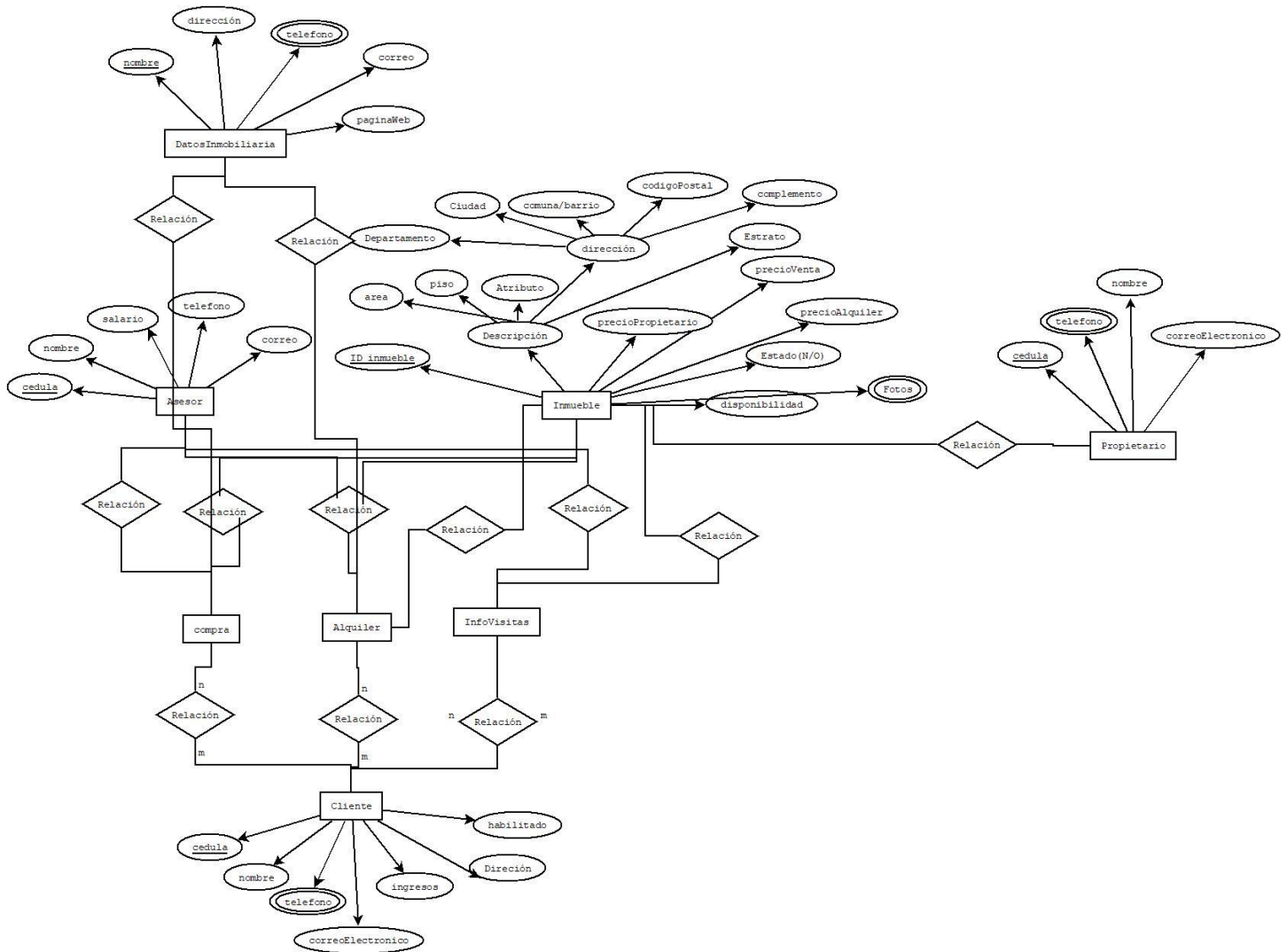


Taller MER Inmobiliaria

Contenido

Modelo Entidad relación inmobiliaria.....	1
Descripción MER inmobiliaria	1
Diagrama MER inmobiliaria	3
Script corregido para crear tablas de base de datos inmobiliaria.	3
Script con datos.....	7
Respuesta a preguntas.....	9

Modelo Entidad relación inmobiliaria



Descripción MER inmobiliaria

El diagrama de la inmobiliaria presenta entidades las cuales se convirtieron en tablas, atributos, relaciones, cardinalidades. En la tabla 1 se describe de la siguiente manera, en la columna tablas se describen las tablas con los respectivos atributos, en la columna relaciones se describen las tablas con las cuales se relaciona con la respectiva cardinalidad.

Tabla 1: Descripción MER inmobiliaria.

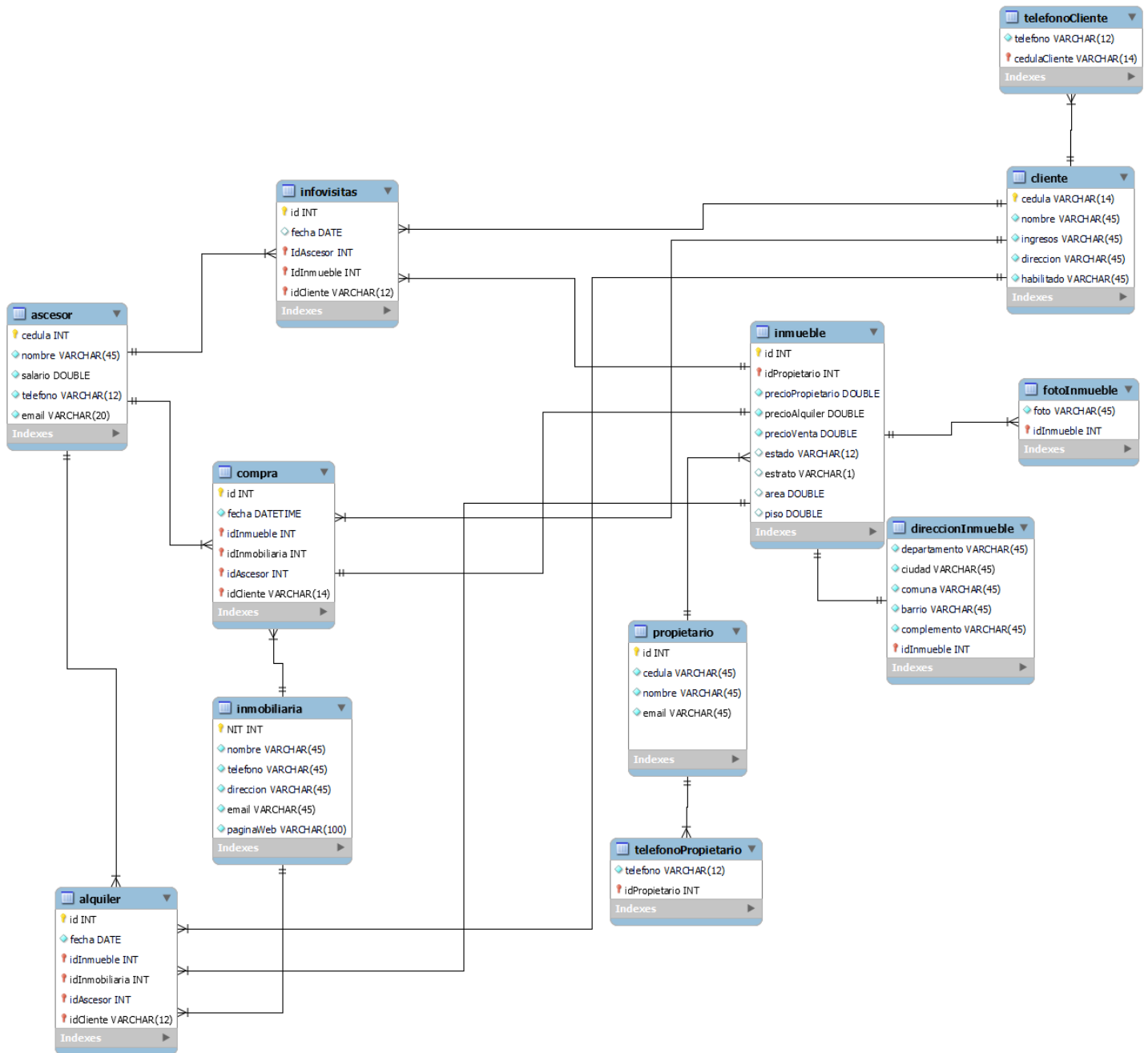
Tablas	Atributos	Relaciones	Cardinalidad
Cliente	cedula, nombre, teléfono (a. multivariado), email, ingresos, dirección, habilitado.	Infovisitas	Un cliente puede realizar una o muchas visitas.
		alquiler	Un cliente puede realizar uno o muchos alquileres.
		compra	Un cliente puede realizar una o más compras.
		telefonoCliente	Un cliente puede tener uno o más teléfonos.
Infovistas	Id, fechaInfovisitas, idInmueble, idInmobiliaria, idAsesor, idCliente.	Asesor	Una infovisita se relaciona con un asesor.
		inmueble	Una infovisita se relaciona con un inmueble.
		cliente	Una infovisita es realizada por un cliente.
alquiler	Id, fechaAlquiler, idInmueble, idInmobiliaria, idAsesor, idCliente.	Inmueble	Un alquiler esta relacionado con un inmueble.
		inmobiliaria	Un alquiler esta relacionado con una inmobiliaria.
		cliente	Un alquiler esta relacionado con un cliente.
compra	Id, fechaCompra, idInmueble, idInmobiliaria, idAsesor, idCliente.	Asesor	Una compra esta relacionada con un asesor.
		Inmueble	Una compra está relacionada con un inmueble.
		inmobiliaria	Una compra esta relacionada con una inmobiliaria.
		asesor	Una compra está relacionada con un asesor.
		cliente	Una compra esta relacionada con un cliente.
Asesor	cedula, nombre, salario, teléfono, correo	Infovisitas	Un asesor esta relacionado con una o más infovisitas.
		alquiler	Un asesor esta relacionado con uno o más alquileres.
		compra.	Un asesor está relacionado con una o más compras.
Inmueble	Id, idPropietario, precioPropietario, precioAlquiler, precioVenta, estado, estrato, área, piso	Propietario	Un inmueble pertenece a un propietario.
		infovisitas	Un inmueble puede recibir una o más infovisitas.
		compra	Un inmueble puede ser comprado una vez.
		propietario	Un inmueble le pertenece a un propietario.
		direccionInmueble	Un inmueble esta relacionado con una dirección.
		fotoInmueble	Un inmueble puede tener una o más fotos.
Propietario	cedula, teléfono(mv), nombre, email.	Inmueble	Un propietario puede tener uno o más inmuebles.
		telefonoPropietario	Un propietario puede tener uno o más teléfonos.
inmobiliaria	nombre, dirección, teléfono, correo, página web	Compra	Una inmobiliaria está relacionada con una o más compras
		alquilar	Una inmobiliaria esta relacionada con uno o más alquileres.

De los atributos multivariados se establecieron las siguientes tablas

Tabla 2:

Tablas	Atributos	Relaciones	cardinalidad
telefonoPropietario	Teléfono, idPropietario	Propietario	Uno o más teléfonos le pertenece a un propietario.
telefonoCliente	Teléfono, idCliente	Cliente	Unos o más teléfonos le pertenecen a un cliente.

Diagrama MER inmobiliaria



Script corregido para crear tablas de base de datos inmobiliaria.

```
-- Table `inmobiliaria`.`cliente`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`cliente` (
  `cedula` VARCHAR(14) NOT NULL,
  `nombre` VARCHAR(45) NOT NULL,
  `ingresos` VARCHAR(45) NOT NULL,
  `direccion` VARCHAR(45) NOT NULL,
  `habilitado` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`cedula`))
ENGINE = InnoDB;
```

```

-----
-- Table `inmobiliaria`.`telefonoCliente`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`telefonoCliente` (
  `telefono` VARCHAR(12) NOT NULL,
  `cedulaCliente` VARCHAR(14) NOT NULL,
  FOREIGN KEY (`cedulaCliente`) REFERENCES `inmobiliaria`.`cliente` (`cedula`)
)
ENGINE = InnoDB;
INSERT INTO inmobiliaria.telefonoCliente (telefono, cedulaCliente)
VALUES ("528909", "90009");

-----
-- Table `inmobiliaria`.`asesor`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`asesor` (
  `cedula` INT NOT NULL,
  `nombre` VARCHAR(45) NOT NULL,
  `salario` DOUBLE NOT NULL,
  `telefono` VARCHAR(12) NOT NULL,
  `email` VARCHAR(20) NOT NULL,
  PRIMARY KEY (`cedula`))
ENGINE = InnoDB;

-----
-- Table `inmobiliaria`.`propietario`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`propietario` (
  `id` INT NOT NULL,
  `cedula` VARCHAR(45) NOT NULL,
  `nombre` VARCHAR(45) NOT NULL,
  `email` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`id`))
ENGINE = InnoDB;

-----
-- Table `inmobiliaria`.`inmueble`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`inmueble` (
  `id` INT NOT NULL,
  `idPropietario` INT NOT NULL,
  `precioPropietario` DOUBLE NOT NULL,
  `precioAlquiler` DOUBLE NOT NULL,
  `precioVenta` DOUBLE NOT NULL,
  `estado` VARCHAR(12) NULL,
  `estrato` VARCHAR(1) NULL,
  `area` DOUBLE NULL,
  `piso` DOUBLE NULL,
  PRIMARY KEY (`id`),
  FOREIGN KEY (`idPropietario`) REFERENCES `inmobiliaria`.`propietario` (`id`)
)
ENGINE = InnoDB;

```

```

-----
-- Table `inmobiliaria`.`fotoInmueble`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`fotoInmueble` (
  `foto` VARCHAR(45) NOT NULL,
  `idInmueble` INT NOT NULL,
  FOREIGN KEY (`idInmueble`) REFERENCES `inmobiliaria`.`inmueble` (`id`)
)
ENGINE = InnoDB;
INSERT INTO inmobiliaria.fotoInmueble (foto, idInmueble)
VALUES ("foto2", "2");
-----
-- Table `inmobiliaria`.`direccionInmueble`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`direccionInmueble` (
  `departamento` VARCHAR(45) NOT NULL,
  `ciudad` VARCHAR(45) NOT NULL,
  `comuna` VARCHAR(45) NOT NULL,
  `barrio` VARCHAR(45) NOT NULL,
  `complemento` VARCHAR(45) NOT NULL,
  `idInmueble` INT NOT NULL,
  FOREIGN KEY (`idInmueble`) REFERENCES `inmobiliaria`.`inmueble` (`id`))
ENGINE = InnoDB;
-----
-- Table `inmobiliaria`.`telefonoPropietario`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`telefonoPropietario` (
  `telefono` VARCHAR(12) NOT NULL,
  `idPropietario` INT NOT NULL,
  FOREIGN KEY (`idPropietario`) REFERENCES `inmobiliaria`.`propietario` (`id`)
)
ENGINE = InnoDB;
-----
-- Table `inmobiliaria`.`inmobiliaria`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`inmobiliaria` (
  `NIT` INT NOT NULL,
  `nombre` VARCHAR(45) NOT NULL,
  `telefono` VARCHAR(45) NOT NULL,
  `direccion` VARCHAR(45) NOT NULL,
  `email` VARCHAR(45) NOT NULL,
  `paginaWeb` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`NIT`))
ENGINE = InnoDB;
-----
-- Table `inmobiliaria`.`compra`
-----
CREATE TABLE IF NOT EXISTS `inmobiliaria`.`compra` (

```

```

`id` INT NOT NULL,
`fecha` DATETIME NOT NULL,
`idInmueble` INT NOT NULL,
`idInmobiliaria` INT NOT NULL,
`idAsesor` INT NOT NULL,
`idCliente` VARCHAR(14) NOT NULL,
PRIMARY KEY (`id`),
  FOREIGN KEY (`idInmueble`) REFERENCES `inmobiliaria`.`inmueble` (`id`),
  FOREIGN KEY (`idInmobiliaria`) REFERENCES `inmobiliaria`.`inmobiliaria` (`NIT`),
  FOREIGN KEY (`idAsesor`) REFERENCES `inmobiliaria`.`asesor` (`cedula`),
  FOREIGN KEY (`idCliente`) REFERENCES `inmobiliaria`.`cliente` (`cedula`)
)
ENGINE = InnoDB;

```

```

-----
-- Table `inmobiliaria`.`alquiler`
-----

```

```

CREATE TABLE IF NOT EXISTS `inmobiliaria`.`alquiler` (
  `id` INT NOT NULL,
  `fecha` DATE NOT NULL,
  `idInmueble` INT NOT NULL,
  `idInmobiliaria` INT NOT NULL,
  `idAsesor` INT NOT NULL,
  `idCliente` VARCHAR(12) NOT NULL,
  PRIMARY KEY (`id`),
  FOREIGN KEY (`idInmueble`) REFERENCES `inmobiliaria`.`inmueble` (`id`),
  FOREIGN KEY (`idInmobiliaria`) REFERENCES `inmobiliaria`.`inmobiliaria` (`NIT`),
  FOREIGN KEY (`idAsesor`) REFERENCES `inmobiliaria`.`asesor` (`cedula`),
  FOREIGN KEY (`idCliente`) REFERENCES `inmobiliaria`.`cliente` (`cedula`)
)
ENGINE = InnoDB;

```

```

-----
-- Table `inmobiliaria`.`infovisitas`
-----

```






```

CREATE TABLE IF NOT EXISTS `inmobiliaria`.`infovisitas` (
  `id` INT NOT NULL,
  `fecha` DATE NULL,
  `IdAsesor` INT NOT NULL,
  `IdInmueble` INT NOT NULL,
  `idCliente` VARCHAR(12) NOT NULL,
  PRIMARY KEY (`id`),
  FOREIGN KEY (`IdAsesor`) REFERENCES `inmobiliaria`.`asesor` (`cedula`),
  FOREIGN KEY (`IdInmueble`) REFERENCES `inmobiliaria`.`inmueble` (`id`),
  FOREIGN KEY (`idCliente`) REFERENCES `inmobiliaria`.`cliente` (`cedula`)
)
ENGINE = InnoDB;

```

Script con datos

SELECT * FROM inmobiliaria.cliente;

Result Grid				Filter Rows:	<input type="text"/>	Edit:	
	cedula	nombre	ingresos	direccion		habilitado	
	11111	Andres	45670000	cr 45 - 56		si	
	12345	Juan	20000000	cr 2a 45 -5		si	
	23456	Ana	30000000	cr 34 45-3		si	
	45678	Yolima	34900000	cr 34 78-45		si	
	90009	Salome	45678900	cr 34-67		si	
	NULL	NULL	NULL	NULL		NULL	

SELECT * FROM inmobiliaria.telefonocliente;

telefono	cedulaCliente
123456	11111
523456	12345
523499	90009
523699	90009
528909	90009

SELECT * FROM inmobiliaria.asesor;

cedula	nombre	salario	telefono	email
120003	Alejan...	700000	505050	a@gmail.com
123000	Daniel	300000	909090	d@gmail.com
132000	Vicky	800000	606060	v@gmail.com
213000	Edin	400000	707070	e@gmail.com
321000	Rosa	300000	808080	r@gmail.com
NULL	NULL	NULL	NULL	NULL

SELECT * FROM inmobiliaria.propietario;

id	cedula	nombre	email
1	1010	Pedro	pedro@gmail.com
2	2020	Maria	Maria@gmail.com
3	3030	Gilbert	g@gmail.com
4	4040	Angela	an@gmail.com
5	3030	Jenny	ail.com
NULL	NULL	NULL	NULL

SELECT * FROM inmobiliaria.inmueble;

id	idPropietario	precioPropietario	precioAlquiler	precioVenta	estado	estrato	area	piso
1	1	6700000	5700000	6600000	activo	3	22	1
2	2	5600000	500000	5500000	activo	3	10	2
3	3	4400000	450000	5660000	activo	5	12	1
4	4	5600000	670000	7800000	activo	4	11	1
5	5	6700000	440000	4400000	activo	5	8	5
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

SELECT * FROM inmobiliaria.fotoinmueble;

foto	idInmueble
foto1	1
foto2	1
foto3	1
foto1	2
foto2	2

SELECT * FROM inmobiliaria.inmobiliaria;

	NIT	nombre	telefono	direccion	email	paginaWeb
	12	casasur	123	cr 34_34	cas...	www.casa...
	13	casan...	321	cr 23-34	nort...	www.norte
	14	tucasa	432	cr 34-23	tc@...	www.tucasa
	15	miCas...	345	cr 45-34-9	mc...	www.casaya
▶	16	lotes	900	cr 45-89	lote...	www.lotes
▲	NULL	NULL	NULL	NULL	NULL	NULL

SELECT * FROM inmobiliaria.telefonopropietario;

	telefono	idPropietario
▶	50009	1
	50008	1
	30008	3
	40008	3
	49000	5

SELECT * FROM inmobiliaria.infovisitas;

	id	fecha	IdAsesor	IdInmueble	idCliente
	1	2022-03-09	321000	1	90009
	2	2022-03-08	120003	2	11111
	3	2022-03-07	120003	3	12345
	4	2022-03-06	123000	4	45678
▶	5	2022-03-09	321000	5	23456
▲	NULL	NULL	NULL	NULL	NULL

SELECT * FROM inmobiliaria.compra;

	id	fecha	idInmueble	idInmobiliaria	idAsesor	idCliente
	1	2023...	1	12	123000	11111
	2	2023...	2	13	120003	12345
	3	2023...	3	14	321000	23456
	4	2023...	4	15	213000	45678
▶	5	2023...	5	16	132000	90009
▲	NULL	NULL	NULL	NULL	NULL	NULL

SELECT * FROM inmobiliaria.alquiler;

	id	fecha	idInmueble	idInmobiliaria	idAsesor	idCliente
	1	2023-02-11	1	12	123000	11111
	2	2023-02-11	2	13	120003	12345
	3	2023-02-10	2	13	321000	23456
	4	2023-02-12	3	14	213000	45678
▶	5	2023-02-13	3	14	132000	90009
▲	NULL	NULL	NULL	NULL	NULL	NULL

alquiler 1 ▼

Respuesta a preguntas

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

Se debería incluir una tabla con los siguientes atributos:

	Atributo	Tipo de dato
	Fecha de Inicio de contrato	Date
	Fecha de Fin de contrato	Date
	Estado	Boolean
FK	idAlquiler	Int

Las fechas de inicio y fin de contrato definen el tiempo de vigencia de contrato, de allí en el atributo estado pasa a ser true mientras el contrato sea vigente y falso mientras el contrato requiera renovación.

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? Puede ser asignando una tabla denominada compraCliente con los siguientes atributos, de tal manera que cuando el id del inmueble se encuentre almacenado en la tabla compraCliente pase a un estado inactivo dentro de un atributo estado en la tabla nmuebles.

	Atributo	Tipo de dato
	Fecha de compra	Date
FK	idCliente	int
FK	idInmueble	Int