

AutoRental

Luis Henao

T2

PEDRO FELIPE GÓMEZ BONILLA

**CAMPUSLANDS
SANDBOX
RUTA JAVA
TIBU
2024**

Tabla de Contenidos

Introducción	3
Caso de Estudio	4
Planificación	5
Construcción del Modelo Conceptual	5
Descripción	5
Gráfica	7
Construcción del Modelo Lógico	7
Descripción	7
Gráfica	9
Normalización del Modelo Lógico.	9
Primera Forma Normal (1FN)	9
Descripción	10
Gráfica	11
Segunda Forma Normal (2FN)	12
Descripción	12
Gráfica	13
Tercera Forma Normal (3FN)	14
Descripción	14
Gráfica	15
Construcción del Modelo Físico	16
Descripción	16
Creación de tablas	16

Introducción

La empresa de AtoRental ha decidido crear un sistema de información, para esto ha decidido escoger una base de datos en MySQL. Esta empresa se dedica al alquiler de vehículos, para lo cual tiene su propia flota de vehículos y sus diferentes sucursales y busca expandirse más. por esto necesitan una base de datos robusta y eficiente.

En este documento se realizarán la construcción de la base de datos con sus diferentes secciones como diagrama conceptual y lógico, normalización, construcción del modelo lógico junto con inserción de datos.

Caso de Estudio

La empresa donde usted trabaja ha sido contratada para desarrollar un sistema de información para una empresa de alquiler de vehículos llamada AutoRental, y usted ha sido designado para diseñar una base de datos para ese sistema de información.

AutoRental cuenta con 5 sucursales en diferentes ciudades y se proyecta a expandirse a otras ciudades del país y cuenta con una flota propia de vehículos de diferentes tipos, modelos (año), capacidad, etc.

Los clientes de AutoRental podrán alquilar un vehículo en una sucursal y entregarlo en otra sucursal.

AutoRental ofrece descuentos sobre diferentes tipos de vehículos a lo largo del año. Los valores de alquiler dependen del tipo de vehículo (sedán, compacto, camioneta platón, camioneta lujo, deportivo, etc) y se cobran por días y/o semanas. Por ejemplo, si un cliente alquila un vehículo por 9 días, el valor cotizado será de 1 semana y 2 días

Si un cliente entrega el vehículo pasada la fecha de entrega contratada, se cobrarán los días adicionales con un incremento del 8%.

La base de datos deberá cumplir almacenar la siguiente información:

- 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

Construcción del Modelo Conceptual

Descripción

Para la construcción del modelo lógico se analizó los requerimientos de la base de datos y se crearon 7 tablas para cumplir esos requerimientos

Tablas y atributos destacables

★ cliente

- **id:** Es él, id único de cada cliente para la fácil identificación dentro de la base de datos

★ sucursal

- **id:** Es el identificador único de cada sucursal para facilitar la búsqueda dentro de la base de datos

★ empleado

- **id:** Es el identificador único de cada empleado para facilitar la búsqueda dentro de la base de datos
- **id_sucursal:** Es él, id de la sucursal en la que está trabajando el empleado

★ vehiculo

- **id:** Es el identificador único de cada vehículo para facilitar la búsqueda dentro de la base de datos

- **id_tipo:** Es él, id de tipo de vehículo que tiene el vehículo, con este se pueden saber los precios por semana y día de cada vehículo junto con su descuento
-
- **sunroof:** Con este atributo se puede saber si el vehículo tiene techo corredizo o no
- **id_sucursal:** Es la sucursal en la que está actualmente el vehículo, ya que este puede cambiar de sucursal cuando llega a algún lado

★ tipo_vehiculo

- **id:** Es el identificador único de cada tipo de vehículo para facilitar la búsqueda dentro de la base de datos
- **precio_semana:** acá está el precio por semana según el tipo de vehículo, este sirve para el momento del cobro del alquiler, saber cuanto cuesta dependiendo del tipo de vehículo
- **precio_dia:** acá está el precio por día según el tipo de vehículo, este sirve para el momento del cobro del alquiler, saber cuanto cuesta dependiendo del tipo de vehículo

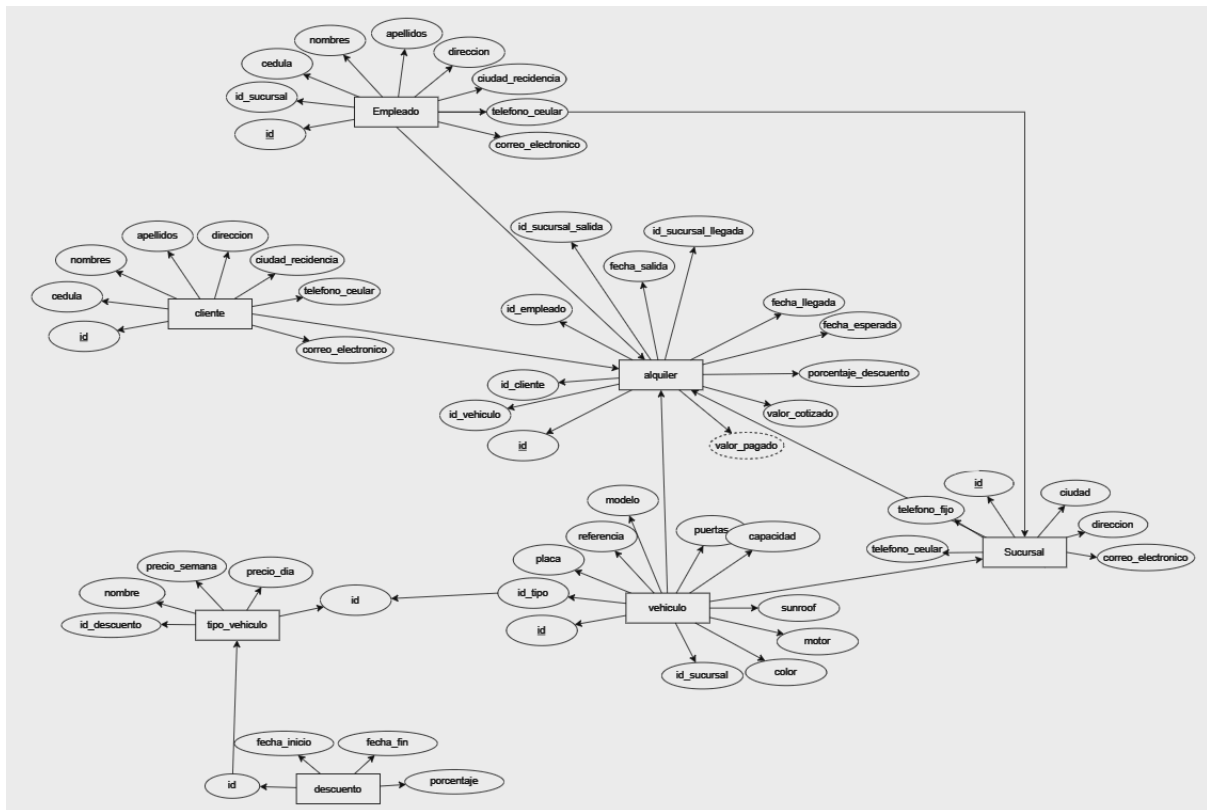
★ descuento

- **id:** Es el identificador único de cada descuento para facilitar la búsqueda dentro de la base de datos
- **fecha_inicio:** Es el inicio de la fecha durante la que estará vigente este descuento
- **fecha_fin:** Es el cierre de la fecha durante la que estará vigente este descuento
- **porcentaje:** Es el porcentaje de descuento que se tendrá en este descuento

★ alquiler

- **id:** Es el identificador único de cada descuento para facilitar la búsqueda dentro de la base de datos
- **id_vehiculo:** Es él, id del vehículo que se está alquilando
- **id_cliente:** Es él, id del cliente que está alquilando el vehículo
- **id_empleado:** Es él, id del empleado que está realizando el alquiler del vehículo
- **id_sucursal_salida:** Es él, id de la sucursal de la cual está saliendo el vehículo
- **fecha_salida:** Es la fecha en la que el vehículo fue alquilado
- **id_sucursal_llegada:** Es él, id de la sucursal en la cual se dejara el carro cuando acabe el alquiler
- **fecha_llegada:** Es la fecha en la que se entregó el vehículo a la sucursal de nuevo
- **fecha_esperada:** Es la fecha en la que se debería entregar el carro hasta esa fecha era el contrato de alquiler, si el cliente se pasa de esa fecha se harán cargos extras al valor final
- **valor_cotizado:** Este es el valor que se cobra por el alquiler basándonos en el tiempo del alquiler del contrato y basándonos en el tipo de vehículo en este no se aplican ni descuentos ni sobrecargos por entrega fuera de plazo
- **valor_pagado:** Este es el valor final que debe pagar el cliente acá se aplicaron los descuentos en caso de tenerlos y los sobrecargos por entrega fuera de plazo (en caso de haber entregado fuera de plazo)

Gráfica



<https://drive.google.com/file/d/1Tv4CtL-q7MYD1GcX7TWjFWHZ3eptxmlf/view?usp=sharing>

Construcción del Modelo Lógico

Descripción

Una vez realizado el diagrama conceptual se creó el diagrama lógico basándose en el mismo, en este se definió el tipo de variable de cada columna en las tablas de la base de datos y la relación que hay entre ellas

Tablas y atributos con sus tipos

★ cliente

- **id:** int primary key
- **cedula:** int(15)
- **nombres:** varchar(50)
- **apellidos:** varchar(50)
- **direccion:** varchar(150)
- **ciudad_residencia:** varchar(40)
- **telefono_celular:** int(15)
- **correo_electronico:** varchar(80)

★ sucursal

- **id:** int primary key

- **direccion:** varchar(150)
- **ciudad:** varchar(40)
- **telefono_fijo:** int(15)
- **telefono_celular:** int(15)
- **correo_electronico:** varchar(80)

★ empleado

- **id:** int primary key
- **id_sucursal:** int foreign key
- **cedula:** int(15)
- **nombres:** varchar(50)
- **apellidos:** varchar(50)
- **direccion:** varchar(150)
- **ciudad_residencia:** varchar(40)
- **telefono_celular:** int(15)
- **correo_electronico:** varchar(80)
-

★ vehiculo

- **id:** int primary key
- **id_tipo:** int foreign key
- **id_sucursal:** int foreign key
- **placa:** varchar(6)
- **referencia:** varchar(20)
- **modelo:** varchar(30)
- **puertas:** int
- **capacidad:** int
- **sunroof:** boolean
- **motor:** varchar(45)
- **color:** varchar(30)

★ tipo_vehiculo

- **id:** int primary key
- **nombre:** varchar(50)
- **id_descuento:** int foreign key
- **precio_semana:** decimal(10,2)
- **precio_dia:** decimal(10,2)

★ descuento

- **id:** int primary key
- **fecha_inicio:** date
- **fecha_fin:** date
- **porcentaje:** int

★ alquiler

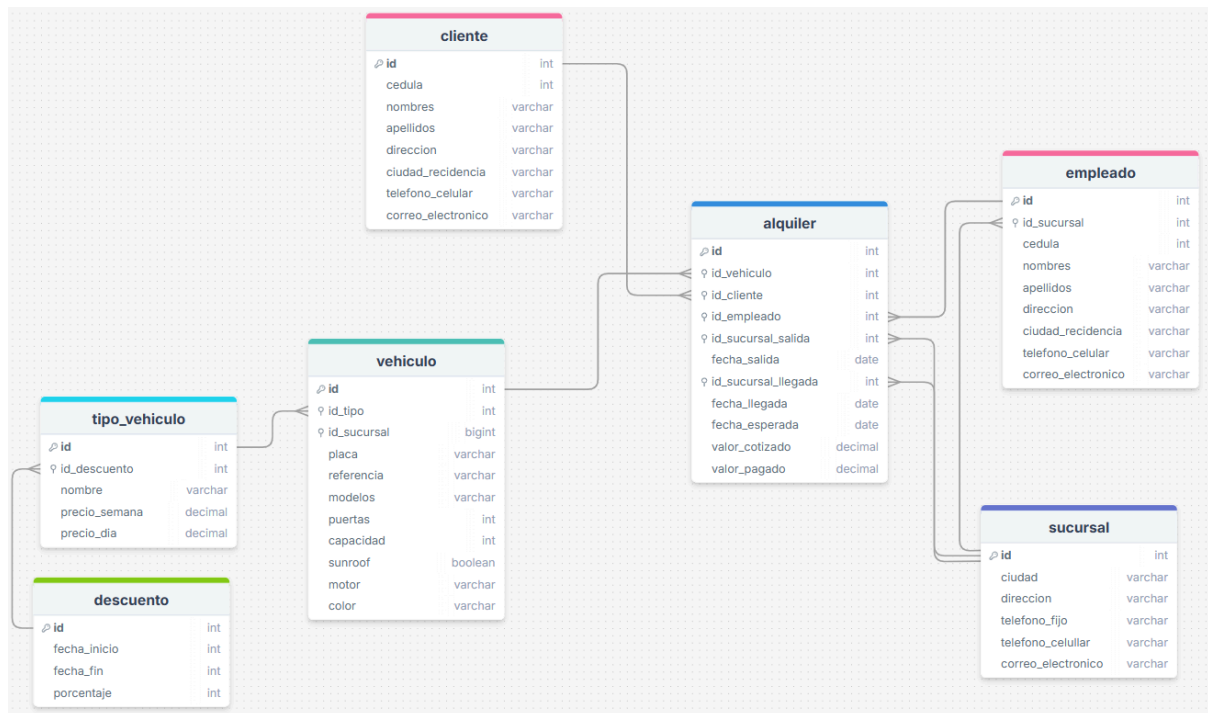
- **id:** int primary key
- **id_vehiculo:** int foreign key
- **id_cliente:** int foreign key
- **id_empleado:** int foreign key
- **id_sucursal_salida:** int foreign key
- **fecha_salida:** date
- **id_sucursal_llegada:** int foreign key
- **fecha_llegada:** date
- **fecha_esperada:** date

- **valor_cotizado:** decimal(10,2)
- **valor_pagado:** decimal(10,2)

Relaciones entre tablas

- ★ **cliente - alquiler:** relación de uno a muchos, ya que un cliente puede realizar muchos alquileres
- ★ **vehiculo - alquiler:** relación de uno a muchos, ya que un vehiculo se puede alquilar muchas veces
- ★ **sucursal - alquiler:** relación de uno a muchos, ya que una sucursal puede alquilar muchos carros, pero un carro no puede salir o llegar a muchas sucursales
- ★ **empleado - alquiler:** relación de uno a muchos, ya que un empleado puede hacer muchos alquileres
- ★ **sucursal - empleado:** relación de uno a muchos, ya que una sucursal puede tener muchos empleados

Gráfica



Normalización del Modelo Lógico.

Primera Forma Normal (1FN)

Para poder cumplir con la normalización se necesita cumplir con unos criterios específicos los cuales son los siguientes:

- Todos los atributos contienen valores atómicos (indivisibles).
- No debe haber grupos repetitivos de columnas.
- Cada columna debe contener un solo valor en cada fila.

Descripción

Para la primera forma normal, solo las tablas de empleado y cliente se separaron las columnas de nombres y apellido por nombre1, nombre2, apellido1, apellido2. Las otras tablas están normalizadas, ya que cumplen con los requerimientos para estar en primera forma normal

Gráfica

cliente	empleado
id int	id int
cedula int	id_sucursal int
nombre1 varchar	cedula int
nombre2 varchar	nombre1 varchar
apellido1 varchar	nombre2 varchar
apellido2 varchar	apellido1 varchar
direccion varchar	apellido2 varchar
ciudad_recidencia varchar	direccion varchar
telefono_celular varchar	ciudad_recidencia varchar
correo_electronico varchar	telefono_celular varchar
	correo_electronico varchar

sucursal	descuento
id int	id int
ciudad varchar	fecha_inicio int
direccion varchar	fecha_fin int
telefono_fijo varchar	porcentaje int
telefono_celular varchar	
correo_electronico varchar	

tipo_vehiculo	
🔑 id	int
♀ id_descuento	int
nombre	varchar
precio_semana	decimal
precio_dia	decimal

vehiculo	
🔑 id	int
♀ id_tipo	int
♀ id_sucursal	bigint
placa	varchar
referencia	varchar
modelos	varchar
puertas	int
capacidad	int
sunroof	boolean
motor	varchar
color	varchar

alquiler	
🔑 id	int
♀ id_vehiculo	int
♀ id_cliente	int
♀ id_empleado	int
♀ id_sucursal_salida	int
fecha_salida	date
♀ id_sucursal_llegada	int
fecha_llegada	date
fecha_esperada	date
valor_cotizado	decimal
valor_pagado	decimal

Segunda Forma Normal (2FN)

Para que una tabla pueda estar en segunda forma normal o 2FN necesita cumplir con los siguientes requerimientos

- Ya debe estar en 1NF
- No tiene dependencia parcial. Es decir, todos los atributos no claves son totalmente dependientes de la clave primaria

Descripción

No se realizan ningún cambio en las tablas, ya que todas cumplen con la segunda forma normal porque ya están en 1NF y no tiene dependencia parcial porque todos los atributos son totalmente dependientes de la clave primaria

Gráfica

cliente		
🔑 id		int
cedula		int
nombres	varchar	
apellidos	varchar	
direccion	varchar	
ciudad_recidencia	varchar	
telefono_celular	varchar	
correo_electronico	varchar	

sucursal		
🔑 id		int
ciudad	varchar	
direccion	varchar	
telefono_fijo	varchar	
telefono_celular	varchar	
correo_electronico	varchar	

empleado		
🔑 id		int
♀ id_sucursal		int
cedula		int
nombres	varchar	
apellidos	varchar	
direccion	varchar	
ciudad_recidencia	varchar	
telefono_celular	varchar	
correo_electronico	varchar	

descuento		
🔑 id		int
fecha_inicio		int
fecha_fin		int
porcentaje		int

tipo_vehiculo		
🔑 id		int
♀ id_descuento		int
nombre	varchar	
precio_semana	decimal	
precio_dia	decimal	

vehiculo		
🔑 id		int
♀ id_tipo		int
♀ id_sucursal		bigint
placa	varchar	
referencia	varchar	
modelos	varchar	
puertas		int
capacidad		int
sunroof		boolean
motor		varchar
color		varchar

alquiler	
 id	int
♀ id_vehiculo	int
♀ id_cliente	int
♀ id_empleado	int
♀ id_sucursal_salida	int
fecha_salida	date
♀ id_sucursal_llegada	int
fecha_llegada	date
fecha_esperada	date
porcentaje_descuento	int
valor_cotizado	decimal
valor_pagado	decimal

Tercera Forma Normal (3FN)

Para que una tabla tenga tercera forma normal o 3FN debe cumplir con los siguientes requerimientos:

- ★ Debe estar en 2NF
- ★ No tiene dependencia parcial transitiva

Descripción

No se realizan ningún cambio en las tablas, ya que todas cumplen con la segunda forma normal porque ya están en 2NF y no tiene dependencia parcial transitiva.

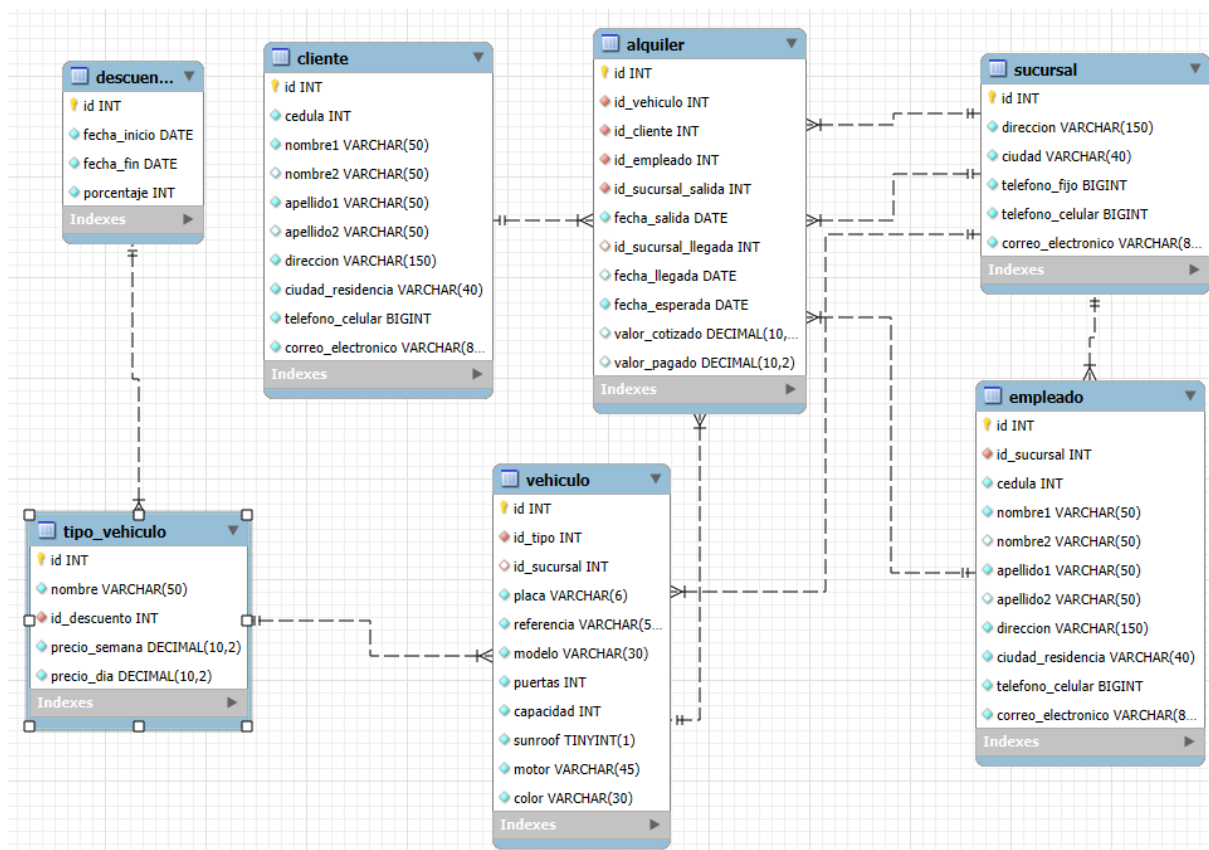
Gráfica



alquiler

 id	int
♀ id_vehiculo	int
♀ id_cliente	int
♀ id_empleado	int
♀ id_sucursal_salida	int
fecha_salida	date
♀ id_sucursal_llegada	int
fecha_llegada	date
fecha_esperada	date
porcentaje_descuento	int
valor_cotizado	decimal
valor_pagado	decimal

Construcción del Modelo Físico



Descripción

ya con la normalización realizada y sabiendo las relaciones que hay entre todas las tablas juntos con los tipos de datos que tiene cada uno se realizara el modelo físico el cual se hará con código de MySQL y con base en ese código se creara el diagrama en la aplicación MySQL workbench

Creación de tablas

```
drop database if exists AutoRenta;

create database AutoRenta;

use AutoRenta;

create table cliente(
    id int primary key not null,
    cedula int(15) not null,
    nombre1 varchar(50) not null,
    nombre2 varchar(50),
    apellido1 varchar(50) not null,
    apellido2 varchar(50),
```

```

        direccion varchar(150) not null,
        ciudad_residencia varchar(40) not null,
        telefono_celular bigint not null,
        correo_electronico varchar(80) not null
    );

create table sucursal(
    id int primary key not null,
    direccion varchar(150) not null,
    ciudad varchar(40) not null,
    telefono_fijo bigint not null,
    telefono_celular bigint not null,
    correo_electronico varchar(80) not null
);

create table empleado(
    id int primary key not null,
    id_sucursal int not null,
    cedula int(15) not null,
    nombre1 varchar(50) not null,
    nombre2 varchar(50),
    apellido1 varchar(50) not null,
    apellido2 varchar(50),
    direccion varchar(150) not null,
    ciudad_residencia varchar(40) not null,
    telefono_celular bigint not null,
    correo_electronico varchar(80) not null,
    foreign key (id_sucursal) references sucursal(id)
);

create table descuento(
    id int primary key not null,
    fecha_inicio date not null,
    fecha_fin date not null,
    porcentaje int not null
);

create table tipo_vehiculo(
    id int primary key not null,
    nombre varchar(50) not null,
    id_descuento int not null,
    precio_semana decimal(10,2) not null,
    precio_dia decimal(10,2) not null,
    foreign key (id_descuento) references descuento(id)
);

```

```

create table vehiculo(
    id int primary key not null,
    id_tipo int not null,
    id_sucursal int,
    placa varchar(6) not null,
    referencia varchar(50) not null,
    modelo varchar(30) not null,
    puertas int not null,
    capacidad int not null,
    sunroof boolean not null,
    motor varchar(45) not null,
    color varchar(30) not null,
    foreign key (id_tipo) references tipo_vehiculo(id),
    foreign key (id_sucursal) references sucursal(id)
);

create table alquiler (
    id int primary key not null,
    id_vehiculo int not null,
    id_cliente int not null,
    id_empleado int not null,
    id_sucursal_salida int not null,
    fecha_salida date not null,
    id_sucursal_llegada int ,
    fecha_llegada date,
    fecha_esperada date not null,
    valor_cotizado decimal(10,2),
    valor_pagado decimal(10,2),
    foreign key (id_vehiculo) references vehiculo(id),
    foreign key (id_cliente) references cliente(id),
    foreign key (id_empleado) references empleado(id),
    foreign key (id_sucursal_salida) references sucursal(id),
    foreign key (id_sucursal_llegada) references sucursal(id)
);

```

Insercion de datos

```

use AutoRenta;

INSERT INTO cliente VALUES
(1, 1023456789, 'Andrea', 'Carolina', 'Ramírez', 'Suárez', 'Calle 127 #45-23 Apto
502', 'Bogotá', 3157894561, 'andrea.ramirez@gmail.com'),
(2, 1098765432, 'Carlos', 'Eduardo', 'Martínez', 'López', 'Carrera 56 #23-45 Torre
3', 'Medellín', 3169874523, 'carlos.martinez@hotmail.com'),

```

(3, 1045678912, 'María', 'Fernanda', 'González', 'Pérez', 'Avenida 68 #89-12', 'Cali', 3178945612, 'mafe.gonzalez@yahoo.com'),
(4, 1056789123, 'Juan', 'Sebastián', 'López', 'Rodríguez', 'Calle 85 #15-45 Edificio Luna', 'Barranquilla', 3189632547, 'jsebastian.lopez@gmail.com'),
(5, 1034567891, 'Diana', 'Patricia', 'Herrera', 'Vargas', 'Carrera 23 #78-90', 'Bucaramanga', 3145896321, 'diana.herrera@outlook.com'),
(6, 1067891234, 'Ricardo', 'Alberto', 'Sánchez', 'Torres', 'Calle 93 #12-34 Int 4', 'Bogotá', 3157893214, 'ricardo.sanchez@gmail.com'),
(7, 1078912345, 'Laura', 'Valentina', 'Torres', 'Muñoz', 'Avenida Caracas #45-67', 'Medellín', 3168947563, 'laura.torres@hotmail.com'),
(8, 1089123456, 'Daniel', 'Felipe', 'García', 'Castro', 'Carrera 7 #82-25', 'Cali', 3179845612, 'daniel.garcia@yahoo.com'),
(9, 1091234567, 'Natalia', 'Andrea', 'Castro', 'Ruiz', 'Calle 116 #17-89', 'Bogotá', 3184567891, 'natalia.castro@gmail.com'),
(10, 1012345678, 'Jorge', 'Luis', 'Ruiz', 'Martínez', 'Carrera 43 #65-43', 'Barranquilla', 3146789123, 'jorge.ruiz@outlook.com'),
(11, 1087654321, 'Carolina', NULL, 'Morales', 'Silva', 'Avenida Suba #145-67', 'Bogotá', 3158912345, 'carolina.morales@gmail.com'),
(12, 1076543219, 'Andrés', 'Felipe', 'Silva', 'Gómez', 'Calle 80 #23-45', 'Medellín', 3167891234, 'andres.silva@hotmail.com'),
(13, 1065432198, 'Mónica', 'Patricia', 'Gómez', 'Pérez', 'Carrera 68 #56-78', 'Cali', 3176543219, 'monica.gomez@yahoo.com'),
(14, 1054321987, 'Santiago', NULL, 'Pérez', 'Ramírez', 'Avenida Boyacá #78-90', 'Bogotá', 3185678912, 'santiago.perez@gmail.com'),
(15, 1043219876, 'Paula', 'Andrea', 'Ramírez', 'González', 'Calle 72 #14-56', 'Bucaramanga', 3147891234, 'paula.ramirez@outlook.com'),
(16, 1032198765, 'Miguel', 'Ángel', 'González', 'García', 'Carrera 15 #93-45', 'Bogotá', 3156789123, 'miguel.gonzalez@gmail.com'),
(17, 1021987654, 'Valentina', NULL, 'García', 'Sánchez', 'Avenida Ciudad de Cali #34-67', 'Medellín', 3165432198, 'valentina.garcia@hotmail.com'),
(18, 1019876543, 'David', 'Santiago', 'Sánchez', 'Castro', 'Calle 147 #13-45', 'Cali', 3174567891, 'david.sanchez@yahoo.com'),
(19, 1098765431, 'Isabella', NULL, 'Castro', 'Torres', 'Carrera 11 #82-71', 'Bogotá', 3183219876, 'isabella.castro@gmail.com'),
(20, 1087654329, 'Juan', 'Pablo', 'Torres', 'López', 'Avenida 19 #123-45', 'Barranquilla', 3149876543, 'juanp.torres@outlook.com'),
(21, 1076543218, 'Camila', 'Andrea', 'López', 'Herrera', 'Calle 100 #15-23', 'Bogotá', 3155432198, 'camila.lopez@gmail.com'),
(22, 1065432187, 'Felipe', NULL, 'Herrera', 'Morales', 'Carrera 9 #115-06', 'Medellín', 3164567891, 'felipe.herrera@hotmail.com'),
(23, 1054321986, 'María', 'Camila', 'Morales', 'Silva', 'Avenida El Dorado #76-23', 'Cali', 3173219876, 'mariac.morales@yahoo.com'),
(24, 1043219875, 'Sebastián', NULL, 'Silva', 'Gómez', 'Calle 134 #19-45', 'Bogotá', 3182198765, 'sebastian.silva@gmail.com'),

```
(25, 1032198764, 'Ana', 'María', 'Gómez', 'Pérez', 'Carrera 21 #45-67',
'Bucaramanga', 3148912345, 'ana.gomez@outlook.com'),
(26, 1021987653, 'Cristian', 'David', 'Pérez', 'García', 'Carrera 8 #45-34',
'Medellín', 3154567891, 'cristian.perez@gmail.com'),
(27, 1098765430, 'Luisa', 'Fernanda', 'García', 'López', 'Calle 67 #89-45',
'Bogotá', 3163219876, 'luisa.garcia@hotmail.com'),
(28, 1087654328, 'Mateo', NULL, 'López', 'Ramírez', 'Avenida 10 #78-90', 'Cali',
3176543281, 'mateo.lopez@yahoo.com'),
(29, 1076543217, 'Juliana', 'Carolina', 'Ramírez', 'Torres', 'Calle 45 #23-56',
'Barranquilla', 3187893216, 'juliana.ramirez@gmail.com'),
(30, 1065432186, 'Manuel', NULL, 'Torres', 'Vargas', 'Carrera 13 #45-12', 'Bogotá',
3149876542, 'manuel.torres@hotmail.com'),
(31, 1054321985, 'Catalina', 'Andrea', 'Vargas', 'Martínez', 'Avenida 45 #12-34',
'Medellín', 3157896541, 'catalina.vargas@gmail.com'),
(32, 1043219874, 'Simón', NULL, 'Martínez', 'Herrera', 'Calle 98 #76-54', 'Cali',
3165432178, 'simon.martinez@hotmail.com'),
(33, 1032198763, 'Gabriela', 'Patricia', 'Herrera', 'Ruiz', 'Carrera 21 #45-12',
'Bogotá', 3174567812, 'gabriela.herrera@gmail.com'),
(34, 1021987652, 'Tomás', NULL, 'Ruiz', 'Gómez', 'Avenida 89 #56-23',
'Bucaramanga', 3183216547, 'tomas.ruiz@outlook.com'),
(35, 1098765429, 'Alejandra', 'María', 'Gómez', 'Sánchez', 'Carrera 34 #78-12',
'Barranquilla', 3145637892, 'alejandra.gomez@yahoo.com'),
(36, 1087654327, 'Nicolás', 'Antonio', 'Sánchez', 'López', 'Calle 45 #23-34',
'Bogotá', 3159876543, 'nicolas.sanchez@hotmail.com'),
(37, 1076543216, 'Mariana', NULL, 'López', 'García', 'Carrera 8 #34-56',
'Medellín', 3165432789, 'mariana.lopez@gmail.com'),
(38, 1065432185, 'Martín', 'David', 'García', 'Ruiz', 'Avenida 23 #45-67', 'Cali',
3176543289, 'martin.garcia@yahoo.com'),
(39, 1054321984, 'Victoria', 'Camila', 'Ruiz', 'Torres', 'Calle 56 #23-12',
'Bogotá', 3187896543, 'victoria.ruiz@gmail.com'),
(40, 1043219873, 'Emilio', NULL, 'Torres', 'Castro', 'Carrera 14 #89-45',
'Barranquilla', 3145698732, 'emilio.torres@outlook.com');
```

```
-- insercion en tala sucursal
```

```
INSERT INTO sucursal VALUES
```

```
(1, 'Calle 45 #23-12', 'Bogotá', 6015784596, 3157894561,
'sucursal1.bogota@rentacar.com'),
(2, 'Carrera 70 #45-89', 'Medellín', 6044563214, 3169874563,
'sucursal1.medellin@rentacar.com'),
(3, 'Avenida 5 #21-45', 'Cali', 6025896547, 3178945612,
'sucursal1.cali@rentacar.com'),
(4, 'Calle 82 #45-23', 'Barranquilla', 6053214578, 3189632587,
'sucursal1.barranquilla@rentacar.com');
```

```
(5, 'Carrera 27 #56-89', 'Bucaramanga', 6076549873, 3145896321,
'sucursal1.bucaramanga@rentacar.com');

-- insercion en tabla de empleados

INSERT INTO empleado VALUES
(1, 3, 79845612, 'Manuel', 'Antonio', 'Pérez', 'Gómez', 'Calle 123 #45-67',
'Bogotá', 3157894561, 'manuel.perez@rentacar.com'),
(2, 5, 80123456, 'Ana', 'Maria', 'López', 'Sánchez', 'Carrera 78 #90-12',
'Medellín', 3169874523, 'ana.lopez@rentacar.com'),
(3, 1, 81234567, 'Carlos', 'Alberto', 'Rodríguez', 'Torres', 'Avenida 45 #23-56',
'Cali', 3178945612, 'carlos.rodriguez@rentacar.com'),
(4, 2, 82345678, 'Laura', 'Sofia', 'García', 'Martínez', 'Calle 67 #89-01',
'Barranquilla', 3189632547, 'laura.garcia@rentacar.com'),
(5, 4, 83456789, 'Diego', 'Alejandro', 'Martínez', 'Ruiz', 'Carrera 34 #56-78',
'Bucaramanga', 3145896321, 'diego.martinez@rentacar.com'),
(6, 2, 84567890, 'María', 'Fernanda', 'Sánchez', 'Pérez', 'Avenida 89 #12-34',
'Bogotá', 3157893214, 'maria.sanchez@rentacar.com'),
(7, 1, 85678901, 'Juan', 'Pablo', 'Torres', 'García', 'Calle 56 #78-90',
'Medellín', 3168947563, 'juan.torres@rentacar.com'),
(8, 5, 86789012, 'Valentina', NULL, 'Ruiz', 'López', 'Carrera 12 #34-56', 'Cali',
3179845612, 'valentina.ruiz@rentacar.com'),
(9, 3, 87890123, 'Andrés', 'Felipe', 'Gómez', 'Martínez', 'Avenida 67 #89-12',
'Barranquilla', 3184567891, 'andres.gomez@rentacar.com'),
(10, 4, 88901234, 'Camila', 'Andrea', 'López', 'Torres', 'Calle 90 #12-34',
'Bucaramanga', 3146789123, 'camila.lopez@rentacar.com'),
(11, 1, 89012345, 'Santiago', NULL, 'Martínez', 'García', 'Carrera 23 #45-67',
'Bogotá', 3158912345, 'santiago.martinez@rentacar.com'),
(12, 3, 90123456, 'Isabella', NULL, 'García', 'Sánchez', 'Avenida 78 #90-12',
'Medellín', 3167891234, 'isabella.garcia@rentacar.com'),
(13, 5, 91234567, 'Daniel', 'Alberto', 'Sánchez', 'Rodríguez', 'Calle 34 #56-78',
'Cali', 3176543219, 'daniel.sanchez@rentacar.com'),
(14, 2, 92345678, 'Mariana', NULL, 'Rodríguez', 'Pérez', 'Carrera 89 #12-34',
'Bogotá', 3185678912, 'mariana.rodriguez@rentacar.com'),
(15, 4, 93456789, 'José', 'Miguel', 'Pérez', 'Torres', 'Avenida 45 #67-89',
'Barranquilla', 3147891234, 'jose.perez@rentacar.com'),
(16, 1, 94567890, 'Gabriela', NULL, 'Torres', 'Ruiz', 'Calle 12 #34-56',
'Bucaramanga', 3156789123, 'gabriela.torres@rentacar.com'),
(17, 3, 95678901, 'Felipe', 'Andrés', 'Ruiz', 'Gómez', 'Carrera 67 #89-12',
'Bogotá', 3165432198, 'felipe.ruiz@rentacar.com'),
(18, 5, 96789012, 'Catalina', NULL, 'Gómez', 'López', 'Avenida 90 #12-34',
'Medellín', 3174567891, 'catalina.gomez@rentacar.com'),
(19, 2, 97890123, 'Eduardo', 'José', 'López', 'Martínez', 'Calle 23 #45-67',
'Cali', 3183219876, 'eduardo.lopez@rentacar.com'),
```

```
(20, 4, 98901234, 'Paula', 'Andrea', 'Martínez', 'García', 'Carrera 78 #90-12',
'Barranquilla', 3149876543, 'paula.martinez@rentacar.com'),
(21, 1, 99012345, 'Ricardo', NULL, 'García', 'Sánchez', 'Avenida 34 #56-78',
'Bogotá', 3155432198, 'ricardo.garcia@rentacar.com'),
(22, 3, 10012345, 'Carolina', NULL, 'Sánchez', 'Torres', 'Calle 89 #12-34',
'Medellín', 3164567891, 'carolina.sanchez@rentacar.com'),
(23, 5, 10123456, 'Sebastián', NULL, 'Torres', 'Ruiz', 'Carrera 45 #67-89', 'Cali',
3173219876, 'sebastian.torres@rentacar.com'),
(24, 2, 10234567, 'Natalia', NULL, 'Ruiz', 'López', 'Avenida 12 #34-56', 'Bogotá',
3182198765, 'natalia.ruiz@rentacar.com'),
(25, 4, 10345678, 'Jorge', 'Luis', 'López', 'Gómez', 'Calle 67 #89-12',
'Barranquilla', 3148912345, 'jorge.lopez@rentacar.com'),
(26, 1, 10456789, 'María', 'José', 'Gómez', 'Martínez', 'Carrera 90 #12-34',
'Bucaramanga', 3154321987, 'mariaj.gomez@rentacar.com'),
(27, 3, 10567890, 'David', 'Santiago', 'Martínez', 'Pérez', 'Avenida 23 #45-67',
'Bogotá', 3163219876, 'david.martinez@rentacar.com'),
(28, 5, 10678901, 'Andrea', 'Carolina', 'Pérez', 'García', 'Calle 78 #90-12',
'Medellín', 3172198765, 'andrea.perez@rentacar.com'),
(29, 2, 10789012, 'Luis', 'Felipe', 'García', 'Sánchez', 'Carrera 34 #56-78',
'Cali', 3181987654, 'luis.garcia@rentacar.com'),
(30, 4, 10890123, 'Sara', 'Valentina', 'Sánchez', 'Torres', 'Avenida 89 #12-34',
'Barranquilla', 3141234567, 'sara.sanchez@rentacar.com'),
(31, 1, 10901234, 'Juan', 'David', 'Torres', 'Ruiz', 'Calle 45 #67-89', 'Bogotá',
3153219876, 'juan.david@rentacar.com'),
(32, 3, 11012345, 'Daniela', NULL, 'Ruiz', 'López', 'Carrera 12 #34-56',
'Medellín', 3162198765, 'daniela.ruiz@rentacar.com'),
(33, 5, 11123456, 'Carlos', 'Eduardo', 'López', 'Gómez', 'Avenida 67 #89-12',
'Cali', 3171987654, 'carlos.eduardo@rentacar.com'),
(34, 2, 11234567, 'Valeria', NULL, 'Gómez', 'Martínez', 'Calle 90 #12-34',
'Bogotá', 3181234567, 'valeria.gomez@rentacar.com'),
(35, 4, 11345678, 'Samuel', NULL, 'Martínez', 'Pérez', 'Carrera 23 #45-67',
'Barranquilla', 3143219876, 'samuel.martinez@rentacar.com');
```

```
-- insercion entabla de descuento
```

```
INSERT INTO descuento VALUES
```

```
(1, '2024-01-01', '2024-02-29', 10),
(2, '2024-03-01', '2024-03-31', 15),
(3, '2024-04-01', '2024-05-31', 20),
(4, '2024-06-15', '2024-07-15', 25),
(5, '2024-07-16', '2024-08-31', 30),
(6, '2024-09-01', '2024-10-31', 12),
(7, '2024-11-01', '2024-11-30', 18),
(8, '2024-12-01', '2024-12-14', 22),
```

```

(9, '2024-12-15', '2024-12-31', 35),
(10, '2024-01-01', '2024-12-31', 5);

-- insercion en tabla de vehiculo

INSERT INTO tipo_vehiculo VALUES
(1, 'Sedán Económico', 1, 450000.00, 85000.00),
(2, 'Sedán Ejecutivo', 2, 650000.00, 120000.00),
(3, 'SUV Compacta', 3, 750000.00, 140000.00),
(4, 'SUV Mediana', 4, 850000.00, 160000.00),
(5, 'SUV Premium', 5, 1200000.00, 220000.00),
(6, 'Pickup Doble Cabina', 6, 900000.00, 170000.00),
(7, 'Van Pasajeros', 7, 1100000.00, 200000.00),
(8, 'Deportivo Básico', 8, 1500000.00, 280000.00),
(9, 'Deportivo Premium', 9, 2000000.00, 380000.00),
(10, 'Lujo Ejecutivo', 6, 2500000.00, 450000.00),
(11, 'Compacto Automático', 1, 500000.00, 95000.00),
(12, 'Compacto Manual', 2, 400000.00, 75000.00),
(13, 'Crossover', 3, 800000.00, 150000.00),
(14, 'Híbrido Sedán', 4, 900000.00, 170000.00),
(15, 'Híbrido SUV', 5, 1300000.00, 240000.00);

-- insercion en la tabla de vehiculo

INSERT INTO vehiculo VALUES
(1, 1, 3, 'ABC123', 'Chevrolet Joy', '2024', 4, 5, false, '1.4L', 'Blanco'),
(2, 2, 5, 'DEF456', 'Toyota Corolla', '2024', 4, 5, true, '2.0L', 'Negro'),
(3, 3, 1, 'GHI789', 'Hyundai Creta', '2023', 5, 5, true, '1.6L', 'Plata'),
(4, 4, 2, 'JKL012', 'Mazda CX-30', '2024', 5, 5, true, '2.0L', 'Rojo'),
(5, 5, 4, 'MNO345', 'BMW X3', '2024', 5, 5, true, '2.5L', 'Gris'),
(6, 6, 3, 'PQR678', 'Toyota Hilux', '2023', 4, 5, false, '2.8L Diesel', 'Blanco'),
(7, 7, 1, 'STU901', 'Hyundai H1', '2024', 5, 12, false, '2.5L Diesel', 'Plata'),
(8, 8, 5, 'VWX234', 'Ford Mustang', '2024', 2, 4, true, '5.0L', 'Rojo'),
(9, 9, 2, 'YZA567', 'Chevrolet Camaro', '2024', 2, 4, true, '6.2L', 'Amarillo'),
(10, 10, 4, 'BCD890', 'Mercedes Benz C200', '2024', 4, 5, true, '2.0L Turbo', 'Negro'),
(11, 11, 3, 'EFG123', 'Kia Picanto', '2024', 4, 5, false, '1.2L', 'Azul'),
(12, 12, 1, 'HIJ456', 'Chevrolet Onix', '2023', 4, 5, false, '1.0L Turbo', 'Rojo'),
(13, 13, 5, 'KLM789', 'Renault Captur', '2024', 5, 5, true, '1.6L', 'Naranja'),
(14, 14, 2, 'NOP012', 'Toyota Corolla Hybrid', '2024', 4, 5, true, '1.8L Hybrid', 'Blanco'),
(15, 15, 4, 'QRS345', 'Toyota RAV4 Hybrid', '2024', 5, 5, true, '2.5L Hybrid', 'Verde'),
(16, 1, 3, 'TUV678', 'Chevrolet Joy', '2023', 4, 5, false, '1.4L', 'Plata'),
(17, 2, 5, 'WXY901', 'Toyota Corolla', '2023', 4, 5, true, '2.0L', 'Gris'),

```



```
(18, 3, 1, 'ZAB234', 'Hyundai Creta', '2024', 5, 5, true, '1.6L', 'Blanco'),
(19, 4, 2, 'CDE567', 'Mazda CX-30', '2023', 5, 5, true, '2.0L', 'Azul'),
(20, 5, 4, 'FGH890', 'BMW X3', '2023', 5, 5, true, '2.5L', 'Negro'),
(21, 6, 3, 'IJK123', 'Toyota Hilux', '2024', 4, 5, false, '2.8L Diesel', 'Plata'),
(22, 7, 1, 'LMN456', 'Hyundai H1', '2023', 5, 12, false, '2.5L Diesel', 'Blanco'),
(23, 8, 5, 'OPQ789', 'Ford Mustang', '2023', 2, 4, true, '5.0L', 'Negro'),
(24, 9, 2, 'RST012', 'Chevrolet Camaro', '2024', 2, 4, true, '6.2L', 'Rojo'),
(25, 10, 4, 'UVW345', 'Mercedes Benz C200', '2023', 4, 5, true, '2.0L Turbo',
'Plata'),
(26, 11, 3, 'XYZ678', 'Kia Picanto', '2024', 4, 5, false, '1.2L', 'Rojo'),
(27, 12, 1, 'ABC901', 'Chevrolet Onix', '2024', 4, 5, false, '1.0L Turbo',
'Blanco'),
(28, 13, 5, 'DEF234', 'Renault Captur', '2023', 5, 5, true, '1.6L', 'Gris'),
(29, 14, 2, 'GHI567', 'Toyota Corolla Hybrid', '2023', 4, 5, true, '1.8L Hybrid',
'Negro'),
(30, 15, 4, 'JKL890', 'Toyota RAV4 Hybrid', '2024', 5, 5, true, '2.5L Hybrid',
'Azul'),
(31, 1, 3, 'MNO123', 'Chevrolet Joy', '2024', 4, 5, false, '1.4L', 'Rojo'),
(32, 2, 5, 'PQR456', 'Toyota Corolla', '2024', 4, 5, true, '2.0L', 'Blanco'),
(33, 3, 1, 'STU789', 'Hyundai Creta', '2023', 5, 5, true, '1.6L', 'Negro'),
(34, 4, 2, 'VWX012', 'Mazda CX-30', '2024', 5, 5, true, '2.0L', 'Plata'),
(35, 5, 4, 'YZA345', 'BMW X3', '2024', 5, 5, true, '2.5L', 'Gris'),
(36, 6, 3, 'BCD678', 'Toyota Hilux', '2023', 4, 5, false, '2.8L Diesel', 'Blanco'),
(37, 7, 1, 'EFG901', 'Hyundai H1', '2024', 5, 12, false, '2.5L Diesel', 'Negro'),
(38, 8, 5, 'HIJ234', 'Ford Mustang', '2024', 2, 4, true, '5.0L', 'Azul'),
(39, 9, 2, 'KLM567', 'Chevrolet Camaro', '2023', 2, 4, true, '6.2L', 'Rojo'),
(40, 10, 4, 'NOP890', 'Mercedes Benz C200', '2024', 4, 5, true, '2.0L Turbo',
'Blanco');
```

```
-- insercion de datos en la tabla alquiler
```

```
INSERT INTO alquiler VALUES
```

```
(1, 12, 20, 5, 1, '2023-07-10', 3, '2023-07-14', '2023-07-12', 180000.00,
180000.00),
(2, 6, 5, 15, 4, '2023-04-02', 1, '2023-04-27', '2023-04-26', 220000.00,
220000.00),
(3, 18, 33, 11, 2, '2023-05-15', 5, '2023-05-20', '2023-05-19', 250000.00,
245000.00),
(4, 2, 4, 19, 3, '2023-06-08', 4, '2023-06-12', '2023-06-10', 190000.00,
190000.00),
(5, 40, 11, 3, 1, '2023-08-20', 2, '2023-08-23', '2023-08-22', 300000.00,
295000.00),
(6, 28, 22, 12, 5, '2023-07-05', 3, '2023-07-19', '2023-07-17', 270000.00,
265000.00),
```

```
(7, 34, 19, 2, 2, '2023-09-12', 1, '2023-09-16', '2023-09-15', 150000.00,
150000.00),
(8, 1, 28, 14, 4, '2023-06-20', 5, '2023-06-25', '2023-06-23', 200000.00,
200000.00),
(9, 30, 7, 7, 3, '2023-05-05', 4, '2023-05-08', '2023-05-07', 230000.00,
225000.00),
(10, 24, 15, 20, 1, '2023-06-25', 2, '2023-06-28', '2023-06-27', 260000.00,
255000.00),
(11, 13, 30, 8, 3, '2023-08-01', 1, '2023-08-06', '2023-08-04', 280000.00,
270000.00),
(12, 8, 27, 4, 5, '2023-07-13', 2, '2023-07-26', '2023-07-26', 220000.00,
215000.00),
(13, 35, 3, 6, 2, '2023-05-22', 5, '2023-05-26', '2023-05-24', 320000.00,
315000.00),
(14, 9, 1, 13, 4, '2023-09-08', 3, '2023-09-12', '2023-09-10', 190000.00,
185000.00),
(15, 32, 24, 18, 1, '2023-08-15', 1, '2023-08-19', '2023-08-17', 220000.00,
220000.00),
(16, 20, 18, 9, 3, '2023-06-30', 2, '2023-07-04', '2023-07-02', 250000.00,
245000.00),
(17, 38, 26, 16, 5, '2023-07-22', 4, '2023-07-27', '2023-07-25', 190000.00,
190000.00),
(18, 14, 10, 17, 2, '2023-09-01', 3, '2023-09-24', '2023-09-24', 300000.00,
295000.00),
(19, 11, 21, 10, 4, '2023-07-19', 1, '2023-07-30', '2023-07-30', 270000.00,
265000.00),
(20, 19, 9, 12, 5, '2023-06-18', 3, '2023-06-22', '2023-06-20', 240000.00,
235000.00),
(21, 3, 31, 23, 2, '2023-09-13', 5, '2023-09-18', '2023-09-16', 260000.00,
255000.00),
(22, 16, 17, 5, 1, '2023-08-12', 4, '2023-08-16', '2023-08-14', 220000.00,
215000.00),
(23, 7, 14, 27, 3, '2023-09-17', 1, '2023-09-21', '2023-09-19', 210000.00,
205000.00),
(24, 22, 5, 25, 4, '2023-06-27', 2, '2023-07-02', '2023-06-30', 200000.00,
200000.00),
(25, 39, 23, 3, 1, '2023-05-10', 4, '2023-05-14', '2023-05-12', 300000.00,
295000.00),
(26, 10, 28, 6, 5, '2023-07-30', 1, '2023-08-21', '2023-08-21', 220000.00,
215000.00),
(27, 4, 25, 11, 2, '2023-06-01', 3, '2023-06-04', '2023-06-03', 240000.00,
235000.00),
(28, 29, 20, 22, 1, '2023-08-28', 5, '2023-09-02', '2023-08-31', 180000.00,
175000.00),
```

```
(29, 23, 34, 8, 4, '2023-05-25', 2, '2023-05-30', '2023-05-28', 210000.00,
205000.00),
(30, 12, 32, 21, 3, '2023-07-04', 5, '2023-07-09', '2023-07-07', 280000.00,
275000.00),
(31, 31, 8, 28, 2, '2023-08-25', 1, '2023-08-30', '2023-08-29', 230000.00,
225000.00),
(32, 27, 12, 24, 5, '2023-09-20', 3, '2023-09-25', '2023-09-23', 270000.00,
265000.00),
(33, 33, 35, 4, 1, '2023-06-15', 2, '2023-06-18', '2023-06-17', 220000.00,
215000.00),
(34, 37, 29, 9, 4, '2023-08-07', 1, '2023-08-11', '2023-08-09', 240000.00,
235000.00),
(35, 25, 16, 30, 3, '2023-07-23', 4, '2023-07-27', '2023-07-25', 300000.00,
295000.00),
(36, 26, 18, 13, 2, '2023-08-04', 5, '2023-08-09', '2023-08-07', 250000.00,
245000.00),
(37, 36, 6, 7, 1, '2023-09-11', 3, '2023-09-14', '2023-09-13', 260000.00,
255000.00),
(38, 21, 3, 5, 4, '2023-06-10', 2, '2023-06-13', '2023-06-12', 230000.00,
225000.00),
(39, 5, 11, 17, 2, '2023-07-02', 4, '2023-07-06', '2023-07-04', 240000.00,
235000.00),
(40, 15, 30, 12, 5, '2023-08-10', 1, '2023-08-22', '2023-08-22', 210000.00,
205000.00),
(41, 4, 22, 9, 3, '2023-09-01', 2, '2023-09-05', '2023-09-03', 210000.00,
205000.00),
(42, 26, 35, 15, 4, '2023-08-12', 1, '2023-08-16', '2023-08-14', 270000.00,
265000.00),
(43, 8, 29, 4, 5, '2023-07-20', 3, '2023-07-23', '2023-07-22', 240000.00,
235000.00),
(44, 21, 10, 13, 1, '2023-06-10', 2, '2023-06-14', '2023-06-12', 230000.00,
225000.00),
(45, 11, 16, 18, 3, '2023-09-25', 5, '2023-09-28', '2023-09-27', 280000.00,
275000.00),
(46, 16, 7, 20, 2, '2023-07-08', 4, '2023-07-12', '2023-07-10', 250000.00,
245000.00),
(47, 31, 33, 5, 1, '2023-08-20', 1, '2023-08-24', '2023-08-22', 220000.00,
215000.00),
(48, 33, 19, 6, 4, '2023-09-15', 2, '2023-09-19', '2023-09-17', 270000.00,
265000.00),
(49, 22, 8, 23, 5, '2023-06-05', 1, '2023-06-09', '2023-06-07', 240000.00,
235000.00),
(50, 12, 11, 24, 2, '2023-05-18', 4, '2023-05-22', '2023-05-20', 200000.00,
195000.00),
```

```
(51, 37, 2, 30, 3, '2023-09-07', 3, '2023-09-11', '2023-09-09', 260000.00,
255000.00),
(52, 39, 26, 21, 4, '2023-07-22', 5, '2023-07-26', '2023-07-24', 250000.00,
245000.00),
(53, 19, 28, 12, 1, '2023-06-15', 4, '2023-06-19', '2023-06-17', 230000.00,
225000.00),
(54, 9, 14, 8, 2, '2023-08-05', 3, '2023-08-17', '2023-08-17', 240000.00,
235000.00),
(55, 28, 18, 7, 5, '2023-07-10', 2, '2023-07-14', '2023-07-12', 250000.00,
245000.00),
(56, 24, 23, 19, 3, '2023-06-11', 1, '2023-06-15', '2023-06-13', 230000.00,
225000.00),
(57, 25, 21, 9, 4, '2023-09-05', 3, '2023-09-09', '2023-09-07', 260000.00,
255000.00),
(58, 6, 1, 14, 2, '2023-05-30', 4, '2023-06-11', '2023-06-11', 240000.00,
235000.00),
(59, 15, 3, 5, 1, '2023-08-18', 2, '2023-08-22', '2023-08-20', 270000.00,
265000.00),
(60, 30, 9, 11, 4, '2023-09-04', 5, '2023-09-08', '2023-09-06', 220000.00,
215000.00),
(61, 10, 30, 12, 3, '2023-07-19', 2, '2023-07-23', '2023-07-21', 210000.00,
205000.00),
(62, 17, 33, 13, 1, '2023-06-23', 5, '2023-06-27', '2023-06-25', 230000.00,
225000.00),
(63, 20, 12, 25, 4, '2023-08-12', 3, '2023-08-16', '2023-08-14', 220000.00,
215000.00),
(64, 34, 27, 6, 2, '2023-07-01', 4, '2023-07-05', '2023-07-03', 240000.00,
235000.00),
(65, 29, 25, 10, 3, '2023-06-17', 5, '2023-06-21', '2023-06-19', 250000.00,
245000.00),
(66, 32, 35, 16, 4, '2023-09-01', 1, '2023-09-13', '2023-09-13', 230000.00,
225000.00),
(67, 14, 11, 8, 5, '2023-07-09', 3, '2023-07-13', '2023-07-11', 220000.00,
215000.00),
(68, 13, 5, 19, 2, '2023-08-22', 2, '2023-08-26', '2023-08-24', 240000.00,
235000.00),
(69, 23, 1, 4, 1, '2023-07-25', 5, '2023-07-29', '2023-07-27', 270000.00,
265000.00),
(70, 5, 22, 12, 3, '2023-08-30', 4, '2023-09-21', '2023-09-21', 250000.00,
245000.00),
(71, 18, 14, 3, 4, '2023-06-12', 1, '2023-06-16', '2023-06-14', 220000.00,
215000.00),
(72, 36, 18, 10, 5, '2023-09-10', 2, '2023-09-14', '2023-09-12', 240000.00,
235000.00),
```

(73, 11, 19, 28, 3, '2023-07-21', 1, '2023-07-25', '2023-07-23', 250000.00, 245000.00),
(74, 32, 20, 22, 2, '2023-08-13', 3, '2023-08-17', '2023-08-15', 270000.00, 265000.00),
(75, 38, 5, 6, 4, '2023-09-04', 4, '2023-09-08', '2023-09-06', 230000.00, 225000.00),
(76, 40, 30, 9, 1, '2023-06-25', 2, '2023-06-29', '2023-06-27', 240000.00, 235000.00),
(77, 27, 8, 15, 3, '2023-08-26', 5, '2023-08-30', '2023-08-28', 220000.00, 215000.00),
(78, 7, 21, 18, 2, '2023-09-02', 1, '2023-09-06', '2023-09-04', 250000.00, 245000.00),
(79, 19, 6, 24, 4, '2023-07-13', 3, '2023-07-17', '2023-07-15', 270000.00, 265000.00),
(80, 10, 33, 7, 1, '2023-09-15', 2, '2023-09-19', '2023-09-17', 220000.00, 215000.00),
(81, 3, 13, 17, 5, '2023-09-01', 4, '2023-09-05', '2023-09-03', 220000.00, 215000.00),
(82, 18, 5, 16, 2, '2023-08-13', 1, '2023-08-25', '2023-08-25', 240000.00, 235000.00),
(83, 25, 11, 9, 4, '2023-06-21', 2, '2023-06-25', '2023-06-23', 250000.00, 245000.00),
(84, 10, 6, 22, 3, '2023-07-05', 3, '2023-07-09', '2023-07-07', 230000.00, 225000.00),
(85, 33, 14, 20, 5, '2023-08-22', 4, '2023-08-26', '2023-08-24', 260000.00, 255000.00),
(86, 6, 16, 12, 1, '2023-09-10', 2, '2023-09-14', '2023-09-12', 210000.00, 205000.00),
(87, 21, 18, 11, 2, '2023-07-17', 5, '2023-07-21', '2023-07-19', 240000.00, 235000.00),
(88, 12, 8, 14, 4, '2023-09-05', 3, '2023-09-09', '2023-09-07', 250000.00, 245000.00),
(89, 34, 30, 10, 1, '2023-06-30', 2, '2023-07-04', '2023-07-02', 230000.00, 225000.00),
(90, 9, 12, 8, 3, '2023-08-02', 4, '2023-08-24', '2023-08-24', 220000.00, 215000.00),
(91, 22, 7, 5, 5, '2023-07-20', 2, '2023-07-24', '2023-07-22', 240000.00, 235000.00),
(92, 26, 25, 18, 3, '2023-08-15', 1, '2023-08-19', '2023-08-17', 230000.00, 225000.00),
(93, 16, 19, 13, 4, '2023-06-17', 5, '2023-06-21', '2023-06-19', 250000.00, 245000.00),
(94, 38, 24, 23, 2, '2023-07-23', 1, '2023-07-27', '2023-07-25', 260000.00, 255000.00),

```
(95, 19, 1, 29, 5, '2023-08-10', 3, '2023-08-14', '2023-08-12', 220000.00,
215000.00),
(96, 20, 21, 4, 1, '2023-06-24', 2, '2023-06-28', '2023-06-26', 230000.00,
225000.00),
(97, 24, 26, 2, 4, '2023-07-14', 5, '2023-07-18', '2023-07-16', 250000.00,
245000.00),
(98, 17, 23, 9, 3, '2023-08-19', 4, '2023-08-23', '2023-08-21', 240000.00,
235000.00),
(99, 28, 9, 3, 5, '2023-06-05', 1, '2023-06-17', '2023-06-17', 230000.00,
225000.00),
(100, 4, 17, 7, 2, '2023-09-20', 3, '2023-09-24', '2023-09-22', 250000.00,
245000.00),
(101, 33, 10, 6, 4, '2023-08-30', 5, '2023-09-03', '2023-09-01', 220000.00,
215000.00),
(102, 15, 6, 15, 2, '2023-07-01', 2, '2023-07-05', '2023-07-03', 230000.00,
225000.00),
(103, 31, 27, 22, 3, '2023-06-22', 4, '2023-06-26', '2023-06-24', 260000.00,
255000.00),
(104, 8, 3, 13, 5, '2023-08-12', 1, '2023-08-16', '2023-08-14', 240000.00,
235000.00),
(105, 12, 14, 25, 1, '2023-07-30', 3, '2023-08-03', '2023-08-01', 220000.00,
215000.00),
(106, 36, 20, 27, 4, '2023-08-09', 2, '2023-08-13', '2023-08-11', 250000.00,
245000.00),
(107, 13, 22, 16, 5, '2023-09-12', 3, '2023-09-16', '2023-09-14', 230000.00,
225000.00),
(108, 27, 28, 17, 1, '2023-06-14', 4, '2023-06-18', '2023-06-16', 240000.00,
235000.00),
(109, 18, 8, 12, 2, '2023-07-09', 5, '2023-07-21', '2023-07-21', 260000.00,
255000.00),
(110, 9, 2, 18, 3, '2023-08-25', 1, '2023-08-29', '2023-08-27', 230000.00,
225000.00),
(111, 14, 11, 21, 4, '2023-07-11', 3, '2023-07-15', '2023-07-13', 240000.00,
235000.00),
(112, 29, 16, 5, 5, '2023-09-08', 4, '2023-09-12', '2023-09-10', 250000.00,
245000.00),
(113, 3, 13, 26, 2, '2023-06-28', 1, '2023-07-02', '2023-06-30', 220000.00,
215000.00),
(114, 30, 18, 9, 1, '2023-08-28', 5, '2023-09-01', '2023-08-30', 240000.00,
235000.00),
(115, 11, 4, 23, 3, '2023-07-18', 2, '2023-07-22', '2023-07-20', 250000.00,
245000.00),
(116, 24, 21, 14, 4, '2023-08-21', 5, '2023-08-25', '2023-08-23', 230000.00,
225000.00),
```

```
(117, 10, 7, 30, 2, '2023-06-23', 3, '2023-06-27', '2023-06-25', 220000.00,
215000.00),
(118, 32, 5, 8, 1, '2023-09-14', 1, '2023-09-18', '2023-09-16', 250000.00,
245000.00),
(119, 22, 15, 28, 5, '2023-08-26', 2, '2023-08-30', '2023-08-28', 240000.00,
235000.00),
(120, 28, 31, 19, 4, '2023-06-04', 4, '2023-06-08', '2023-06-06', 230000.00,
225000.00),
(121, 6, 10, 11, 3, '2023-08-13', 5, '2023-08-25', '2023-08-25', 240000.00,
235000.00),
(122, 14, 18, 22, 2, '2023-07-11', 4, '2023-07-15', '2023-07-13', 250000.00,
245000.00),
(123, 19, 7, 30, 1, '2023-09-02', 3, '2023-09-06', '2023-09-04', 220000.00,
215000.00),
(124, 12, 3, 9, 5, '2023-08-20', 1, '2023-08-24', '2023-08-22', 230000.00,
225000.00),
(125, 24, 25, 17, 4, '2023-07-29', 2, '2023-08-02', '2023-07-31', 240000.00,
235000.00),
(126, 30, 9, 14, 3, '2023-06-28', 5, '2023-07-02', '2023-06-30', 250000.00,
245000.00),
(127, 16, 6, 28, 2, '2023-08-15', 4, '2023-08-19', '2023-08-17', 220000.00,
215000.00),
(128, 28, 16, 11, 1, '2023-09-01', 3, '2023-09-05', '2023-09-03', 230000.00,
225000.00),
(129, 25, 14, 19, 4, '2023-07-03', 2, '2023-07-07', '2023-07-05', 240000.00,
235000.00),
(130, 20, 12, 10, 5, '2023-08-17', 1, '2023-08-21', '2023-08-19', 250000.00,
245000.00),
(131, 8, 20, 22, 3, '2023-06-19', 4, '2023-06-23', '2023-06-21', 220000.00,
215000.00),
(132, 10, 2, 18, 5, '2023-08-05', 1, '2023-08-27', '2023-08-27', 230000.00,
225000.00),
(133, 21, 8, 13, 2, '2023-07-26', 3, '2023-07-30', '2023-07-28', 240000.00,
235000.00),
(134, 33, 27, 24, 4, '2023-08-11', 5, '2023-08-15', '2023-08-13', 250000.00,
245000.00),
(135, 22, 4, 27, 1, '2023-09-14', 2, '2023-09-18', '2023-09-16', 220000.00,
215000.00),
(136, 12, 13, 5, 3, '2023-06-13', 4, '2023-06-17', '2023-06-15', 230000.00,
225000.00),
(137, 9, 3, 21, 5, '2023-07-14', 2, '2023-07-18', '2023-07-16', 240000.00,
235000.00),
(138, 17, 26, 6, 1, '2023-08-01', 3, '2023-08-05', '2023-08-03', 250000.00,
245000.00),
```

```
(139, 23, 11, 8, 4, '2023-06-25', 2, '2023-06-29', '2023-06-27', 220000.00,
215000.00),
(140, 28, 19, 7, 3, '2023-08-19', 1, '2023-08-23', '2023-08-21', 230000.00,
225000.00),
(141, 26, 17, 10, 5, '2023-09-02', 2, '2023-09-06', '2023-09-04', 240000.00,
235000.00),
(142, 15, 5, 23, 4, '2023-06-07', 3, '2023-06-11', '2023-06-09', 250000.00,
245000.00),
(143, 30, 20, 3, 1, '2023-07-20', 5, '2023-07-24', '2023-07-22', 220000.00,
215000.00),
(144, 32, 18, 25, 2, '2023-06-12', 1, '2023-06-16', '2023-06-14', 230000.00,
225000.00),
(145, 34, 9, 16, 4, '2023-08-21', 2, '2023-08-25', '2023-08-23', 240000.00,
235000.00),
(146, 18, 1, 8, 3, '2023-09-04', 4, '2023-09-26', '2023-09-26', 250000.00,
245000.00),
(147, 11, 30, 24, 5, '2023-07-28', 2, '2023-08-01', '2023-07-30', 220000.00,
215000.00),
(148, 27, 28, 6, 1, '2023-08-11', 3, '2023-08-15', '2023-08-13', 230000.00,
225000.00),
(149, 12, 14, 17, 4, '2023-07-02', 5, '2023-07-24', '2023-07-24', 240000.00,
235000.00),
(150, 21, 15, 12, 2, '2023-09-07', 3, '2023-09-11', '2023-09-09', 250000.00,
245000.00);
```

Funciones

```
use autorenta;

-- FUNCIONES

-- calcula el valor cotizado para ello debes saber cuantas semanas y dias se
alquilar el vehiculo y con eso
-- revisar el precio semanal y diario de ese vehiculo en el tipo de vehiculo

DELIMITER //
CREATE FUNCTION valor_cotizado (id_vehiculoF int, fecha_salida date, fecha_esperada
date)
RETURNS DECIMAL(10,2) DETERMINISTIC
BEGIN
    DECLARE valor_cotizado DECIMAL(10,2); -- Crear una variable interna
    DECLARE valor_semanal DECIMAL(10,2);
    DECLARE valor_diario DECIMAL(10,2);
    DECLARE dias_alquiler INT;
```



```

    DECLARE dias int;
    declare semanas int;
    declare valor_dias decimal(10,2);
    declare valor_semanas decimal(10,2);

    SET valor_semanal=(SELECT distinct tv.precio_semana from alquiler inner join
vehiculo on alquiler.id_vehiculo=vehiculo.id inner join tipo_vehiculo tv on
vehiculo.id_tipo=tv.id where alquiler.id_vehiculo=id_vehiculoF);
    SET valor_diario=(SELECT distinct tv.precio_dia from alquiler inner join
vehiculo on alquiler.id_vehiculo=vehiculo.id inner join tipo_vehiculo tv on
vehiculo.id_tipo=tv.id where alquiler.id_vehiculo=id_vehiculoF);
    SET dias_alquiler= timestampdiff(day,fecha_salida,fecha_esperada);
    set dias=dias_alquiler % 7;
    set semanas= floor(dias_alquiler/7);
    set valor_dias= dias*valor_diario;
    set valor_semanas=semanas*valor_semanal;
    set valor_cotizado= valor_dias+valor_semanas;

    RETURN valor_cotizado;
END //
DELIMITER ;

select id as id_alquiler, valor_cotizado(id_vehiculo,fecha_salida,fecha_esperada)
as valor_cotizado from alquiler;

-- calcula el numero de dias y semanas que el cliente tuvo el vehiculo

DELIMITER //
CREATE FUNCTION tiempo_alquiler (fecha_salida date, fecha_llegada date)
RETURNS varchar(200) DETERMINISTIC
BEGIN

    declare tiempo_alquiler varchar(200);
    declare dias_alquiler INT;
    declare dias int;
    declare semanas int;

    SET dias_alquiler= timestampdiff(day,fecha_salida,fecha_llegada);
    set dias=dias_alquiler % 7;
    set semanas= floor(dias_alquiler/7);
    set tiempo_alquiler= concat('dias: ',dias,' semanas: ',semanas);

```

```

        RETURN tiempo_alquiler;
END //
DELIMITER ;

select id,tiempo_alquiler (fecha_salida, fecha_llegada) as tiempo_alquiler from
alquiler;

-- calcular el valor final de pago añadiendo descuentos y cobros extra por retraso

DELIMITER //
CREATE FUNCTION valor_final (id_vehiculoF int, fecha_salida date, fecha_esperada
date,fecha_llegada date)
RETURNS DECIMAL(10,2) DETERMINISTIC
BEGIN
    declare valor_cotizado decimal(10,2); -- Crear una variable interna
    declare valor_diario decimal(10,2);
    declare valor_diario_retraso decimal(10,2);
    declare dias_retraso int;
    declare valor_dias_retraso int;
    declare fecha_inicio_descuento date;
    declare fecha_fin_descuento date;
    declare descuento int;
    declare valor_final decimal(10,2);

    set valor_cotizado=valor_cotizado(id_vehiculoF,fecha_salida,fecha_esperada);

    set valor_diario=(SELECT distinct tv.precio_dia from alquiler
    inner join vehiculo on alquiler.id_vehiculo=vehiculo.id
    inner join tipo_vehiculo tv on vehiculo.id_tipo=tv.id where
alquiler.id_vehiculo=id_vehiculoF);

    set valor_diario_retraso=valor_diario +(valor_diario*0.08);

    set dias_retraso=timestampdiff(day, fecha_esperada, fecha_llegada);

    if dias_retraso>1 then
        set valor_dias_retraso=dias_retraso*valor_diario_retraso;
    else
        set valor_dias_retraso=0;
    end if;

    set fecha_inicio_descuento=(select distinct d.fecha_inicio from descuento d

```

```

        inner join tipo_vehiculo tv on d.id=tv.id_descuento inner join vehiculo v on
tv.id=v.id_tipo where v.id=id_vehiculoF);

        set fecha_fin_descuento=(select distinct d.fecha_fin from descuento d
        inner join tipo_vehiculo tv on d.id=tv.id_descuento inner join vehiculo v on
tv.id=v.id_tipo where v.id=id_vehiculoF);

        set descuento=(select distinct d.porcentaje from descuento d
        inner join tipo_vehiculo tv on d.id=tv.id_descuento inner join vehiculo v on
tv.id=v.id_tipo where v.id=id_vehiculoF);

        if fecha_llegada>=fecha_inicio_descuento and fecha_llegada<=fecha_fin_descuento
then
            set valor_cotizado = valor_cotizado - (valor_cotizado*descuento/100);
        else
            set valor_cotizado = valor_cotizado;
        end if;

        set valor_final= valor_cotizado + valor_dias_retraso;

        RETURN valor_final;
END //
DELIMITER ;

select id as id_alquiler,
valor_final(id_vehiculo,fecha_salida,fecha_esperada,fecha_llegada) as valor_a_pagar
from alquiler;

-- funcion para concatenar el nombre completo en una sola columna

DELIMITER //
CREATE FUNCTION nombre_completo (nombre1 varchar(50),nombre2 varchar(50),apellido1
varchar(50),apellido2 varchar(50))
RETURNS varchar(200) DETERMINISTIC
BEGIN

    declare nombre_completo varchar(200);

    set nombre_completo= concat_ws(' ',nombre1,' ',nombre2,' ',apellido1,'
',apellido2);

    RETURN nombre_completo;
END //

```

```

DELIMITER ;

select id,nombre_completo(nombre1,nombre2,apellido1,apellido2) as nombre_completo
from cliente;

-- funcion para saber si un vehiculo tiene descuento

DELIMITER //
CREATE FUNCTION saber_descuento (id_vehiculoF int, fecha_actual date)
RETURNS varchar(200) DETERMINISTIC
BEGIN

    declare fecha_inicio_descuento date;
    declare fecha_fin_descuento date;
    declare descuento int;
    declare info_descuento varchar(200);

    set fecha_inicio_descuento=(select distinct d.fecha_inicio from descuento d
        inner join tipo_vehiculo tv on d.id=tv.id_descuento inner join vehiculo v on
tv.id=v.id_tipo where v.id=id_vehiculoF);

    set fecha_fin_descuento=(select distinct d.fecha_fin from descuento d
        inner join tipo_vehiculo tv on d.id=tv.id_descuento inner join vehiculo v on
tv.id=v.id_tipo where v.id=id_vehiculoF);

    set descuento=(select distinct d.porcentaje from descuento d
        inner join tipo_vehiculo tv on d.id=tv.id_descuento inner join vehiculo v on
tv.id=v.id_tipo where v.id=id_vehiculoF);

    if fecha_actual>=fecha_inicio_descuento and fecha_actual<=fecha_fin_descuento
then
        set info_descuento = concat('El vehiculo tiene ',descuento,'% de
descuento');
    else
        set info_descuento = 'El vehiculo no tiene descuento actualmente';
    end if;

    RETURN info_descuento;
END //
DELIMITER ;

select saber_descuento(7,'2024-11-15') as Descuento;

```

Consultas

```
-- CONSULTAS

-- 1. Obtener todos los vehículos que tienen sunroof.

select * from vehiculo where sunroof=1;

-- 2. Listar los empleados que trabajan la sucursal de Bogotá.

select e.* from empleado e inner join sucursal s on e.id_sucursal = s.id where
s.ciudad='Bogotá';

-- 3. Consultar todos los vehículos disponibles la sucursal de Cali.

select v.* from vehiculo v inner join sucursal s on v.id_sucursal = s.id where
s.ciudad='Cali';

-- 4. Contar la cantidad de vehículos que tiene cada tipo de vehiculo.

select tv.nombre, count(v.id_tipo) as cantidad_vehiculos from tipo_vehiculo tv
inner join vehiculo v on tv.id = v.id_tipo group by 1;

-- 5. Obtener los nombre y correo de todos los clientes.

select nombrel1,nombre2,correo_electronico from cliente;

-- 6. Consultar el total de ingresos generados por los alquileres.

select sum(valor_pagado) from alquiler;

-- 7. Listar las sucursales que no tienen empleados asignados.

select s.* from sucursal s where s.id not in (select id_sucursal from empleado);

-- 8. Consultar el valor total pagado en alquileres por cliente .

select c.nombrel1,c.nombre2,sum(a.valor_pagado) as Total_pagos from cliente c inner
join alquiler a on c.id = a.id_cliente group by 1,2;
```

```

-- 9. Listar todos los vehículos que superan la capacidad de 4 pasajeros.

select * from vehiculo where capacidad>4;

-- 10. Listar la sucursal con el mayor número de vehículos registrados.

select s.*,count(v.id_sucursal) as cantidad_vehiculos from sucursal s inner join
vehiculo v on s.id=v.id_sucursal group by 1 order by 2 desc limit 1;

-- 11. Verificar qué vehículos fueron alquilados en más de una ocasión.

select v.referencia, count(a.id_vehiculo) as 'cantidad_alquilado' from vehiculo v
inner join alquiler a on v.id=a.id_vehiculo group by 1 having cantidad_alquilado>1;

-- 12. Consultar el promedio de valor cotizado en los alquileres.

select round(avg(valor_cotizado)) as promedio_valor_cotizado from alquiler;

-- 13. Listar todos los clientes que han alquilado un vehículo en la sucursal con
id 5.

select distinct c.* from cliente c inner join alquiler a on c.id=a.id_cliente where
a.id_sucursal_salida=5;

-- 14. Obtener los datos de los vehículos alquilados en el mes de julio.

select v.*,a.id as id_alquiler,a.fecha_salida from vehiculo v inner join alquiler
a on v.id=a.id_vehiculo where month(a.fecha_salida)=7;

-- 15. Contar cuántos vehículos hay de cada color en el sistema.

select color,count(color) as cantidad_vehiculos from vehiculo group by 1 ;

-- 16. Consultar el historial de alquileres del cliente con el numero de cedula
1087654328.

select c.id,c.cedula,c.nombre1,c.apellido1,a.* from cliente c inner join alquiler a
on c.id=a.id_cliente where c.cedula=1087654328;

-- 17. Obtener los nombres y correos electrónicos de empleados con teléfono
celular.

select nombre1,nombre2,correo_electronico,telefono_celular from empleado;

```

```

-- 18. Listar todos los tipos de vehículos y sus precios por día y por semana.

select nombre,precio_dia,precio_semana from tipo_vehiculo;

-- 19. Mostrar el porcentaje de descuento activo y tipo de vehiculo para el mes de
diciembre.

select d.porcentaje,tv.nombre from descuento d inner join tipo_vehiculo tv on d.id
= tv.id_descuento where month(d.fecha_inicio)=12;

-- 20. Listar los empleados que comparten la misma ciudad de residencia.

select ciudad_residencia,id,nombrel1,apellidol1 from empleado order by 1;

-- 21. Consultar las fechas de salida y llegada de alquileres realizados por el
cliente de id 10.

select c.id,c.nombrel1,c.apellidol1,a.fecha_salida,a.fecha_llegada from cliente c
inner join alquiler a on c.id= a.id_cliente where c.id=10;

-- 22. Lista los nombres de los clientes junto con el id y la ciudad de las
sucursales en las cuales a alquilado vehiculos.

select distinct c.nombrel1,c.nombre2,s.id,s.ciudad from cliente c inner join
alquiler a on c.id=a.id_cliente inner join sucursal s on a.id_sucursal_salida=s.id
order by 1;

-- 23. Lista los nombres de los empleados junto con los nombres del cliente al que
le a realizado alquileres.

select distinct concat(e.nombrel1,' ',e.apellidol1) as
nombres_empleado,concat(c.nombrel1,' ',c.apellidol1) as nombres_cliente
from empleado e inner join alquiler a on e.id=a.id_empleado
inner join cliente c on a.id_cliente=c.id;

-- 24. Lista el top 3 de empleados con mayor cantidad de alquileres realizados.

select e.*,count(a.id_empleado) as Cantidad_alquileres from empleado e inner join
alquiler a on e.id=a.id_empleado group by 1 order by 12 desc limit 3;

-- 25. Lista los nombres de los clientes junto con cuantos alquileres ha realizado
select count(a.id_cliente) as cantidad_alquileres,c.nombrel1,c.nombre2 from cliente
c inner join alquiler a on c.id=a.id_cliente group by 2,3;

```

Procedimientos

```
--
PROCEDIMIENTOS-----

-- asignar valor_cotizado a un alquiler
delimiter //
create procedure asignar_valor_cotizado(in id_alquiler int,in valor_cotizadoP
decimal(10,2))
begin

    update alquiler      set valor_cotizado=valor_cotizadoP where id=id_alquiler;
end
// delimiter ;

call asignar_valor_cotizado(2,(select
valor_cotizado(id_vehiculo,fecha_salida,fecha_esperada) from alquiler where id=2))
;

-- corregir todos los valores cotizados de la tabla alquiler

delimiter //
create procedure asignar_todos_valores_cotizados()
begin
    declare contador int default 1;

    while contador<(select count(*) from alquiler)+1 do

        call asignar_valor_cotizado(contador,(select
valor_cotizado(id_vehiculo,fecha_salida,fecha_esperada) from alquiler where
id=contador));
        set contador=contador+1;

    end while;
end
// delimiter ;

call asignar_todos_valores_cotizados();

-- asignar valor_pagado a un alquiler

delimiter //
```



```

create procedure asignar_valor_pagado(in id_alquiler int,in valor_pagadoP
decimal(10,2))
begin
    update alquiler      set valor_pagado=valor_pagadoP where id=id_alquiler;
end // delimiter ;

call asignar_valor_pagado(1,(select
valor_final(id_vehiculo,fecha_salida,fecha_esperada,fecha_llegada) from alquiler
where id=1));

-- corregir todos los valores pagados de la tabla alquiler

delimiter //
create procedure asignar_todos_valores_pagados()
begin
    declare contador int default 1;

    while contador<(select count(*) from alquiler)+1 do

        call asignar_valor_pagado(contador,(select
valor_final(id_vehiculo,fecha_salida,fecha_esperada,fecha_llegada) from alquiler
where id=contador));

        set contador=contador+1;

    end while;
end
// delimiter ;

call asignar_todos_valores_pagados();

-- registrar cliente

delimiter //
create procedure registrar_cliente(
    in p_cedula int(15),
    in p_nombre1 varchar(50),
    in p_nombre2 varchar(50),
    in p_apellido1 varchar(50),
    in p_apellido2 varchar(50),
    in p_direccion varchar(150),
    in p_ciudad_residencia varchar(40),
    in p_telefono_celular bigint,

```

```

        in p_correo_electronico varchar(80),
        in p_user_cliente varchar(30),
        in p_password_cliente varchar(40))
begin
    insert into cliente(cedula, nombre1, nombre2, apellido1, apellido2,
direccion,ciudad_residencia, telefono_celular, correo_electronico,user_cliente,
password_cliente)
        values (p_cedula, p_nombre1, p_nombre2, p_apellido1, p_apellido2,
p_direccion,p_ciudad_residencia, p_telefono_celular,
p_correo_electronico,p_user_cliente, p_password_cliente);
end
// delimiter ;

call registrar_cliente(1093904929,'Luis','Orlando','Henao','Bermon','CR 3e
#1-24','Tibú',3003426875,'luis_or2454@gmail.com','Lucho03','Luchin123');

-- registrar empleado
delimiter //
create procedure registrar_empleado(
    in p_id_sucursal int,
    in p_cedula int(15),
    in p_nombre1 varchar(50),
    in p_nombre2 varchar(50),
    in p_apellido1 varchar(50),
    in p_apellido2 varchar(50),
    in p_direccion varchar(150),
    in p_ciudad_residencia varchar(40),
    in p_telefono_celular bigint,
    in p_correo_electronico varchar(80),
    in p_user_empleado varchar(30),
    in p_password_empleado varchar(40))
begin
    insert into
empleado(id_sucursal,cedula,nombre1,nombre2,apellido1,apellido2,direccion,ciudad_re
sidencia,telefono_celular,correo_electronico,user_empleado,password_empleado)
        values
(p_id_sucursal,p_cedula,p_nombre1,p_nombre2,p_apellido1,p_apellido2,p_direccion,p_c
iudad_residencia,p_telefono_celular,p_correo_electronico,p_user_empleado,p_password
_empleado);
end // delimiter ;

call registrar_empleado(4,1093909,'Luis','Miguel','Caicedo','Bermon','CR 3e
#1-24','Tibú',30554861,'luis_m2454@gmail.com','Lucho03','Luchin456');

```

```

-- ingresar un nuevo alquiler

delimiter //
create procedure ingresar_alquiler(
    in r_id_vehiculo int,
    in r_id_cliente int,
    in r_id_empleado int,
    in r_id_sucursal_salida int,
    in r_fecha_salida date,
    in r_id_sucursal_llegada int,
    in r_fecha_llegada date,
    in r_fecha_esperada date,
    in r_valor_cotizado decimal(10,2),
    in r_valor_pagado decimal(10,2)
)
begin

    insert into
alquiler(id_vehiculo,id_cliente,id_empleado,id_sucursal_salida,fecha_salida,id_sucursal_llegada,fecha_llegada,fecha_esperada,valor_cotizado,valor_pagado)
values
(r_id_vehiculo,r_id_cliente,r_id_empleado,r_id_sucursal_salida,r_fecha_salida,r_id_sucursal_llegada,r_fecha_llegada,r_fecha_esperada,r_valor_cotizado,r_valor_pagado);
end

// delimiter ;

call
ingresar_alquiler(12,41,11,1,current_date(),3,null,current_date()+9,valor_cotizado(12,current_date(),current_date()+9),null);

-- Registrar vehiculo

delimiter //
create procedure registrar_vehiculo(
    in r_id_tipo int,
    in r_id_sucursal int,
    in r_placa varchar(6),
    in r_referencia varchar(50),
    in r_modelo varchar(30),
    in r_puertas int,
    in r_capacidad int,
    in r_sunroof tinyint(1),
    in r_motor varchar(45),
    in r_color varchar(30))

```

```

begin
    insert into
vehiculo(id_tipo,id_sucursal,placa,referencia,modelo,puertas,capacidad,sunroof,moto
r,color)

values(r_id_tipo,r_id_sucursal,r_placa,r_referecnia,r_modelo,r_puertas,r_capacidad,
r_sunroof,r_motor,r_color);
end
// delimiter ;

    call registrar_vehiculo(3,2,'HYR3564','Chevrolet Joy','2022',4,5,1,'5.0L',
'Rojo');

-- insertar tipo de vehiculo
delimiter //
create procedure insertar_tipo_vehiculo(
    in r_nombre varchar(50),
    in r_id_descuento int,
    in r_precio_semana decimal(10,2),
    in r_precio_dia decimal(10,2))
begin
    insert into tipo_vehiculo(nombre,id_descuento,precio_semana,precio_dia)
    values(r_nombre,r_id_descuento,r_precio_semana,r_precio_dia);
end
//delimiter ;

call insertar_tipo_vehiculo('4x4',7,600000.00,100000.00)

-- insertar una nueva sucursal

delimiter //
create procedure insertar_sucursal(
    in r_direccion varchar(150),
    in r_ciudad varchar(40),
    in r_telefono_fijo bigint,
    in r_telefono_celular bigint,
    in r_correo_electronico varchar(80))
begin

    insert into sucursal
(direccion,ciudad,telefono_fijo,telefono_celular,correo_electronico)

```

```
values(r_direccion,r_ciudad,r_telefono_fijo,r_telefono_celular,r_correo_electronico
);

end
// delimiter ;

call insertar_sucursal('CLL 5 AVN
2','Cucuta',56889532,3254875962,'autorental_cucuta@gmail.com');
```