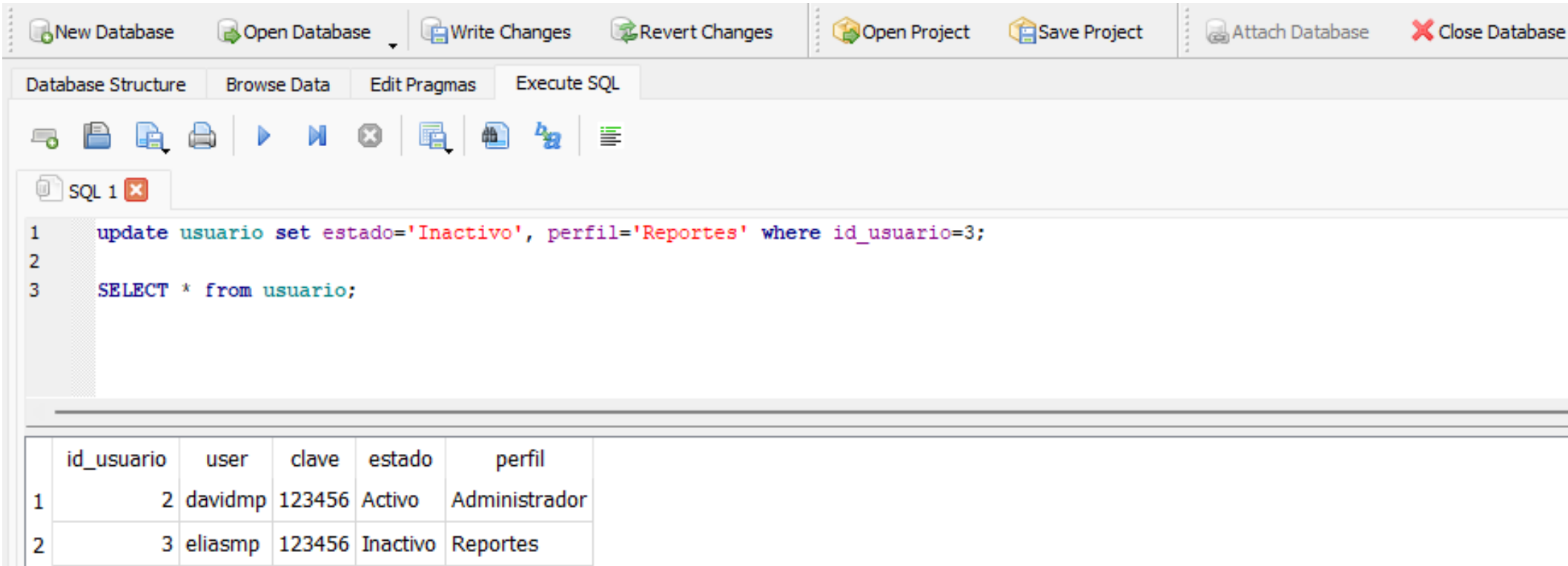


<https://www.sqlitetutorial.net/>

UPDATE employees SET city = 'Toronto', state = 'ON', postcode = 'M5P 2N7' WHERE employeeid = 4;

update **usuario** set **estado**='Inactivo', **perfil**='Reportes' where id\_usuario=3;

SELECT \* from **usuario**;



The screenshot shows a SQLite IDE interface. The top toolbar includes buttons for 'New Database', 'Open Database', 'Write Changes', 'Revert Changes', 'Open Project', 'Save Project', 'Attach Database', and 'Close Database'. Below the toolbar is a tabbed interface with 'Database Structure', 'Browse Data', 'Edit Pragmas', and 'Execute SQL'. The 'Execute SQL' tab is active, showing a text area with the following SQL code:

```
1 update usuario set estado='Inactivo', perfil='Reportes' where id_usuario=3;
2
3 SELECT * from usuario;
```

Below the SQL editor, a table of results is displayed:

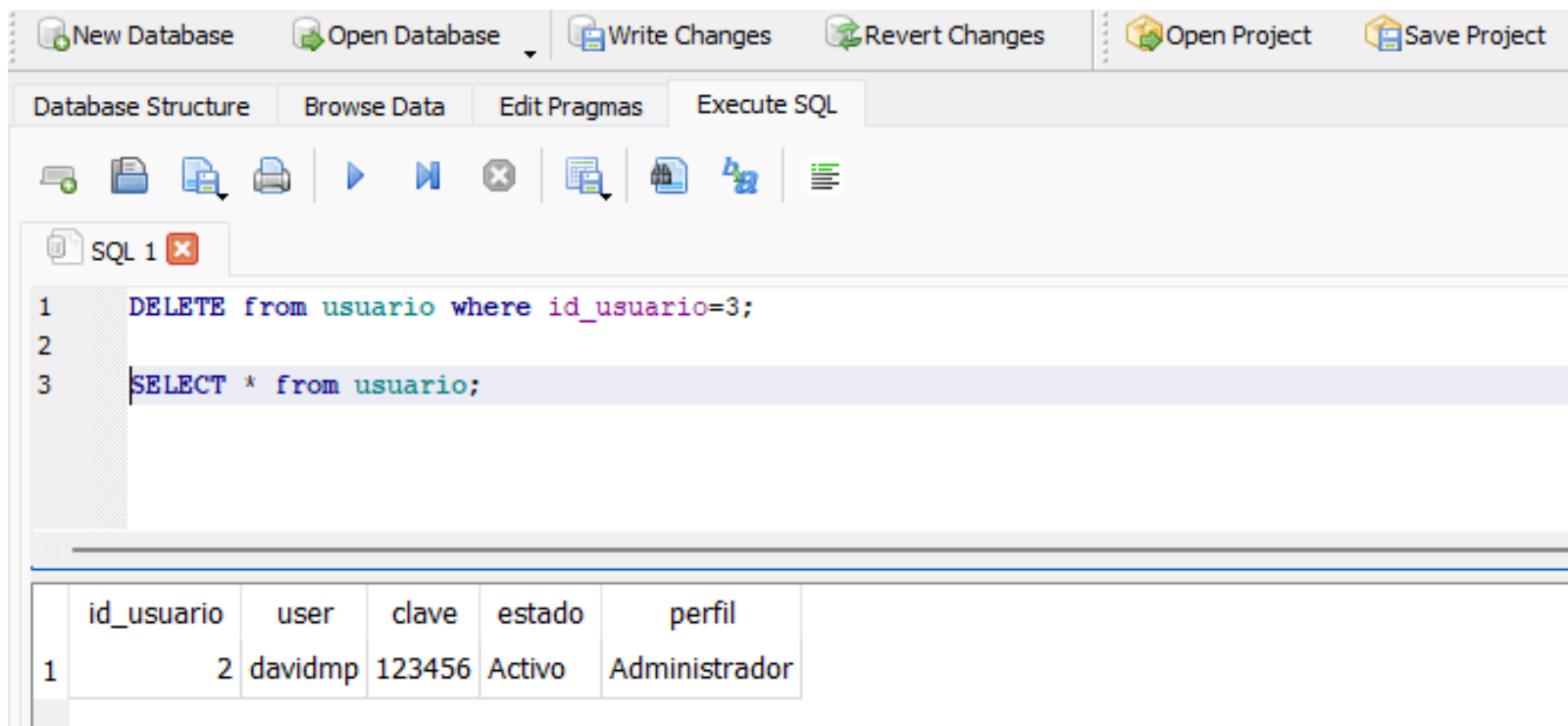
	id_usuario	user	clave	estado	perfil
1	2	davidmp	123456	Activo	Administrador
2	3	eliasmp	123456	Inactivo	Reportes

<https://www.sqlitetutorial.net/>

DELETE FROM table WHERE search\_condition;

DELETE from **usuario** where **id\_usuario=3**;

SELECT \* from **usuario**;





# MUCHAS GRACIAS

*Sé Íntegro, Sé Misionero, Sé Innovador*

---

[www.upeu.edu.pe](http://www.upeu.edu.pe)