

actividad1

May 26, 2022



Universidad
Internacional
de Valencia

Integrantes del grupo:

- Alexander Raúl Ortiz Insuasti
- Dámaris Daniela Núñez Pérez
- Jimmy Saúl Yar Carvajal

Índice

1. Introducción
2. Necesidades y requisitos del cliente (análisis de requisitos)
3. Diseño de base de datos
 - 3.1. Diseño conceptual
 - 3.1.1. Pasos para realizar el diseño conceptual
 - 3.2. Diseño lógico
 - 3.3. Diseño físico
 - 3.3.1. Pasos para realizar el diseño físico
 - 3.3.1.1. Conexión en local
4. Despliegue de la base de datos
 - 4.1. Cadena de conexión local
 - 4.2. Cadena de conexión en la nube
 - 4.2.1. Ejecutar el diseño físico en la nube
 - 4.3. Inserción de datos
 - 4.4. Inserción de datos
5. Consultas a base de datos
 - 5.1 Consultas multitable (con JOIN)
 - 5.2 Consultas con función de agregación (operaciones sobre conjuntos de filas)
6. Conclusiones

1 Introducción

Empresa pequeña dedicada al alquiler de automóviles. Se realiza reservación del vehículo de distintas gamas, especificando las fechas de inicio y final del alquiler, es decir, el día que el auto sale del patio de carros y el día que ingresa. No vende autos. Además, se encarga de llevar un control de mantenimiento preventivo, por ejemplo: cambio de aceite a los 5000 km, alineación de llantas a los 10000 km.. El encargado de la empresa requiere una base de datos para gestionar a los clientes y el alquiler de los carros, llevar un control adecuado del mantenimiento de los vehículos, además, necesitan automatizar el proceso, ya que, se ha estado realizando manualmente.

2 Necesidades y requisitos del cliente (análisis de requisitos)

El primer contacto que se obtuvo con el propietario de la empresa RentaCar, ha tomado la decisión de realizar un sistema web, para promocionar su variedad de vehículos que posee para el alquiler, y con el encargado de llevar el control de la revisión de cada uno de los vehículos, de igual forma con la secretaria que lleva el control del proceso de alquiler. Tanto el propietario, como la secretaria y

el encargado de mantenimiento, son personas que conocen como funciona el negocio, además, de ser expertos en las tareas que realiza cada uno, al contar con más de cinco años de experiencia en la empresa.

El propietario necesita tener un base de datos, para llevar un listado de todos los vehículos con sus características, de igual manera desea llevar un seguimiento del estado de los vehículos que se encuentran en alquiler, el mantenimiento y comprobación de disponibilidad. La secretaria necesita llevar un control de la información de los clientes y los vehículos que alquila cada uno. La persona encargada de llevar al taller los vehículos y de dar limpieza, necesita llevar un control preventivo y correctivo de cada vehículo.

Actualmente la empresa no tiene ningun sistema que lleve el registro de los clientes y de los vehiculos, la secretaria de la empresa maneja todos los registros mediante hojas de excel e impresas, esto con lleva a una duplicidad de informacion de los clientes o perdida de algunos registros al no tener centralizada en un solo lugar. En el formulario apunta los datos de los clientes y del vehículo que alquila, y dificulta el consultar los vehiculos que estas disponibles y durante cuanto tiempo no va estar disponible, el sistema ayudaria a los clientes a realizar reservas en linea. De igual manera, plasma la información necesaria en cuadernos para llevar el control de mantenimiento de cada vehículo, lo que dificulta llevar el control de mantenimiento de cada vehículo, para que no presente ninguna averia al prestar el servicio.

Lista de tareas:

- Crear y editar la información de los clientes
- Eliminar clientes
- Visualizar una lista de todos los clientes
- Crear y editar la información de los vehículos
- Eliminar vehículos
- Visualizar una lista de todos los vehículos
- Crear y editar la información de las categorías de los vehículos
- Eliminar las categorías de los vehículos
- Visualizar una lista de todas las categorías de los vehículos
- Crear y editar la información de las reservaciones
- Eliminar reservaciones
- Visualizar una lista de todas las reservaciones
- Saber si un vehículo se encuentra reservado
- Crear y editar la información de las facturas
- Visualizar una lista de todas las facturas
- Crear y editar ficha de mantenimiento vehículos
- Eliminar ficha de mantenimiento
- Visualizar una lista con todas las fichas de mantenimiento

- Visualizar cuantos kilómetros ha recorrido cada vehículo
- Visualizar cuando un vehículo requiera cambiar el aceite
- Visualizar cuando un vehículo requiera alinear sus llantar

Las operaciones que realiza la empresa y los datos que utiliza con los siguientes:

Tarea	Datos necesarios
Crear y editar la información de los clientes	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Eliminar clientes	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Visualizar una lista de todos los clientes	Cédula, nombre, apellido, email, teléfono, dirección
Crear y editar la información de los vehículos	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Eliminar vehículos	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Visualizar una lista de todos los vehículos	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Crear y editar las categorías de los vehículos	Nombre, gama
Eliminar categorías de los vehículos	Nombre, gama
Visualizar una lista de todas las categorías de los vehículos	Nombre, gama
Crear y editar la información de las reservaciones	Fecha retiro, fecha entrega, kilometraje (libre, controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, vehículo, cliente
Eliminar reservaciones	Fecha retiro, fecha entrega, kilometraje (libre, controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, vehículo, cliente
Visualizar una lista de todas las reservaciones	Fecha retiro, fecha entrega, kilometraje (libre, controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, vehículo, cliente, estado
Saber si un vehículo se encuentra reservado	Información del vehículo, información de reservación
Crear y editar la información de las facturas	Información del cliente, información del vehículo, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total
Visualizar una lista de todas las facturas	Información del cliente, información del vehículo, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total

Tarea	Datos necesarios
Crear y editar ficha de mantenimiento vehículos	Información del vehículo, fecha, servicio, kilometraje, comentario
Eliminar ficha de mantenimiento	Información del vehículo, fecha, servicio, kilometraje, comentario
Visualizar una lista de todas las fichas de mantenimiento	Información del vehículo, fecha, servicio, kilometraje, comentario
Visualizar una lista con todas las fichas de mantenimiento	Información del vehículo, fecha, servicio, kilometraje, comentario
Visualizar cuantos kilómetros ha recorrido cada vehículo	Información de vehículo
Visualizar cuando un vehículo requiera cambiar el aceite	Información de vehículo
Visualizar cuando un vehículo requiera alinear sus llantas	Información de vehículo
Iniciar sesión	Usuario y contraseña

Requisitos

Requisito	Datos necesarios
Crear cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Editar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Eliminar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Consultar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Listar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Crear categoría de vehículos	Nombre, gama
Editar categoría de vehículos	Nombre, gama
Eliminar categoría de vehículos	Nombre, gama
Consultar categoría de vehículos	Nombre, gama
Listar categoría de vehículos	Nombre, gama
Crear vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)

Requisito	Datos necesarios
Editar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Eliminar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Consultar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Listar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Crear factura	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total
Editar factura	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total
Consultar factura	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total

Requisito	Datos necesarios
Listar facturas	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total
Crear reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Editar reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Eliminar reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Consultar reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor

Requisito	Datos necesarios
Listar reservaciones	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre, controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Crear ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Editar ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Eliminar ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Consultar ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Listar fichas de mantenimiento de vehículos	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Consultar el kilometraje de los vehículos	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Listar vehículos que requieran cambiar el aceite	Marca, modelo, costo_diario, kilometraje > 5000, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Listar vehículos que requieran alinear llantas	Marca, modelo, costo_diario, kilometraje > 10000, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)

Requisito	Datos necesarios
Iniciar sesión	Usuario y contraseña

Requisitos finales

Considerando el presupuesto de la empresa, se llevará a cabo los siguientes requisitos:

Requisitos Finales	Datos necesarios
Crear cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Editar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Eliminar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Consultar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Listar cliente	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Crear categoría de vehículos	Nombre, gama
Editar categoría de vehículos	Nombre, gama
Eliminar categoría de vehículos	Nombre, gama
Consultar categoría de vehículos	Nombre, gama
Listar categoría de vehículos	Nombre, gama
Crear vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Editar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Eliminar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Consultar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Listar vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)

Requisitos Finales	Datos necesarios
Crear factura	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total
Editar factura	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total
Consultar factura	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total
Listar facturas	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor, fecha, cantidad, descripción, valor unitario, valor total, subtotal, iva, total

Requisitos Finales	Datos necesarios
Crear reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Editar reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Eliminar reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Consultar reservación	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor
Listar reservaciones	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña, marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha retiro, fecha entrega, kilometraje (libre,controlado), comentario, vehículo, cliente, estado (aceptado, cancelado), valor

Requisitos Finales	Datos necesarios
Crear ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Editar ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Eliminar ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Consultar ficha de mantenimiento de un vehículo	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Listar fichas de mantenimiento de vehículos	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Consultar el kilometraje de los vehículos	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Listar vehículos que requieran cambiar el aceite	Marca, modelo, costo_diario, kilometraje > 5000, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Listar vehículos que requieran alinear llantas	Marca, modelo, costo_diario, kilometraje > 10000, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Iniciar sesión	Usuario y contraseña

3 Diseño de base de datos

3.1 Diseño conceptual

Para realizar el diseño conceptual hay que seguir una serie de pasos que facilitan su construcción, los cuales se definen a continuación:

3.1.1 Pasos para realizar el diseño conceptual

Paso 1: Identificación de entidades y atributos.

Nº	Entidad	Atributos
Entidad 1	CLIENTE	Cédula, nombre, apellido, email, teléfono, dirección, usuario, contraseña
Entidad 2	VEHÍCULO	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible)
Entidad 3	FICHA_MANTENIMIENTO	Marca, modelo, costo_diario, kilometraje, número de puertas, placa, color, tipo de combustible (Super, extra, diesel), número de plazas, estado (reservado, disponible), fecha, servicio, kilometraje, comentario
Entidad 4	ADMINISTRADOR	Usuario, contraseña, email
Entidad 5	RESERVACIÓN	Fecha retiro, comentario, tipo kilometraje(libre, controlado), estado(aceptado, cancelado), entrega, valor
Entidad 6	CATEGORIA	nombre, gama
Entidad 7	FACTURA_ENCABEZADO	fecha, subtotal, iva, total
Entidad 8	FACTURA_DETALLA	cantidad, descripción, valor unitario, valor total

Paso 2: Identificación de relaciones Cliente realiza reservación, relación REALIZA. Cardinalidad máxima: un cliente realiza muchas reservaciones y una reservación es realizada por un cliente. Cardinalidad mínima: un cliente realiza como mínimo 0 reservaciones y una reservación es realizada por lo menos por 1 cliente, es decir, no hay obligatoriedad. Cliente (0, max) realizar (0, max) reservación

Administrador registra vehículo, relación REGISTRA. Cardinalidad máxima: un administrador registra muchos vehículos y un vehículo es registrado por un administrador. Cardinalidad mínima: una secretaria registra como mínimo un vehículo y un vehículo es registrado como mínimo por una secretaria, es decir, hay obligatoriedad.

Secretaria (1, max) registra (1, max) vehículo

Reservación contiene vehículo, relación CONTIENE. Cardinalidad máxima: una reservación contiene un vehículo y un vehículo es contenido en varias reservaciones. Cardinalidad mínima: una reservación contiene como mínimo un vehículo y un vehículo es contenido como mínimo por 0

reservaciones, es decir, no hay obligatoriedad.

Reservación (0, max) contiene (0, max) vehículo

Vehículo tiene ficha mantenimiento, relación TIENE. Cardinalidad máxima: un vehículo tiene muchas fichas de mantenimiento y una ficha de mantenimiento es de un vehículo, Cardinalidad mínima: un vehículo tiene como mínimo una ficha de mantenimiento y una ficha de mantenimiento tiene como mínimo un vehículo. Vehículo (1, max) tiene (1, max) ficha mantenimiento

Vehículo pertenece categoría, relación PERTENECE. Cardinalidad máxima: un vehículo pertenece a una categoría y una categoría está en varios vehículos. Cardinalidad mínima: un vehículo pertenece como mínimo a una categoría y una categoría tiene 0 vehículos, es decir, no hay obligatoriedad.

Vehículo (max, 1) pertenece (max, 1) categoría

Reservación genera factura encabezado, relación GENERA. Cardinalidad máxima: una reservación genera una factura encabezado y una factura encabezado es generada por una reservación. Cardinalidad mínima: una reservación genera como mínimo 0 facturas encabezado y una factura encabezado genera como mínimo una reservación, es decir, no hay obligatoriedad. Reservación (0, max) genera (0, max) factura encabezado

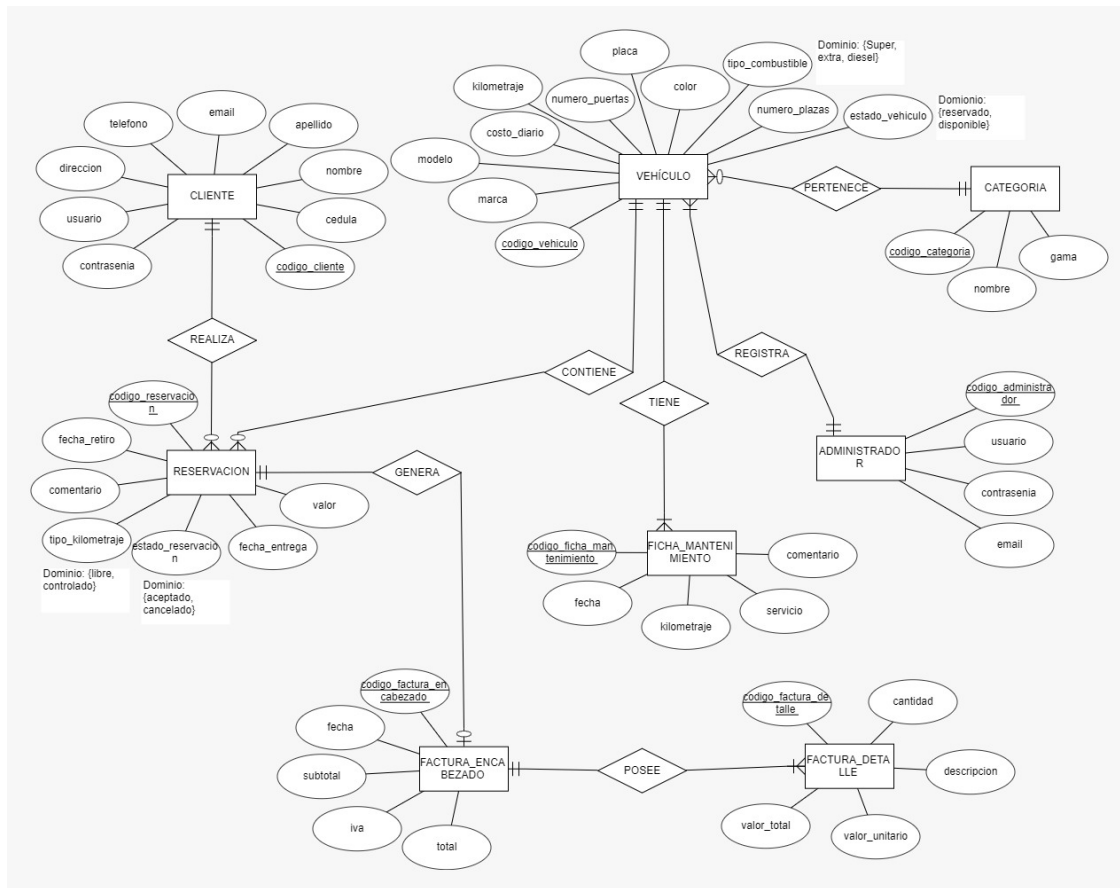
Factura encabezado posee factura detalle, relación POSEE. Cardinalidad máxima: una factura encabezado posee muchas facturas detalle y una factura detalle pertenece a una factura encabezado. Cardinalidad mínima: una factura encabezado posee como mínimo una factura detalle y una factura detalle es poseída como mínimo por una factura encabezado, es decir, hay obligatoriedad Factura encabezado (1, max) posee (1, max) factura detalle

Paso 3: Determinar dominios a los atributos

Entidad	Atributo	Dominio
VEHÍCULO	tipo_combustible	Super, extra y diesel
VEHÍCULO	estado	Disponible y reservado
RESERVA	estado	Aceptado y cancelado
RESERVA	tipo_kilometraj	Libre y controlado

Paso 4: Determinar identificadores

Entidad	Identificador
CLIENTE	codigo_cliente
VEHÍCULO	codigo_vehiculo
RESERVACIÓN	codigo_reservacion
CATEGORÍA	codigo_categoria
ADMINISTRADOR	codigo_administrador
FICHA_MANTENIMIENTO	codigo_ficha_mantenimiento
FACTURA_ENCABEZADO	codigo_factura_encabezado
FACTURA_DETALLE	codigo_factura_detalle



Paso 5: Dibujar el diagrama E/R

3.2 Diseño lógico

Paso 1: Convertir entidades a tablas

- **CLIENTE:** (codigo_cliente, cedula, nombre, apellido, email, telefono, direccion, usuario, contrasenia)
PK: {codigo_cliente}
- **RESERVACION:** (codigo_reservacion, valor, fecha_retiro, fecha_entrega, comentario, tipo_kilometraje, estado_reservacion)
Dominio estado_reservacion: [aceptado, cancelado]
Dominio tipo_kilometraje: [libre, controlado]
PK: {codigo_reservacion}
- **VEHICULO:** (codigo_vehiculo, marca, placa, modelo, color, tipo_combustible, numero_plazas, numero_puertas, estado_vehiculo)
Dominio estado_vehiculo: [reservado, disponible]
Dominio tipo_combustible: [super, extra, diesel]
PK: {codigo_vehiculo}

- FICHA_MANTENIMIENTO: (codigo_ficha_mantenimiento, fecha, kilometraje, servicio, comentario)
PK: {codigo_ficha_mantenimiento}
- CATEGORIA: (codigo_categoria, nombre, gama)
PK: {codigo_categoria}
- FACTURA_ENCABEZADO: (codigo_factura_encabezado, fecha, subtotal, iva, total)
PK: {codigo_factura_encabezado}
- FACTURA_DETALLE: (codigo_factura_detalle, total, total_unitario, cantidad, descripcion)
PK: {codigo_factura_detalle}
- ADMINISTRADOR: (codigo_administrador, usuario, contrasenia, email)
PK: {codigo_administrador}

Paso 2: Convertir relaciones a tablas

N a M No hay ninguna relación de muchos a muchos en el diagrama conceptual.

1 a N

- RESERVACION: (codigo_reserva, valor, fecha_retiro, fecha_entrega, comentario, tipo_kilometraje, estado_reserva)
PK: {codigo_reserva}
FK1: {RESERVACION.codigo_cliente -> CLIENTE.codigo_cliente}
FK2: {RESERVACION.codigo_vehiculo -> VEHICULO.codigo_vehiculo}
- VEHICULO: (codigo_vehiculo, marca, placa, modelo, color, tipo_combustible, numero_plazas, numero_puertas)
PK: {codigo_vehiculo}
FK1: {VEHICULO.codigo_categoria -> CATEGORIA.codigo_categoria}
FK2: {VEHICULO.codigo_administrador -> ADMINISTRADOR.codigo_administrador}
- FICHA_MANTENIMIENTO: (codigo_ficha_mantenimiento, fecha, kilometraje, servicio, comentario, codigo_vehiculo)
PK: {codigo_ficha_mantenimiento}
FK1: {FICHA_MANTENIMIENTO.codigo_vehiculo -> VEHICULO.codigo_vehiculo}
- FACTURA_DETALLE: (codigo_factura_detalle, total, total_unitario, cantidad, descripcion, codigo_factura_encabezado)
PK: {codigo_factura_detalle}
FK1: {FACTURA_DETALLE.codigo_factura_encabezado -> FACTURA_ENCABEZADO.codigo_factura_encabezado}

1 a 1

- FACTURA_ENCABEZADO: (codigo_factura_encabezado, fecha, subtotal, iva, total, codigo_reservacion)
PK: {codigo_enc}
FK: {FACTURA_ENCABEZADO.codigo_reservacion -> RESERVACION.codigo_reservacion}

Paso 3: Reglas de integridad

- RESERVACION: (codigo_reservacion, valor, fecha_retiro, fecha_entrega, comentario, tipo_kilometraje, estado)
PK: {codigo_reservacion}
FK1: {RESERVACION.codigo_cliente -> CLIENTE.codigo_cliente}
¿Acepta nulos? No
Borrado: Restringir
Modificado: Propagar
FK2: {RESERVACION.codigo_vehiculo -> VEHICULO.codigo_vehiculo}
¿Acepta nulos? No
Borrado: Restringir
Modificado: Propagar
- VEHICULO: (codigo_vehiculo, marca, placa, modelo, color, tipo_combustible, numero_plazas, numero_puertas)
PK: {codigo_vehiculo}
FK1: {VEHICULO.codigo_categoria -> CATEGORIA.codigo_categoria}
¿Acepta nulos? No
Borrado: Restringir
Modificado: Propagar
FK2: {VEHICULO.codigo_administrador -> ADMINISTRADOR.codigo_administrador}
¿Acepta nulos? No
Borrado: Restringir
Modificado: Propagar
- FICHA_MANTENIMIENTO: (codigo_ficha_mantenimiento, fecha, kilometraje, servicio, comentario, codigo_vehiculo)
PK: {codigo_ficha_mantenimiento}
FK1: {FICHA_MANTENIMIENTO.codigo_vehiculo -> VEHICULO.codigo_vehiculo}
¿Acepta nulos? No
Borrado: Restringir
Modificado: Propagar

- **FACTURA_DETALLE:** (codigo_factura_detalle,total,total_unitario,cantidad,descripcion,codigo_factura_detalle)
 PK: {codigo_factura_detalle}
 FK1: {FACTURA_DETALLE.codigo_factura_encabezado -> FACTURA_ENCABEZADO.codigo_factura_encabezado}
 ¿Acepta nulos? No
 Borrado: Restringir
 Modificado: Propagar
- **FACTURA_ENCABEZADO:** (codigo_factura_encabezado,fecha,subtotal,iva,total,codigo_reservacion)
 PK: {codigo_factura_encabezado}
 FK: {FACTURA_ENCABEZADO.codigo_reservacion -> RESERVACION.codigo_reservacion}
 ¿Acepta nulos? Si
 Borrado: Restringir
 Modificado: Propagar

Paso 4: Normalización

- Primera forma normal (1FN)
 Cumplen todas las tablas la primera forma normal, todos los atributos son atomicos.
- Segunda forma normal (2FN)
 Cumplen todas las tablas, no existe ninguna clave primaria con dos o mas atributos.
- Tercera forma normal (3FN)
 Cumplen todas las tablas, no atributos que dependan de otros.

Paso 5: Desnormalización No hemos realizado porque no lo creemos conveniente hacerlo.

Paso 6: Esquema lógico final

- **CLIENTE:** (codigo_cliente, cedula, nombre, apellido, email, telefono, direccion, usuario, contrasenia)
 PK: {codigo_cliente}
- **RESERVACION:** (codigo_reservacion, valor, fecha_retiro, fecha_entrega,comentario, tipo_kilometraje, estado_reservacion, codigo_cliente, codigo_vehiculo)
 Dominio estado_reservacion: [aceptado,cancelado]
 Dominio tipo_kilometraje: [libre, controlado]
 PK: {codigo_reservacion}
 FK1: {RESERVACION.codigo_cliente -> CLIENTE.codigo_cliente}

¿Acepta nulos? No

Borrado: Restringir

Modificado: Propagar

FK2: {RESERVACION.codigo_vehiculo -> VEHICULO.codigo_vehiculo}

¿Acepta nulos? No

Borrado: Restringir

Modificado: Propagar

- VEHICULO: (codigo_vehiculo, marca, placa, modelo, color, tipo_combustible, numero_plazas, numero_puertas)

Dominio estado_vehiculo: [reservado, disponible]

Dominio tipo_combustible: [super, extra, diesel]

PK: {codigo_vehiculo}

FK1: {VEHICULO.codigo_categoria -> CATEGORIA.codigo_categoria}

¿Acepta nulos? No

Borrado: Restringir

Modificado: Propagar

FK2: {VEHICULO.codigo_administrador -> ADMINISTRADOR.codigo_administrador}

¿Acepta nulos? No

Borrado: Restringir

- FICHA_MANTENIMIENTO: (codigo_mantenimiento, fecha, kilometraje, servicio, comentario, codigo_vehiculo)

PK: {codigo_mantenimiento}

FK1: {FICHA_MANTENIMIENTO.codigo_vehiculo -> VEHICULO.codigo_vehiculo}

¿Acepta nulos? No

Borrado: Restringir

Modificado: Propagar

- CATEGORIA: (codigo_categoria, nombre, gama)

PK: {codigo_categoria}

- FACTURA_DETALLE: (codigo_factura_detalle, valor_total, valor_unitario, cantidad, descripcion, codigo_factura_encabezado)

PK: {codigo_factura_detalle}

FK1: {FACTURA_DETALLE.codigo_factura_encabezado -> FACTURA_ENCABEZADO.codigo_factura_encabezado}

¿Acepta nulos? No

Borrado: Restringir

Modificado: Propagar

- FACTURA_ENCABEZADO: (codigo_factura_encabezado, fecha, subtotal, iva, total, codigo_reservacion)

PK: {codigo_enc}

FK: {FACTURA_ENCABEZADO.codigo_reservacion -> RESERVA-
CION.codigo_reservacion}

¿Acepta nulos? Si

Borrado: Restringir

Modificado: Propagar

- ADMINISTRADOR: (codigo_administrador, usuario, contrasenia, email)

PK: {codigo_administrador}

3.3 Diseño físico

3.3.1 Pasos para realizar el diseño físico

Paso 1: Identificar el Sistema de Gestión de Base de Datos Para el desarrollo de este proyecto se va a utilizar PostgreSQL versión 13

Paso 2: Convertir el diseño lógico en código

```
[ ]: # Instalar libreria psycopg2
!pip install psycopg2
```

Conexión en local

```
[ ]: # Importar la libreria
import psycopg2
```

```
[ ]: # LOCAL

# Se obtiene la URL de la base de datos de ElephantSQL,
# que tiene formato postgres://usuario:contrasenia@maquina:puerto/bbdd

# bbdd: el nombre de la base de datos, en este caso: dddd

bbdd = 'actividad_1' #cambiar por tu bbdd

# Usuario: nombre de usuario de la base de datos, en este caso: uuuuu

usuario = 'postgres' #cambiar por tu usuario
```

```

# Contraseña: contraseña de la base de datos, en este caso: ppppp
contrasenia = '<contraseña>' #cambiar por tu contraseña

# Máquina: nombre de la máquina/host (el que aloja) la base de datos, en este
↳ caso: mmmm

maquina = 'localhost'

# Puerto: puerto de la base de datos, en este caso: 5432 (puerto por defecto)

puerto = '5432'

# Conectar a la base de datos

conexion = psycopg2.connect(dbname=bbdd,

                             user=usuario,

                             password=contrasenia,

                             host=maquina,

                             port=puerto

                             )

```

```

[ ]: # Abrir un cursor para poder realizar operaciones sobre la base de datos
      cursor = conexion.cursor()

```

```

[ ]: # Crear un schema
      cursor.execute("""
                     CREATE SCHEMA rentacar;
                     """)

```

```

[ ]: # CREAMOS TABLA CLIENTE

      cursor.execute("""
                     CREATE TABLE rentacar.cliente(
                     ↳
                     ↳
                     ↳
                     cedula character varying(10) NOT NULL,
                     nombre character varying(20) NOT NULL,

```

```

        apellido character varying(20) NOT NULL,
        email character varying(50) NOT NULL,
        telefono character varying(10) NOT NULL,
        direccion character varying(100) NOT NULL,
        usuario character varying(15) NOT NULL,
        contrasenia character varying(15) NOT NULL,
        CONSTRAINT cp_cliente PRIMARY KEY (codigo_cliente)
    );
    """
conexion.commit()

```

[]: *# CREAMOS TABLA CATEGORIA*

```

cursor.execute("""
    CREATE TABLE rentacar.categoria(
        codigo_categoria serial NOT NULL,
        nombre character varying(50) NOT NULL,
        gama character varying(50) NOT NULL,
        CONSTRAINT cp_categoria PRIMARY KEY (codigo_categoria)
    );
    """)
conexion.commit()

```

[]: *# CREAMOS TABLA ADMINISTRADOR*

```

cursor.execute("""
    CREATE TABLE rentacar.administrador(
        codigo_administracion serial NOT NULL,
        usuario character varying(15) NOT NULL,
        contrasenia character varying(15) NOT NULL,
        email character varying(50) NOT NULL,
        CONSTRAINT cp_administracion PRIMARY KEY (codigo_administracion)
    );
    """)
conexion.commit()

```

[]: *# CREAMOS TABLA VEHICULO*

```

cursor.execute("""
    CREATE DOMAIN rentacar.tipo_combustible_vehiculo as VARCHAR(10)
    CHECK (VALUE IN ('super','extra','diesel'));
    """)

cursor.execute("""
    CREATE DOMAIN rentacar.estado_vehiculo as VARCHAR(10)
    CHECK (VALUE IN ('reservado','disponible'));
    """)

```

```

cursor.execute("""
        CREATE TABLE rentacar.vehiculo(
            codigo_vehiculo serial NOT NULL,
            marca character varying(100) NOT NULL,
            placa character varying(10) NOT NULL,
            modelo character varying(50) NOT NULL,
            color character varying(50) NOT NULL,
            tipo_combustible rentacar.tipo_combustible_vehiculo NOT NULL
        ↪default 'super',
            numero_plazas int not null,
            numero_puertas int not null,
            kilometraje decimal not null,
            costo_diario decimal not null,
            estado_vehiculo rentacar.estado_vehiculo not null default
        ↪'disponible',
            codigo_categoria int NOT NULL,
            codigo_administrador int NOT NULL,
            CONSTRAINT cp_vehiculo PRIMARY KEY (codigo_vehiculo),
            CONSTRAINT caj_vehiculo_categoria FOREIGN KEY
        ↪(codigo_categoria) references rentacar.categoria
                ON DELETE RESTRICT
                ON UPDATE CASCADE,
            CONSTRAINT caj_vehiculo_administrador FOREIGN KEY
        ↪(codigo_administrador) references rentacar.administrador
                ON DELETE RESTRICT
                ON UPDATE CASCADE
        );
    """)
conexion.commit()

```

[]: *# CREAMOS TABLA FICHA_MANTENIMIENTO*

```

cursor.execute("""
        CREATE TABLE rentacar.ficha_mantenimiento(
            codigo_ficha_mantenimiento serial NOT NULL,
            fecha date NOT NULL,
            kilometraje decimal NOT NULL,
            servicio character varying(200) NOT NULL,
            comentario character varying(200) NOT NULL,
            codigo_vehiculo int NOT NULL,
            CONSTRAINT cp_ficha_mantenimiento PRIMARY KEY
        ↪(codigo_ficha_mantenimiento),
            CONSTRAINT caj_vehiculo_mantenimiento FOREIGN KEY
        ↪(codigo_vehiculo) references rentacar.vehiculo
                ON DELETE RESTRICT
    """)

```

```

        ON UPDATE CASCADE
    );
    """)
conexion.commit()

```

[]: # CREAMOS TABLA RESERVACION

```

cursor.execute("""
    CREATE DOMAIN rentacar.estado_reservacion VARCHAR(10)
    CHECK (VALUE IN ('aceptado','cancelado'));
    """)

cursor.execute("""
    CREATE DOMAIN rentacar.tipo_kilometraje_reservacion VARCHAR(10)
    CHECK (VALUE IN ('libre', 'controlado'));
    """)

cursor.execute("""
    CREATE TABLE rentacar.reservacion(
        codigo_reservacion serial NOT NULL,
        valor decimal NOT NULL,
        fecha_retiro date NOT NULL,
        fecha_entrega date NOT NULL,
        comentario text NOT NULL,
        tipo_kilometraje rentacar.tipo_kilometraje_reservacion NOT NULL,
        ⇨default 'libre',
        estado_reservacion rentacar.estado_reservacion NOT NULL default
        ⇨'aceptado',
        codigo_cliente int NOT NULL,
        codigo_vehiculo int NOT NULL,
        CONSTRAINT cp_reservacion PRIMARY KEY (codigo_reservacion),
        CONSTRAINT caj_reservacion_cliente FOREIGN KEY (codigo_cliente)
        ⇨references rentacar.cliente
            ON DELETE RESTRICT
            ON UPDATE CASCADE,
        CONSTRAINT caj_reservacion_vehiculo FOREIGN KEY
        ⇨(codigo_vehiculo) references rentacar.vehiculo
            ON DELETE RESTRICT
            ON UPDATE CASCADE
    );
    """)
conexion.commit()

```

[]: # CREAMOS TABLA FACTURA_ENCABEZADO

```

cursor.execute("""

```



```

CREATE TABLE rentacar.factura_encabezado(
    codigo_factura_encabezado serial NOT NULL,
    fecha date NOT NULL,
    subtotal decimal NOT NULL,
    iva decimal NOT NULL,
    total decimal NOT NULL,
    codigo_reservacion int NOT NULL,
    CONSTRAINT cp_factura_encabezado PRIMARY KEY,
    ↪(codigo_factura_encabezado),
    CONSTRAINT caj_factura_encabezado_reservacion FOREIGN KEY,
    ↪(codigo_reservacion) references rentacar.reservacion
        ON DELETE RESTRICT
        ON UPDATE CASCADE
);
"""
conexion.commit()

```

[]: *# CREAMOS TABLA FACTURA_DETALLE*

```

cursor.execute("""
    CREATE TABLE rentacar.factura_detalle(
        codigo_factura_detalle serial NOT NULL,
        total decimal NOT NULL,
        total_unitario decimal NOT NULL,
        cantidad int NOT NULL,
        descripcion character varying(200) NOT NULL,
        codigo_factura_encabezado int NOT NULL,
        CONSTRAINT cp_factura_detalle PRIMARY KEY,
    ↪(codigo_factura_detalle),
        CONSTRAINT caj_factura_detalle_encabezado FOREIGN KEY,
    ↪(codigo_factura_encabezado) references rentacar.factura_encabezado
        ON DELETE RESTRICT
        ON UPDATE CASCADE
    );
    """)
conexion.commit()

```

Paso 3: Realizar vista

[]: CREATE VIEW rentacar.view_vehiculos_disponibles AS
 SELECT v.codigo_vehiculo, v.marca, v.placa, v.modelo, v.costeo_diario, v.
 ↪estado_vehiculo FROM rentacar.vehiculo v
 WHERE v.estado_vehiculo = 'disponible';

Paso 4: Esquema físico

```

[ ]: CREATE SCHEMA rentacar;

CREATE TABLE rentacar.cliente(
    codigo_cliente serial NOT NULL,
    cedula character varying(10) NOT NULL,
    nombre character varying(20) NOT NULL,
    apellido character varying(20) NOT NULL,
    email character varying(50) NOT NULL,
    telefono character varying(10) NOT NULL,
    direccion character varying(100) NOT NULL,
    usuario character varying(15) NOT NULL,
    contrasenia character varying(15) NOT NULL,
    CONSTRAINT cp_cliente PRIMARY KEY (codigo_cliente)
);

CREATE TABLE rentacar.categoria(
    codigo_categoria serial NOT NULL,
    nombre character varying(50) NOT NULL,
    gama character varying(50) NOT NULL,
    CONSTRAINT cp_categoria PRIMARY KEY (codigo_categoria)
);

CREATE TABLE rentacar.administrador(
    codigo_administracion serial NOT NULL,
    usuario character varying(15) NOT NULL,
    contrasenia character varying(15) NOT NULL,
    email character varying(50) NOT NULL,
    CONSTRAINT cp_administracion PRIMARY KEY (codigo_administracion)
);

CREATE DOMAIN rentacar.tipo_combustible_vehiculo as VARCHAR(10)
    CHECK (VALUE IN ('super','extra','diesel'));

CREATE DOMAIN rentacar.estado_vehiculo as VARCHAR(10)
    CHECK (VALUE IN ('reservado','disponible'));

CREATE TABLE rentacar.vehiculo(
    codigo_vehiculo serial NOT NULL,
    marca character varying(100) NOT NULL,
    placa character varying(10) NOT NULL,
    modelo character varying(50) NOT NULL,
    color character varying(50) NOT NULL,
    tipo_combustible rentacar.tipo_combustible_vehiculo NOT NULL default ↵
    ↵ 'super',
    numero_plazas int not null,
    numero_puertas int not null,
    kilometraje decimal not null,

```

```

        costo_diario decimal not null,
        estado_vehiculo rentacar.estado_vehiculo not null default 'disponible',
        codigo_categoria int NOT NULL,
        codigo_administrador int NOT NULL,
        CONSTRAINT cp_vehiculo PRIMARY KEY (codigo_vehiculo),
        CONSTRAINT caj_vehiculo_categoria FOREIGN KEY (codigo_categoria)
↳references rentacar.categoria
            ON DELETE RESTRICT
            ON UPDATE CASCADE,
        CONSTRAINT caj_vehiculo_administrador FOREIGN KEY
↳(codigo_administrador) references rentacar.administrador
            ON DELETE RESTRICT
            ON UPDATE CASCADE
    );

CREATE DOMAIN rentacar.estado_reservacion VARCHAR(10)
    CHECK (VALUE IN ('aceptado', 'cancelado'));

CREATE DOMAIN rentacar.tipo_kilometraje_reservacion VARCHAR(10)
    CHECK (VALUE IN ('libre', 'controlado'));

CREATE TABLE rentacar.reservacion(
    codigo_reservacion serial NOT NULL,
    valor decimal NOT NULL,
    fecha_retiro date NOT NULL,
    fecha_entrega date NOT NULL,
    comentario text NOT NULL,
    tipo_kilometraje rentacar.tipo_kilometraje_reservacion NOT NULL default
↳'libre',
    estado_reservacion rentacar.estado_reservacion NOT NULL default
↳'aceptado',
    codigo_cliente int NOT NULL,
    codigo_vehiculo int NOT NULL,
    CONSTRAINT cp_reservacion PRIMARY KEY (codigo_reservacion),
    CONSTRAINT caj_reservacion_cliente FOREIGN KEY (codigo_cliente)
↳references rentacar.cliente
            ON DELETE RESTRICT
            ON UPDATE CASCADE,
    CONSTRAINT caj_reservacion_vehiculo FOREIGN KEY (codigo_vehiculo)
↳references rentacar.vehiculo
            ON DELETE RESTRICT
            ON UPDATE CASCADE
);

CREATE TABLE rentacar.ficha_mantenimiento(
    codigo_ficha_mantenimiento serial NOT NULL,

```

```

        fecha date NOT NULL,
        kilometraje decimal NOT NULL,
        servicio character varying(200) NOT NULL,
        comentario character varying(200) NOT NULL,
        codigo_vehiculo int NOT NULL,
        CONSTRAINT cp_ficha_mantenimiento PRIMARY KEY
↪(codigo_ficha_mantenimiento),
        CONSTRAINT caj_vehiculo_mantenimiento FOREIGN KEY (codigo_vehiculo)
↪references rentacar.vehiculo
            ON DELETE RESTRICT
            ON UPDATE CASCADE
    );

CREATE TABLE rentacar.factura_encabezado(
    codigo_factura_encabezado serial NOT NULL,
    fecha date NOT NULL,
    subtotal decimal NOT NULL,
    iva decimal NOT NULL,
    total decimal NOT NULL,
    codigo_reservacion int NOT NULL,
    CONSTRAINT cp_factura_encabezado PRIMARY KEY
↪(codigo_factura_encabezado),
    CONSTRAINT caj_factura_encabezado_reservacion FOREIGN KEY
↪(codigo_reservacion) references rentacar.reservacion
            ON DELETE RESTRICT
            ON UPDATE CASCADE
);

CREATE TABLE rentacar.factura_detalle(
    codigo_factura_detalle serial NOT NULL,
    total decimal NOT NULL,
    total_unitario decimal NOT NULL,
    cantidad int NOT NULL,
    descripcion character varying(200) NOT NULL,
    codigo_factura_encabezado int NOT NULL,
    CONSTRAINT cp_factura_detalle PRIMARY KEY (codigo_factura_detalle),
    CONSTRAINT caj_factura_detalle_encabezado FOREIGN KEY
↪(codigo_factura_encabezado) references rentacar.factura_encabezado
            ON DELETE RESTRICT
            ON UPDATE CASCADE
);

CREATE VIEW rentacar.view_vehiculos_disponibles AS
SELECT v.codigo_vehiculo, v.marca, v.placa, v.modelo, v.costeo_diario, v.
↪estado_vehiculo FROM rentacar.vehiculo v
WHERE v.estado_vehiculo = 'disponible';

```

4 Despliegue de la base de datos

4.1 Cadena de conexión local

bbdd: el nombre de la base de datos, en este caso: ddddd

```
bbdd = 'actividad_1' #cambiar por tu bdd
```

Usuario: nombre de usuario de la base de datos, en este caso: uuuuu

```
usuario = 'postgres' #cambiar por tu usuario
```

Contraseña: contraseña de la base de datos, en este caso: ppppp

```
contrasena = '' #cambiar por tu contraseña
```

Máquina: nombre de la máquina/host (el que aloja) la base de datos, en este caso: mmmmm

```
maquina = 'localhost'
```

Puerto: puerto de la base de datos, en este caso: 5432 (puerto por defecto)

```
puerto = '5432'
```

Conectar a la base de datos

```
conexion = psycopg2.connect(dbname=bbdd,  
                             user=usuario,  
  
                             password=contrasena,  
  
                             host=maquina,  
  
                             port=puerto  
)
```

4.2 Conexión en la nube

Se obtiene la URL de la base de datos de ElephantSQL,

que tiene formato postgres://usuario:contrasena@maquina:puerto/bbdd

bbdd: el nombre de la base de datos, en este caso: ddddd

```
bbdd = 'xqlnzzfq' #cambiar por tu bdd
```

Usuario: nombre de usuario de la base de datos, en este caso: uuuuu

```
usuario = 'xqlnzzfq' #cambiar por tu usuario
```

Contraseña: contraseña de la base de datos, en este caso: ppppp

```
contrasena = '' #cambiar por tu contraseña
```

Máquina: nombre de la máquina/host (el que aloja) la base de datos, en este caso: mmmmm

```
maquina = 'drona.db.elephantsql.com'
```

Puerto: puerto de la base de datos, en este caso: 5432 (puerto por defecto)

```
puerto = '5432'
```

Conectar a la base de datos

```
conexion = psycopg2.connect(dbname=bdd,  
                             user=usuario,  
                             password=contrasena,  
                             host=maquina,  
                             port=puerto  
)
```

4.2.1 Ejecutar el diseño físico en la nube

```
[ ]: # Importar la libreria  
import psycopg2
```

```
[ ]: # NUBE  
  
# Se obtiene la URL de la base de datos de ElephantSQL,  
# que tiene formato postgres://usuario:contrasena@maquina:puerto/bdd  
  
# bdd: el nombre de la base de datos, en este caso: dddd  
  
bdd = 'xqlnzzfq' #cambiar por tu bdd  
  
# Usuario: nombre de usuario de la base de datos, en este caso: uuuuu  
  
usuario = 'xqlnzzfq' #cambiar por tu usuario  
  
# Contraseña: contraseña de la base de datos, en este caso: ppppp  
  
contrasena = '<contraseña>' #cambiar por tu contraseña  
  
# Máquina: nombre de la máquina/host (el que aloja) la base de datos, en este caso: mmmmm  
maquina = 'drona.db.elephantsql.com'
```

```

# Puerto: puerto de la base de datos, en este caso: 5432 (puerto por defecto)_
puerto = '5432'

# Conectar a la base de datos

conexion = psycopg2.connect(dbname=bdd,

                             user=usuario,

                             password=contrasena,

                             host=maquina,

                             port=puerto

)

```

```

[ ]: # Abrir un cursor para poder realizar operaciones sobre la base de datos
cursor = conexion.cursor()

```

```

[ ]: # Crear un schema
cursor.execute("""
               CREATE SCHEMA rentacar;
               """)

```

```

[ ]: # CREAMOS TABLA CLIENTE

cursor.execute("""
               CREATE TABLE rentacar.cliente(
               ↵
               ↵
               ↵
               codigo_cliente serial NOT NULL,
               ↵
               ↵
               cedula character varying(10) NOT NULL,
               nombre character varying(20) NOT NULL,
               apellido character varying(20) NOT NULL,
               email character varying(50) NOT NULL,
               telefono character varying(10) NOT NULL,
               direccion character varying(100) NOT NULL,
               usuario character varying(15) NOT NULL,
               contrasena character varying(15) NOT NULL,
               CONSTRAINT cp_cliente PRIMARY KEY (codigo_cliente)
               );
               """)
conexion.commit()

```

```
[ ]: # CREAMOS TABLA CATEGORIA
```

```
cursor.execute("""
    CREATE TABLE rentacar.categoria(
        codigo_categoria serial NOT NULL,
        nombre character varying(50) NOT NULL,
        gama character varying(50) NOT NULL,
        CONSTRAINT cp_categoria PRIMARY KEY (codigo_categoria)
    );
""")
conexion.commit()
```

```
[ ]: # CREAMOS TABLA ADMINISTRADOR
```

```
cursor.execute("""
    CREATE TABLE rentacar.administrador(
        codigo_administracion serial NOT NULL,
        usuario character varying(15) NOT NULL,
        contrasenia character varying(15) NOT NULL,
        email character varying(50) NOT NULL,
        CONSTRAINT cp_administracion PRIMARY KEY (codigo_administracion)
    );
""")
conexion.commit()
```

```
[ ]: # CREAMOS TABLA VEHICULO
```

```
cursor.execute("""
    CREATE DOMAIN rentacar.tipo_combustible_vehiculo as VARCHAR(10)
    CHECK (VALUE IN ('super','extra','diesel'));
""")
```

```
cursor.execute("""
    CREATE DOMAIN rentacar.estado_vehiculo as VARCHAR(10)
    CHECK (VALUE IN ('reservado','disponible'));
""")
```

```
cursor.execute("""
    CREATE TABLE rentacar.vehiculo(
        codigo_vehiculo serial NOT NULL,
        marca character varying(100) NOT NULL,
        placa character varying(10) NOT NULL,
        modelo character varying(50) NOT NULL,
        color character varying(50) NOT NULL,
        tipo_combustible rentacar.tipo_combustible_vehiculo NOT NULL,
        ↪default 'super',
        numero_plazas int not null,
```



```

        numero_puertas int not null,
        kilometraje decimal not null,
        costo_diario decimal not null,
        estado_vehiculo rentacar.estado_vehiculo not null default 1
↪ 'disponible',
        codigo_categoria int NOT NULL,
        codigo_administrador int NOT NULL,
        CONSTRAINT cp_vehiculo PRIMARY KEY (codigo_vehiculo),
        CONSTRAINT caj_vehiculo_categoria FOREIGN KEY 1
↪ (codigo_categoria) references rentacar.categoria
        ON DELETE RESTRICT
        ON UPDATE CASCADE,
        CONSTRAINT caj_vehiculo_administrador FOREIGN KEY 1
↪ (codigo_administrador) references rentacar.administrador
        ON DELETE RESTRICT
        ON UPDATE CASCADE
    );
    """
conexion.commit()

```

[]: # CREAMOS TABLA FICHA_MANTENIMIENTO

```

cursor.execute("""
    CREATE TABLE rentacar.ficha_mantenimiento(
        codigo_ficha_mantenimiento serial NOT NULL,
        fecha date NOT NULL,
        kilometraje decimal NOT NULL,
        servicio character varying(200) NOT NULL,
        comentario character varying(200) NOT NULL,
        codigo_vehiculo int NOT NULL,
        CONSTRAINT cp_ficha_mantenimiento PRIMARY KEY 1
↪ (codigo_ficha_mantenimiento),
        CONSTRAINT caj_vehiculo_mantenimiento FOREIGN KEY 1
↪ (codigo_vehiculo) references rentacar.vehiculo
        ON DELETE RESTRICT
        ON UPDATE CASCADE
    );
    """)
conexion.commit()

```

[]: # CREAMOS TABLA RESERVACION

```

cursor.execute("""
    CREATE DOMAIN rentacar.estado_reservacion VARCHAR(10)
    CHECK (VALUE IN ('aceptado','cancelado'));
    """)

```

```

cursor.execute("""
    CREATE DOMAIN rentacar.tipo_kilometraje_reservacion VARCHAR(10)
    CHECK (VALUE IN ('libre', 'controlado'));
""")

cursor.execute("""
    CREATE TABLE rentacar.reservacion(
        codigo_reservacion serial NOT NULL,
        valor decimal NOT NULL,
        fecha_retiro date NOT NULL,
        fecha_entrega date NOT NULL,
        comentario text NOT NULL,
        tipo_kilometraje rentacar.tipo_kilometraje_reservacion NOT NULL,
↪default 'libre',
        estado_reservacion rentacar.estado_reservacion NOT NULL default,
↪'aceptado',
        codigo_cliente int NOT NULL,
        codigo_vehiculo int NOT NULL,
        CONSTRAINT cp_reservacion PRIMARY KEY (codigo_reservacion),
        CONSTRAINT caj_reservacion_cliente FOREIGN KEY (codigo_cliente)
↪references rentacar.cliente
            ON DELETE RESTRICT
            ON UPDATE CASCADE,
        CONSTRAINT caj_reservacion_vehiculo FOREIGN KEY
↪(codigo_vehiculo) references rentacar.vehiculo
            ON DELETE RESTRICT
            ON UPDATE CASCADE
    );
""")
conexion.commit()

```

[]: # CREAMOS TABLA FACTURA_ENCABEZADO

```

cursor.execute("""
    CREATE TABLE rentacar.factura_encabezado(
        codigo_factura_encabezado serial NOT NULL,
        fecha date NOT NULL,
        subtotal decimal NOT NULL,
        iva decimal NOT NULL,
        total decimal NOT NULL,
        codigo_reservacion int NOT NULL,
        CONSTRAINT cp_factura_encabezado PRIMARY KEY
↪(codigo_factura_encabezado),
        CONSTRAINT caj_factura_encabezado_reservacion FOREIGN KEY
↪(codigo_reservacion) references rentacar.reservacion
    );
""")

```

```

        ON DELETE RESTRICT
        ON UPDATE CASCADE
    );
    """
conexion.commit()

```

```
[ ]: # CREAMOS TABLA FACTURA_DETALLE
```

```

cursor.execute("""
    CREATE TABLE rentacar.factura_detalle(
        codigo_factura_detalle serial NOT NULL,
        total decimal NOT NULL,
        total_unitario decimal NOT NULL,
        cantidad int NOT NULL,
        descripcion character varying(200) NOT NULL,
        codigo_factura_encabezado int NOT NULL,
        CONSTRAINT cp_factura_detalle PRIMARY KEY
        ↪(codigo_factura_detalle),
        CONSTRAINT caj_factura_detalle_encabezado FOREIGN KEY
        ↪(codigo_factura_encabezado) references rentacar.factura_encabezado
        ON DELETE RESTRICT
        ON UPDATE CASCADE
    );
    """)
conexion.commit()

```

```
[ ]: # Cerrar conexion y cursor
```

```

cursor.close()
conexion.close()

```

4.2.2 Inserción de datos en local

```
[ ]: # Importar la libreria
```

```
import psycopg2
```

```
[ ]: # LOCAL
```

```

# Se obtiene la URL de la base de datos de ElephantSQL,
# que tiene formato postgres://usuario:contrasena@maquina:puerto/bbdd

# bbdd: el nombre de la base de datos, en este caso: dddd

bbdd = 'actividad_1' #cambiar por tu bbdd

```

```

# Usuario: nombre de usuario de la base de datos, en este caso: uuuuu
usuario = 'postgres' #cambiar por tu usuario

# Contraseña: contraseña de la base de datos, en este caso: ppppp
contrasenia = '<contraseña>' #cambiar por tu contraseña

# Máquina: nombre de la máquina/host (el que aloja) la base de datos, en este caso: mmmmm
maquina = 'localhost'

# Puerto: puerto de la base de datos, en este caso: 5432 (puerto por defecto)
puerto = '5432'

# Conectar a la base de datos
conexion = psycopg2.connect(dbname=bbdd,

                             user=usuario,

                             password=contrasenia,

                             host=maquina,

                             port=puerto

                             )

```

```

[ ]: # Insercion de clientes

cursor.execute("""
    INSERT INTO rentacar.CLIENTE(cedula, nombre, apellido, email,
    telefono, direccion, usuario, contrasenia)
    VALUES
    ('1002003000','Pedro','Lopez','pedro@gmail.com','0987654321','C/
    Teodoro gomez 2-33 Cubilche','pedro1545','jfMe345'),
    ('1002003001','Hugo','Benítez','Hugo@gmail.com','0987654322','C/
    Teodoro gomez 2-33 Cubilche','Hugo143','vvzzdv'),
    ('1002003002','Martín','Contreras','Contreras@gmail.
    com','0987654323','C/ Teodoro gomez 2-33 Contreras143','pedro8','juunt'),
    ('1002003003','Lucas','Díaz','Lucas@gmail.com','0987654324','C/
    Teodoro gomez 2-33 Cubilche','Lucas334','jfMs45'),

```

```

                ('1002003004','Pedro','Contreras','Lucas2@gmail.
↪com','0987654325','C/ Teodoro gomez 2-33_
↪Cubilche','Contreras2r4','34242vddf'),
                ('1002003005','Mateo','Contreras','Mateo3@gmail.
↪com','0987654326','C/ Teodoro gomez 2-33 Cubilche','Mateo1ew','cbcbxd'),
                ('1002003006','Leo','Lopez','Leo@gmail.com','0987654327','C/_
↪Teodoro gomez 2-33 Cubilche','Lower','vdivfwww'),
                ('1002003007','Daniel','Álvarez','Daniel5@gmail.
↪com','0987654328','C/ Teodoro gomez 2-33 Cubilche','Danielrwrw','fsfsfd'),
                ('1002003008','Alejandro','Andrade','Andrad6e@gmail.
↪com','0987654329','C/ Teodoro gomez 2-33 Cubilche','Alejandrorwer','nncnb'),
                ('1002003009','Pablo','Lopez','Pablo2@gmail.
↪com','0987654334','C/ Teodoro gomez 2-33 Cubilche','Lopez43','miuiuiu'),
                ('2002003001','Alan','Lopez','Alan8@gmail.com','0987654345','C/_
↪Teodoro gomez 2-33 Cubilche','Alan43435','qwq3wsa'),
                ('2002003002','Pedro','Álvarez','pedro16@gmail.
↪com','0987654354','C/ Teodoro gomez 2-33 Cubilche','pedro122','4erdfvv'),
                ('2002003003','Hugo','Castro','Hugo33@gmail.
↪com','0987623321','C/ Teodoro gomez 2-33 Cubilche','pedro344','jfMe345'),
                ('2002003004','Omar','Benítez','Omar66@gmail.
↪com','0987654321','C/ Teodoro gomez 2-33 Cubilche','Omar2223','er234'),
                ('2002003005','Leo','Duarte','Duarte43@gmail.
↪com','0923654321','C/ Teodoro gomez 2-33 Cubilche','Duarte666','re435tyg'),
                ('2002003006','Pedro','Lopez','pedro32@gmail.
↪com','0987444321','C/ Teodoro gomez 2-33 Cubilche','pedro4644','vfdw34'),
                ('2002003007','Omar','Espinoza','Omar34@gmail.
↪com','0987654356','C/ Teodoro gomez 2-33 Cubilche','Omar3646','2wsdf'),
                ('2002003008','Pedro','Lopez','pedro422@gmail.
↪com','0934654321','C/ Teodoro gomez 2-33 Cubilche','pedro5646','87tfg'),
                ('2002003009','Lucas','Díaz','Lucas35@gmail.
↪com','0987656321','C/ Teodoro gomez 2-33 Cubilche','Lucas6646','0987ygf'),
                ('2002003000','Mateo','Andrade','Andrade67@gmail.
↪com','0986254321','C/ Teodoro gomez 2-33 Cubilche','Andrade3464','45rfg');
↪
                """)
conexion.commit()

```

[]: # Insercion de CATEGORIA

```

cursor.execute("""
                INSERT INTO rentacar.CATEGORIA(nombre, gama)
                VALUES
                ('Deportivos','Alta'),
                ('Deportivos','Media'),
                ('Compactos','Media'),
                ('Compactos','Baja'),

```

```

        ('Subcompactos','Alta'),
        ('Subcompactos','Media'),
        ('Subcompactos','Baja');
    """
conexion.commit()

```

[]: *# Insercion de ADMINISTRADOR*

```

cursor.execute("""
    INSERT INTO rentacar.ADMINISTRADOR(usuario, contrasenia, email)
    VALUES
    ('joseAdmin','32453','joseAdmin@gmail.com'),
    ('mariaAdmin','34533','mariaAdmin@gmail.com'),
    ('marcosAdmin','43535','marcosAdmin@gmail.com');
    """)
conexion.commit()

```

[]: *# Insercion de VEHICULO*

```

cursor.execute("""
    INSERT INTO rentacar.VEHICULO(
        marca,
        modelo,
        costo_diario,
        kilometraje,
        numero_puertas,
        placa,
        color,
        tipo_combustible,
        numero_plazas,
        estado_vehiculo,
        codigo_categoria,
        codigo_administrador
    )
    VALUES
    ('Ford_
↪Fiesta','LS','12','123456','4','pbi1234','rojo','super',5,'disponible',1,2),
    ('Toyota_
↪Corolla','ES','43','123567','3','qwe234','verde','extra',4,'reservado',2,2),
    ('Honda_
↪Civic','HEV','54','123678','4','poi456','negro','super',2,'reservado',1,2),
    ('Ford_
↪Fiesta','IS','65','123789','4','lkj432','negro','diesel',5,'disponible',1,2),
    ('Kia_
↪Rio','CT','102','123890','3','mnb456','verde','diesel',5,'disponible',3,2),

```

```

        ('Toyota_
↪Corolla','LC','64','234567','3','ghj576','amarillo','extra',4,'reservado',1,2),
        ('Ford_
↪Fiesta','RX','53','234678','4','sdf970','rojo','super',2,'disponible',1,1),
        ('Honda_
↪Civic','GT','22','234789','3','xcv234','negro','super',4,'reservado',2,1),
        ('Kia_
↪Rio','S','65','234890','4','ret456','verde','extra',5,'reservado',2,2),
        ('Lexus_
↪ES','X','76','345678','4','dfe284','blanco','diesel',4,'disponible',2,1);
        """)
conexion.commit()

```

[]: # Insercion de FICHA_MANTENIMIENTO

```

cursor.execute("""
        INSERT INTO rentacar.FICHA_MANTENIMIENTO(
            fecha,
            kilometraje,
            servicio,
            comentario,
            codigo_vehiculo
        )
        VALUES
        (DATE '2022-01-20','020556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',1),
        (DATE '2021-02-23','120556','Reemplazar las Bujías','sin_
↪ninguna observación',1),
        (DATE '2013-03-25','127556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',1),
        (DATE '2023-04-26','121524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',1),
        (DATE '2009-05-20','136556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',1),
        (DATE '2011-06-12','163556','Servicio de Dirección_
↪Asistida','sin ninguna observación',1),
        (DATE '2022-07-20','167556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',1),
        (DATE '2022-08-20','120195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',1),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',2),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',2),

```

```

        (DATE '2021-03-24','227556','Cambiar el Liquido de
↪Transmission','sin ninguna observación',2),
        (DATE '2005-06-12','263556','Servicio de Dirección
↪Asistida','sin ninguna observación',2),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de
↪Combustible','sin ninguna observación',2),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula
↪PCV','sin ninguna observación',2),
        (DATE '2022-08-20','221195','Revise y Reemplace su Válvula
↪PCV','sin ninguna observación',2),

        (DATE '2012-01-20','120556','Alineación y balanceo de
↪ruedas','sin ninguna observación',3),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin
↪ninguna observación',3),
        (DATE '2021-03-24','227556','Cambiar el Liquido de
↪Transmission','sin ninguna observación',3),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para
↪Engranajes','sin ninguna observación',3),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de
↪Combustible','sin ninguna observación',3),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula
↪PCV','sin ninguna observación',3),
        (DATE '2022-03-24','237556','Cambiar el Liquido de
↪Transmission','sin ninguna observación',3),

        (DATE '2012-01-20','120556','Alineación y balanceo de
↪ruedas','sin ninguna observación',4),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin
↪ninguna observación',4),
        (DATE '2021-03-24','227556','Cambiar el Liquido de
↪Transmission','sin ninguna observación',4),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para
↪Engranajes','sin ninguna observación',4),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin
↪ninguna observación',4),
        (DATE '2005-06-12','263556','Servicio de Dirección
↪Asistida','sin ninguna observación',4),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula
↪PCV','sin ninguna observación',4),
        (DATE '2004-01-23','221556','Reemplazar las Bujías','sin
↪ninguna observación',4),

        (DATE '2012-01-20','120556','Alineación y balanceo de
↪ruedas','sin ninguna observación',5),

```



```

        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',5),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',5),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',5),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',5),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',5),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',5),
        (DATE '2015-01-20','220556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',5),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',6),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',6),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',6),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',6),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',6),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',6),
        (DATE '2006-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',6),

        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',7),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',7),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',7),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',7),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',7),
        (DATE '2015-04-26','221624','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',7),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',8),

```

```

        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',8),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',8),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',8),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',8),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',8),
        (DATE '2014-12-26','224524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',8),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',9),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',9),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',9),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',9),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',9),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',9),
        (DATE '2021-05-24','228556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',9),
        (DATE '2021-07-24','237556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',9);
    """)
conexion.commit()

```

[]: *# Insercion de RESERVACION*

```

cursor.execute("""
        INSERT INTO rentacar.RESERVACION(
            fecha_retiro,
            fecha_entrega,
            comentario,
            tipo_kilometraje,
            valor,
            estado_reservacion,
            codigo_cliente,
            codigo_vehiculo
        )
        VALUES

```

```

        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',1,1),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',2,1),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','libre',243,'aceptado',3,2),
        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','controlado',253,'aceptado',4,3),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','controlado',263,'cancelado',5,4),
        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','libre',273,'aceptado',6,5),
        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',283,'cancelado',7,6),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','libre',213,'aceptado',8,7),

        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',9,6),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',2,10),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','libre',243,'aceptado',4,9),
        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','controlado',253,'aceptado',3,3),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','libre',263,'cancelado',1,8),
        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','controlado',273,'aceptado',4,5),
        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','libre',283,'cancelado',5,6),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','controlado',213,'aceptado',1,7),

        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',2,3),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',2,4),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','controlado',243,'aceptado',3,2),
        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','libre',253,'aceptado',4,5),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','controlado',263,'cancelado',5,4),

```

```

        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','libre',273,'aceptado',6,3),
        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',283,'cancelado',7,1),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','libre',213,'aceptado',8,10),

        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',12,3),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',14,4),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','controlado',243,'aceptado',15,2),
        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','libre',253,'aceptado',14,5),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','controlado',263,'cancelado',15,4),
        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','libre',273,'aceptado',16,3),
        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',283,'cancelado',17,1),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','libre',213,'aceptado',18,10);
    """)
conexion.commit()

```

[]: *# Insercion de FACTURA_ENCABEZADO*

```

cursor.execute("""
        INSERT INTO rentacar.FACTURA_ENCABEZADO(
            fecha,
            subtotal,
            iva,
            total,
            codigo_reservacion
        )
        VALUES
        (DATE '2022-01-20',23,0.12,200,1),
        (DATE '2021-02-23',22,0.12,210,2),
        (DATE '2013-03-25',34,0.12,230,3),
        (DATE '2023-04-26',65,0.12,250,4),
        (DATE '2009-05-20',43,0.12,300,5),
        (DATE '2011-06-12',63,0.12,250,6),
        (DATE '2022-07-20',43,0.12,340,7),
        (DATE '2022-08-20',23,0.12,550,8),

```

```

        (DATE '2022-01-20',14,0.12,400,9),
        (DATE '2021-02-23',33,0.12,252,10),
        (DATE '2013-03-25',24,0.12,320,11),
        (DATE '2023-04-26',54,0.12,120,12),
        (DATE '2009-05-20',46,0.12,110,13),
        (DATE '2011-06-12',56,0.12,120,14),
        (DATE '2022-07-20',46,0.12,210,15),
        (DATE '2022-08-20',56,0.12,200,16),

        (DATE '2022-01-20',34,0.12,220,17),
        (DATE '2021-02-23',43,0.12,430,18),
        (DATE '2013-03-25',54,0.12,120,19),
        (DATE '2023-04-26',65,0.12,440,20),
        (DATE '2009-05-20',45,0.12,500,21),
        (DATE '2011-06-12',26,0.12,230,22),
        (DATE '2022-07-20',45,0.12,330,23),
        (DATE '2022-08-20',26,0.12,440,24),

        (DATE '2022-01-20',45,0.12,290,25),
        (DATE '2021-02-23',65,0.12,230,26),
        (DATE '2013-03-25',76,0.12,150,27),
        (DATE '2023-04-26',87,0.12,160,28),
        (DATE '2009-05-20',66,0.12,150,29),
        (DATE '2011-06-12',89,0.12,150,30),
        (DATE '2022-07-20',66,0.12,160,31),
        (DATE '2022-08-20',29,0.12,610,32);
    """)

```

```
conexion.commit()
```

```
[ ]: # Insercion de FACTURA_DETALLE
```

```

cursor.execute("""
    INSERT INTO rentacar.FACTURA_DETALLE(
        cantidad,
        total_unitario,
        total,
        descripcion,
        codigo_factura_encabezado
    )
    VALUES
    (1,23,200,'alquiler vehiculo',1),
    (1,100,100,'alquiler gps',1),
    (1,22,210,'alquiler vehiculo',2),
    (1,50,50,'alquiler silla bebe',2),
    (1,34,230,'alquiler vehiculo',3),
    (1,50,50,'alquiler silla bebe',3),
    (1,65,250,'alquiler vehiculo',4),

```

(1,100,100,'alquiler gps',4),
(1,43,300,'alquiler vehiculo',5),
(1,50,50,'alquiler silla bebe',5),
(1,63,250,'alquiler vehiculo',6),
(1,100,100,'alquiler gps',6),
(1,43,340,'alquiler vehiculo',7),
(1,23,550,'alquiler vehiculo',8),
(1,100,100,'alquiler gps',8),

(1,14,400,'alquiler vehiculo',9),
(1,33,252,'alquiler vehiculo',10),
(1,100,100,'alquiler gps',10),
(1,24,320,'alquiler vehiculo',11),
(1,50,50,'alquiler silla bebe',11),
(1,54,120,'alquiler vehiculo',12),
(1,46,110,'alquiler vehiculo',13),
(1,100,100,'alquiler gps',13),
(1,56,120,'alquiler vehiculo',14),
(1,50,50,'alquiler silla bebe',14),
(1,46,210,'alquiler vehiculo',15),
(1,100,100,'alquiler gps',15),
(1,56,200,'alquiler vehiculo',16),
(1,100,100,'alquiler gps',16),

(1,34,220,'alquiler vehiculo',17),
(1,43,430,'alquiler vehiculo',18),
(1,100,100,'alquiler gps',18),
(1,54,120,'alquiler vehiculo',19),
(1,50,50,'alquiler silla bebe',19),
(1,100,100,'alquiler gps',19),
(1,65,440,'alquiler vehiculo',20),
(1,50,50,'alquiler silla bebe',20),
(1,45,500,'alquiler vehiculo',21),
(1,26,230,'alquiler vehiculo',22),
(1,100,100,'alquiler gps',22),
(1,45,330,'alquiler vehiculo',23),
(1,50,50,'alquiler silla bebe',23),
(1,26,440,'alquiler vehiculo',24),
(1,100,100,'alquiler gps',24),

(1,45,290,'alquiler vehiculo',25),
(1,50,50,'alquiler silla bebe',25),
(1,65,230,'alquiler vehiculo',26),
(1,100,100,'alquiler gps',26),
(1,76,150,'alquiler vehiculo',27),
(1,87,160,'alquiler vehiculo',28),
(1,100,100,'alquiler gps',28),

```

        (1,50,50,'alquiler silla bebe',28),
        (1,66,150,'alquiler vehiculo',29),
        (1,89,150,'alquiler vehiculo',30),
        (1,100,100,'alquiler gps',30),
        (1,66,160,'alquiler vehiculo',31),
        (1,100,100,'alquiler gps',31),
        (1,29,610,'alquiler vehiculo',32),
        (1,50,50,'alquiler silla bebe',32);
    """
    conexion.commit()

```

```

[ ]: # Cerrar conexion y cursor
      cursor.close()
      conexion.close()

```

4.2.3 Inserción de datos en la nube

```

[ ]: # Importar la libreria
      import psycopg2

```

```

[ ]: # NUBE

      # Se obtiene la URL de la base de datos de ElephantSQL,

      # que tiene formato postgres://usuario:contrasenya@maquina:puerto/bbdd

      # bbdd: el nombre de la base de datos, en este caso: dddd

      bbdd = 'xqlnzzfq' #cambiar por tu bbdd

      # Usuario: nombre de usuario de la base de datos, en este caso: uuuuu

      usuario = 'xqlnzzfq' #cambiar por tu usuario

      # Contraseña: contraseña de la base de datos, en este caso: ppppp

      contrasenya = '<contraseña>' #cambiar por tu contraseña

      # Máquina: nombre de la máquina/host (el que aloja) la base de datos, en este
      ↪ caso: mmmmm

      maquina = 'drona.db.elephantsql.com'

      # Puerto: puerto de la base de datos, en este caso: 5432 (puerto por defecto)_

```

```

puerto = '5432'

# Conectar a la base de datos

conexion = psycopg2.connect(dbname=bdd,

                             user=usuario,

                             password=contrasena,

                             host=maquina,

                             port=puerto

)

```

[]: *# Insercion de clientes*

```

cursor.execute("""
    INSERT INTO rentacar.CLIENTE(cedula, nombre, apellido, email,
    ↪telefono, direccion, usuario, contrasena)
    VALUES
    ('1002003000','Pedro','Lopez','pedro@gmail.com','0987654321','C/
    ↪ Teodoro gomez 2-33 Cubilche','pedro1545','jfMe345'),
    ('1002003001','Hugo','Benítez','Hugo@gmail.com','0987654322','C/
    ↪ Teodoro gomez 2-33 Cubilche','Hugo143','vvzzdv'),
    ('1002003002','Martín','Contreras','Contreras@gmail.
    ↪com','0987654323','C/ Teodoro gomez 2-33 Contreras143','pedro8','juunt'),
    ('1002003003','Lucas','Díaz','Lucas@gmail.com','0987654324','C/
    ↪Teodoro gomez 2-33 Cubilche','Lucas334','jfMs45'),
    ('1002003004','Pedro','Contreras','Lucas2@gmail.
    ↪com','0987654325','C/ Teodoro gomez 2-33
    ↪Cubilche','Contreras2r4','34242vddf'),
    ('1002003005','Mateo','Contreras','Mateo3@gmail.
    ↪com','0987654326','C/ Teodoro gomez 2-33 Cubilche','Mateo1ew','cbcbxd'),
    ('1002003006','Leo','Lopez','Leo@gmail.com','0987654327','C/
    ↪Teodoro gomez 2-33 Cubilche','Leower','vdrvfvww'),
    ('1002003007','Daniel','Álvarez','Daniel5@gmail.
    ↪com','0987654328','C/ Teodoro gomez 2-33 Cubilche','Danielrwrw','fsfsfd'),
    ('1002003008','Alejandro','Andrade','Andrad6e@gmail.
    ↪com','0987654329','C/ Teodoro gomez 2-33 Cubilche','Alejandrorwer','nnbnb'),
    ('1002003009','Pablo','Lopez','Pablo2@gmail.
    ↪com','0987654334','C/ Teodoro gomez 2-33 Cubilche','Lopez43','miuiuiu'),

```



```

                ('2002003001','Alan','Lopez','Alan8@gmail.com','0987654345','C/
↪Teodoro gomez 2-33 Cubilche','Alan43435','qwq3wsa'),
                ('2002003002','Pedro','Álvarez','pedro16@gmail.
↪com','0987654354','C/ Teodoro gomez 2-33 Cubilche','pedro122','4erdfvv'),
                ('2002003003','Hugo','Castro','Hugo33@gmail.
↪com','0987623321','C/ Teodoro gomez 2-33 Cubilche','pedro344','jfMe345'),
                ('2002003004','Omar','Benítez','Omar66@gmail.
↪com','0987654321','C/ Teodoro gomez 2-33 Cubilche','Omar2223','er234'),
                ('2002003005','Leo','Duarte','Duarte43@gmail.
↪com','0923654321','C/ Teodoro gomez 2-33 Cubilche','Duarte666','re435tyg'),
                ('2002003006','Pedro','Lopez','pedro32@gmail.
↪com','0987444321','C/ Teodoro gomez 2-33 Cubilche','pedro4644','vfdw34'),
                ('2002003007','Omar','Espinoza','Omar34@gmail.
↪com','0987654356','C/ Teodoro gomez 2-33 Cubilche','Omar3646','2wsdf'),
                ('2002003008','Pedro','Lopez','pedro422@gmail.
↪com','0934654321','C/ Teodoro gomez 2-33 Cubilche','pedro5646','87tfg'),
                ('2002003009','Lucas','Díaz','Lucas35@gmail.
↪com','0987656321','C/ Teodoro gomez 2-33 Cubilche','Lucas6646','0987ygf'),
                ('2002003000','Mateo','Andrade','Andrade67@gmail.
↪com','0986254321','C/ Teodoro gomez 2-33 Cubilche','Andrade3464','45rfg');
↪
            """)
conexion.commit()

```

[]: *# Insercion de CATEGORIA*

```

cursor.execute("""
                INSERT INTO rentacar.CATEGORIA(nombre, gama)
                VALUES
                ('Deportivos','Alta'),
                ('Deportivos','Media'),
                ('Compactos','Media'),
                ('Compactos','Baja'),
                ('Subcompactos','Alta'),
                ('Subcompactos','Media'),
                ('Subcompactos','Baja');
                """)
conexion.commit()

```

[]: *# Insercion de ADMINISTRADOR*

```

cursor.execute("""
                INSERT INTO rentacar.ADMINISTRADOR(usuario, contrasenia, email)
                VALUES
                ('joseAdmin','32453','joseAdmin@gmail.com'),
                ('mariaAdmin','34533','mariaAdmin@gmail.com'),

```

```

                ('marcosAdmin','43535','marcosAdmin@gmail.com');
            """)
conexion.commit()

```

[]: # Insercion de VEHICULO

```

cursor.execute("""
    INSERT INTO rentacar.VEHICULO(
        marca,
        modelo,
        costo_diario,
        kilometraje,
        numero_puertas,
        placa,
        color,
        tipo_combustible,
        numero_plazas,
        estado_vehiculo,
        codigo_categoria,
        codigo_administrador
    )
    VALUES
    ('Ford_
↪Fiesta','LS','12','123456','4','pbi1234','rojo','super',5,'disponible',1,2),
    ('Toyota_
↪Corolla','ES','43','123567','3','qwe234','verde','extra',4,'reservado',2,2),
    ('Honda_
↪Civic','HEV','54','123678','4','poi456','negro','super',2,'reservado',1,2),
    ('Ford_
↪Fiesta','IS','65','123789','4','lkj432','negro','diesel',5,'disponible',1,2),
    ('Kia_
↪Rio','CT','102','123890','3','mnb456','verde','diesel',5,'disponible',3,2),
    ('Toyota_
↪Corolla','LC','64','234567','3','ghj576','amarillo','extra',4,'reservado',1,2),
    ('Ford_
↪Fiesta','RX','53','234678','4','sdf970','rojo','super',2,'disponible',1,1),
    ('Honda_
↪Civic','GT','22','234789','3','xcv234','negro','super',4,'reservado',2,1),
    ('Kia_
↪Rio','S','65','234890','4','ret456','verde','extra',5,'reservado',2,2),
    ('Lexus_
↪ES','X','76','345678','4','dfe284','blanco','diesel',4,'disponible',2,1);
    """)
conexion.commit()

```

```
[ ]: # Insercion de FICHA_MANTENIMIENTO
```

```
cursor.execute("""
    INSERT INTO rentacar.FICHA_MANTENIMIENTO(
        fecha,
        kilometraje,
        servicio,
        comentario,
        codigo_vehiculo
    )
    VALUES
        (DATE '2022-01-20','020556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',1),
        (DATE '2021-02-23','120556','Reemplazar las Bujías','sin_
↪ninguna observación',1),
        (DATE '2013-03-25','127556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',1),
        (DATE '2023-04-26','121524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',1),
        (DATE '2009-05-20','136556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',1),
        (DATE '2011-06-12','163556','Servicio de Dirección_
↪Asistida','sin ninguna observación',1),
        (DATE '2022-07-20','167556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',1),
        (DATE '2022-08-20','120195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',1),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',2),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',2),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',2),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',2),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',2),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',2),
        (DATE '2022-08-20','221195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',2),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',3),
```

```

        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',3),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',3),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',3),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',3),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',3),
        (DATE '2022-03-24','237556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',3),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',4),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',4),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',4),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',4),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',4),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',4),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',4),
        (DATE '2004-01-23','221556','Reemplazar las Bujías','sin_
↪ninguna observación',4),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',5),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',5),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',5),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',5),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',5),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',5),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',5),

```

```

        (DATE '2015-01-20','220556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',5),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',6),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',6),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',6),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',6),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',6),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',6),
        (DATE '2006-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',6),

        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',7),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',7),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',7),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',7),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',7),
        (DATE '2015-04-26','221624','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',7),

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',8),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',8),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',8),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',8),
        (DATE '2015-05-20','236556','Cambiar el Líquido de Frenos','sin_
↪ninguna observación',8),
        (DATE '2005-06-12','263556','Servicio de Dirección_
↪Asistida','sin ninguna observación',8),
        (DATE '2014-12-26','224524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',8),

```

```

        (DATE '2012-01-20','120556','Alineación y balanceo de_
↪ruedas','sin ninguna observación',9),
        (DATE '2002-02-23','220556','Reemplazar las Bujías','sin_
↪ninguna observación',9),
        (DATE '2021-03-24','227556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',9),
        (DATE '2014-04-26','221524','Cambiar el Lubricante para_
↪Engranajes','sin ninguna observación',9),
        (DATE '2008-07-20','267556','Limpiar los Inyectores de_
↪Combustible','sin ninguna observación',9),
        (DATE '2020-08-20','220195','Revise y Reemplace su Válvula_
↪PCV','sin ninguna observación',9),
        (DATE '2021-05-24','228556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',9),
        (DATE '2021-07-24','237556','Cambiar el Liquido de_
↪Transmission','sin ninguna observación',9);
    """)
conexion.commit()

```

[]: *# Insercion de RESERVACION*

```

cursor.execute("""
        INSERT INTO rentacar.RESERVACION(
            fecha_retiro,
            fecha_entrega,
            comentario,
            tipo_kilometraje,
            valor,
            estado_reservacion,
            codigo_cliente,
            codigo_vehiculo
        )
        VALUES
        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',1,1),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',2,1),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','libre',243,'aceptado',3,2),
        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','controlado',253,'aceptado',4,3),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','controlado',263,'cancelado',5,4),
        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','libre',273,'aceptado',6,5),

```

```

        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',283,'cancelado',7,6),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','libre',213,'aceptado',8,7),

        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',9,6),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',2,10),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','libre',243,'aceptado',4,9),
        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','controlado',253,'aceptado',3,3),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','libre',263,'cancelado',1,8),
        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','controlado',273,'aceptado',4,5),
        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','libre',283,'cancelado',5,6),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','controlado',213,'aceptado',1,7),

        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',2,3),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',2,4),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','controlado',243,'aceptado',3,2),
        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','libre',253,'aceptado',4,5),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','controlado',263,'cancelado',5,4),
        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','libre',273,'aceptado',6,3),
        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',283,'cancelado',7,1),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','libre',213,'aceptado',8,10),

        (DATE '2022-01-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',233,'aceptado',12,3),
        (DATE '2021-02-23',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',223,'cancelado',14,4),
        (DATE '2013-03-25',DATE '2015-01-20','sin ninguna_
↪novedad','controlado',243,'aceptado',15,2),

```

```

        (DATE '2023-04-26',DATE '2024-01-20','sin ninguna_
↪novedad','libre',253,'aceptado',14,5),
        (DATE '2009-05-20',DATE '2010-01-20','sin ninguna_
↪novedad','controlado',263,'cancelado',15,4),
        (DATE '2011-06-12',DATE '2012-01-20','sin ninguna_
↪novedad','libre',273,'aceptado',16,3),
        (DATE '2022-07-20',DATE '2023-01-20','sin ninguna_
↪novedad','controlado',283,'cancelado',17,1),
        (DATE '2022-08-20',DATE '2022-12-20','sin ninguna_
↪novedad','libre',213,'aceptado',18,10);
        """)
conexion.commit()

```

[]: *# Insercion de FACTURA_ENCABEZADO*

```

cursor.execute("""
        INSERT INTO rentacar.FACTURA_ENCABEZADO(
            fecha,
            subtotal,
            iva,
            total,
            codigo_reservacion
        )
        VALUES
        (DATE '2022-01-20',23,0.12,200,1),
        (DATE '2021-02-23',22,0.12,210,2),
        (DATE '2013-03-25',34,0.12,230,3),
        (DATE '2023-04-26',65,0.12,250,4),
        (DATE '2009-05-20',43,0.12,300,5),
        (DATE '2011-06-12',63,0.12,250,6),
        (DATE '2022-07-20',43,0.12,340,7),
        (DATE '2022-08-20',23,0.12,550,8),

        (DATE '2022-01-20',14,0.12,400,9),
        (DATE '2021-02-23',33,0.12,252,10),
        (DATE '2013-03-25',24,0.12,320,11),
        (DATE '2023-04-26',54,0.12,120,12),
        (DATE '2009-05-20',46,0.12,110,13),
        (DATE '2011-06-12',56,0.12,120,14),
        (DATE '2022-07-20',46,0.12,210,15),
        (DATE '2022-08-20',56,0.12,200,16),

        (DATE '2022-01-20',34,0.12,220,17),
        (DATE '2021-02-23',43,0.12,430,18),
        (DATE '2013-03-25',54,0.12,120,19),
        (DATE '2023-04-26',65,0.12,440,20),
        (DATE '2009-05-20',45,0.12,500,21),

```



```

        (DATE '2011-06-12',26,0.12,230,22),
        (DATE '2022-07-20',45,0.12,330,23),
        (DATE '2022-08-20',26,0.12,440,24),

        (DATE '2022-01-20',45,0.12,290,25),
        (DATE '2021-02-23',65,0.12,230,26),
        (DATE '2013-03-25',76,0.12,150,27),
        (DATE '2023-04-26',87,0.12,160,28),
        (DATE '2009-05-20',66,0.12,150,29),
        (DATE '2011-06-12',89,0.12,150,30),
        (DATE '2022-07-20',66,0.12,160,31),
        (DATE '2022-08-20',29,0.12,610,32);
    """
)

```

```

conexion.commit()

```

```

[ ]: # Insercion de FACTURA_DETALLE

```

```

cursor.execute("""
    INSERT INTO rentacar.FACTURA_DETALLE(
        cantidad,
        total_unitario,
        total,
        descripcion,
        codigo_factura_encabezado
    )
    VALUES
    (1,23,200,'alquiler vehiculo',1),
    (1,100,100,'alquiler gps',1),
    (1,22,210,'alquiler vehiculo',2),
    (1,50,50,'alquiler silla bebe',2),
    (1,34,230,'alquiler vehiculo',3),
    (1,50,50,'alquiler silla bebe',3),
    (1,65,250,'alquiler vehiculo',4),
    (1,100,100,'alquiler gps',4),
    (1,43,300,'alquiler vehiculo',5),
    (1,50,50,'alquiler silla bebe',5),
    (1,63,250,'alquiler vehiculo',6),
    (1,100,100,'alquiler gps',6),
    (1,43,340,'alquiler vehiculo',7),
    (1,23,550,'alquiler vehiculo',8),
    (1,100,100,'alquiler gps',8),

    (1,14,400,'alquiler vehiculo',9),
    (1,33,252,'alquiler vehiculo',10),
    (1,100,100,'alquiler gps',10),
    (1,24,320,'alquiler vehiculo',11),
    (1,50,50,'alquiler silla bebe',11),

```

```

(1,54,120,'alquiler vehiculo',12),
(1,46,110,'alquiler vehiculo',13),
(1,100,100,'alquiler gps',13),
(1,56,120,'alquiler vehiculo',14),
(1,50,50,'alquiler silla bebe',14),
(1,46,210,'alquiler vehiculo',15),
(1,100,100,'alquiler gps',15),
(1,56,200,'alquiler vehiculo',16),
(1,100,100,'alquiler gps',16),

(1,34,220,'alquiler vehiculo',17),
(1,43,430,'alquiler vehiculo',18),
(1,100,100,'alquiler gps',18),
(1,54,120,'alquiler vehiculo',19),
(1,50,50,'alquiler silla bebe',19),
(1,100,100,'alquiler gps',19),
(1,65,440,'alquiler vehiculo',20),
(1,50,50,'alquiler silla bebe',20),
(1,45,500,'alquiler vehiculo',21),
(1,26,230,'alquiler vehiculo',22),
(1,100,100,'alquiler gps',22),
(1,45,330,'alquiler vehiculo',23),
(1,50,50,'alquiler silla bebe',23),
(1,26,440,'alquiler vehiculo',24),
(1,100,100,'alquiler gps',24),

(1,45,290,'alquiler vehiculo',25),
(1,50,50,'alquiler silla bebe',25),
(1,65,230,'alquiler vehiculo',26),
(1,100,100,'alquiler gps',26),
(1,76,150,'alquiler vehiculo',27),
(1,87,160,'alquiler vehiculo',28),
(1,100,100,'alquiler gps',28),
(1,50,50,'alquiler silla bebe',28),
(1,66,150,'alquiler vehiculo',29),
(1,89,150,'alquiler vehiculo',30),
(1,100,100,'alquiler gps',30),
(1,66,160,'alquiler vehiculo',31),
(1,100,100,'alquiler gps',31),
(1,29,610,'alquiler vehiculo',32),
(1,50,50,'alquiler silla bebe',32);
"""

```

```

conexion.commit()

```

```

[ ]: # Cerrar conexion y cursor
      cursor.close()
      conexion.close()

```

5 Consultas a base de datos

5.1 Consultas multitabla (con JOIN)

- Mostrar los datos del cliente (codigo_cliente,nombre,apellido,cedula) y saber cuantas reservaciones tienen cada uno

Se espera visualizar

codigo_cliente	nombre	apellido	cedula	numero_alquiler
9	"Alejandro"	"Andrade"	"1002003008"	1
15	"Leo"	"Duarte"	"2002003005"	2
3	"Martín"	"Contreras"	"1002003002"	3
17	"Omar"	"Espinoza"	"2002003007"	1
5	"Pedro"	"Contreras"	"1002003004"	3
4	"Lucas"	"Díaz"	"1002003003"	4
6	"Mateo"	"Contreras"	"1002003005"	2
14	"Omar"	"Benítez"	"2002003004"	2
2	"Hugo"	"Benítez"	"1002003001"	4
16	"Pedro"	"Lopez"	"2002003006"	1
7	"Leo"	"Lopez"	"1002003006"	2
12	"Pedro"	"Álvarez"	"2002003002"	1
1	"Pedro"	"Lopez"	"1002003000"	3
18	"Pedro"	"Lopez"	"2002003008"	1
8	"Daniel"	"Álvarez"	"1002003007"	2

```
[ ]: SELECT c.codigo_cliente, c.nombre, c.apellido, c.cedula, COUNT(c.  
      ↪codigo_cliente) as numero_alquiler FROM rentacar.cliente as c  
RIGHT JOIN rentacar.reservacion as r ON c.codigo_cliente = r.codigo_cliente  
GROUP BY c.codigo_cliente;
```

- Mostrar los datos del vehiculo (codigo_vehiculo,marca,modelo,placa) y saber el numero de veces alquilado

Se espera visualizar

codigo_vehiculo	marca	modelo	placa	numero_alquiler
9	"Kia Rio"	"S"	"ret456"	1
3	"Honda Civic"	"HEV"	"poi456"	6
5	"Kia Rio"	"CT"	"mnb456"	4
4	"Ford Fiesta"	"IS"	"lkj432"	5
10	"Lexus ES"	"X"	"dfe284"	3
6	"Toyota Corolla"	"LC"	"ghj576"	3
2	"Toyota Corolla"	"ES"	"qwe234"	3
7	"Ford Fiesta"	"RX"	"sdf970"	2
1	"Ford Fiesta"	"LS"	"pbi1234"	4
8	"Honda Civic"	"GT"	"xcv234"	1

```
[ ]: SELECT v.codigo_vehiculo, v.marca, v.modelo, v.placa, COUNT(v.codigo_vehiculo)
      ↪as numero_alquiler FROM rentacar.vehiculo as v
RIGHT JOIN rentacar.reservacion as r ON v.codigo_vehiculo = r.codigo_vehiculo
GROUP BY v.codigo_vehiculo;
```

- Mostrar los datos del vehiculo (codigo_vehiculo,marca,modelo,placa) y saber el numero de mantenimientos

Se espera visualizar

codigo_vehiculo	marca	modelo	placa	numero_mantenimiento
4	"Ford Fiesta"	"IS"	"lkj432"	8
6	"Toyota Corolla"	"LC"	"ghj576"	7
2	"Toyota Corolla"	"ES"	"qwe234"	7
9	"Kia Rio"	"S"	"ret456"	8
7	"Ford Fiesta"	"RX"	"sdf970"	6
3	"Honda Civic"	"HEV"	"poi456"	7
1	"Ford Fiesta"	"LS"	"pbi1234"	8
5	"Kia Rio"	"CT"	"mnb456"	8
8	"Honda Civic"	"GT"	"xcv234"	7

```
[ ]: SELECT v.codigo_vehiculo, v.marca, v.modelo, v.placa, COUNT(v.codigo_vehiculo)
      ↪as numero_mantenimiento FROM rentacar.vehiculo as v
RIGHT JOIN rentacar.ficha_mantenimiento as fm ON v.codigo_vehiculo = fm.
      ↪codigo_vehiculo
GROUP BY v.codigo_vehiculo;
```

5.2 Consultas con función de agregación (operaciones sobre conjuntos de filas)

- Obtener el precio mas alto agrupados por el color del vehiculo, y ordenar de mayor a menor precio

Se espera visualizar

color	costo_diario
"verde"	102
"blanco"	76
"negro"	65
"verde"	65
"amarillo"	64
"negro"	54
"rojo"	53
"verde"	43
"negro"	22
"rojo"	12

```
[ ]: SELECT v.color, MAX(v.costo_diario) as costo_diario FROM rentacar.vehiculo AS v
GROUP BY v.color, v.costo_diario
ORDER BY v.costo_diario DESC;
```

- Mostrar los datos del vehículo (codigo_vehiculo,marca,modelo,placa) y mostrar el costo total recaudado por cada vehículo alquilado

Se espera visualizar

codigo_vehiculo	marca	modelo	placa	costo_recaudado
9	"Kia Rio"	"S"	"ret456"	65
3	"Honda Civic"	"HEV"	"poi456"	324
5	"Kia Rio"	"CT"	"mnb456"	408
4	"Ford Fiesta"	"IS"	"lkj432"	325
10	"Lexus ES"	"X"	"dfe284"	228
6	"Toyota Corolla"	"LC"	"ghj576"	192
2	"Toyota Corolla"	"ES"	"qwe234"	129
7	"Ford Fiesta"	"RX"	"sdf970"	106
1	"Ford Fiesta"	"LS"	"pbi1234"	48
8	"Honda Civic"	"GT"	"xcv234"	22

```
[ ]: SELECT v.codigo_vehiculo, v.marca, v.modelo, v.placa, SUM(v.costo_diario) as
↳ costo_recaudado FROM rentacar.vehiculo as v
RIGHT JOIN rentacar.reservacion as r ON v.codigo_vehiculo = r.codigo_vehiculo
GROUP BY v.codigo_vehiculo;
```

6 Conclusiones

- Se necesita tener claro la necesidad del cliente para poder realizar un buen diseño de la base de datos.
- No es necesario realizar todas las formas normales para el diseño de la base de datos.
- Si surge alguna modificación en el diseño físico, se tiene que volver al diseño lógico y conceptual a corregir.

[]: