

Taller 4 bases de datos Sofka

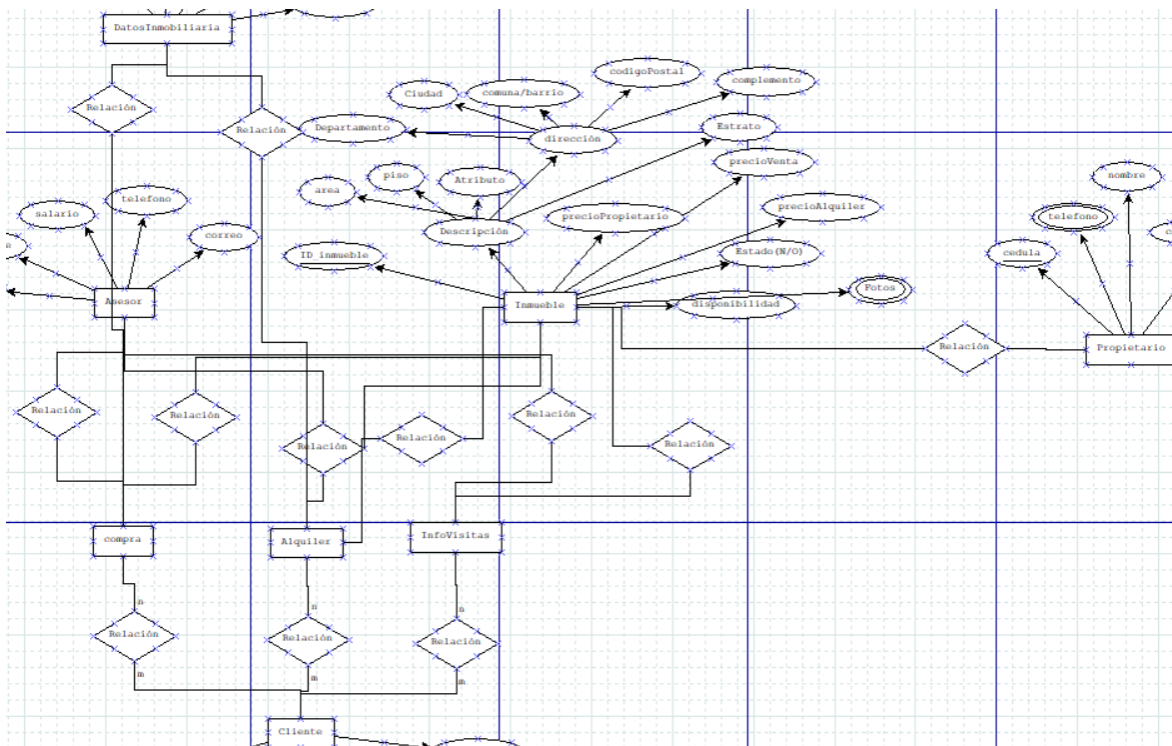
Mer Inmobiliaria SofkaU

Requerimientos:

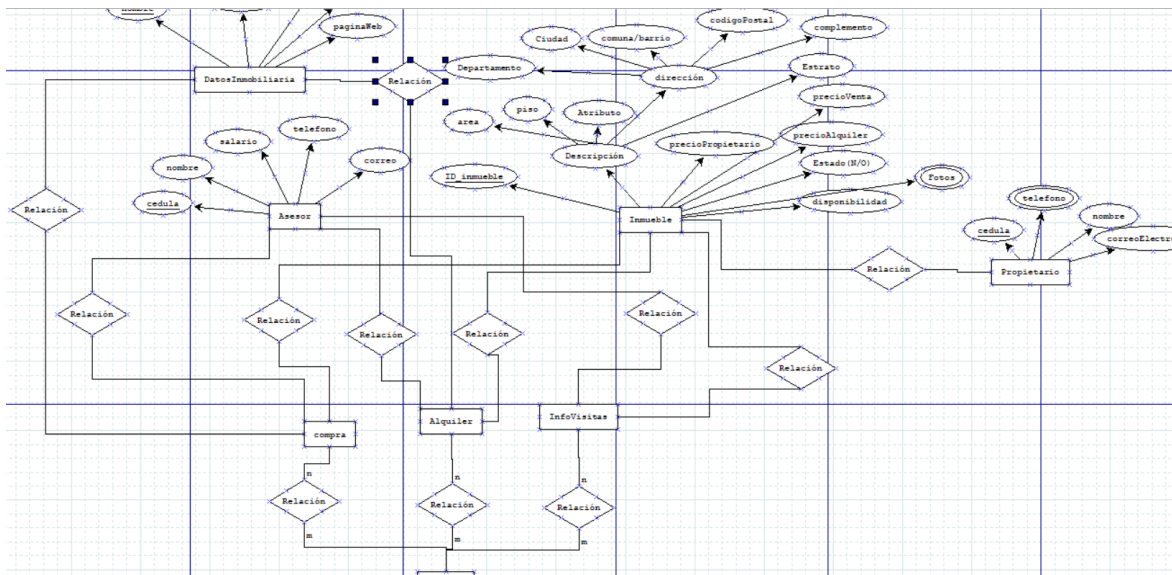
- Complete únicamente las cardinalidades del modelo entidad relacional adjunto (diagrama1.dia).
 - Pase el diagrama a Workbench donde incluya todas las cardinalidades que especifique en el punto anterior.
 - Genere el Script aplicando ingeniería hacia adelante.
 - Corra el Script y genere la base de datos.
 - Poblar todas las tablas de la base de datos al menos con 5 registros.
-
- Después de haber realizado todo lo anterior mencionado responda:
 1. ¿Qué debería cambiar o agregar para incluir la renovación de contratos de alquiler?
 2. ¿Cómo podría controlar que la misma persona que compra un inmueble con identificador X y se lo entrega a la agencia Inmobiliaria SofkaU, NO pueda alquilar el inmueble con identificador X?
-
- Genere un archivo PDF con la documentación clara que indique el paso a paso de como realizo la solución del taller y las respuestas de las dos preguntas anteriores.
 - Emplee el uso adecuado de los commits.

Solución:

Debido a lo complejo que se ve el diagrama del ejercicio en cuanto a sus relaciones:



Decidí organizar un poco el diagrama de tal manera que fuese más legible permitiendo un mejor entendimiento, quedando así:



Una vez realizado este proceso de organización, procedo a completar las cardinalidades como fue propuesto en el ejercicio.

Este modelo Entidad relación, cuenta con un total de 12 relaciones, 8 entidades.

Entidades:

- Datos inmobiliaria
- Asesor
- Cliente
- Compra
- Alquiler
- InfoVisita
- Inmueble
- Propietario

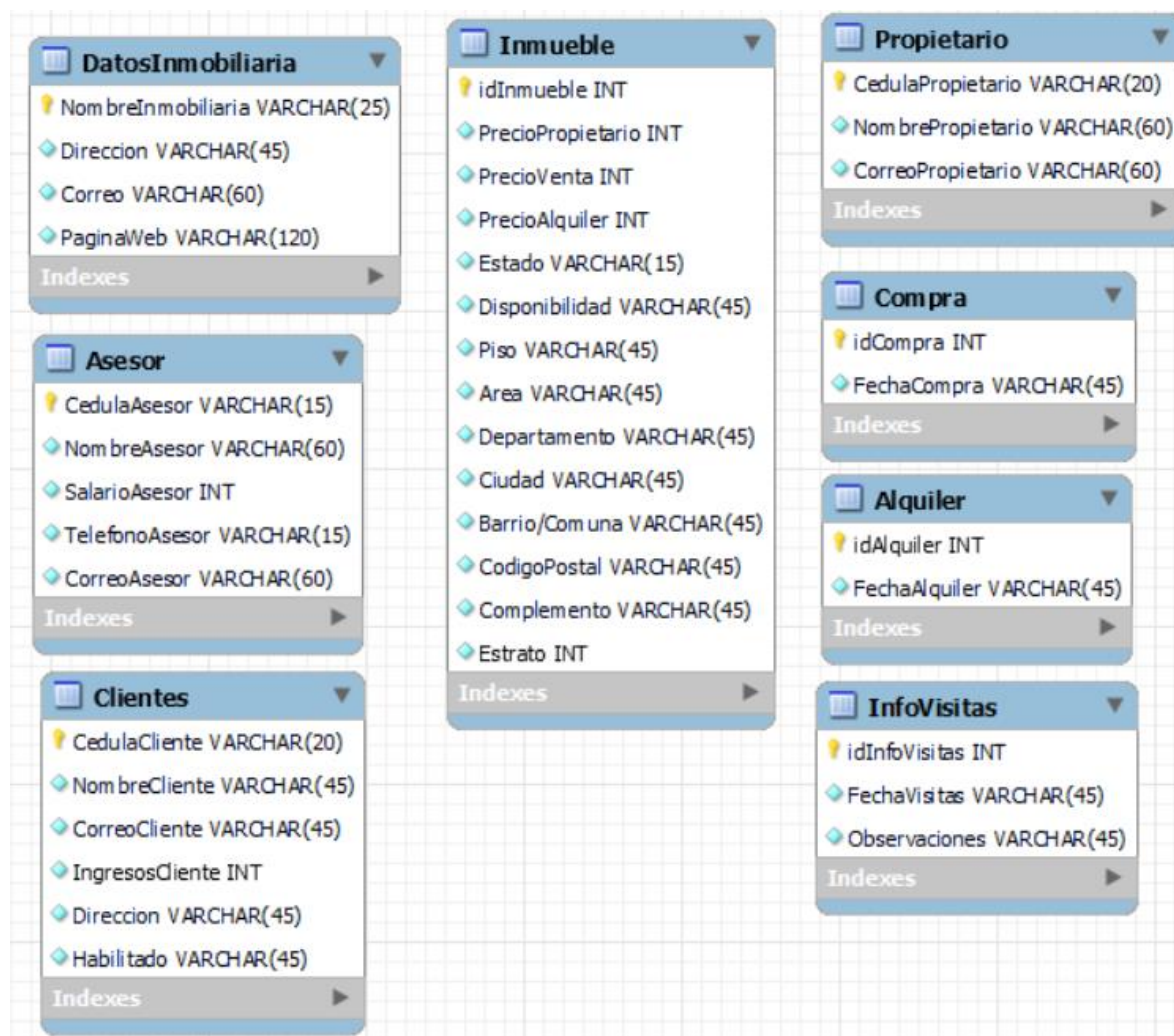
Cardinalidad:

- Relación entre la entidad Cliente y compra: (N, M)
- Relación entre la entidad Cliente y Alquiler: (N, M)
- Relación entre la entidad Cliente y infovisitas: (N, M)
- Relación entre la entidad Asesor y Compra: (1, N)
- Relación entre la entidad Asesor y Alquiler: (1, N)
- Relación entre la entidad Asesor y Infovisitas: (1, N)
- Relación entre la entidad Inmueble y Compra: (1, N)
- Relación entre la entidad Inmueble y Alquiler: (1, N)
- Relación entre la entidad Inmueble y Infovisitas: (1, N)
- Relación entre la entidad Inmueble y Propietarios: (N, M)
- Relación entre la entidad Datos Inmobiliaria y Compra: (1, N)
- Relación entre la entidad Datos Inmobiliaria y Alquiler: (1, N)

Modelo Relacional:

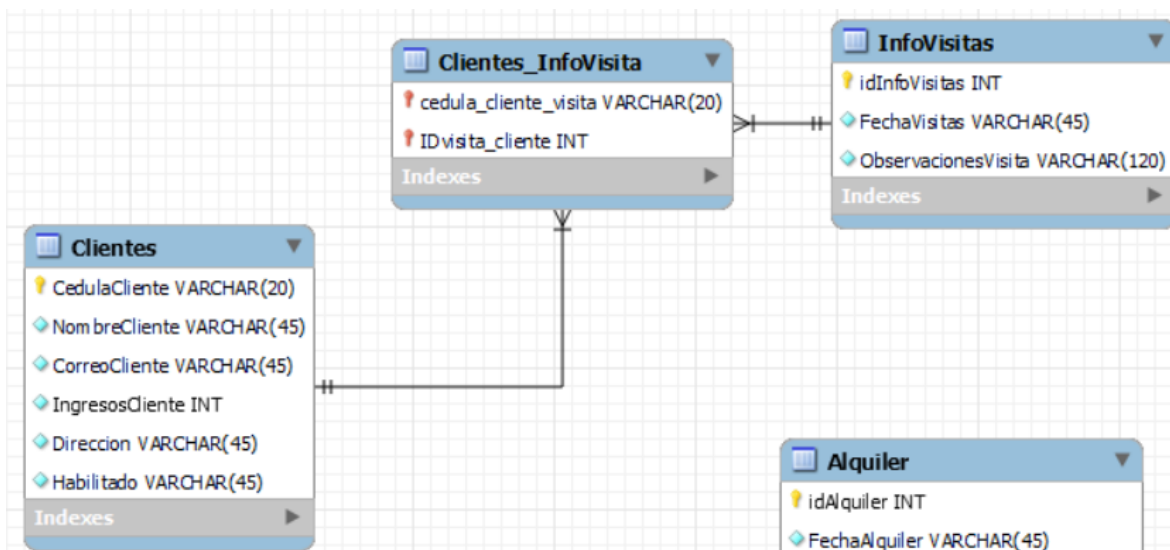
Una vez completado el procedimiento en el diagrama Entidad-Relación, procedemos a crear el modelo relacional.

Se comenzó el proceso creando una tabla para cada una de la entidades de la base de datos, de esta manera:

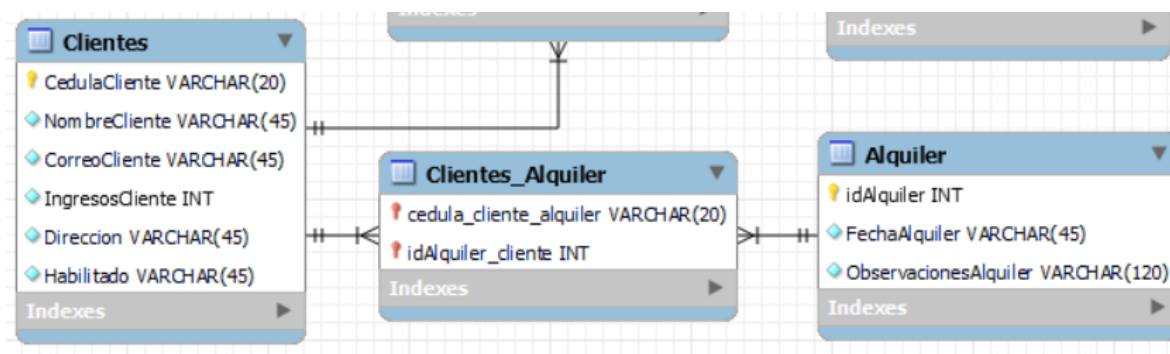


De esta manera ya es posible empezar a relacionarlos en el modelo con las relaciones previamente establecidas en el diagrama ER.

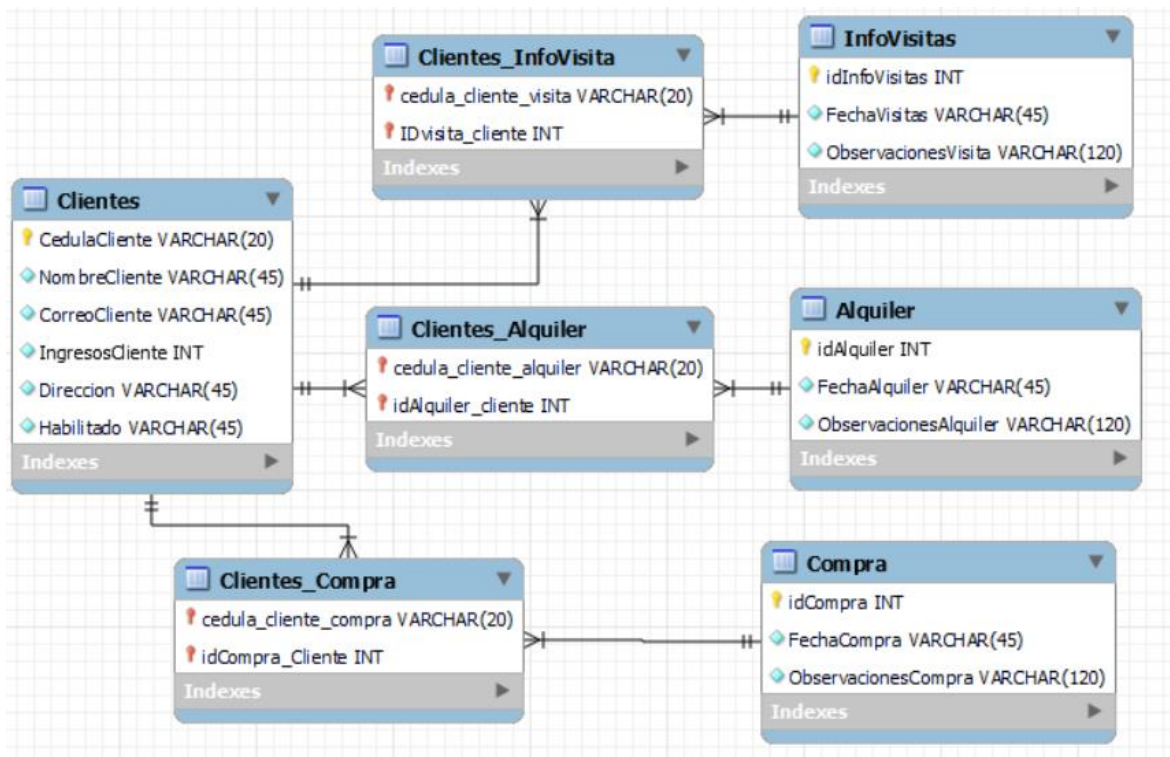
Comenzamos con la relación entre clientes y visita, la cual es una relación de muchos a muchos y queda de esta manera:



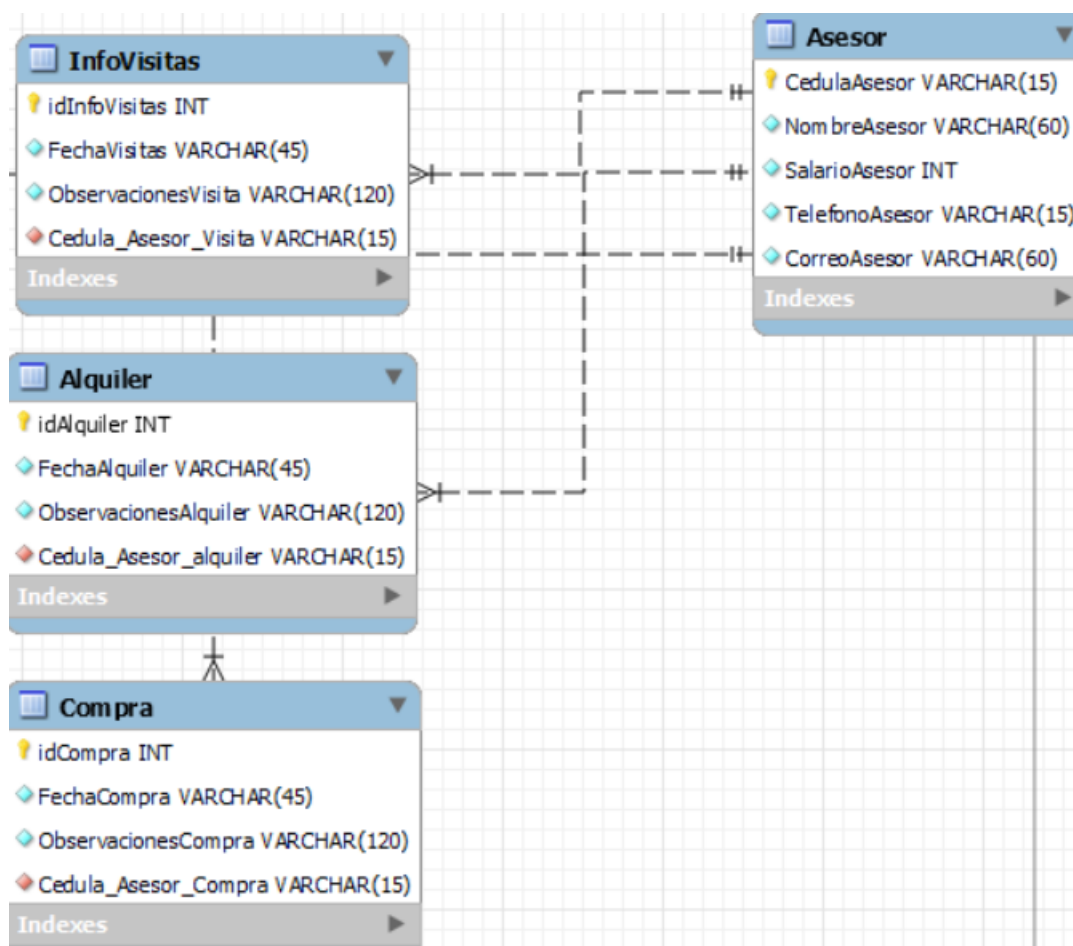
Seguidamente completamos la relación entre clientes y alquiler, la cual también es una relación de uno a muchos, quedando de esta manera:



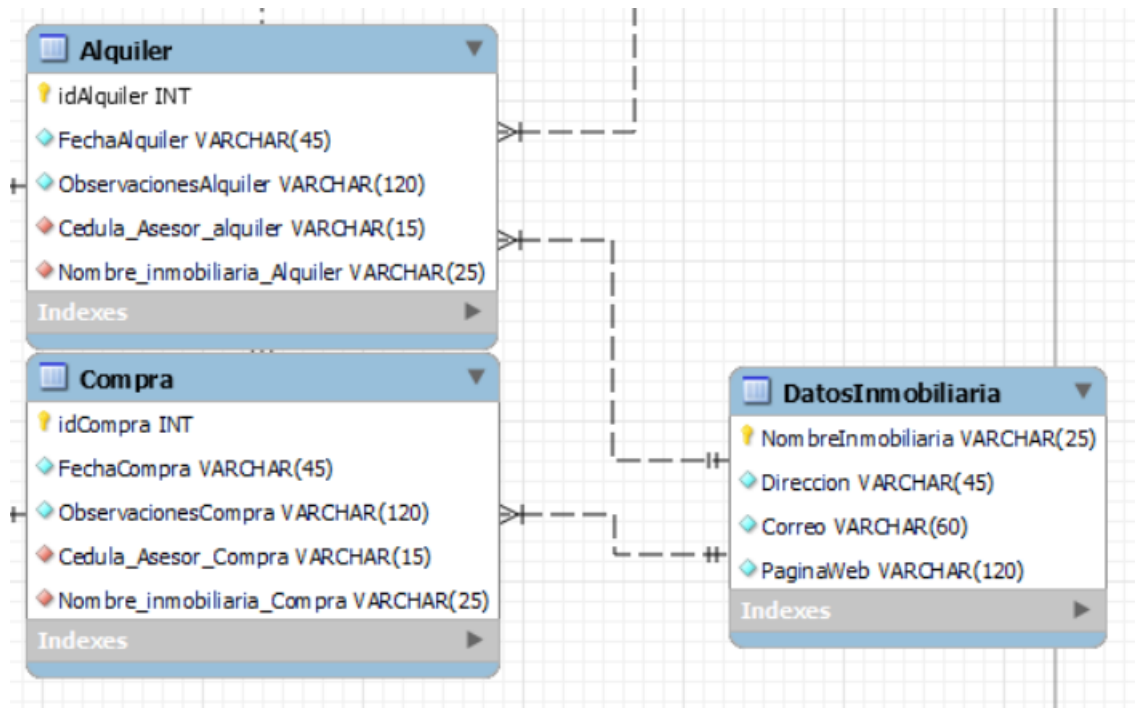
Se realiza el mismo procedimiento para la relación entre compra y cliente dando este resultado para todas las relaciones de clientes:



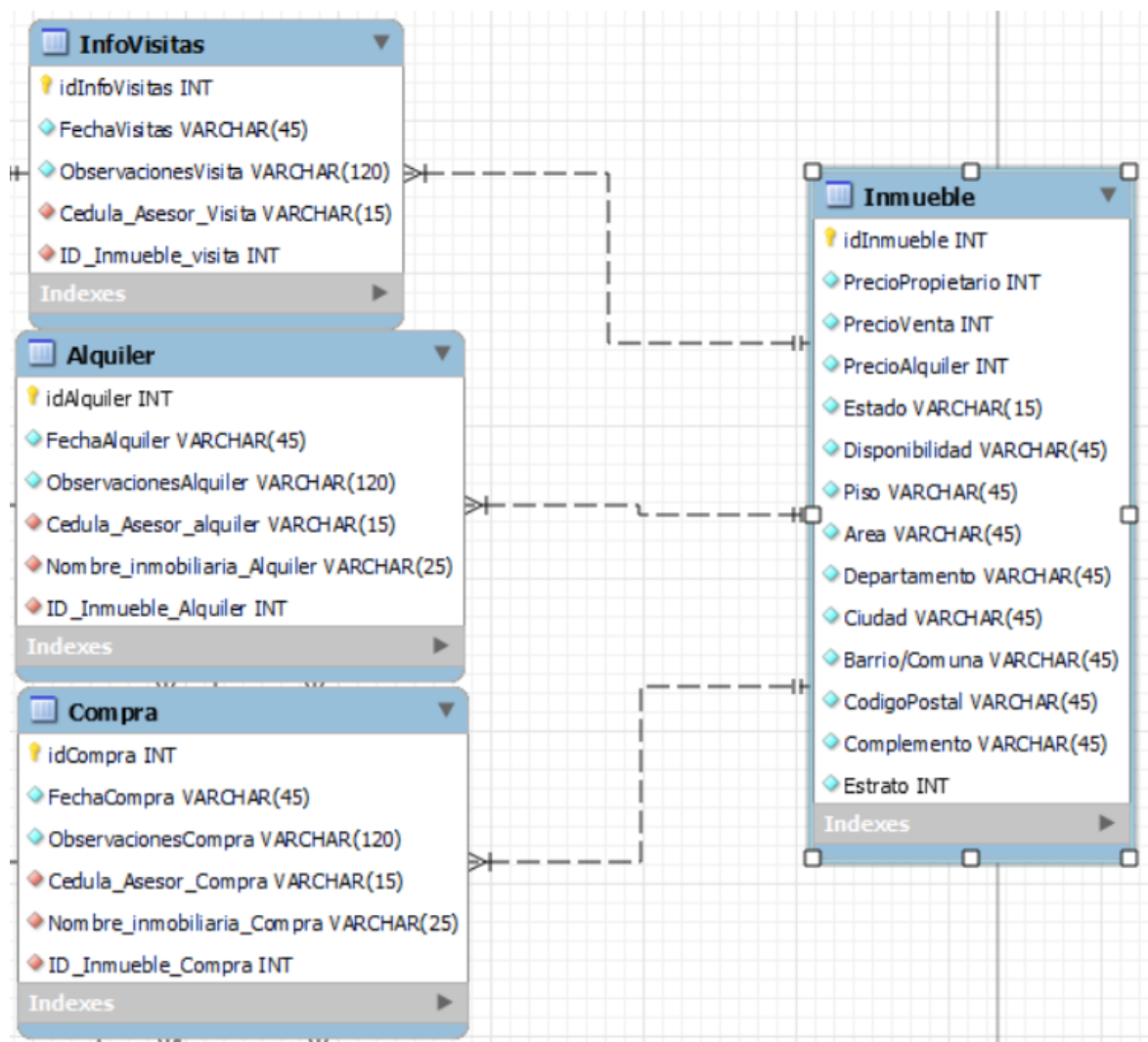
A continuación, crearemos las respectivas relaciones del asesor con Infovisitas, Alquiler, Compra, estas son relaciones de 1 a muchos, en este caso se agrega la cedula del asesor en las otras entidades como una llave foránea.



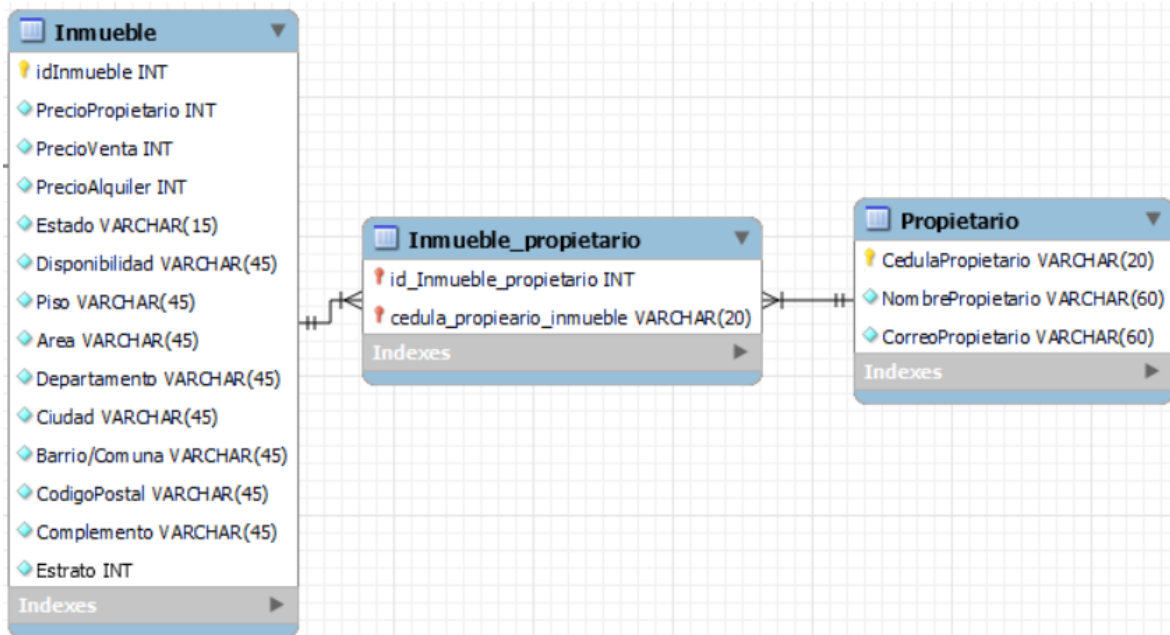
Siguiendo el diagrama relacional, los datos de las compras y los alquileres, se registran bajo una inmobiliaria, por ello se realiza una relación 1 a muchos, en la cual al registro de las compras o alquileres se les asigna el nombre de la inmobiliaria como llave foránea.



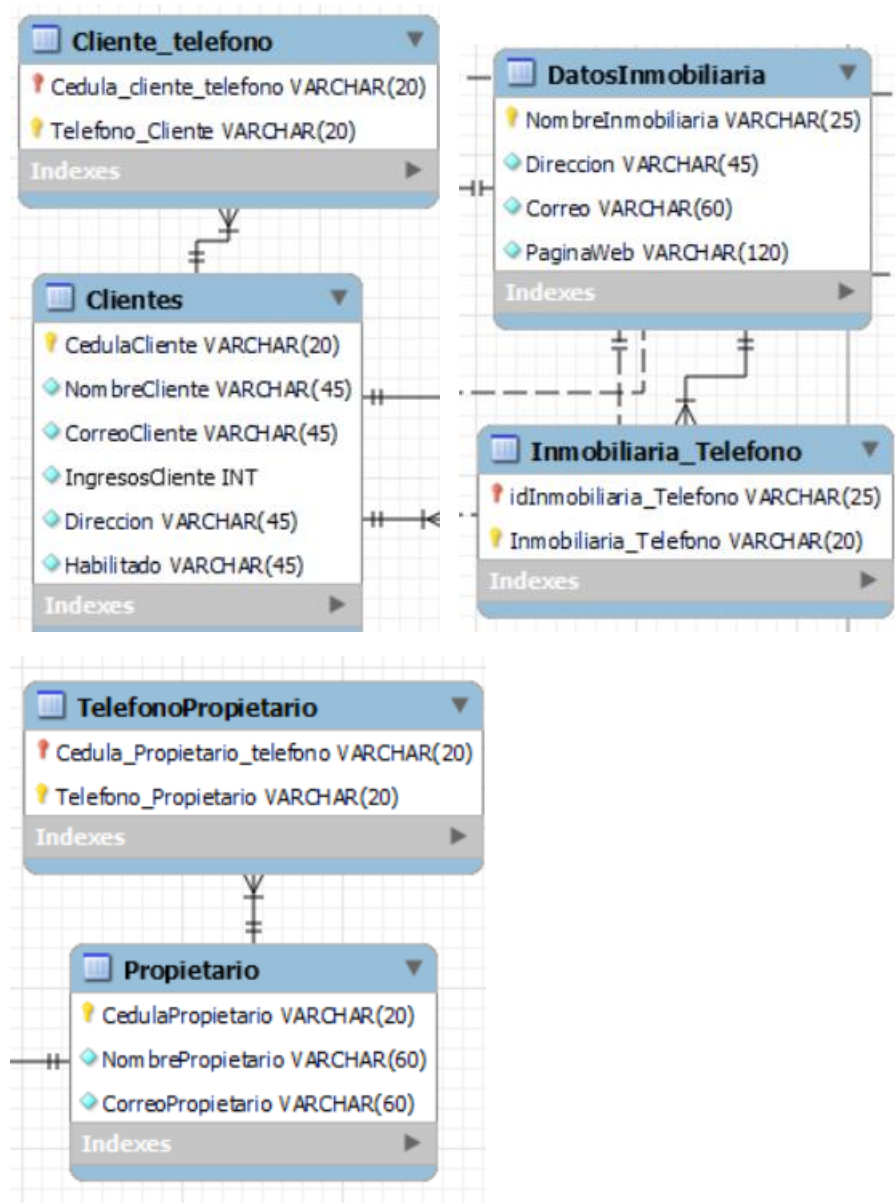
Seguendo el diagrama un inmueble puede ser visitado, alquilado y comprado en una o muchas ocasiones, por ello se crea una relación de uno a muchos entre inmueble, infovistas, alquiler y compra, agregando la ID del inmueble en cada registro de estas tres tablas como una llave foránea, así:

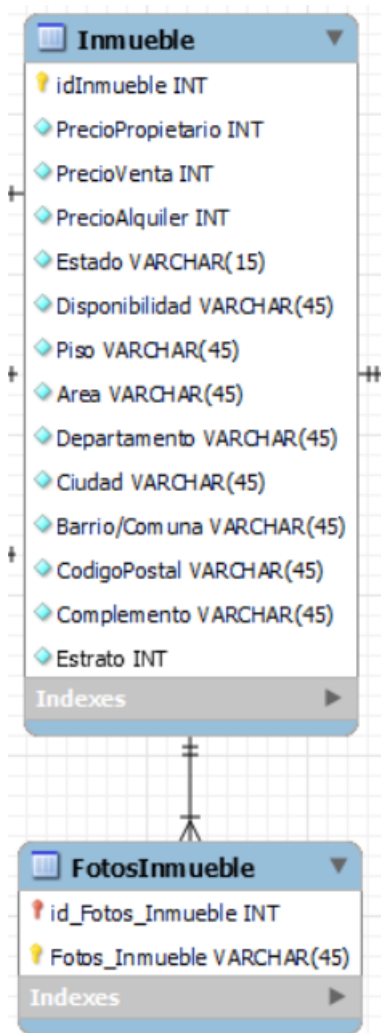


Completando las relaciones, un inmueble puede tener uno o muchos propietarios, y un propietario puede tener muchos inmuebles, por ello se crea una relación de muchos a muchos en el modelo relacional, así:



Para completar el modelo relacional, se crearán los distintos atributos multivariados representados en el diagrama ER, para ello se crea una tabla en la cual se almacenan estos datos utilizando la clave primaria de la entidad a la cual pertenecen estos atributos.





Una vez hecho este proceso, procedo a pasar el chequeo de normalización con cada una de las formas normales, evaluando el modelo relacional, en este caso se cumplen los requisitos.

Normalización:

N1:

Normas	Estado
Todos los atributos tienen valores atómicos	Cumple
No hay atributos multivaluados	Cumple
No existen registros duplicados	Cumple
Se eliminaron todas las columnas repetidas	Cumple
Definir clave principal	Cumple

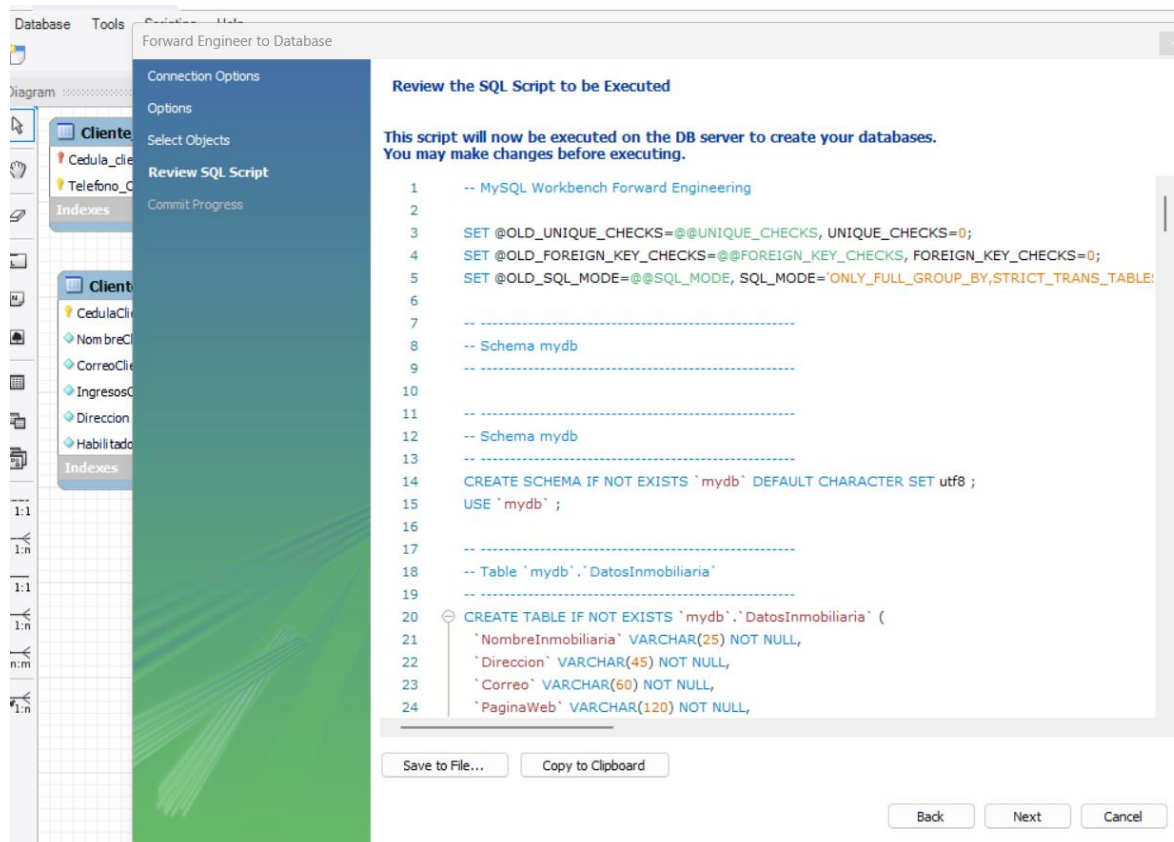
N2:

Normas	Estado
Cumple con la norma 1	Cumple
Los valores de las columnas dependen solo de la llave primaria	Cumple
Las tablas tienen una única llave primaria que las identifique	Cumple

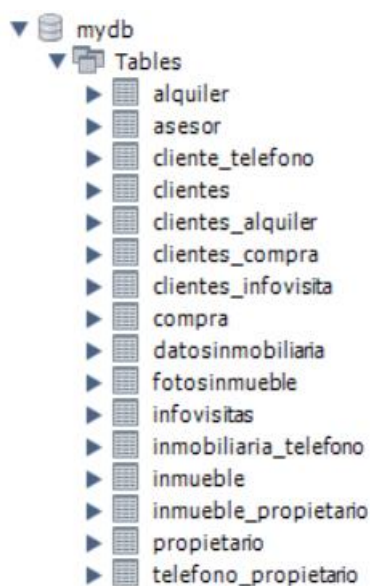
N3:

Normas	Estado
Cumple la norma 2	Cumple
Los atributos no incluidos en la clave primaria no dependen transitivamente la clave primaria	Cumple

Una vez realizados todos estos pasos procedo a generar un script con ingeniería hacia adelante y generar la base de datos.



Base de datos:



Una vez creada la base de datos procederemos a generar registros en cada tabla:

Tabla Inmueble:

Ejemplo sentencia SQL:

```
4 • INSERT INTO `mydb`.`inmueble` (`idInmueble`, `PrecioPropietario`, `PrecioVenta`,
5   `PrecioAlquiler`, `Estado`, `Disponibilidad`,
6   `Piso`, `Area`, `Departamento`, `Ciudad`, `Barrio/Comuna`,
7   `CodigoPostal`, `Complemento`, `Estrato`)
8   VALUES ('1', '64000', '60000', '1500', 'Nuevo', 'Disponible',
9   'Madera', '200', 'Antioquia', 'Medellin', 'comuna 15', '1151', 'sur', '2');
```

Ejemplo registros Inmueble:

idInmueble	PrecioPropietario	PrecioVenta	PrecioAlquiler	Estado	Disponibilidad	Piso	Area	Departamento	Ciudad	Barrio/Comuna	CodigoPostal	Complemento	Estrato
1	64000	60000	1500	Nuevo	Disponible	Mad...	200	Antioquia	Medellin	comuna 15	1151	sur	2
2	70000	65000	1600	Viejo	Disponible	Bald...	300	Cundinamarca	Chia	Esperanza	1245	Norte	3
3	70000	72000	2000	Nuevo	Disponible	Mad...	500	Cundinamarca	Pacho	Tintal	1321	Occidente	1
4	32000	40000	1200	Nuevo	Disponible	Mad...	200	Boyaca	Chiqui...	Centro	3212	Centro	3
5	40000	43000	1600	Nuevo	Disponible	Mad...	200	Santander	Bucar...	Primavera	1115	Norte	3

Tabla Propietario:

Ejemplo sentencia SQL:

```
• INSERT INTO `mydb`.`propietario` (`CedulaPropietario`, `NombrePropietario`,
  `CorreoPropietario`) VALUES ('123456789', 'David Ruiz', 'david@gmail.com');
```

Ejemplo registros:

Result Grid	Filter Rows:	Edit:
CedulaPropietario	NombrePropietario	CorreoPropietario
12345	Paula Cortez	paula@gmail.com
123456	Andrea Perez	andrea@gmail.com
1234567	Fabian lopez	fabian@gmail.com
123456789	David Ruiz	david@gmail.com
321321	Carlos yepes	carlos@gmail.com
HULL	HULL	HULL

Tabla Inmueble_Propietario:

Ejemplo sentencia SQL:

```
6 • INSERT INTO `mydb`.`inmueble_propietario` (`id_Inmueble_propietario`,  
7      `cedula_propieario_inmueble`) VALUES ('1', '12345');
```

Ejemplo Registros:

Result Grid	Filter Rows:	Edi
id_Inmueble_propietario		cedula_propieario_inmueble
1		12345
2		123456
3		1234567
4		123456789
5		321321

Tabla Telefono_Propietario:

Ejemplo sentencia SQL:

```
6 • INSERT INTO `mydb`.`telefono_propietario` (`cedulaPropietario-Telefono`,  
7      `Telefono_Propietariocol`) VALUES ('12345', '7500500');
```

Ejemplo registros:

Result Grid	Filter Rows:	E
cedulaPropietario-Telefono		Telefono_Propietariocol
12345		3202103560
12345		7500500
123456		8500500
1234567		3123215500
321321		9500500

Tabla fotos Inmueble:

Ejemplo sentencia SQL:

```
6 • INSERT INTO `mydb`.`fotosinmueble` (`id_Fotos_Inmueble`, `Fotos_Inmueble`) VALUES ('1', 'Fachada1');
```

Ejemplo registros:

Result Grid	Filter Rows:	
id_Fotos_Inmueble		Fotos_Inmueble
1		Fachada1
2		Fachada2
3		Fachada3
4		Fachada4
5		Fachada5
NULL		NULL

Tabla datos inmobiliaria:

Ejemplo sentencia SQL:

```
4 • INSERT INTO `mydb`.`datosinmobiliaria` (`NombreInmobiliaria`, `Direccion`,  
5   `Correo`, `PaginaWeb`) VALUES ('SofkaU', 'El poblado',  
6   'sofkau@gmail.com', 'sofka.com');
```

Ejemplo registros:

	NombreInmobiliaria	Direccion	Correo	PaginaWeb
	arriendaya	suba	arrienda@gmail.com	arriendaya.com
	Contar	calle 70	contar@gmail.com	contar.com
	formarte	cr 40	formarte@gmail.com	formarte.com
	SofkaU	El poblado	sofkau@gmail.com	sofka.com
▶	tucasa	tu casa	tucasa@gmail.com	tucasa.com
*	NULL	NULL	NULL	NULL

Tabla teléfono inmobiliaria:

Ejemplo sentencia SQL:

```
• INSERT INTO `mydb`.`inmobiliaria_telefono` (`idInmobiliaria_Telefono`,  
  `Inmobiliaria_Telefono`) VALUES ('arriendaya', '320456');
```

Ejemplo registros:

	idInmobiliaria_Telefono	Inmobiliaria_Telefono
	arriendaya	320456
	Contar	321456
	formarte	450321
	SofkaU	32156
▶	tucasa	999991
	NULL	NULL

Tabla Clientes:

Ejemplo sentencia SQL:

```
INSERT INTO `mydb`.`clientes` (`CedulaCliente`, `NombreCliente`,  
`CorreoCliente`, `IngresosCliente`, `Direccion`, `Habilitado`) VALUES ('656522',  
'Paolo Morales', 'paolo@gmail.com', '3500', 'cra 12 # 10', 'Si');
```

Ejemplo registros:








Result Grid			 Filter Rows:	<input type="text"/>	Edit:				Export/Import:		
	CedulaCliente	NombreCliente	CorreoCliente	IngresosCliente	Direccion	Habilitado					
	123321	Luis novoa	luis@gmail.com	4000	CL 26 # 25	Si					
	656512	Claudio vega	claudia@gmail.com	4000	d 65 # 14	Si					
	656522	Paolo Morales	paolo@gmail.com	3500	cra 12 # 10	Si					
	757752	Jose Ortiz	jose@gmail.com	3000	CL 11 # 11	Si					
	876494	Maria bernal	maria@gmail.com	2900	d 12 # 12	Si					

Tabla cliente teléfono:

Ejemplo sentencia SQL:

```
5 INSERT INTO `mydb`.`cliente_telefono` (`Cedula_cliente_telefono`,  
6 `Telefono_Cliente`) VALUES ('123321', '3100100');
```

Ejemplo registros:

Cedula_cliente_telefono	Telefono_Cliente
123321	3100100
656512	2100100
656522	6100100
757752	5100100
876494	4100100

Tabla Asesores:

Ejemplo sentencia SQL:

```
INSERT INTO `mydb`.`asesor` (`CedulaAsesor`, `NombreAsesor`,  
`SalarioAsesor`, `TelefonoAsesor`, `CorreoAsesor`) VALUES ('654654', 'Paola correa',  
'2500', '8900900', 'paola@hotmail.com');
```

Ejemplo registros:

	CedulaAsesor	NombreAsesor	SalarioAsesor	TelefonoAsesor	CorreoAsesor
	147258	leonel alvarez	2500	6900900	leo@hotmail.com
	159357	camila cabello	3500	4900900	camila@hotmail.c...
	456321	emiliano marti...	2500	7900900	emilio@hotmail.com
	654654	Paola correa	2500	8900900	paola@hotmail.com
	963258	bruno soriano	2500	5900900	bruno@hotmail.com
	NULL	NULL	NULL	NULL	NULL

Tabla Alquiler:

Ejemplo sentencia SQL:

```
3 • INSERT INTO `mydb`.`alquiler` (`idAlquiler`, `FechaAlquiler`,  
4   `ObservacionesAlquiler`, `Cedula_Asesor_alquiler`,  
5   `Nombre_inmobiliaria_Alquiler`, `ID_Inmueble_Alquiler`) VALUES ('25', '12/02/2023',  
6   'Acuerdo concertado', '147258', 'SofkaU', '1');
```

Ejemplo registros:

	idAlquiler	FechaAlquiler	ObservacionesAlquiler	Cedula_Asesor_alquiler	Nombre_inmobiliaria_Alquiler	ID_Inmueble_Alquiler
	25	12/02/2023	Acuerdo concertado	147258	SofkaU	1
	26	12/02/2023	Acuerdo concertado	456321	formarte	2
	27	12/02/2023	Acuerdo concertado	147258	SofkaU	3
	28	12/02/2023	Acuerdo concertado	456321	formarte	4
	29	12/02/2023	Acuerdo concertado	147258	SofkaU	5

Tabla compra:

Ejemplo sentencia SQL:

```
• INSERT INTO `mydb`.`compra` (`idCompra`, `FechaCompra`,  
  `ObservacionesCompra`, `Cedula_Asesor_Compra`,  
  `Nombre_inmobiliaria_Compra`, `ID_Inmueble_Compra`) VALUES ('78', '12/02/2023',  
  'Completado', '159357', 'Contar', '1');
```

Ejemplo registros:

	idCompra	FechaCompra	ObservacionesCompra	Cedula_Asesor_Compra	Nombre_inmobiliaria_Compra	ID_Inmueble_Compra
	78	12/02/2023	Completado	159357	Contar	1
	79	12/02/2023	Completado	963258	Contar	2
	80	12/02/2023	Completado	159357	tucasa	3
	81	12/02/2023	Completado	963258	tucasa	4
	82	12/02/2023	Completado	159357	Contar	5

Tabla info Visitas:

Ejemplo sentencia SQL:

```
INSERT INTO `mydb`.`infovisitas` (`idInfoVisitas`, `FechaVisitas`,  
`ObservacionesVisita`, `Cedula_Asesor_Visita`,  
`ID_Inmueble_visita`) VALUES ('101', '13/02/2023',  
'No le gusto', '456321', '1');
```

Ejemplo registros:

Result Grid

Filter Rows:

Edit:

Export/Import:

	idInfoVisitas	FechaVisitas	ObservacionesVisita	Cedula_Asesor_Visita	ID_Inmueble_visita
	101	13/02/2023	No le gusto	456321	1
	102	13/02/2023	No le gusto	456321	2
	103	13/02/2023	No le gusto	654654	3
	104	13/02/2023	No le gusto	456321	4
	105	13/02/2023	No le gusto	654654	5
	NULL	NULL	NULL	NULL	NULL

Tabla clientes alquiler:

Ejemplo sentencia SQL:

```
INSERT INTO `mydb`.`clientes_alquiler` (`cedula_cliente_alquiler`,  
`idAlquiler_cliente`) VALUES ('123321', '25');
```

Ejemplo registros:

cedula_cliente_alquiler	idAlquiler_cliente
123321	25
656512	26
656522	27
757752	28
876494	29

Tabla clientes compra:

Ejemplo sentencia SQL:

```
INSERT INTO `mydb`.`clientes_compra` (`cedula_cliente_compra`,  
`idCompra_Cliente`) VALUES ('123321', '78');
```

Ejemplo registros:

Result Grid		Filter Rows:
	cedula_cliente_compra	idCompra_Cliente
	123321	78
	656512	79
	656522	80
	757752	81
▶	876494	82

Tabla clientes info visitas:

Ejemplo sentencia SQL:

```
INSERT INTO `mydb`.`clientes_infovisita` (`cedula_cliente_visita`,  
`IDvisita_cliente`) VALUES ('123321', '101');
```

Ejemplo registros:

	cedula_cliente_visita	IDvisita_cliente
	123321	101
	656512	102
	656522	103
	757752	104
▶	876494	105

Preguntas:

- ¿Qué debería cambiar o agregar para incluir la renovación de contratos de alquiler?

R:

Se debería agregar una entidad mas entre las entidades alquiler y cliente, esta entidad se llamará contrato y contará con los detalles de la duración del alquiler, y será el medio por el cual el cliente adquirirá un alquiler.

- ¿Cómo podría controlar que la misma persona que compra un inmueble con identificador X y se lo entrega a la agencia Inmobiliaria SofkaU, NO pueda alquilar el inmueble con identificador X?

R:

Se podría controlar agregando un atributo a cliente, este atributo se llamara propiedad_adquirida, se configurara de tal manera que permita un valor nulo en caso de que el cliente no tenga ninguna propiedad, y se configurara en alquiler de tal manera que si el atributo propiedad adquirida coincide con el que desea alquilar; no lo deje completar el proceso.