

Entrega Proyecto 2 CODERHOUSE

Ventas Autos Usados Emiratos Árabes Unidos

Autor: Miguel Angel Di Rocco

Objetivo del Proyecto:

El objetivo de este proyecto es crear una base de datos relacional para gestionar información sobre las ventas de autos usados en Emiratos Árabes Unidos. El sistema permitirá almacenar y analizar datos como la marca, modelo, año, precio, kilometraje, tipo de carrocería, cilindrada, transmisión, tipo de combustible, color, ubicación y condición del vehículo. Esta base de datos servirá como base para análisis de tendencias de ventas, identificación de patrones de compra y predicción de precios.

Alcance:

Este proyecto tiene como objetivo el diseño e implementación de una base de datos relacionales que permita almacenar y gestionar información detallada sobre las ventas de autos usados en Emiratos Árabes Unidos. El sistema está enfocado en facilitar la recopilación, análisis y consulta de datos clave, como las características de los vehículos (marca, modelo, año, kilometraje, tipo de carrocería, etc.) y las transacciones de venta (precio, ubicación, etc.)

La base de datos permitirá realizar los siguientes análisis:

- Identificación de tendencias en el mercado de autos usados, calculando en atributos como el precio y el año de fabricación.
- Evaluación de la demanda de diferentes tipos de vehículos en diversas ubicaciones.
- Análisis de la relación entre el kilometraje de los vehículos y su precio de venta.
- Generación de informes sobre el rendimiento de ventas en función de la ubicación, el tipo de combustible, la condición del vehículo, entre otros.

Usuario Final y Nivel de Aplicación del Análisis:

Usuario final: Analistas de datos, responsables de ventas y gerentes de vehículos usados.

Nivel de Aplicación del Análisis:

1. Operativo:

- Análisis de datos de ventas individuales, comparaciones de precios y seguimiento de ventas: El dataset contiene información sobre vehículos vendidos, incluyendo su precio, tipo de transmisión, condición del vehículo y más. Esto permite realizar un análisis operativo donde puedes:
 - Rastrear cada venta individual.
 - Comparar precios según características como el modelo, el tipo de combustible, el kilometraje, etc.

2. Táctico:

- Identificación de patrones de venta y evaluación de la demanda en diferentes ubicaciones: El dataset El conjunto de datos incluye columnas de ubicación y tipo de vehículo, lo que te permite identificar patrones de venta en función de la región geográfica o las características de los vehículos más vendidos.

3. Estratégico:

- Toma de decisiones sobre precios y oferta de autos en función de las tendencias del mercado:

Listado de Tablas con Descripción:

• Tabla	• Descripción
• Vehicle	• Tabla principal. Contiene información general del vehículo.
• Model	• Lista de modelos. Cada modelo pertenece a una marca (Make).
• Make	• Marcas de autos (ej. Toyota, Ford).
• BodyType	• Tipo de carrocería (SUV, Sedán, etc.).
• Transmission	• Tipo de transmisión (Automática, Manual).
• Color	• Colores de los vehículos.
• FuelType	• Tipo de combustible (Gasoline, Diesel, Electric).
• Cylinders	• Cantidad de cilindros del motor.
• Location	• Ubicación del vehículo.
• VehicleCondition	• Condición del vehículo.
•	•

Identificación de claves primarias y foráneas:

Tabla Vehicle: PK: VehicleID FK: ModelID → Tabla Model BodyTypeID → Tabla BodyType CylindersID → Tabla Cylinders TransmissionID → Tabla Transmission FuelTypeID → Tabla FuelType ColorID → Tabla Color LocationID → Tabla Location ConditionID → Tabla Condition	Tabla Model: PK: ModelID FK: MakeID→ Tabla Make Tabla Make: PK: MakeID Tabla BodyType: PK: BodyTypeID	Tabla Color: PK: ColorID Tabla Cylinders: PK: CylindersID Tabla Condition: PK: ConditionID Tabla Transmission: PK: TransmissionID	Tabla FuelType: PK: FuelTypeID Tabla Location: PK: LocationID
---	--	--	--

Cada relación entre tablas sigue las reglas de integridad referencial, donde las claves foráneas aseguran que los datos en una tabla tengan correspondencia válida en otra.

Columnas de cada tabla:

Vehicle: VehicleID (PK), ModelID (FK), Year, Price, Mileage, BodyTypeID (FK), CylindersID (FK), TransmissionID (FK), FuelTypeID (FK), ColorID (FK), LocationID (FK), ConditionID (FK)
Model: ModelID (PK), ModelName, MakeID (FK)
Make: MakeID (PK), MakeName
BodyType: BodyTypeID (PK), BodyTypeName
Color: ColorID (PK), ColorName
Cylinders: CylindersID (PK), CylindersCount
Transmission: TransmissionID (PK), TransmissionType
FuelType: FuelTypeID (PK), FuelTypeName
Condition: ConditionID (PK), ConditionName
Location: LocationID (PK), LocationName

Tipo de datos sugeridos:

IDs: INT
Year: YEAR o INT
Price: DECIMAL(10,2)
Mileage: INT
Nombres y descripciones: VARCHAR(50) a VARCHAR(100)

DESCRIPCION DE RELACION ENTRE TABLAS:

Vehicle ↔ Model:
Relación: Muchos a Uno (N:1).
Cada vehículo pertenece a un modelo específico, y esto se vincula mediante ModelID como llave foránea en la tabla Vehicle.

Model ↔ Make:
Relación: Muchos a Uno (N:1).
Cada modelo pertenece a una marca específica, y esto se establece mediante MakeID como llave foránea en la tabla Model.

Vehicle ↔ BodyType:
Relación: Muchos a Uno (N:1).
Cada vehículo tiene un tipo de carrocería definido por BodyTypeID como llave foránea en la tabla Vehicle.

Vehicle ↔ Cylinders:
Relación: Muchos a Uno (N:1).
Cada vehículo tiene un número de cilindros especificado por CylindersID como llave foránea en Vehicle.

Vehicle ↔ Transmission:
Relación: Muchos a Uno (N:1).
El tipo de transmisión de cada vehículo se define mediante TransmissionID como llave foránea.

Vehicle ↔ FuelType:
Relación: Muchos a Uno (N:1).
Cada vehículo utiliza un tipo de combustible identificado por FuelTypeID como llave foránea.

Vehicle ↔ Color:
Relación: Muchos a Uno (N:1).
El color del vehículo está definido por ColorID como llave foránea

Vehicle ↔ Condition:
Relación: Muchos a Uno (N:1).
La condición de cada vehículo (nuevo, usado, etc.) está especificada por ConditionID.

Vehicle ↔ Location:
Relación: Muchos a Uno (N:1).
La ubicación de venta o almacenamiento del vehículo se define por LocationID como llave foránea.

Tablas independientes:
BodyType, Color, Cylinders, Condition, Transmission, FuelType, Location, y Make no tienen relaciones adicionales más allá de servir como referencia para las llaves foráneas mencionadas.

Estas relaciones definen una estructura clara y escalable para la base de datos, asegurando integridad referencial.

No son claves primarias ni foráneas:
Price, Mileage y Year no actúan como identificadores únicos (claves primarias) que podrían ser referenciados en otras tablas, ni representan relaciones con otras entidades. Solo describen características específicas de un vehículo.

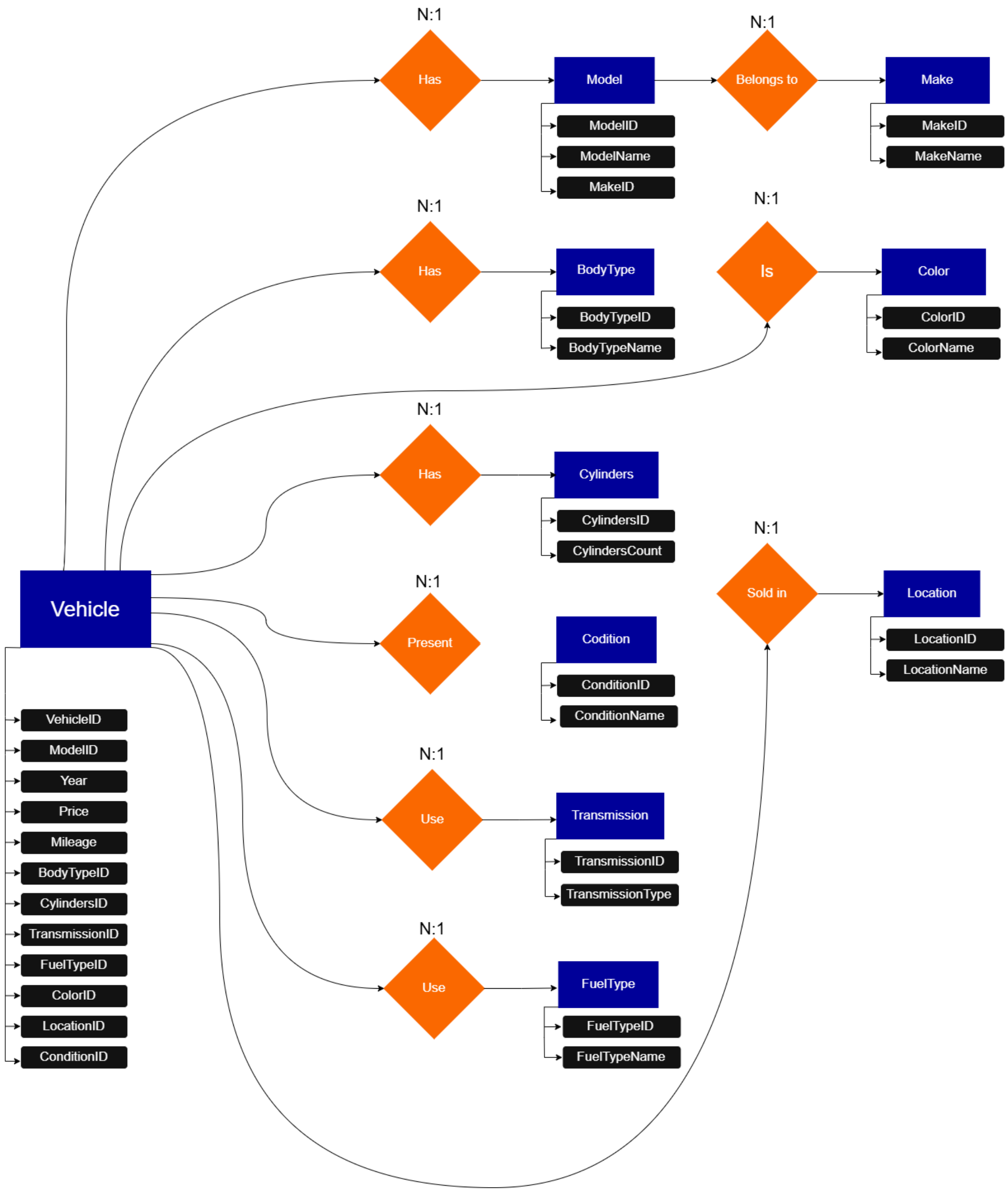
Atributos dependientes de la entidad:
Estas columnas son atributos intrínsecos de la tabla **Vehicle**. Por ejemplo, **Price** representa el costo de un vehículo, **Mileage** indica el kilometraje, y **Year** refleja el año de fabricación. Son datos independientes que no requieren relación con otras tablas.

No hay necesidad de normalización:
Normalizar estos atributos en tablas separadas no aporta ventajas, ya que no son repetitivos o compartidos entre distintas entidades. Su lugar natural es dentro de la tabla **Vehicle**, donde aportan contexto sin generar redundancia.

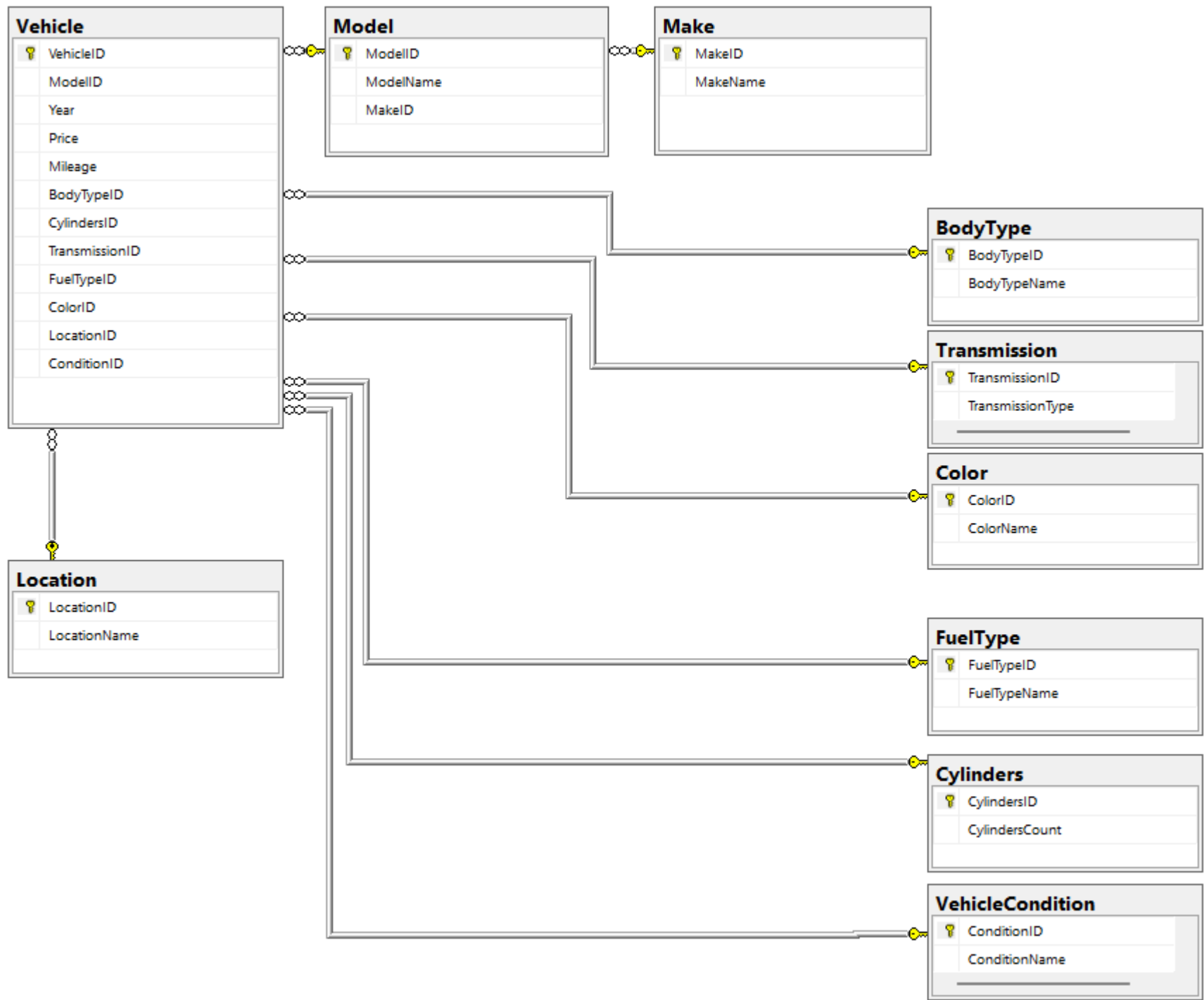
Propósito descriptivo:
Estas columnas no tienen un propósito de conexión, sino que enriquecen la descripción del vehículo. Como tales, no necesitan referencias externas.

Si algún día se requiere incorporar análisis relacionados con estos atributos (como rangos de precios, segmentación por año, etc.), esto se haría directamente desde la tabla **Vehicle** sin necesidad de relaciones adicionales.

A CONTINUACION SE ADJUNTA EL DIAGRAMA ENTIDAD RELACION REALIZADO:



TAMBIEN ADJUNTO EL DIAGRAMA REALIZADO EN SQL SERVER A MODO DE PRACTICA:



ESTA ES MI BASE DE DATOS PARA LA ENTREGA 2 CON SUS TABLAS:

- Entrega2
 - Database Diagrams
 - dbo.Diagram_0
 - Tables
 - System Tables
 - FileTables
 - External Tables
 - Graph Tables
 - dbo.BodyType
 - dbo.Color
 - dbo.Cylinders
 - dbo.FuelType
 - dbo.Location
 - dbo.Make
 - dbo.Model
 - dbo.Transmission
 - dbo.Vehicle
 - dbo.VehicleCondition