

Santiago Ramirez Arenas

Mer Inmobiliaria SofkaU

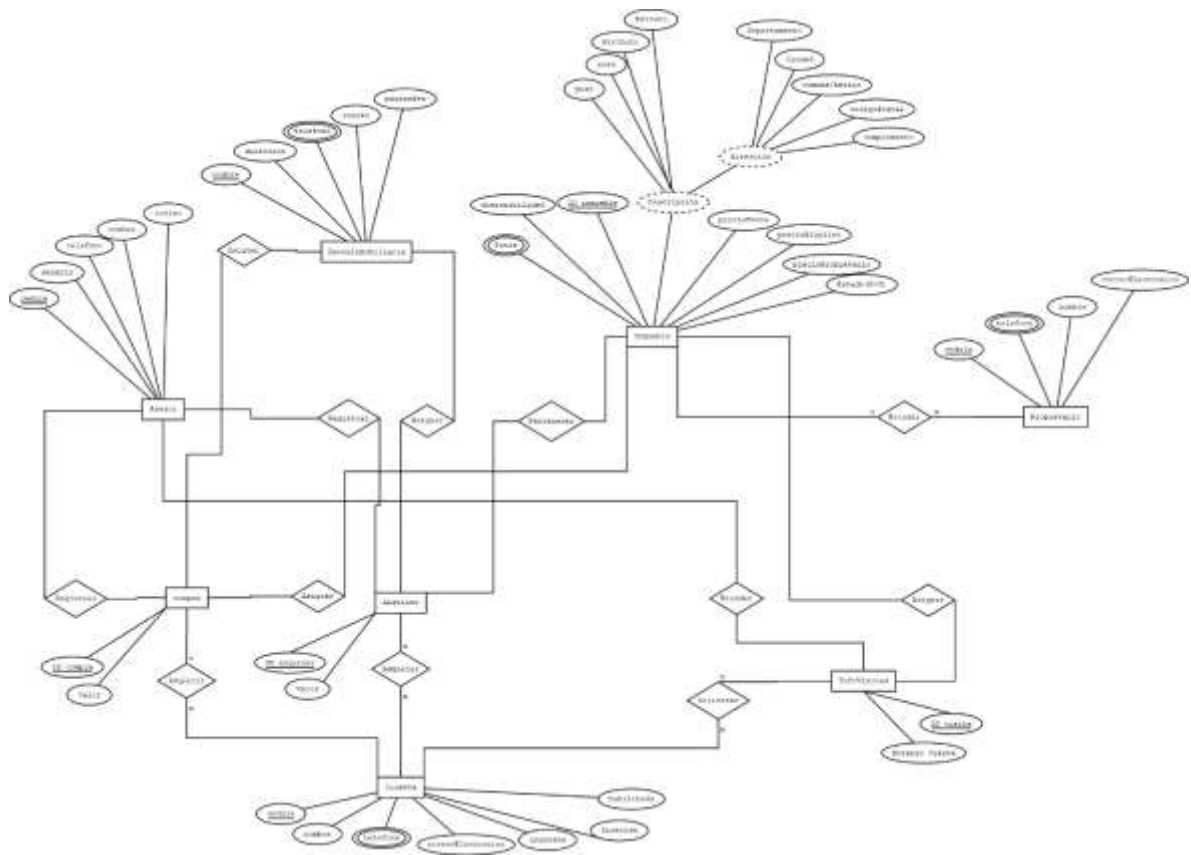
Requerimientos:

- Complete ***unicamente*** 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.
- Gere el Script aplicando ingeniería hacia adelante.
- Corra el Script y genere la base de datos.
- Poblar todas las tabla de la base de datos al menos con 5 registros.
- Después de haber realizado todo lo anterior mencionado responda:
 - ¿Qué debería cambiar o agregar para incluir la renovación de contratos de 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?
- 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

Primero organizo el MER proporcionado ya que es muy complicado entenderlo de la forma en la que estaba.

Imagen 1: MER de la inmobiliaria organizado

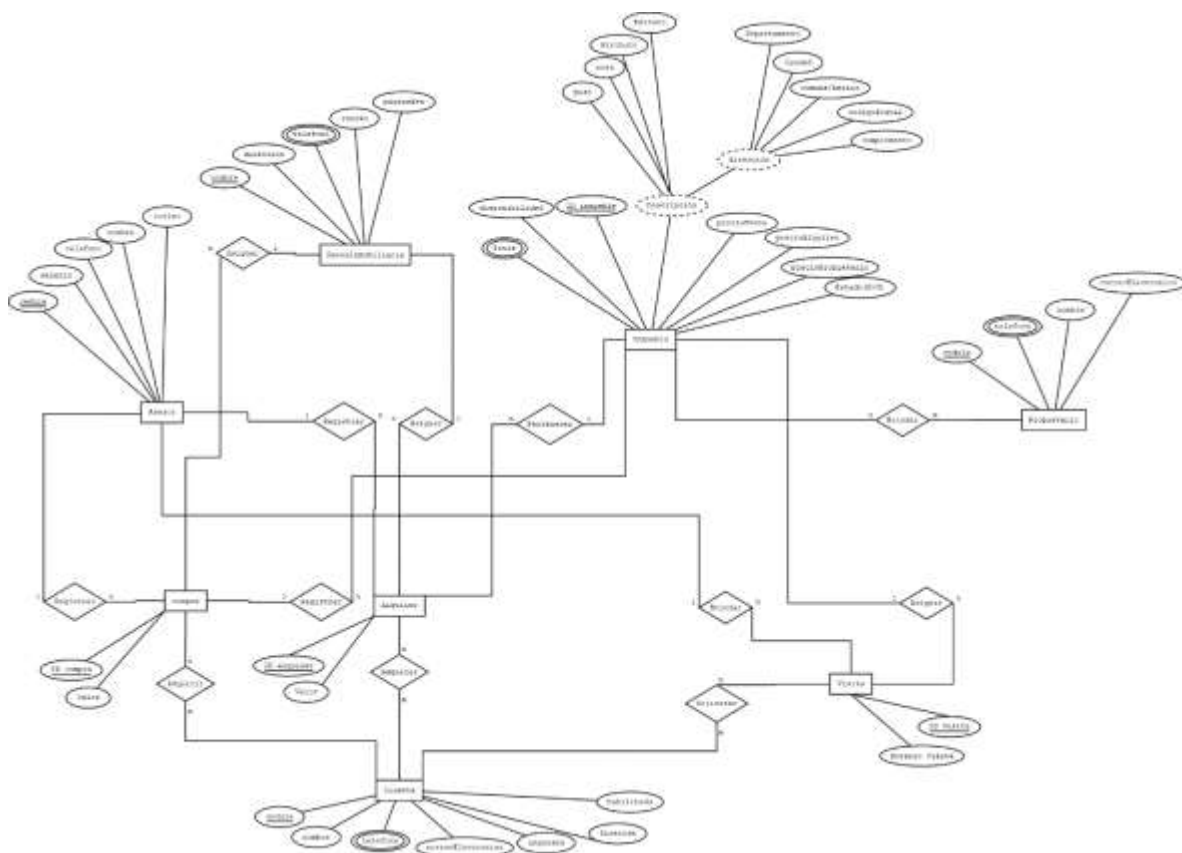


Explicación de las cardinalidades:

- Una compra puede ser registrada por un asesor, mientras que un asesor puede registrar muchas compras. (N,M)
- Un inmueble puede ser registrado por una compra, mientras que una compra puede registrar muchos inmuebles (si una persona desea adquirir múltiples propiedades al tiempo, este tiene la posibilidad de realizar una compra conjunta). (1,N)
- Un cliente puede adquirir muchos alquileres, mientras que un alquiler puede ser adquirido por muchos clientes (Si por ejemplo un núcleo familia desea adquirir en conjunto una casa, ambos pueden ser registrados como arrendatarios). (N,M)
- Un cliente puede solicitar una muchas visitas, mientras que una visita puede ser solicitada por muchos clientes (Puede darse el caso en el que varias personas interesadas en un mismo inmueble soliciten una visita para la misma fecha). (N,M)
- Un asesor puede brindar muchas visitas, mientras que una visita puede ser brindada por un solo asesor. (1,N)

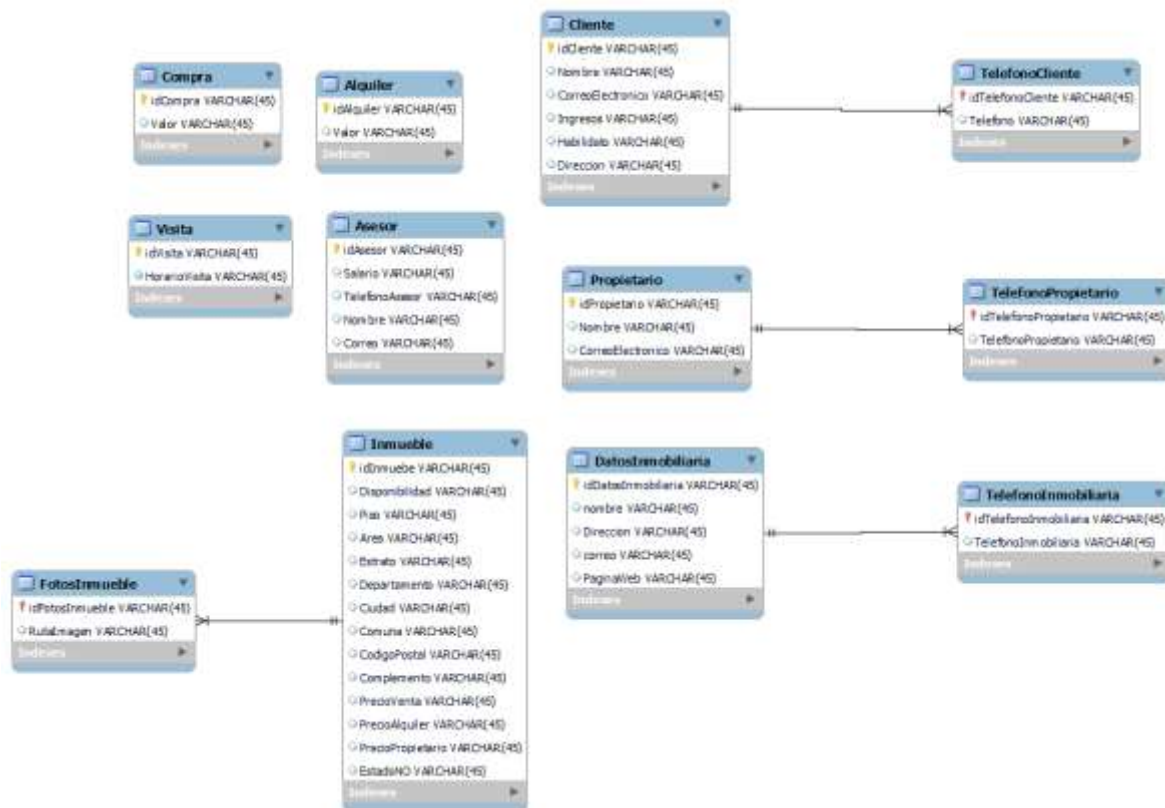
- Una visita puede ser asignada a un inmueble, mientras que un inmueble puede tener asignadas múltiples visitas (Por ejemplo, para visitas grupales de múltiples interesados). (1,N)
- Un propietario puede brindar muchos inmuebles, y un inmueble puede ser brindado por muchos propietarios (Por ejemplo, para propiedades que son patrimonio familiar y desean venderse). (N,M)
- Un inmueble puede pertenecer a muchos alquileres y un alquiler solo puede pertenecer a un inmueble (Si por ejemplo es una casa de varios pisos, bien podría alquilarse cada piso por separado). (1,N)
- Los datos de una inmobiliaria pueden ser asignados a muchos alquileres, mientras que un alquiler solo puede ser asignado a los datos de una inmobiliaria. (1,N)
- Un asesor puede registrar muchas alquileres, mientras que un alquiler solo puede ser registrado por un asesor (asumiendo que el modelo de negocios de la inmobiliaria se basa en comisiones individuales según rendimiento). (1,N)
- Los datos de la inmobiliaria pueden ser asignados a una compra, mientras que una compra solo puede ser asignada a los datos de una inmobiliaria. (1,N)
- Un cliente puede adquirir muchas compras y una compra puede ser adquirida por muchos clientes. (N,M)

Imagen 2: MER de la inmobiliaria con las cardinalidades



Después de diagramar el MER con sus cardinalidades lo siguiente fue representar las entidades con su respectiva llave primaria, además de crear las tablas para los atributos **Multivaluados** como lo son los teléfonos, además para los atributos derivados se agregaron en las tablas de su respectiva entidad

Imagen 3: Entidades representadas en Workbench



Explicación

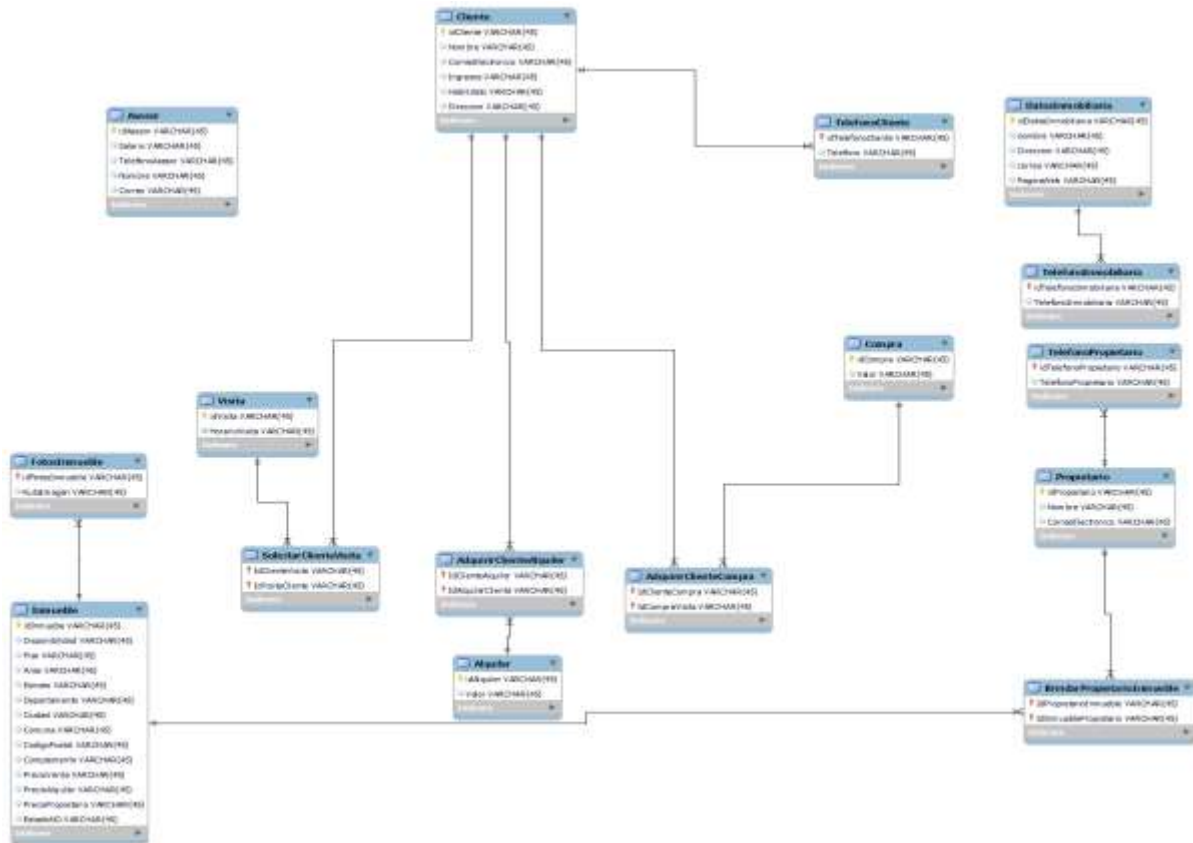
Lo siguiente es representar las cardinalidades (N,M) creando las respectivas claves foráneas que conectan las dos entidades mediante esta relación, para asignar el nombre de la tabla usé la notación que me recomendó un conocido, para llevar un mejor orden.

La tabla se nombra primero con el nombre de la relación **Brindar y** después con el nombre de las dos entidades que intervienen **Propietario Inmueble**, por lo tanto quedaría **BrindarPropietarioInmueble**.

Algo que me confunde es el nombre que le debo asignar a las llaves foráneas, ya que si dos llaves foráneas tienen el mismo nombre resulta en un error, para ejercicios que contienen muchos elementos puede resultar confuso por lo que usé la notación **IdEntidad1Entidad2**, por lo tanto

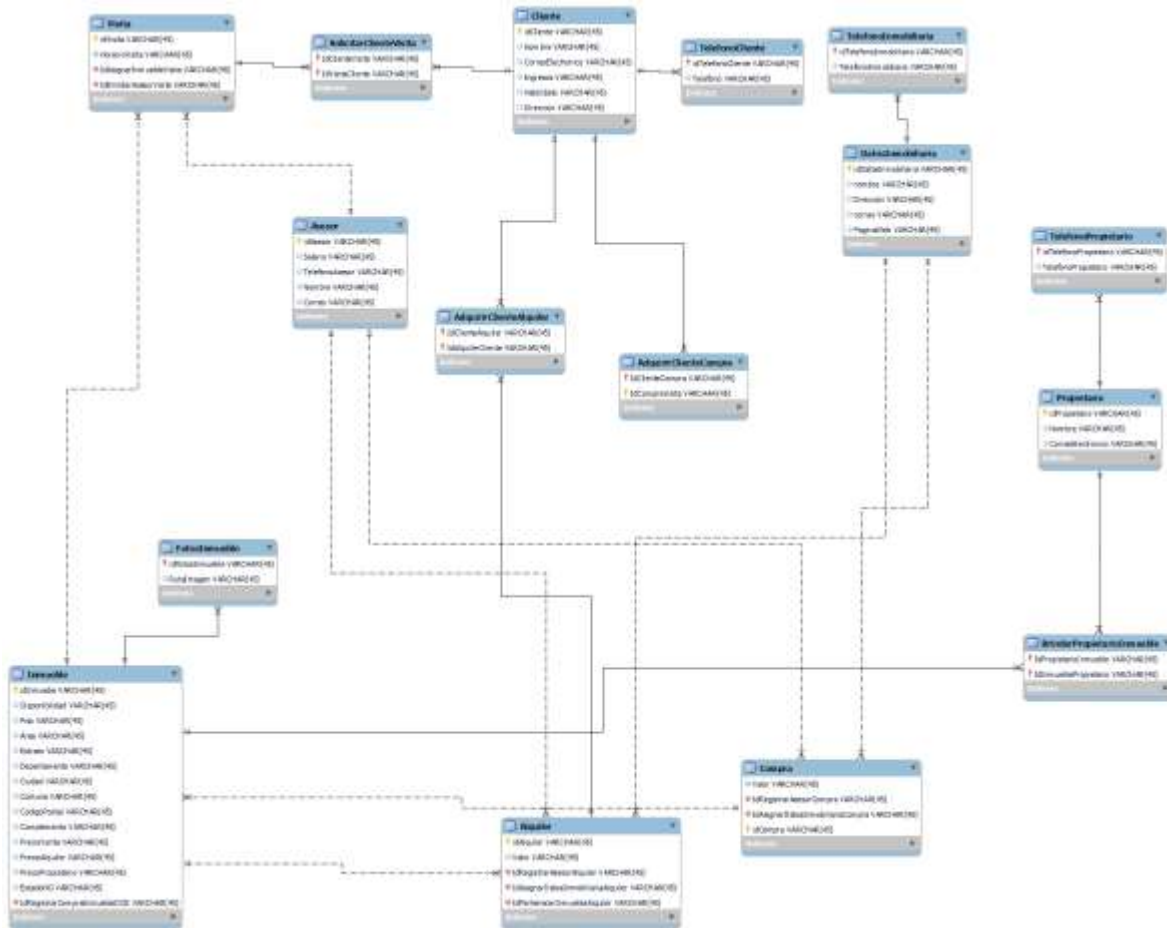
quedaría **IdPropietarioInmueble** y **IdInmueblePropietario**, indicando la primer palabra a la entidad a la cual está conectada.

Imagen 4: Representación cardinalidades (N,M) en Workbench



Para las relaciones que tienen una cardinalidad de (1,N) se le asignó la llave primaria de la entidad que tiene como alcance el 1 a la entidad que tiene como alcance N como una llave foranea. Usando la misma nomenclatura anteriormente mencionada.

Imagen 5: Representación cardinalidades (1,N) en Workbench



Código generado mediante la opción Forward Engineer de WorkBench

```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_
DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

--
-- Schema mydb
--

--
-- Schema mydb
--

CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;

--
-- Table `mydb`.`Asesor`
--

CREATE TABLE IF NOT EXISTS `mydb`.`Asesor` (
  `idAsesor` VARCHAR(45) NOT NULL,
  `Salario` VARCHAR(45) NULL,
  `TelefonoAsesor` VARCHAR(45) NULL,
  `Nombre` VARCHAR(45) NULL,
  `Correo` VARCHAR(45) NULL,
  PRIMARY KEY (`idAsesor`))
ENGINE = InnoDB;

--
-- Table `mydb`.`DatosInmobiliaria`
--

CREATE TABLE IF NOT EXISTS `mydb`.`DatosInmobiliaria` (
  `idDatosInmobiliaria` VARCHAR(45) NOT NULL,
  `nombre` VARCHAR(45) NULL,
  `Direccion` VARCHAR(45) NULL,
  `correo` VARCHAR(45) NULL,
  `PaginaWeb` VARCHAR(45) NULL,
  PRIMARY KEY (`idDatosInmobiliaria`))
ENGINE = InnoDB;

--
-- Table `mydb`.`Compra`
--

CREATE TABLE IF NOT EXISTS `mydb`.`Compra` (
  `Valor` VARCHAR(45) NOT NULL,
  `IdRegistrarAsesorCompra` VARCHAR(45) NOT NULL,
  `IdAsignarDatosInmobiliariaCompra` VARCHAR(45) NOT NULL,
```

```

    `idCompra` VARCHAR(45) NOT NULL,
    INDEX `IdRegistrarAsesorCompra_idx` (`IdRegistrarAsesorCompra` ASC)
VISIBLE,
    INDEX `IdAsignarDatosInmobiliariaCompra_idx`
(`IdAsignarDatosInmobiliariaCompra` ASC) VISIBLE,
    PRIMARY KEY (`idCompra`),
    CONSTRAINT `IdRegistrarAsesorCompra`
        FOREIGN KEY (`IdRegistrarAsesorCompra`)
        REFERENCES `mydb`.`Asesor` (`idAsesor`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    CONSTRAINT `IdAsignarDatosInmobiliariaCompra`
        FOREIGN KEY (`IdAsignarDatosInmobiliariaCompra`)
        REFERENCES `mydb`.`DatosInmobiliaria` (`idDatosInmobiliaria`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`Inmueble`
-- -----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`Inmueble` (
    `idInmuebe` VARCHAR(45) NOT NULL,
    `Disponibilidad` VARCHAR(45) NULL,
    `Piso` VARCHAR(45) NULL,
    `Area` VARCHAR(45) NULL,
    `Estrato` VARCHAR(45) NULL,
    `Departamento` VARCHAR(45) NULL,
    `Ciudad` VARCHAR(45) NULL,
    `Comuna` VARCHAR(45) NULL,
    `CodigoPostal` VARCHAR(45) NULL,
    `Complemento` VARCHAR(45) NULL,
    `PrecioVenta` VARCHAR(45) NULL,
    `PrecioAlquiler` VARCHAR(45) NULL,
    `PrecioPropietario` VARCHAR(45) NULL,
    `EstadoNO` VARCHAR(45) NULL,
    `IdRegistrarCompraInmuebleDOS` VARCHAR(45) NOT NULL,
    PRIMARY KEY (`idInmuebe`),
    INDEX `IdRegistrarCompraInmuebleDOS_idx`
(`IdRegistrarCompraInmuebleDOS` ASC) VISIBLE,
    CONSTRAINT `IdRegistrarCompraInmuebleDOS`
        FOREIGN KEY (`IdRegistrarCompraInmuebleDOS`)
        REFERENCES `mydb`.`Compra` (`idCompra`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`Visita`
-- -----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`Visita` (

```



```

`idVisita` VARCHAR(45) NOT NULL,
`HorarioVisita` VARCHAR(45) NOT NULL,
`IdAsignarInmuebleVisita` VARCHAR(45) NOT NULL,
`IdBrindarAsesorVisita` VARCHAR(45) NOT NULL,
PRIMARY KEY (`idVisita`),
INDEX `IdAsignarInmuebleVisita_idx` (`IdAsignarInmuebleVisita` ASC)
VISIBLE,
INDEX `IdBrindarAsesorVisita_idx` (`IdBrindarAsesorVisita` ASC)
VISIBLE,
CONSTRAINT `IdAsignarInmuebleVisita`
FOREIGN KEY (`IdAsignarInmuebleVisita`)
REFERENCES `mydb`.`Inmueble` (`idInmuebe`)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `IdBrindarAsesorVisita`
FOREIGN KEY (`IdBrindarAsesorVisita`)
REFERENCES `mydb`.`Asesor` (`idAsesor`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`Cliente`
-- -----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`Cliente` (
  `idCliente` VARCHAR(45) NOT NULL,
  `Nombre` VARCHAR(45) NULL,
  `CorreoElectronico` VARCHAR(45) NULL,
  `Ingresos` VARCHAR(45) NULL,
  `Habilitado` VARCHAR(45) NULL,
  `Direccion` VARCHAR(45) NULL,
  PRIMARY KEY (`idCliente`))
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`TelefonoCliente`
-- -----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`TelefonoCliente` (
  `idTelefonoCliente` VARCHAR(45) NOT NULL,
  `Telefono` VARCHAR(45) NULL,
  PRIMARY KEY (`idTelefonoCliente`),
  CONSTRAINT `idTelefonoCliente`
FOREIGN KEY (`idTelefonoCliente`)
REFERENCES `mydb`.`Cliente` (`idCliente`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-- -----
-- Table `mydb`.`TelefonoInmobiliaria`
-- -----

```

```

-----
CREATE TABLE IF NOT EXISTS `mydb`.`TelefonoInmobiliaria` (
  `idTelefonoInmobiliaria` VARCHAR(45) NOT NULL,
  `TelefonoInmobiliaria` VARCHAR(45) NULL,
  PRIMARY KEY (`idTelefonoInmobiliaria`),
  CONSTRAINT `idTelefonoInmobiliaria`
    FOREIGN KEY (`idTelefonoInmobiliaria`)
      REFERENCES `mydb`.`DatosInmobiliaria` (`idDatosInmobiliaria`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`FotosInmueble`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`FotosInmueble` (
  `idFotosInmueble` VARCHAR(45) NOT NULL,
  `RutaImagen` VARCHAR(45) NULL,
  PRIMARY KEY (`idFotosInmueble`),
  CONSTRAINT `idFotosInmueble`
    FOREIGN KEY (`idFotosInmueble`)
      REFERENCES `mydb`.`Inmueble` (`idInmuebe`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`Propietario`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`Propietario` (
  `idPropietario` VARCHAR(45) NOT NULL,
  `Nombre` VARCHAR(45) NULL,
  `CorreoElectronico` VARCHAR(45) NULL,
  PRIMARY KEY (`idPropietario`))
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`TelefonoPropietario`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`TelefonoPropietario` (
  `idTelefonoPropietario` VARCHAR(45) NOT NULL,
  `TelefonoPropietario` VARCHAR(45) NULL,
  PRIMARY KEY (`idTelefonoPropietario`),
  CONSTRAINT `idTelefonoPropietario`
    FOREIGN KEY (`idTelefonoPropietario`)
      REFERENCES `mydb`.`Propietario` (`idPropietario`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```
-- Table `mydb`.`Alquiler`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Alquiler` (  
  `idAlquiler` VARCHAR(45) NOT NULL,  
  `Valor` VARCHAR(45) NULL,  
  `IdRegistrarAsesorAlquiler` VARCHAR(45) NOT NULL,  
  `IdAsignarDatosInmobiliariaAlquiler` VARCHAR(45) NOT NULL,  
  `IdPertenerInmuebleAlquiler` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`idAlquiler`),  
  INDEX `IdRegistrarAsesorAlquiler_idx` (`IdRegistrarAsesorAlquiler` ASC)  
  VISIBLE,  
  INDEX `IdAsignarDatosInmobiliariaAlquiler_idx`  
  (`IdAsignarDatosInmobiliariaAlquiler` ASC) VISIBLE,  
  INDEX `IdPertenerInmuebleAlquiler_idx`  
  (`IdPertenerInmuebleAlquiler` ASC) VISIBLE,  
  CONSTRAINT `IdRegistrarAsesorAlquiler`  
    FOREIGN KEY (`IdRegistrarAsesorAlquiler`)  
    REFERENCES `mydb`.`Asesor` (`idAsesor`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `IdAsignarDatosInmobiliariaAlquiler`  
    FOREIGN KEY (`IdAsignarDatosInmobiliariaAlquiler`)  
    REFERENCES `mydb`.`DatosInmobiliaria` (`idDatosInmobiliaria`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `IdPertenerInmuebleAlquiler`  
    FOREIGN KEY (`IdPertenerInmuebleAlquiler`)  
    REFERENCES `mydb`.`Inmueble` (`idInmuebe`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- Table `mydb`.`SolicitarClienteVisita`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`SolicitarClienteVisita` (  
  `IdClienteVisita` VARCHAR(45) NOT NULL,  
  `IdVisitaCliente` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`IdClienteVisita`, `IdVisitaCliente`),  
  INDEX `IdVisitaCliente_idx` (`IdVisitaCliente` ASC) VISIBLE,  
  CONSTRAINT `IdClienteVisita`  
    FOREIGN KEY (`IdClienteVisita`)  
    REFERENCES `mydb`.`Cliente` (`idCliente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `IdVisitaCliente`  
    FOREIGN KEY (`IdVisitaCliente`)  
    REFERENCES `mydb`.`Visita` (`idVisita`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)
```

```
ENGINE = InnoDB;
```

```
-- Table `mydb`.`AdquirirClienteAlquiler`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`AdquirirClienteAlquiler` (  
  `IdClienteAlquiler` VARCHAR(45) NOT NULL,  
  `IdAlquilerCliente` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`IdClienteAlquiler`, `IdAlquilerCliente`),  
  INDEX `IdAlquilerCliente_idx` (`IdAlquilerCliente` ASC) VISIBLE,  
  CONSTRAINT `IdClienteAlquiler`  
    FOREIGN KEY (`IdClienteAlquiler`)  
    REFERENCES `mydb`.`Cliente` (`idCliente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `IdAlquilerCliente`  
    FOREIGN KEY (`IdAlquilerCliente`)  
    REFERENCES `mydb`.`Alquiler` (`idAlquiler`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- Table `mydb`.`AdquirirClienteCompra`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`AdquirirClienteCompra` (  
  `IdClienteCompra` VARCHAR(45) NOT NULL,  
  `IdCompraVisita` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`IdClienteCompra`, `IdCompraVisita`),  
  CONSTRAINT `IdClienteCompra`  
    FOREIGN KEY (`IdClienteCompra`)  
    REFERENCES `mydb`.`Cliente` (`idCliente`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
-- Table `mydb`.`BrindarPropietarioInmueble`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`BrindarPropietarioInmueble` (  
  `IdPropietarioInmueble` VARCHAR(45) NOT NULL,  
  `IdInmueblePropietario` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`IdPropietarioInmueble`, `IdInmueblePropietario`),  
  INDEX `IdInmueblePropietario_idx` (`IdInmueblePropietario` ASC)  
  VISIBLE,  
  CONSTRAINT `IdPropietarioInmueble`  
    FOREIGN KEY (`IdPropietarioInmueble`)  
    REFERENCES `mydb`.`Propietario` (`idPropietario`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,
```

```

CONSTRAINT `IdInmueblePropietario`
  FOREIGN KEY (`IdInmueblePropietario`)
  REFERENCES `mydb`.`Inmueble` (`idInmuebe`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

INSERCIÓN DE REGISTROS

```

INSERT INTO `mydb`.`telefonocliente` (`IdClienteCliente`, `telefono`)
VALUES ('1', '3105986251');
INSERT INTO `mydb`.`telefonocliente` (`IdClienteCliente`, `telefono`)
VALUES ('2', '3105986252');
INSERT INTO `mydb`.`telefonocliente` (`IdClienteCliente`, `telefono`)
VALUES ('3', '3105986253');
INSERT INTO `mydb`.`telefonocliente` (`IdClienteCliente`, `telefono`)
VALUES ('4', '3105986254');
INSERT INTO `mydb`.`telefonocliente` (`IdClienteCliente`, `telefono`)
VALUES ('5', '3105986255');

INSERT INTO `mydb`.`inmueble` (`idInmuebe`, `Disponibilidad`, `Piso`,
`Area`, `Estrato`, `Departamento`, `Ciudad`, `Comuna`, `CodigoPostal`,
`Complemento`, `PrecioVenta`, `PrecioAlquiler`, `PrecioPropietario`)
VALUES ('1', 'si', '1', '23', '4', '2', 'Pereira', 'San cristobla', '4',
'No hay', '2222', '111', '111');
INSERT INTO `mydb`.`inmueble` (`idInmuebe`, `Disponibilidad`, `Piso`,
`Area`, `Estrato`, `Departamento`, `Ciudad`, `Comuna`, `CodigoPostal`,
`Complemento`, `PrecioVenta`, `PrecioAlquiler`, `PrecioPropietario`)
VALUES ('2', 'si', '2', '22', '4', '1', 'Bogota', 'Las brias', '3', 'No
hay', '2222', '111', '112');
INSERT INTO `mydb`.`inmueble` (`idInmuebe`, `Disponibilidad`, `Piso`,
`Area`, `Estrato`, `Departamento`, `Ciudad`, `Comuna`, `CodigoPostal`,
`Complemento`, `PrecioVenta`, `PrecioAlquiler`, `PrecioPropietario`)
VALUES ('3', 'si', '3', '46', '3', '2', 'Armedia', 'Montesuma', '3', 'no
hay', '222', '102', '122');
INSERT INTO `mydb`.`inmueble` (`idInmuebe`, `Disponibilidad`, `Piso`,
`Area`, `Estrato`, `Departamento`, `Ciudad`, `Comuna`, `CodigoPostal`,
`Complemento`, `PrecioVenta`, `PrecioAlquiler`, `PrecioPropietario`)
VALUES ('4', 'si', '4', '24', '2', '3', 'Pereira', 'Monte', '2', 'no
hay', '2333', '12', '2212');
INSERT INTO `mydb`.`inmueble` (`idInmuebe`, `Disponibilidad`, `Piso`,
`Area`, `Estrato`, `Departamento`, `Ciudad`, `Comuna`, `CodigoPostal`,
`Complemento`, `PrecioVenta`, `PrecioAlquiler`, `PrecioPropietario`)

```

```
VALUES ('5', 'si', '5', '35', '1', 'e', 'Pereira', 'Viento', '1', 'no  
hay', '100', '11', '11223');
```

```
INSERT INTO `mydb`.`propietario` (`idPropietario`, `Nombre`,  
`CorreoElectronico`) VALUES ('1', 'Pepe1', 'ejemplo@gmail.com');  
INSERT INTO `mydb`.`propietario` (`idPropietario`, `Nombre`,  
`CorreoElectronico`) VALUES ('2', 'Pepe2', 'ejemplo@gmail.com');  
INSERT INTO `mydb`.`propietario` (`idPropietario`, `Nombre`,  
`CorreoElectronico`) VALUES ('3', 'Pepe3', 'ejemplo@gmail.com');  
INSERT INTO `mydb`.`propietario` (`idPropietario`, `Nombre`,  
`CorreoElectronico`) VALUES ('4', 'Pepe4', 'ejemplo@gmail.com');  
INSERT INTO `mydb`.`propietario` (`idPropietario`, `Nombre`,  
`CorreoElectronico`) VALUES ('5', 'Pepe5', 'ejemplo@gmail.com');
```

```
INSERT INTO `mydb`.`asesor` (`idAsesor`, `Salario`, `TelefonoAsesor`,  
`Nombre`, `Correo`) VALUES ('1', '100', '322716201', 'Pepe1',  
'ejemplo@gmail.com');  
INSERT INTO `mydb`.`asesor` (`idAsesor`, `Salario`, `TelefonoAsesor`,  
`Nombre`, `Correo`) VALUES ('2', '200', '322716202', 'Pepe2',  
'ejemplo@gm');  
INSERT INTO `mydb`.`asesor` (`idAsesor`, `Salario`, `TelefonoAsesor`,  
`Nombre`, `Correo`) VALUES ('3', '300', '322716203', 'Pepe3',  
'ejemplo@gm');  
INSERT INTO `mydb`.`asesor` (`idAsesor`, `Salario`, `TelefonoAsesor`,  
`Nombre`, `Correo`) VALUES ('4', '400', '322716204', 'Pepe4',  
'ejemplo@gm');  
INSERT INTO `mydb`.`asesor` (`idAsesor`, `Salario`, `TelefonoAsesor`,  
`Nombre`, `Correo`) VALUES ('5', '500', '322716205', 'Pepe5',  
'ejemplo@gm');
```

```
INSERT INTO `mydb`.`visita` (`idVisita`, `HorarioVisita`,  
`IdAsignarInmuebleVisita`, `IdBrindarAsesorVisita`) VALUES ('1', '9 am',  
'1', '1');  
INSERT INTO `mydb`.`visita` (`idVisita`, `HorarioVisita`,  
`IdAsignarInmuebleVisita`, `IdBrindarAsesorVisita`) VALUES ('2', '7 am',  
'2', '2');  
INSERT INTO `mydb`.`visita` (`idVisita`, `HorarioVisita`,  
`IdAsignarInmuebleVisita`, `IdBrindarAsesorVisita`) VALUES ('3', '10 am',  
'3', '3');  
INSERT INTO `mydb`.`visita` (`idVisita`, `HorarioVisita`,  
`IdAsignarInmuebleVisita`, `IdBrindarAsesorVisita`) VALUES ('4', '12  
am', '4', '4');  
INSERT INTO `mydb`.`visita` (`idVisita`, `HorarioVisita`,  
`IdAsignarInmuebleVisita`, `IdBrindarAsesorVisita`) VALUES ('5', '11 pm',  
'5', '5');
```

```

INSERT INTO `mydb`.`fotosinmueble` (`idFotosInmueble`, `RutaImagen`)
VALUES ('1', '12292');
INSERT INTO `mydb`.`fotosinmueble` (`idFotosInmueble`, `RutaImagen`)
VALUES ('2', '12292');
INSERT INTO `mydb`.`fotosinmueble` (`idFotosInmueble`, `RutaImagen`)
VALUES ('3', '12292');
INSERT INTO `mydb`.`fotosinmueble` (`idFotosInmueble`, `RutaImagen`)
VALUES ('4', '12292');
INSERT INTO `mydb`.`fotosinmueble` (`idFotosInmueble`, `RutaImagen`)
VALUES ('5', '12292');

```

```

INSERT INTO `mydb`.`datosinmobiliaria` (`idDatosInmobiliaria`, `nombre`,
`Direccion`, `correo`, `PaginaWeb`) VALUES ('1', 'La samaritana1',
'carrera 14', 'ejemplo@gmail.com', 'LaSamaritana.com');
INSERT INTO `mydb`.`datosinmobiliaria` (`idDatosInmobiliaria`, `nombre`,
`Direccion`, `correo`, `PaginaWeb`) VALUES ('2', 'La samaritana2',
'carrera 15', 'ejemplo@gmail.com', 'LaSamaritana.com');
INSERT INTO `mydb`.`datosinmobiliaria` (`idDatosInmobiliaria`, `nombre`,
`Direccion`, `correo`, `PaginaWeb`) VALUES ('3', 'La samaritana3',
'carrera 16', 'ejemplo@gmail.com', 'LaSamaritana.com');
INSERT INTO `mydb`.`datosinmobiliaria` (`idDatosInmobiliaria`, `nombre`,
`Direccion`, `correo`, `PaginaWeb`) VALUES ('4', 'La samaritana4',
'carrera 17', 'ejemplo@gmail.com', 'LaSamaritana.com');
INSERT INTO `mydb`.`datosinmobiliaria` (`idDatosInmobiliaria`, `nombre`,
`Direccion`, `correo`, `PaginaWeb`) VALUES ('5', 'La samaritana5',
'carrera 19', 'ejemplo@gmail.com', 'LaSamaritana.com');

```

```

INSERT INTO `mydb`.`telefonoinmobiliaria` (`idTelefonoInmobiliaria`,
`TelefonoInmobiliaria`) VALUES ('1', '1111122');
INSERT INTO `mydb`.`telefonoinmobiliaria` (`idTelefonoInmobiliaria`,
`TelefonoInmobiliaria`) VALUES ('2', '111123');
INSERT INTO `mydb`.`telefonoinmobiliaria` (`idTelefonoInmobiliaria`,
`TelefonoInmobiliaria`) VALUES ('3', '3313233');
INSERT INTO `mydb`.`telefonoinmobiliaria` (`idTelefonoInmobiliaria`,
`TelefonoInmobiliaria`) VALUES ('4', '12133');
INSERT INTO `mydb`.`telefonoinmobiliaria` (`idTelefonoInmobiliaria`,
`TelefonoInmobiliaria`) VALUES ('5', '124');

```

```

INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,
`IdInmueblePropietario`) VALUES ('1', '1');
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,
`IdInmueblePropietario`) VALUES ('2', '2');
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,
`IdInmueblePropietario`) VALUES ('3', '3');

```

```
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,  
`IdInmueblePropietario`) VALUES ('4', '4');  
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,  
`IdInmueblePropietario`) VALUES ('5', '5');
```

```
INSERT INTO `mydb`.`telefonopropietario` (`idTelefonoPropietario`,  
`TelefonoPropietario`) VALUES ('1', '14444444442');  
INSERT INTO `mydb`.`telefonopropietario` (`idTelefonoPropietario`,  
`TelefonoPropietario`) VALUES ('2', '132214');  
INSERT INTO `mydb`.`telefonopropietario` (`idTelefonoPropietario`,  
`TelefonoPropietario`) VALUES ('3', '14243244');  
INSERT INTO `mydb`.`telefonopropietario` (`idTelefonoPropietario`,  
`TelefonoPropietario`) VALUES ('4', '143334');  
INSERT INTO `mydb`.`telefonopropietario` (`idTelefonoPropietario`,  
`TelefonoPropietario`) VALUES ('5', '14344');
```

```
INSERT INTO `mydb`.`solicitarclientevisita` (`IdClienteVisita`,  
`IdVisitaCliente`) VALUES ('1', '1');  
INSERT INTO `mydb`.`solicitarclientevisita` (`IdClienteVisita`,  
`IdVisitaCliente`) VALUES ('2', '2');  
INSERT INTO `mydb`.`solicitarclientevisita` (`IdClienteVisita`,  
`IdVisitaCliente`) VALUES ('3', '3');  
INSERT INTO `mydb`.`solicitarclientevisita` (`IdClienteVisita`,  
`IdVisitaCliente`) VALUES ('4', '4');  
INSERT INTO `mydb`.`solicitarclientevisita` (`IdClienteVisita`,  
`IdVisitaCliente`) VALUES ('5', '5');
```

```
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,  
`IdInmueblePropietario`) VALUES ('1', '1');  
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,  
`IdInmueblePropietario`) VALUES ('2', '2');  
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,  
`IdInmueblePropietario`) VALUES ('3', '3');  
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,  
`IdInmueblePropietario`) VALUES ('4', '4');  
INSERT INTO `mydb`.`brindarpropietarioinmueble` (`IdPropietarioInmueble`,  
`IdInmueblePropietario`) VALUES ('5', '5');
```

```
INSERT INTO `mydb`.`alquiler` (`idAlquiler`, `Valor`,  
`IdRegistrarAsesorAlquiler`, `IdAsignarDatosInmobiliariaAlquiler`,  
`IdPertenerInmuebleAlquiler`) VALUES ('1', '200', '1', '1', '1');  
INSERT INTO `mydb`.`alquiler` (`idAlquiler`, `Valor`,  
`IdRegistrarAsesorAlquiler`, `IdAsignarDatosInmobiliariaAlquiler`,  
`IdPertenerInmuebleAlquiler`) VALUES ('2', '300', '2', '2', '2');
```



```

INSERT INTO `mydb`.`alquiler` (`idAlquiler`, `Valor`,
`IdRegistrarAsesorAlquiler`, `IdAsignarDatosInmobiliariaAlquiler`,
`IdPertenerInmuebleAlquiler`) VALUES ('3', '400', '3', '3', '3');
INSERT INTO `mydb`.`alquiler` (`idAlquiler`, `Valor`,
`IdRegistrarAsesorAlquiler`, `IdAsignarDatosInmobiliariaAlquiler`,
`IdPertenerInmuebleAlquiler`) VALUES ('4', '500', '4', '4', '4');
INSERT INTO `mydb`.`alquiler` (`idAlquiler`, `Valor`,
`IdRegistrarAsesorAlquiler`, `IdAsignarDatosInmobiliariaAlquiler`,
`IdPertenerInmuebleAlquiler`) VALUES ('5', '600', '5', '5', '5');

INSERT INTO `mydb`.`adquirircliente compra` (`IdClienteCompra`,
`IdCompraVisita`) VALUES ('1', '1');
INSERT INTO `mydb`.`adquirircliente compra` (`IdClienteCompra`,
`IdCompraVisita`) VALUES ('2', '2');
INSERT INTO `mydb`.`adquirircliente compra` (`IdClienteCompra`,
`IdCompraVisita`) VALUES ('3', '3');
INSERT INTO `mydb`.`adquirircliente compra` (`IdClienteCompra`,
`IdCompraVisita`) VALUES ('4', '4');
INSERT INTO `mydb`.`adquirircliente compra` (`IdClienteCompra`,
`IdCompraVisita`) VALUES ('5', '5');

```

Se adjunta captura de los datos ingresados

Imagen 5: Visualización registros

| Result Grid | | | |
|--------------|---------------|--------|-------------------|
| Filter Rows: | | | |
| | idPropietario | Nombre | CorreoElectronico |
| ▶ | 1 | Pepe1 | ejemplo@gmail.com |
| | 2 | Pepe2 | ejemplo@gmail.com |
| | 3 | Pepe3 | ejemplo@gmail.com |
| | 4 | Pepe4 | ejemplo@gmail.com |
| | 5 | Pepe5 | ejemplo@gmail.com |
| ✱ | NULL | NULL | NULL |

Imagen 6: Visualización de registros

| Result Grid | | | | | | | | | | | | |
|---|-----------|----------------|------|------|---------|--------------|---------|------------|--------------|-------------|-------------|----|
| Filter Rows: | | | | | | | | | | | | |
| Edit: Export/Import: Wrap Cell Content: | | | | | | | | | | | | |
| | idInmuebe | Disponibilidad | Piso | Area | Estrato | Departamento | Ciudad | Comuna | CodigoPostal | Complemento | PrecioVenta | P |
| | 1 | si | 1 | 23 | 4 | 2 | Pereira | San cri... | 4 | No hay | 2222 | 1: |
| | 2 | si | 2 | 22 | 4 | 1 | Bogota | Las brias | 3 | No hay | 2222 | 1: |
| ▶ | 3 | si | 3 | 46 | 3 | 2 | Armedia | Montes... | 3 | no hay | 222 | 10 |
| | 4 | si | 4 | 24 | 2 | 3 | Pereira | Monte | 2 | no hay | 2333 | 1: |
| | 5 | si | 5 | 35 | 1 | e | Pereira | Viento | 1 | no hay | 100 | 1: |
| ✱ | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NU |

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

Quizás agregaría un atributo a la entidad alquiler que se llame “Renovacion” de tipo bool y que en un inicio pueda ser Null para la primer vez que la persona adquiere un contrato, y la segunda vez que se llene la información se llenaría el campo renovación, me tocaría comprobarlo.

- ¿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?

Quizás crearía otra entidad que realice la consulta para comparar los ID si son iguales y si son iguales asignaría un valor de “No es posible “ para vetarlo del alquiler con el mismo ID