

PY518: Implementación del modelo físico

*Diseño funcional y Carga de Datos
(DDL/DML)*

Autores: Alexander, Sergio, Carlos



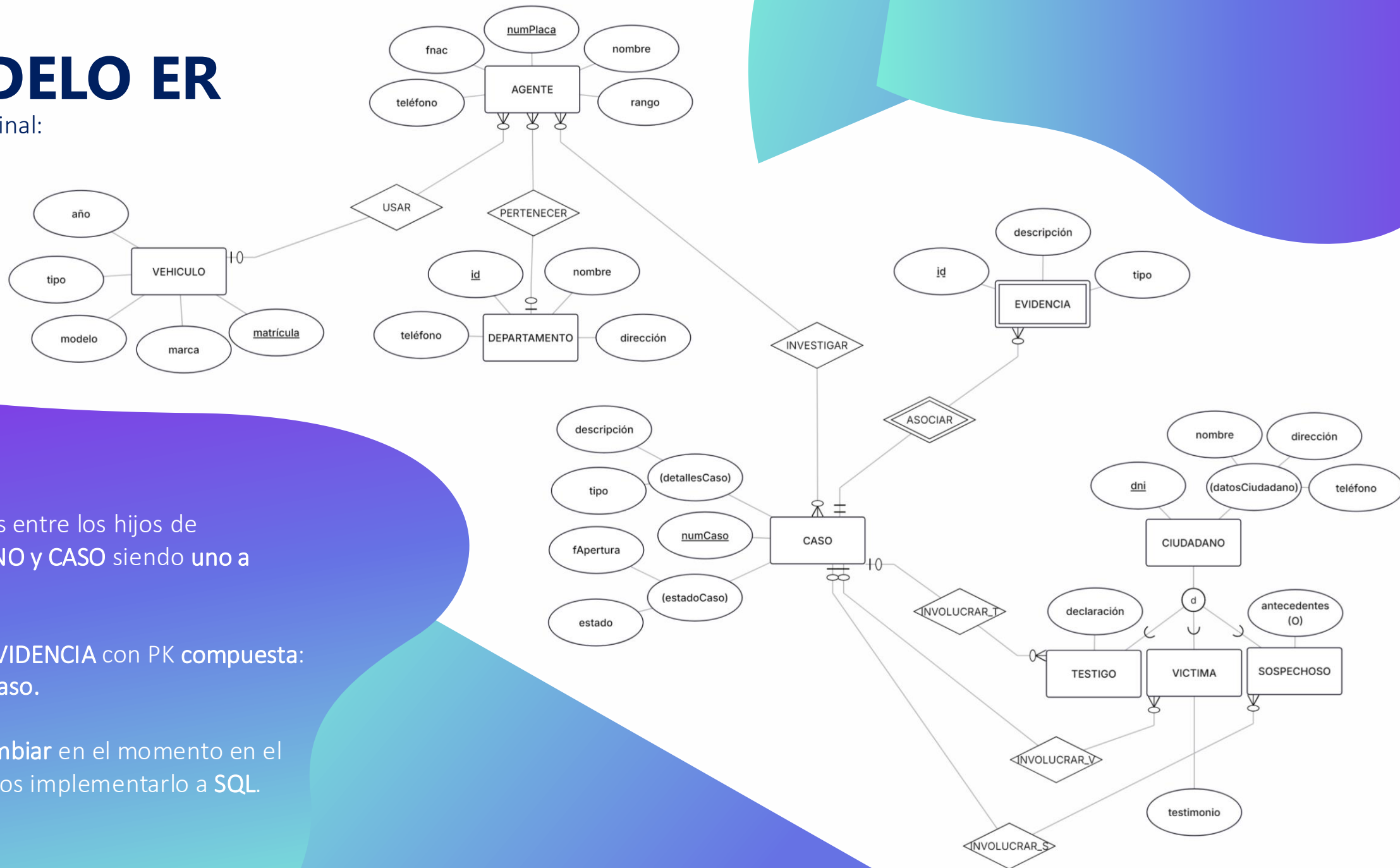
ÍNDICE

- ☐ *MER: Diseño conceptual*
- ☐ *MR: Modelo Relacional y Adaptaciones*
- ☐ *DDL: Implementación, Claves y Restricciones*
- ☐ *DML: Carga de Datos y Validación*



MODELO ER

Diseño original:



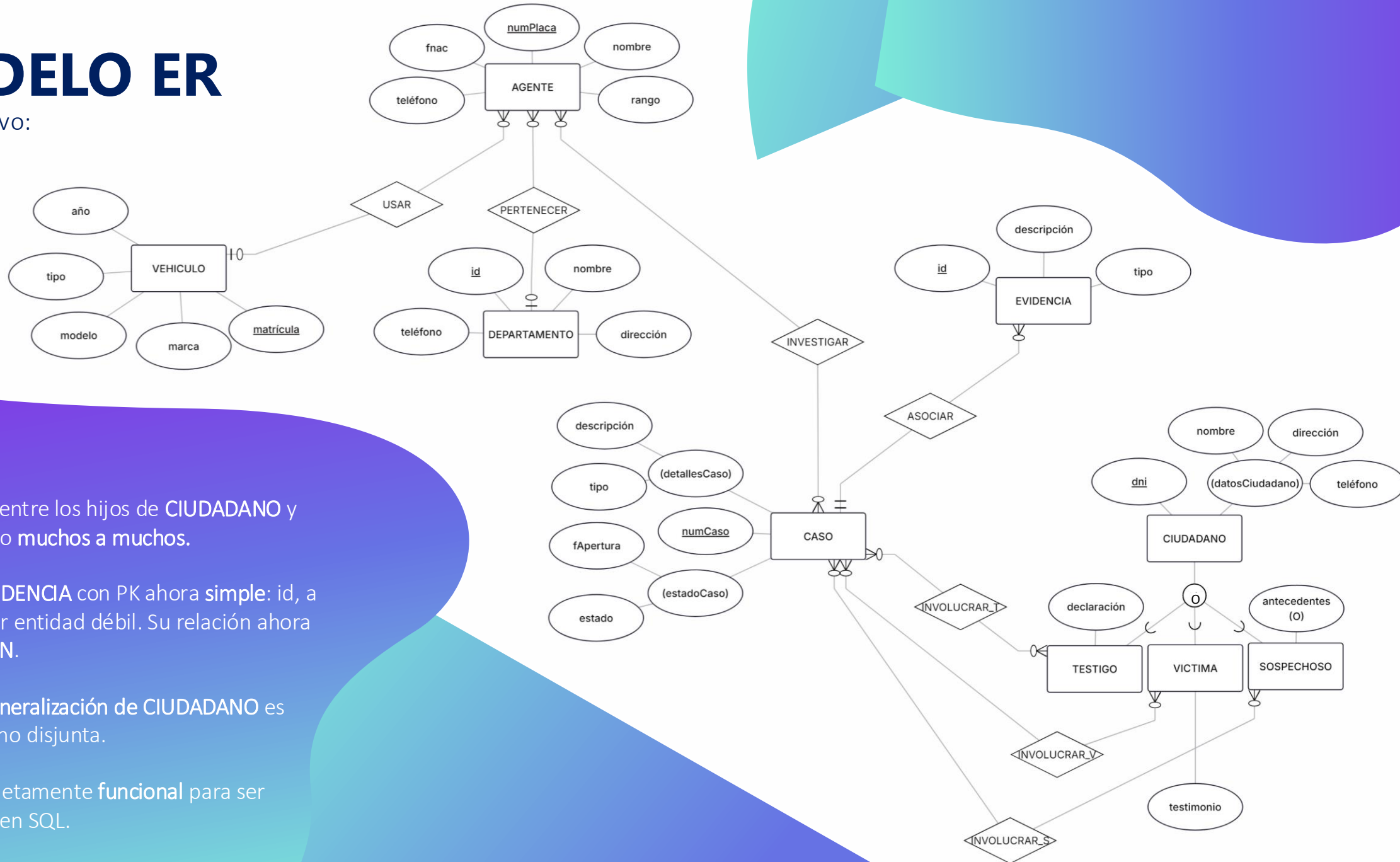
Puntos clave:

- Relaciones entre los hijos de **CIUDADANO** y **CASO** siendo uno a muchos.
- Entidad **EVIDENCIA** con PK compuesta: id y numCaso.

Esto debe **cambiar** en el momento en el que intentamos implementarlo a **SQL**.

MODELO ER

Diseño nuevo:



Puntos clave:

- Relaciones entre los hijos de **CIUDADANO** y **CASO** siendo muchos a muchos.
- Entidad **EVIDENCIA** con PK ahora simple: id, a pesar de ser entidad débil. Su relación ahora permite VNN.
- Ahora la generalización de **CIUDADANO** es solapada y no disjunta.

Ahora es completamente funcional para ser implementado en SQL.

MODELO MR

Diseño original:

Puntos clave:

- PK de TESTIGO, VICTIMA, SOSPECHOSO siendo simple.

Correcto conceptualmente, pero no válido para SQL (no pueden pertenecer a más de un caso).

TESTIGO (dni*, numCaso*, declaración)

- PK (dni)
- FK (dni) → CIUDADANO
- FK (numCaso) → CASO

VICTIMA (dni*, numCaso*, testimonio)

- PK (dni)
- FK (dni) → CIUDADANO
- FK (numCaso) → CASO

SOSPECHOSO (dni*, numCaso*, antecedentes)

- PK (dni)
- FK (dni) → CIUDADANO
- FK (numCaso) → CASO

EVIDENCIA (id, numCaso*, descripción, tipo)

- PK (id, numCaso)
- FK (numCaso) → CASO

Puntos clave:

- PK compuesta, con id y numCaso.

Correcto para una entidad débil en el modelo MR, pero no válido para SQL (id tiene AUTO_INCREMENT).

MODELO MR

Diseño nuevo:

Puntos clave:

- PK de TESTIGO, VICTIMA, SOSPECHOSO siendo compuesta junto a numCaso.

Correcto para SQL, ahora es una relación N:M donde la propia tabla hijo actúa como **tabla intermedia** entre CIUDADANO y CASO, y permite que un CIUDADANO pertenezca a más de un caso.

TESTIGO (dni*, numCaso*, declaración)

- PK (dni)
- FK (dni) → CIUDADANO
- FK (numCaso) → CASO

VICTIMA (dni*, numCaso*, testimonio)

- PK (dni)
- FK (dni) → CIUDADANO
- FK (numCaso) → CASO

SOSPECHOSO (dni*, numCaso*, antecedentes)

- PK (dni)
- FK (dni) → CIUDADANO
- FK (numCaso) → CASO

EVIDENCIA (id, numCaso*, descripción, tipo)

- PK (id, numCaso)
- FK (numCaso) → CASO

Puntos clave:

- PK simple, con id.

Deja de ser **entidad débil** y ahora permite que id tenga **AUTO_INCREMENT**.

SCRIPT DDL

SQL más destacable:

```
CREATE TABLE TESTIGO (  
    dni VARCHAR(9),  
    declaracion VARCHAR(512),  
    numCaso INT UNSIGNED,  
  
    -- La clave primaria compuesta permite al mismo DNI ser testigo en distintos casos.  
    CONSTRAINT PK_TESTIGO PRIMARY KEY (dni, numCaso),  
    CONSTRAINT FK_TES_CIU FOREIGN KEY (dni) REFERENCES CIUDADANO(dni),  
    CONSTRAINT FK_TES_CAS FOREIGN KEY (numCaso) REFERENCES CASO(numCaso)  
);  
  
CREATE TABLE VICTIMA (  
    dni VARCHAR(9),  
    testimonio VARCHAR(512),  
    numCaso INT UNSIGNED,  
  
    -- Clave compuesta para ser víctima en múltiples casos.  
    CONSTRAINT PK_VICTIMA PRIMARY KEY (dni, numCaso),  
    CONSTRAINT FK_VIC_CIU FOREIGN KEY (dni) REFERENCES CIUDADANO(dni),  
    CONSTRAINT FK_VIC_CAS FOREIGN KEY (numCaso) REFERENCES CASO(numCaso)  
);  
  
CREATE TABLE SOSPECHOSO (  
    dni VARCHAR(9),  
    antecedentes VARCHAR(512),  
    numCaso INT UNSIGNED,  
  
    -- Clave compuesta para ser sospechoso en múltiples casos.  
    CONSTRAINT PK_SOSPECHOSO PRIMARY KEY (dni, numCaso),  
    CONSTRAINT FK_SOS_CIU FOREIGN KEY (dni) REFERENCES CIUDADANO(dni),  
    CONSTRAINT FK_SOS_CAS FOREIGN KEY (numCaso) REFERENCES CASO(numCaso)  
);
```





SCRIPT DDL

SQL más destacable:

```
CREATE TABLE EVIDENCIA (  
    id INT UNSIGNED AUTO_INCREMENT,  
    numCaso INT UNSIGNED NOT NULL,  
    descripcion VARCHAR(512),  
    tipo VARCHAR(64),  
  
    CONSTRAINT PK_EVIDENCIA PRIMARY KEY (id),  
    CONSTRAINT FK_EVI_CAS FOREIGN KEY (numCaso) REFERENCES CASO(numCaso)  
);
```

```
CREATE TABLE DEPARTAMENTO (  
    id INT UNSIGNED AUTO_INCREMENT,  
    nombre VARCHAR(128),  
    telefono VARCHAR(16),  
    direccion VARCHAR(128),  
  
    CONSTRAINT PK_DEPARTAMENTO PRIMARY KEY (id)  
);
```

```
ALTER TABLE DEPARTAMENTO AUTO_INCREMENT = 100;
```


SCRIPT DML

SQL más destacable:

```
INSERT INTO VEHICULO (matricula, marca, modelo, tipo, anyo) VALUES
('ABC1234', 'Toyota', 'Corolla', 'Sedán', 2019);
```

```
INSERT INTO DEPARTAMENTO (nombre, telefono, direccion) VALUES
('Recursos Humanos', '+34 612 345 678', 'Calle Mayor 12, Madrid');
```

```
INSERT INTO AGENTE (numPlaca, nombre, fnac, telefono, rango, matriculaVehiculo, idDepartamento) VALUES
('A1001', 'Carlos Méndez', '1985-03-12', '+34 612 345 678', 'Sargento', 'ABC1234', 1);
```

```
INSERT INTO CASO (descripcionCaso, tipoCaso, fApertura, estadoCaso) VALUES
('Robo en establecimiento comercial con denuncia de testigos.', 'Robo', '2023-01-15', 'Abierto');
```

```
INSERT INTO EVIDENCIA (numCaso, descripcion, tipo) VALUES
(1, 'Cámara de seguridad del local que muestra al sospechoso.', 'Video');
```

```
INSERT INTO INVESTIGAR (numPlaca, numCaso) VALUES
('A1001', 1);
```

```
INSERT INTO TESTIGO (dni, declaracion, numCaso) VALUES
('12345678A', 'Vi al sospechoso saliendo del local.', 1);
```

Las tablas **CASO**, **DEPARTAMENTO** y **EVIDENCIA** generan automáticamente sus **PK** al insertar registros (**AUTO_INCREMENT**).

Tablas como **INVESTIGAR**, **TESTIGO**, **VICTIMA** y **SOSPECHOSO** usan **PK compuesta**. Esto permite que un ciudadano o agente participe en **múltiples casos** sin duplicar información.



VACIAR TABLAS

```
-- 1. Deshabilitar temporalmente la verificación de claves foráneas.
-- Esto permite truncar tablas con relaciones sin error.
SET FOREIGN_KEY_CHECKS = 0;

-- 2. TRUNCAR LAS TABLAS

-- Tablas de relaciones y entidades débiles (se pueden truncar primero)
TRUNCATE TABLE INVESTIGAR;
TRUNCATE TABLE TESTIGO;
TRUNCATE TABLE VICTIMA;
TRUNCATE TABLE SOSPECHOSO;
TRUNCATE TABLE EVIDENCIA;

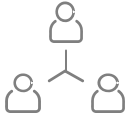
-- Tablas de entidades principales (en un orden seguro)
TRUNCATE TABLE AGENTE;
TRUNCATE TABLE CASO;
TRUNCATE TABLE DEPARTAMENTO;
TRUNCATE TABLE VEHICULO;
TRUNCATE TABLE CIUDADANO;

-- 3. REINICIAR CONTADORES AUTO_INCREMENT
-- Si hay un valor inicial (como 100 para DEPARTAMENTO), se restablece ese valor.

ALTER TABLE DEPARTAMENTO AUTO_INCREMENT = 100;
ALTER TABLE CASO AUTO_INCREMENT = 1;
ALTER TABLE EVIDENCIA AUTO_INCREMENT = 1;

-- 4. Habilitar de nuevo la verificación de claves foráneas.
SET FOREIGN_KEY_CHECKS = 1;
```





Gracias