**LABORATORIO AUTORENTA**

**ZULLY FERNANDA ORTIZ AVENDAÑO**

**DOCENTE**

**PEDRO FELIPE GÓMEZ BONILLA**

**CAMPUSLANDS**

**SANDBOX  
RUTA JAVA**

**TIBU**

**2024**

[**Introducción 3**](#_ap04fxaj64vk)

[**Caso de Estudio 4**](#_x09akgyuxdnp)

[**Planificación 5**](#_8muzr1pu3mlz)

[Ejecución 5](#_fyw0g34ysmvp)

[Construcción del Modelo Conceptual 5](#_akgvn7l12xs2)

[Descripción 5](#_qxuss0va0z8)

[Las entidades y atributos 5](#_xa0q6tdkczce)

[Relaciones y Cardinalidades 7](#_61tfywmysud)

[Gráfica 8](#_u45t31dnlow2)

[Construcción del Modelo Lógico 9](#_nzoi4crbe63p)

[Descripción 9](#_46z7wrz43fjb)

[Relaciones y Cardinalidades 10](#_o7okxpgwj6gm)

[Gráfica 12](#_3bhc49u9qkwy)

[Normalización del Modelo Lógico 13](#_j975ki11pvw9)

[Primera Forma Normal (1FN) 13](#_ipsnvw5v2v0u)

[Descripción 13](#_yudex7qmv5l6)

[Descripción técnica 13](#_me85rlu9frjf)

[Gráfica 14](#_u1g2e96ymz30)

[Forma Normal (2FN) 15](#_3fyocgo001fx)

[Descripción 15](#_n31b5bkefr23)

[Descripción técnica 15](#_75tsbwacwvzn)

[Gráfica 16](#_mywvu5xf7mjo)

[Tercera Forma Normal (3FN) 17](#_aeip5fb7t3s5)

[Descripción 17](#_v6plqfy3484u)

[Descripción técnica 17](#_jsds51upy8af)

[Gráfica 18](#_bcqznd6b310i)

[Construcción del Modelo Físico 20](#_muq8ro9qbje6)

[Descripción 20](#_5g32igevz4r8)

[Construcción del Diagrama UML 23](#_fd69ln6ssi1a)

[Descripción 23](#_8ovy9d4paa4u)

[**Inserciones de Datos 26**](#_f7douwo8slk8)

[Descripción 26](#_5q2qacmjwv31)

# 

# **Introducción**

En este proyecto se presentará una guía detallada de la estructura e implementación de una base de datos para la empresa de alquiler de vehículos llamada AutoRenta, donde el objetivo principal es gestionar un sistema de información.

Primero se realizará un caso de estudio con sus requerimientos específicos, donde a partir de estos análisis se procederá a desarrollar un modelo conceptual detallado donde se identificará las entidades principales, sus atributos y relaciones de todas ellas, donde este paso es realizado para comprender la estructura que se llevará a cabo para la empresa AutoRenta.

Prosiguiendo se realizará la conversión del modelo conceptual al lógico, donde este tiene como función representar de manera más concisa de la organización de toda la base de datos, facilitando si compression, también se aplicará el proceso de normalización hasta la tercera forma normal (3FN) donde se reducira los datos redundantes.

Finalmente se llevará a cabo la conversión del modelo lógico al modelo físico, el cual definirá las entidades, atributos y relaciones.

Con estos pasos se garantizara un proyecto completo y efectivo para el diseño de los datos de datos para la empresa AutoRenta.

# **Caso de Estudio**

La empresa Autorenta ha solicitado el desarrollo de un sistema de información para gestionar su operación de alquiler de vehículos. Actualmente, con 5 sucursales en distintas ciudades y planes de expansión, requiere un sistema para gestionar el alquiler de su flota de vehículos. Los clientes podrán recoger un vehículo en una sucursal y devolverlo en otra. Las tarifas se calculan por días y/o semanas según el tipo de vehículo, con posibles descuentos durante el año. Si se excede la fecha de entrega, se aplicará un recargo del 8% por cada día adicional. El sistema busca optimizar la gestión de inventarios, reservas y pagos, mejorando la eficiencia y satisfacción del cliente, por lo que tendrá los siguientes requerimientos para la elaboración:

* *Sucursales:*

ciudad y dirección donde se ubica, teléfono fijo, celular y correo electrónico.

* *Empleados:*

sucursal donde labora, cédula, nombres, apellidos, dirección y ciudad de

residencia, celular y correo electrónico.

* *Clientes:*

cédula, nombres, apellidos, dirección y ciudad de residencia, celular y correo

Electrónico.

* *Vehículos:*

tipo de vehículo, placa, referencia, modelo, puertas, capacidad, sunroof, motor,

Color.

* *Alquileres:*

vehículo, cliente, empleado, sucursal y fecha de salida, sucursal y fecha de

llegada, fecha esperada de llegada, valor de alquiler por semana, valor de alquiler por día,

porcentaje de descuento, valor cotizado y valor pagado.

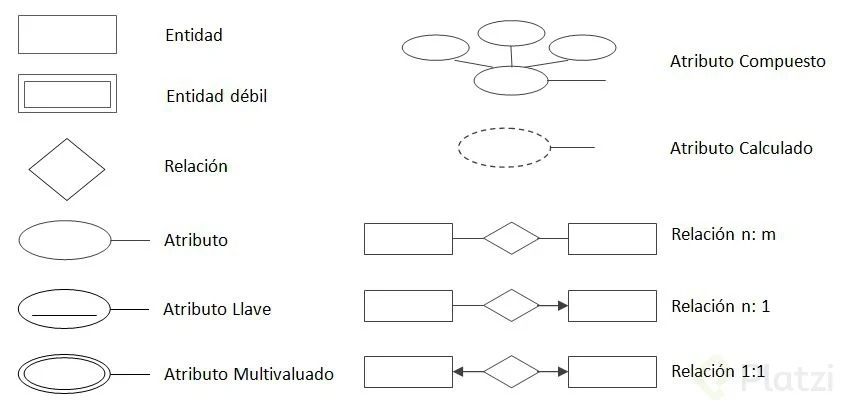
# **Planificación**

## Ejecución

Una vez haya sido analizado toda la información requerida para la elaboración de base de datos de la empresa Autorenta, se inició la elaboración del modelo conceptual, la cual este proceso proporciona un alta descripción donde representará los conceptos principales de la base de datos y las relaciones que hay entre ellos.

## Construcción del Modelo Conceptual

Se realizó el Modelo conceptual para identificar cada una de las entidades, sus atributos y las relaciones entre ellas, donde se proporcionó una visión más clara y estructurada de cómo se organizó y conecto los diferentes modelos de la base de dato teniendo en cuenta lo siguiente:



### Descripción

### Las entidades y atributos

1. Sucursales:

* Id\_sucursal.
* Ubicación.
* Teléfono fijo.
* Celular.
* Correo Electronico.

1. Empleados:

* Id\_empleado.
* id\_sucursal.
* Cédula.
* Nombres.
* Apellidos.
* Ubicación.
* Celular.
* Correo electrónico.

1. Vehículos:

* Id\_vehiculo.
* Tipo vehiculo.
* Placa.
* Referencia.
* Modelo.
* Puertas.
* Capacidad.
* Sunroof.
* Motor.
* Color.

1. Clientes:

* Id\_cliente.
* Cédula.
* Nombres.
* Apellidos.
* Ubicación.
* Celular.
* Correo electrónico.

1. Alquileres:

* Id\_alquiler.
* Id\_vehiculo.
* Id\_empleado.
* Id\_sucursal.
* Fecha salida
* Fecha esperada entrega.
* Fecha entrega.
* Valor cotizado.
* Valor pagado.

1. tipo\_vehiculo:

* Id\_tipo\_vehiculo.
* Tipo
* Valor\_alquiler\_dia.
* Valor\_alquiler\_semana.

1. Descuento:

* Id\_descuento.
* Id\_tipo\_vehiculo
* Fecha\_inicio
* Fecha\_fin
* porcentaje \_descuento.

### Relaciones y Cardinalidades

1. Sucursales - Empleados:

* Relación: “Tiene”, Un sucursal tiene varios empleados y un empleado tiene una sucursal.
* Cardinalidad: 1-N (uno a muchos).

1. Empleados - Alquileres:

* Relación: “Adquirir”,Un empleado puede adquirir varios alquileres y un alquiler puede ser adquirido por un empleado.
* Cardinalidad: 1:N (uno a muchos).

1. Vehículo - Alquileres:

* Relación: ‘Alquilar’, Un vehículo puede ser alquilado varias veces y en cada alquiler se puede alquilar un vehículo.
* Cardinalidad: 1:N (uno a muchos).

1. Tipo\_vehiculo - vehiculo:

* Relación: ‘Pertenece’, Muchos vehículos le pertenecen a un tipo de vehículo y un tipo vehículo le pertenece a muchos vehículos.
* Cardinalidad: 1:N (Uno a muchos).

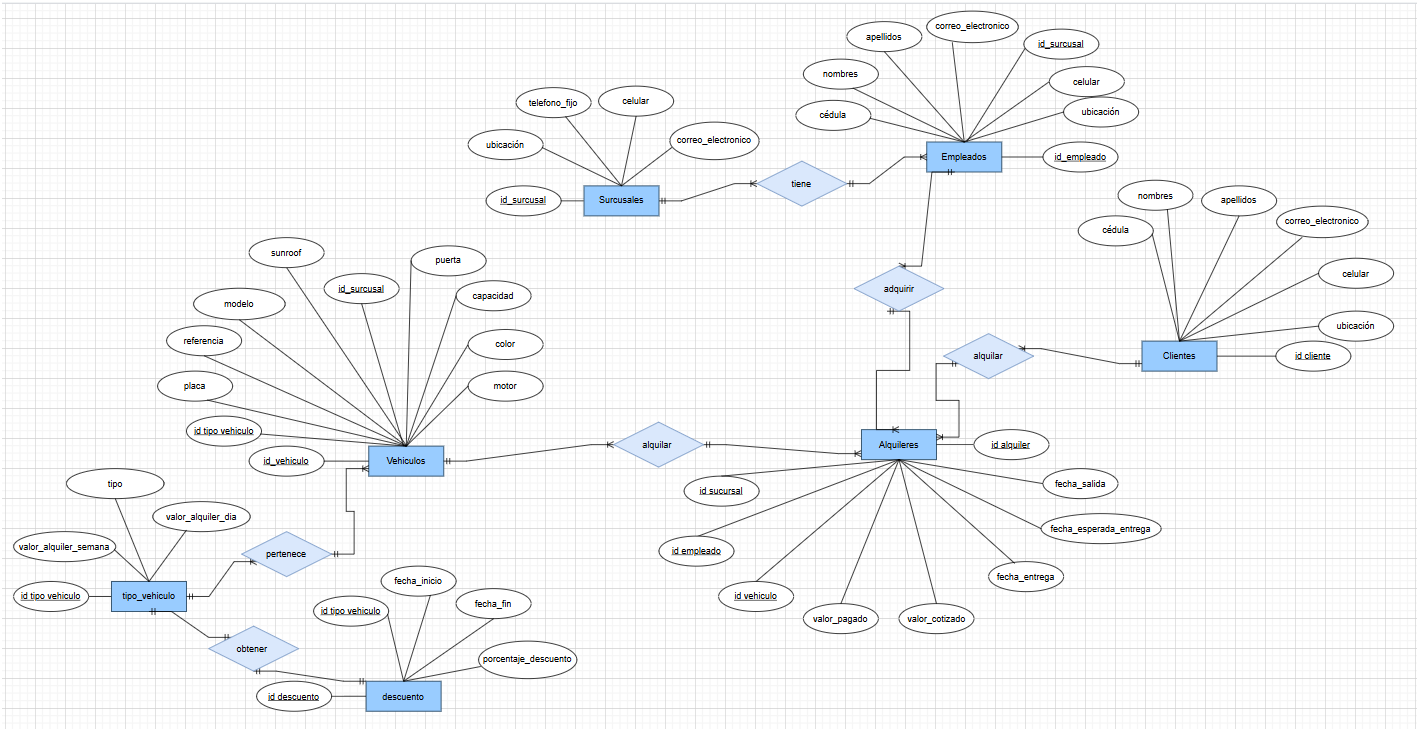
1. Tipo\_vehiculo - descuento:

* Relación: ‘Obtener’, Un tipo de vehículo puede tener un descuento y un descuento está relacionado con un tipo de vehículo.
* Cardinalidad: 1:1 (Uno a uno ).

1. Clientes - Alquileres:

* Relación: ‘Realizar, Un cliente puede realizar varios alquileres pero un alquiler solo lo puede hacer un cliente.
* Cardinalidad: 1:N (Uno a Muchos ).

### Gráfica

****

## Construcción del Modelo Lógico

Se realizó el modelo lógico teniendo en cuenta el modelo conceptual, donde se incorporaron detalles más específicos como las características de cada atributo, incluidas las claves primarias, foráneas y las relaciones de cardinalidad.

### Descripción

1. Sucursales:

* Id\_sucursal int primary key.
* Ubicación varchar(50).
* Teléfono fijo int.
* Celular int.
* Correo Electronico varchar(50).

1. Empleados:

* Id\_empleado int primary key.
* Id\_sucursal foreign key.
* Cédula int.
* Nombres varchar(50).
* Apellidos varchar(50).
* Ubicación varchar(50).
* Celular int.
* Correo electrónico varchar(50).

1. Vehículos:

* Id\_vehiculo primary key.
* Tipo vehículo varchar(50).
* Placa varchar(50).
* Referencia int.
* Modelo varchar(50).
* Puertas int.
* Capacidad int.
* Sunroof varchar(50).
* Motor varchar(50).
* Color varchar(50).

1. Clientes:

* Id\_cliente primary key.
* Cédula int.
* Nombres varchar(50).
* Apellidos varchar(50).
* Ubicación varchar(50).
* Celular int.
* Correo electrónico varchar(50).

1. Alquileres:

* Id\_alquiler Primary key.
* Id\_vehiculo foreign key.
* Id\_empleado foreign key.
* Id\_sucursa foreign key.
* Fecha salida Date
* Fecha esperada entrega date.
* Fecha entrega Date
* Valor cotizado int.
* Valor pagado int.

1. tipo\_vehiculo:

* Id\_tipo\_vehiculo Primary key.
* Tipo varchar(50).
* Valor\_alquiler\_dia Date.
* Valor\_alquiler\_semana Date.

1. Descuento:

* Id\_descuento Primary key.
* Id\_tipo\_vehiculo
* Fecha\_inicio Date.
* Fecha\_fin Date.
* porcentaje \_descuento double.

### Relaciones y Cardinalidades

Se realizó las relaciones y cardinalidades respectivas del modelo lógico con sus entidades para tener mejor visualización de la base de datos:

1. Sucursales - Empleados:

* Una sucursal tiene varios empleados y un empleado tiene una sucursal. 1-N (uno a muchos).



1. Empleados - Alquileres:

* Un empleado puede adquirir varios alquileres y un alquiler puede ser adquirido por un empleado. 1:N (uno a muchos).



1. Vehículo - Alquileres:

* Un vehículo puede ser alquilado varias veces y en cada alquiler se puede alquilar un vehículo. 1:N (uno a muchos).



1. Tipo\_vehiculo - vehiculo:

* Muchos vehículos le pertenece a un tipo de vehiculo y un tipo vehículo le pertenece a muchos vehículos. 1:N (Uno a muchos).



1. Tipo\_vehiculo - descuento:

* Un tipo de vehículo puede tener un descuento y un descuento está relacionado con un tipo de vehículo. 1:1 (Uno a uno ).

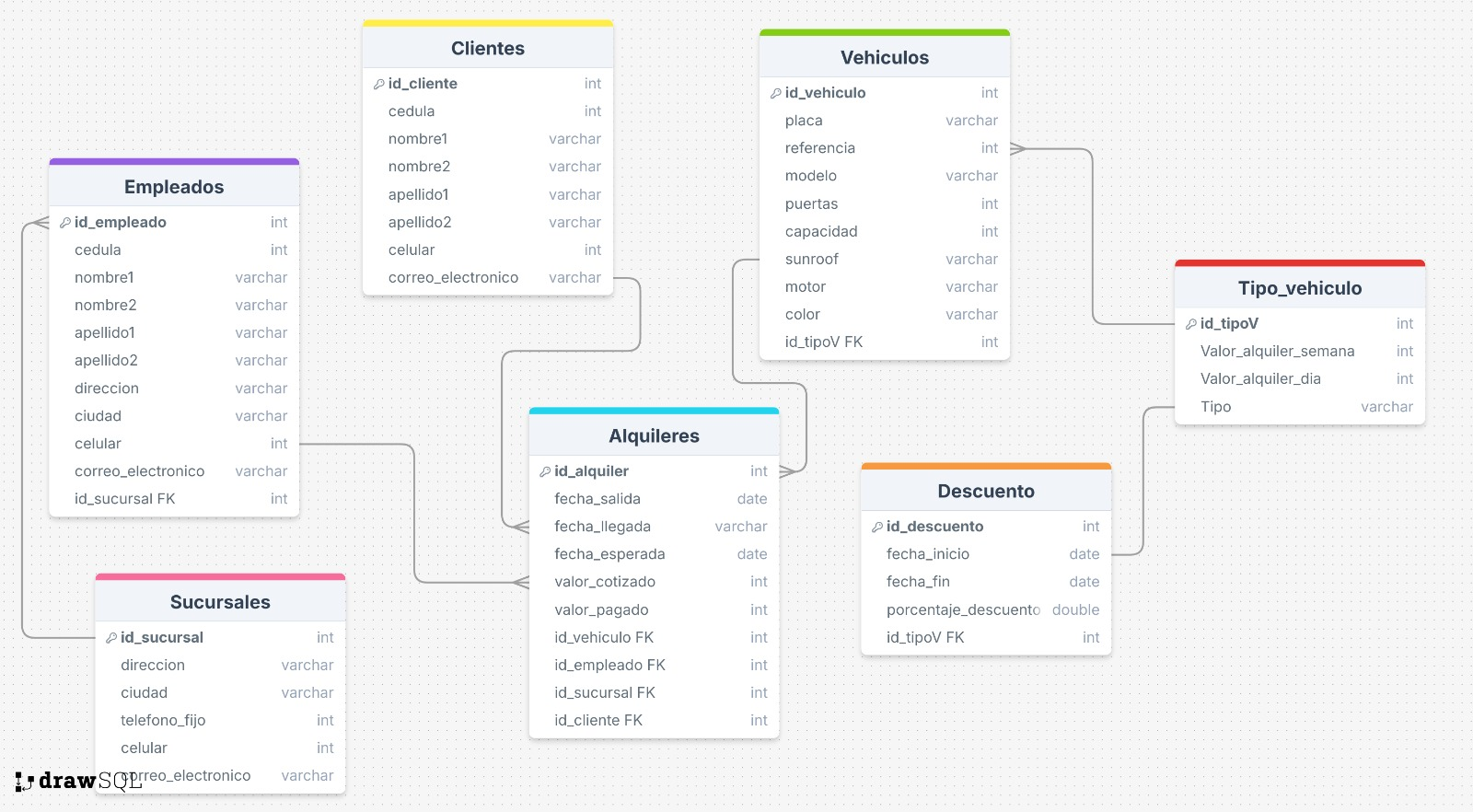


1. Clientes - Alquileres:

* Un cliente puede realizar varios alquileres pero un alquiler solo lo puede hacer un cliente. 1:N (Uno a muchos).



### Gráfica



## Normalización del Modelo Lógico

Se realizó el proceso de la normalización de las tablas anteriormente visualizadas para

organizar los datos de manera más eficiente,minimizando redundancias y dependencias

transitivas en la base de datos en desarrollo.

### Primera Forma Normal (1FN)

Una tabla está en 1FN si cumple con los siguientes criterios:

* Cada atributo debe tener valores atómicos.
* Cada fila la misma tabla debe ser única
* Debe prevalecer un crecimiento vertical de los datos y no horizontal
* No deben existir grupos repetidos de datos.

#### Descripción

La primera forma normal es el primer nivel de normalización en base de datos donde se le aplicará a las tablas de las bases de datos para garantizar una mejor organización donde se evita redundancias y asegurará la consistencia de la información.

#### Descripción técnica

1. Sucursales:

❖ No encuentra en 1FN, ya que cada columna no tiene valores únicos y son repetitivos pero cuenta con una clave primaria única.

1. Empleados :

❖No encuentra en 1FN, ya que cada columna no tiene valores únicos y son repetitivos pero cuenta con una clave primaria única.

3. Clientes :

❖ No encuentra en 1FN, ya que cada columna no tiene valores únicos y son repetitivos pero cuenta con una clave primaria única.

4. Alquileres :

❖ No encuentra en 1FN, ya que cada columna no tiene valores únicos y son repetitivos pero cuenta con una clave primaria única.

5. Vehiculos:

❖ No encuentra en 1FN, ya que cada columna no tiene valores únicos y son repetitivos pero cuenta con una clave primaria única.

6. Descuentos:

❖ No encuentra en 1FN, ya que cada columna no tiene valores únicos y son repetitivos pero cuenta con una clave primaria única.

7. Tipo\_vehiculo:

❖ No encuentra en 1FN, ya que cada columna no tiene valores únicos y son repetitivos pero cuenta con una clave primaria única.

#### Gráfica

| Sucursales | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| id\_sucursal | titulo | ciudad | dirección | telefono\_fijo | Celular | correo\_electronico |
| PK |  |  |  |  |  |  |

| Empleados | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_empleado | id\_sucursal | cedula | nombre1 | nombre2 | apellido1 | apellido2 | ciudad | ubicación | celular | correo\_electronico |
| PK | FK |  |  |  |  |  |  |  |  |  |

| Clientes | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_cliente | cedula | nombre1 | nombre2 | apellido1 | apellido2 | ciudad | ubicación | celular | correo\_electronico |
| PK |  |  |  |  |  |  |  |  |  |

| Alquileres | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_alquiler | id\_sucursal | id\_empleado | id\_vehiculo | valor\_pagado | valor\_cotizado | fecha\_entrega | fecha\_esperada\_entrega | fecha\_salida |
| PK | FK | FK | FK |  |  |  |  |  |

| Vehiculos | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_vehiculo | id\_tipo\_vehiculo | id\_sucursal | placa | modelo | sunroof | referencia | puerta | capacidad | color | motor |
| PK | FK | FK |  |  |  |  |  |  |  |  |

| Descuento | | | | |
| --- | --- | --- | --- | --- |
| id\_descuento | id\_tipo\_vehiculo | fecha\_inicio | fecha\_fin | porcentaje\_descuento |
| PK | FK |  |  |  |

| Tipo\_vehiculo | | | |
| --- | --- | --- | --- |
| id\_tipo\_vehiculo | valor\_alquiler\_semanal | valor\_alquiler\_dia | tipo |
| PK |  |  |  |

### Forma Normal (2FN)

Una tabla está en 2FN si cumple con los siguientes criterios:

* Estar en 1FN.
* La relación debe tener una clave principal, de preferencia simple.
* Cada atributo de la tabla debe depender totalmente del atributo clave.

#### Descripción

La segunda forma normal, es el segundo nivel de normalización en el diseño de la base de

datos que se aplicará a las tablas de una base de datos que ya cumplen con la primera forma

normal y lleva a cabo la eliminación de dependencias parciales dentro de una tabla.

#### Descripción técnica

1. Sucursales :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Empleados :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Clientes :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Alquileres :

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Vehiculos:

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Descuento:

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

1. Tipo\_vehiculo:

❖ Se encuentra en 2FN, ya que cuenta con una clave primaria única y cada

La columna depende completamente de esa clave primaria.

#### Gráfica

| Sucursales | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| id\_sucursal | titulo | ciudad | dirección | telefono\_fijo | Celular | correo\_electronico |
| PK |  |  |  |  |  |  |

| Empleados | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_empleado | id\_sucursal | cedula | nombre1 | nombre2 | apellido1 | apellido2 | ciudad | ubicación | celular | correo\_electronico |
| PK | FK |  |  |  |  |  |  |  |  |  |

| Clientes | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_cliente | cedula | nombre1 | nombre2 | apellido1 | apellido2 | ciudad | ubicación | celular | correo\_electronico |
| PK |  |  |  |  |  |  |  |  |  |

| Alquileres | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_alquiler | id\_sucursal | id\_empleado | id\_vehiculo | valor\_pagado | valor\_cotizado | fecha\_entrega | fecha\_esperada\_entrega | fecha\_salida |
| PK | FK | FK | FK |  |  |  |  |  |

| Vehiculos | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_vehiculo | id\_tipo\_vehiculo | id\_sucursal | placa | modelo | sunroof | referencia | puerta | capacidad | color | motor |
| PK | FK | FK |  |  |  |  |  |  |  |  |

| Descuento | | | | |
| --- | --- | --- | --- | --- |
| id\_descuento | id\_tipo\_vehiculo | fecha\_inicio | fecha\_fin | porcentaje\_descuento |
| PK | FK |  |  |  |

| Tipo\_vehiculo | | | |
| --- | --- | --- | --- |
| id\_tipo\_vehiculo | valor\_alquiler\_semanal | valor\_alquiler\_dia | tipo |
| PK |  |  |  |

### Tercera Forma Normal (3FN)

Una tabla está en 3NF si cumple con los siguientes criterios:

* Debe estar en 2FN.
* No deben existir atributos no principales que dependen transitivamente del atributo clave.

#### Descripción

La tercera forma normal, es el tercer nivel de normalización en el diseño de la base de datos

que se aplicará a las tablas de una base de datos que ya cumplen con la segunda forma

normal y se enfoca en la eliminación de dependencias transitivas, evitando que un atributo no

clave depende de otro no clave.

#### Descripción técnica

1. Sucursales :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Empleados :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Clientes :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Alquileres :

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Vehiculos:

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Descuento:

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

1. Tipo\_vehiculo:

❖ Se encuentra en 3FN, ya que está en la 2FN y en cada columna no hay

dependencias transitivas con la clave primaria.

#### Gráfica

| Sucursales | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| id\_sucursal | titulo | ciudad | dirección | telefono\_fijo | Celular | correo\_electronico |
| PK |  |  |  |  |  |  |

| Empleados | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_empleado | id\_sucursal | cedula | nombre1 | nombre2 | apellido1 | apellido2 | ciudad | ubicación | celular | correo\_electronico |
| PK | FK |  |  |  |  |  |  |  |  |  |

| Clientes | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_cliente | cedula | nombre1 | nombre2 | apellido1 | apellido2 | ciudad | ubicación | celular | correo\_electronico |
| PK |  |  |  |  |  |  |  |  |  |

| Alquileres | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_alquiler | id\_sucursal | id\_empleado | id\_vehiculo | valor\_pagado | valor\_cotizado | fecha\_entrega | fecha\_esperada\_entrega | fecha\_salida |
| PK | FK | FK | FK |  |  |  |  |  |

| Vehiculos | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id\_vehiculo | id\_tipo\_vehiculo | id\_sucursal | placa | modelo | sunroof | referencia | puerta | capacidad | color | motor |
| PK | FK | FK |  |  |  |  |  |  |  |  |

| Descuento | | | | |
| --- | --- | --- | --- | --- |
| id\_descuento | id\_tipo\_vehiculo | fecha\_inicio | fecha\_fin | porcentaje\_descuento |
| PK | FK |  |  |  |

| Tipo\_vehiculo | | | |
| --- | --- | --- | --- |
| id\_tipo\_vehiculo | valor\_alquiler\_semanal | valor\_alquiler\_dia | tipo |
| PK |  |  |  |

# Construcción del Modelo Físico

Se realizó el modelo físico acorde con la elaboración del modelo lógico donde incluye las entidades, sus atributos y las relaciones entre ellas. El objetivo de este modelo es incorporar los tipos de datos de los atributos previamente definidos.

### Descripción

El modelo físico se trabaja en MySQL, en el cual cada entidad se representa como una tabla compuesta por sus atributos correspondientes, organizados en columnas con tipos de datos específicos según sea necesario

Tablas

Para crear la base de datos utilice el siguiente comando:

create database AutoRenta\_OrtizZully;

Para utilizar la base de datos ocupe el siguiente comando:

use AutoRenta\_OrtizZully;

Comenzaremos creando las tablas junto con sus tipos de datos correspondientes. Para esto,

utiliza los siguientes comandos:

1. Creación de la tabla Sucursales:

CREATE TABLE sucursales (

id\_sucursal INT PRIMARY KEY,

ubicacion VARCHAR(255),

ciudad VARCHAR(50) ,

dirección VARCHAR(50) ,

telefono\_fijo VARCHAR(50) ,

celular VARCHAR(50) ,

correo\_electronico VARCHAR(255)

);

2. Creación de la tabla tipo\_vehiculo:

CREATE TABLE tipo\_vehiculo(

id\_tipo\_vehiculoINT PRIMARY KEY,

tipo VARCHAR(250) ,

valor\_alquiler\_dia DOUBLE,

valor\_alquiler\_semana DOUBLE

);

3. Creación de la tabla vehiculos:

CREATE TABLE vehiculos (

id\_vehiculo INT PRIMARY KEY,

id\_tipo\_vehiculo INT,

id\_sucursal INT,

placa VARCHAR(50) ,

referencia VARCHAR(255),

modelo VARCHAR(255),

puertas int,

capacidad int,

sunroof BOOLEAN,

motor VARCHAR(255),

color VARCHAR(255),

foreign key (id\_tipo\_vehiculo) references tipo\_vehiculo(id\_tipo\_vehiculo),

foreign key (id\_sucursal) references sucursales(id\_sucursal)

);

4. Creación de la tabla descuento:

CREATE TABLE descuento (

id\_descuento INT PRIMARY KEY,

id\_tipo\_vehiculo INT ,

fecha\_inicio DATE,

fecha\_fin DATE,

porcentaje\_descuento DOUBLE,

foreign key (id\_tipo\_vehiculo) references tipo\_vehiculo(id\_tipo\_vehiculo)

);

5. Creación de la tabla clientes:

CREATE TABLE clientes (

id\_cliente INT PRIMARY KEY,

cedula VARCHAR(50) ,

nombre1 VARCHAR(255) ,

nombre2 VARCHAR(255) ,

apellido1 VARCHAR(255) ,

apellido2 VARCHAR(255) ,

ubicacion VARCHAR(255),

ciudad VARCHAR(50) ,

celular VARCHAR(50) ,

correo\_electronico VARCHAR(255)

);

6. Creación de la tabla empleados:

CREATE TABLE empleados (

id\_empleado INT PRIMARY KEY ,

Id\_sucursal INT ,

cedula VARCHAR(50) ,

nombre1 VARCHAR(255) ,

nombre2 VARCHAR(255) ,

apellido1 VARCHAR(255) ,

apellido2 VARCHAR(255) ,

ubicacion VARCHAR(255),

ciudad VARCHAR(50) ,

celular VARCHAR(50) ,

correo\_electronico VARCHAR(255) ,

foreign key (id\_sucursal) references sucursales(id\_sucursal)

);

7. Creación de la tabla empleados:

CREATE TABLE alquileres (

id\_alquiler INT PRIMARY KEY,

id\_vehiculo INT ,

id\_empleado INT ,

id\_sucursal INT ,

fecha\_salida DATE ,

fecha\_esperada\_entrega DATE,

fecha\_entrega DATE ,

valor\_cotizado DECIMAL(10, 2) ,

valor\_pagado DECIMAL(10, 2),

foreign key (id\_vehiculo) references vehiculos(id\_vehiculo),

foreign key (id\_empleado) references empleados(id\_empleado),

foreign key (id\_sucursal) references sucursales(id\_sucursal)

);

## Construcción del Diagrama UML

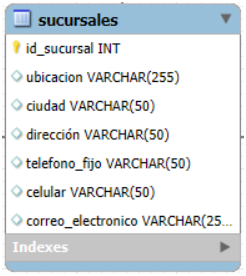
Se realizó un diagrama UML tomando como referencia la normalización para poder entender mejor cada diseño, la arquitectura del código y la implementación. Dónde nos permitirá tener una visión clara y detallada de cómo se maneja cada consulta de la la empresa AutoRenta.

### Descripción

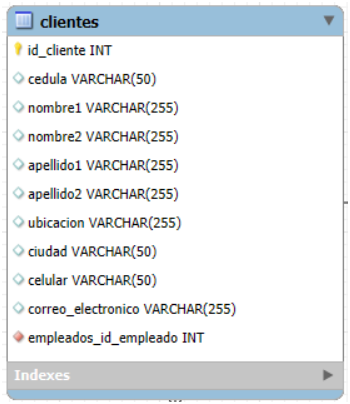
El diagrama UML se diseñó con el objetivo de representar detalladamente la estructura de cada tabla y sus relaciones, donde tendrá así más ganadas unas llaves primarias y foráneas, además especifica obligatoriamente los atributos, dando un nivel de detalle facilitada para poder entenderse con más facilidad.

Comenzaremos creando las tablas junto con sus tipos de datos correspondientes:

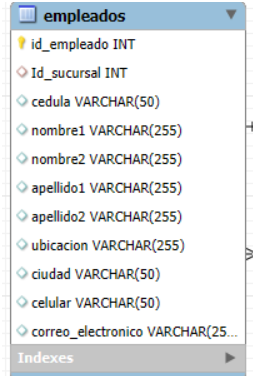
1.Tabla Sucursales:



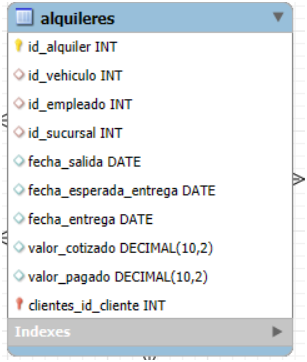
2. Tabla Clientes:



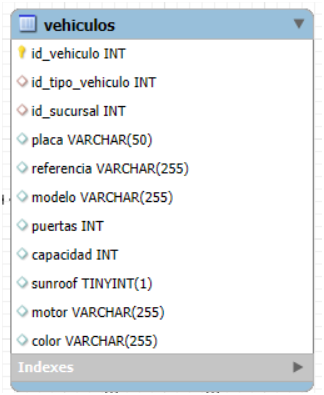
3. Tabla Empleados:



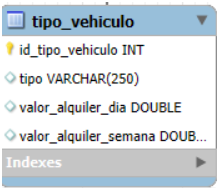
4. Tabla Alquileres:



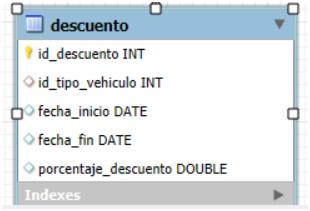
5. Tabla Vehículos:



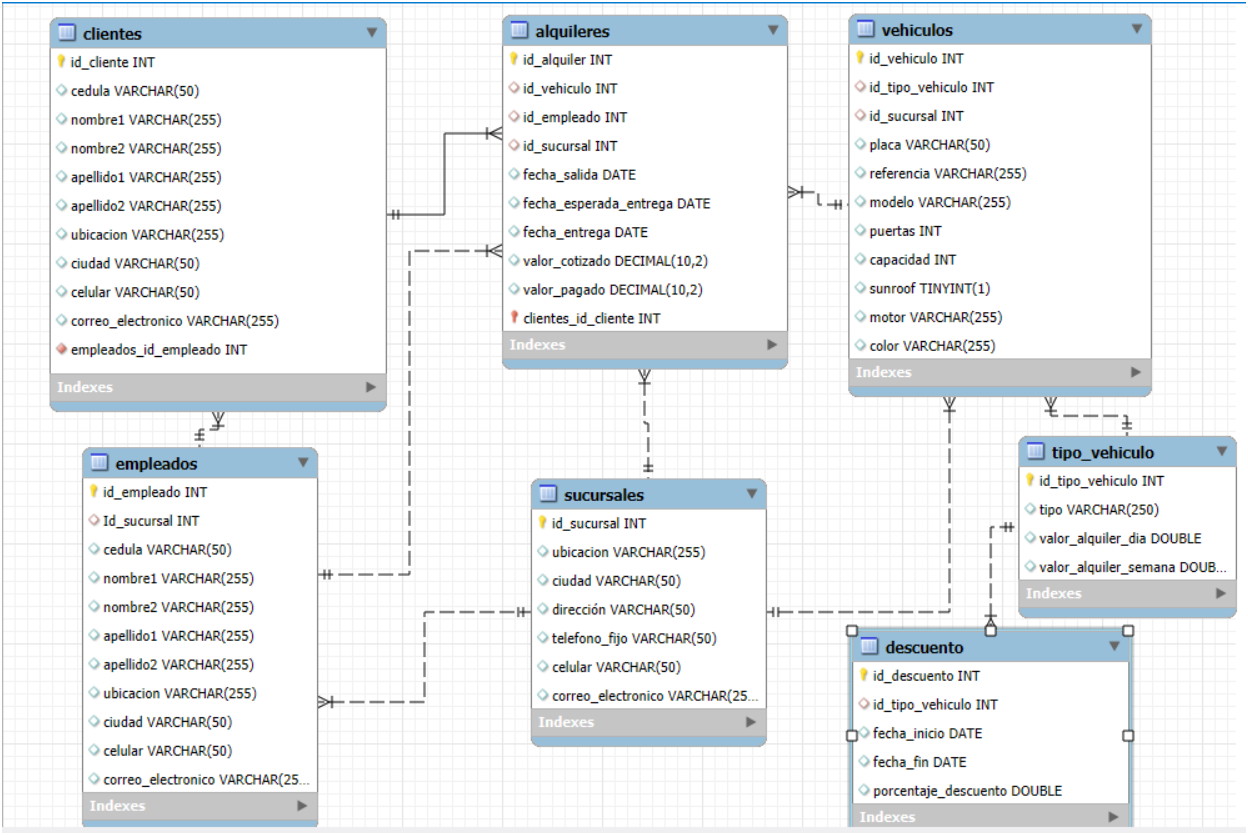
6. Tabla tipo\_vehiculo



7. Tabla tipo descuento:



Gráfica

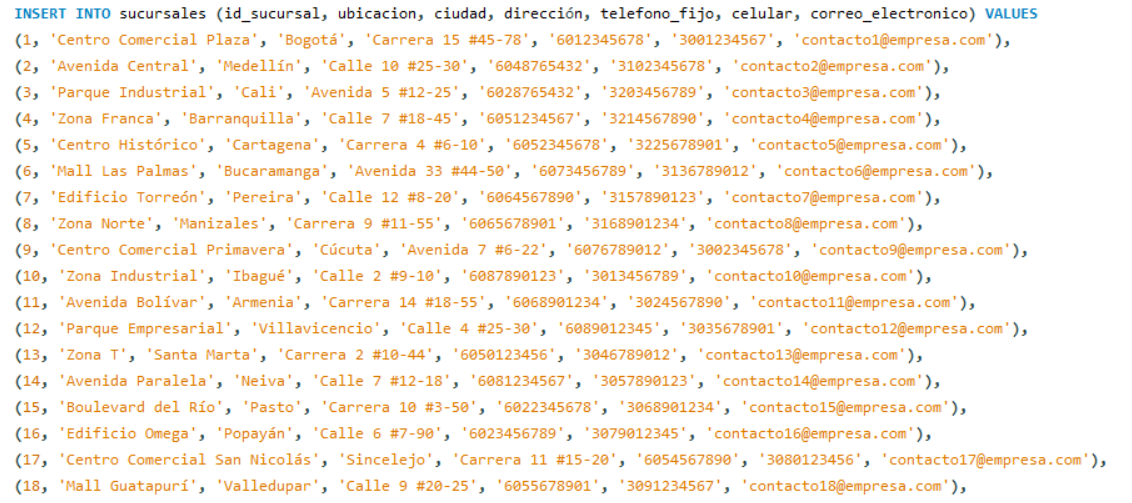


# Inserciones de Datos

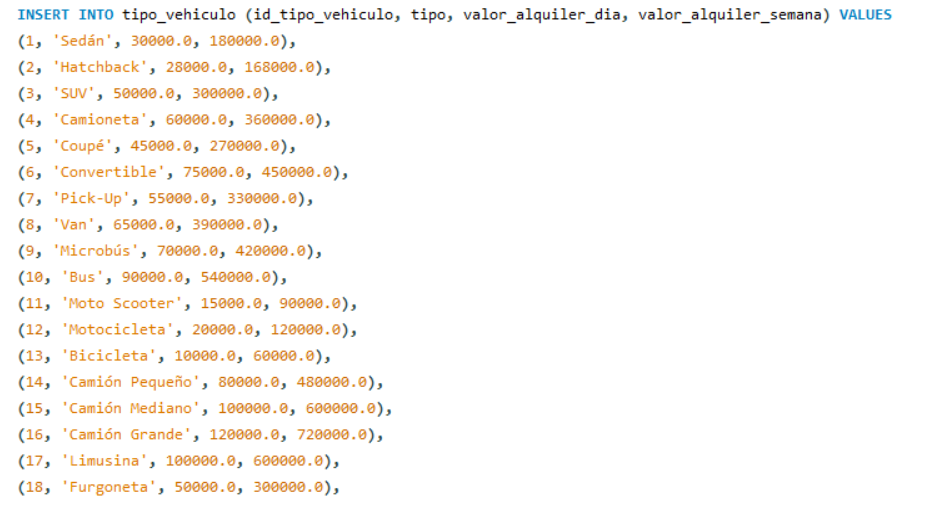
### Descripción

Para poder entender mejor el proceso de inserción de datos en la tabla de datos de AutoRental, se deberá insertar datos en todos lo datos creados

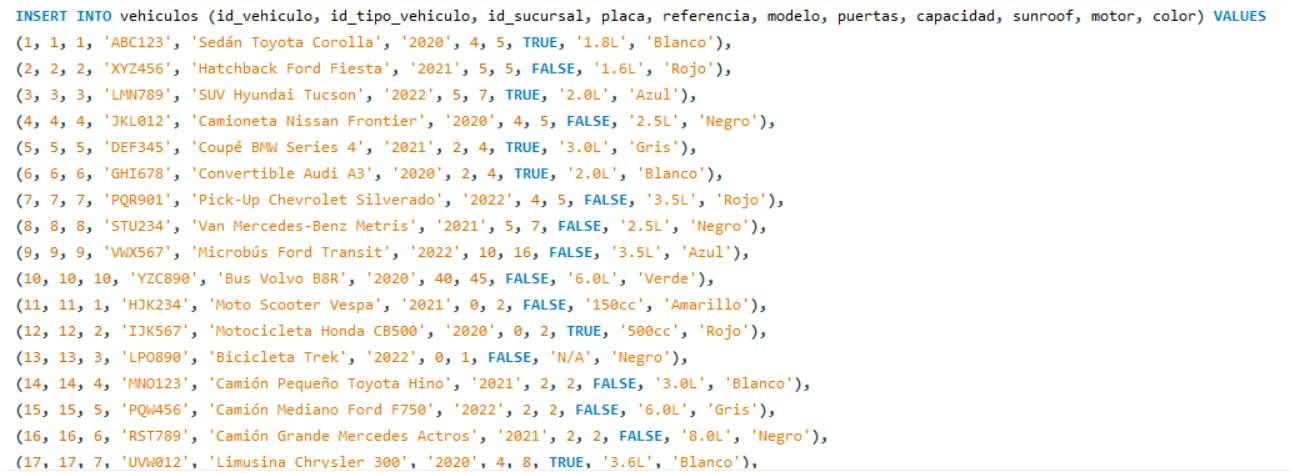
1. Tabla Sucursales



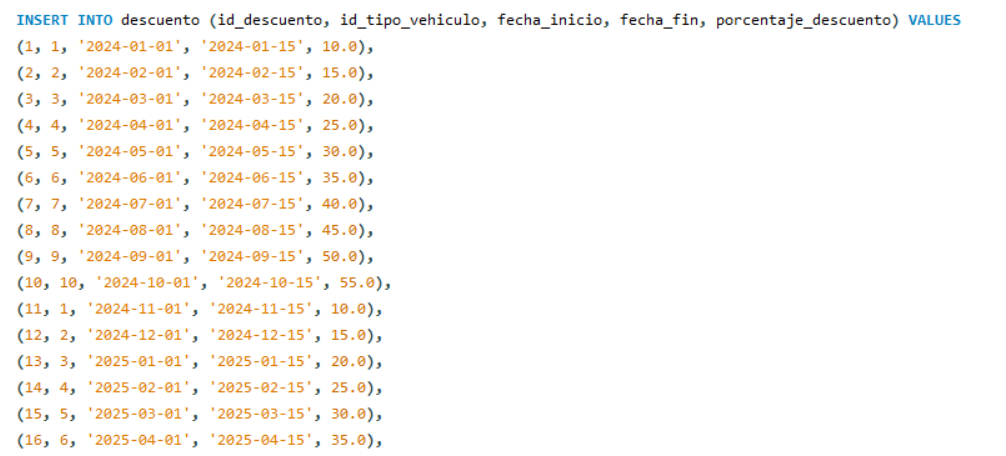
1. Tabla de tipo\_vehiculo:



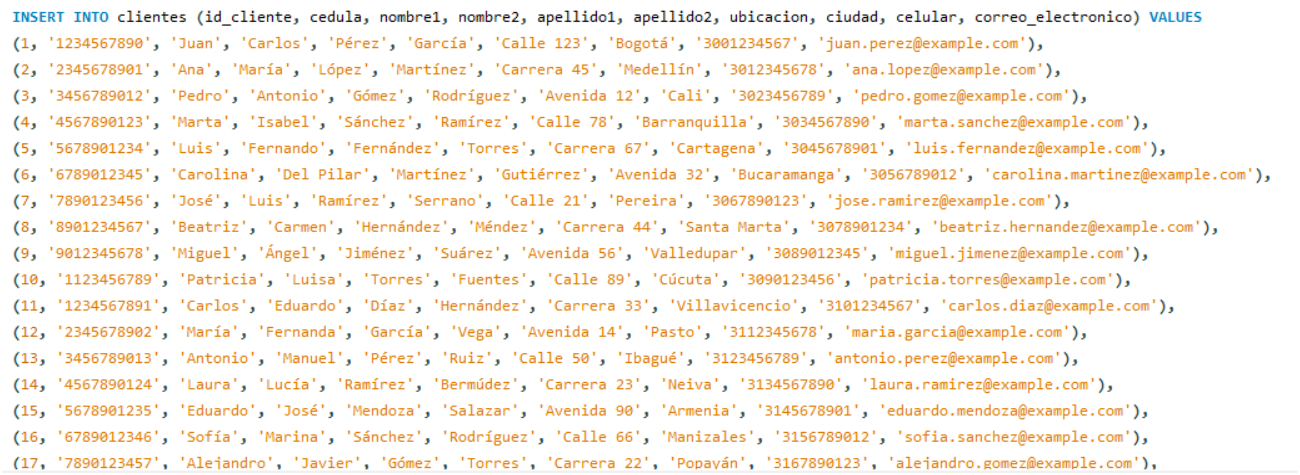
1. Tabla vehículos:



1. Tabla descuento:



1. Tabla clientes:



1. Tabla empleados:



1. Tabla Alquileres:

