Case Study-Designing an EV Database on MySQL

Prepared by:

Pratik Chhimpi

Instructor: Dr. Junaid Qazi

Data Science - DS 037-009 – Assignment

Southern Alberta Institute of Technology

June 04, 2025

Table of Contents

Introduction:

The project outlines the design and implementation of a database system for managing EV sales, and dealership operations for a hypothetical organization.

Organization Selection:

Organization: Tee – Electrified Vehicles Division (Hypothetical)

Domain: Automotive sales and infrastructure for electric vehicles (including BEVs, PHEVs, and Hybrids).

Purpose and Goals:

Purpose: To create a database system that tracks sales, inventory, employees, and dealership data for Tee's EV business.

Goals:

- Sales Analysis: Capture and analyze sales transactions and trends over any time-period.
- Dealership Performance: Monitor and evaluate dealership performance and regional trends.
- Employee & Marketing Insights: Track sales representatives and buying trend to support targeted.

Entity Identification

In the proposed star schema, we used one fact table and four-dimension tables:

Dimension Tables:

`dim_vehicle` - Electric vehicle details.

`dim_dealership` – Dealership data.

`dim_time` – Date, month, quarter, year, holiday status.

`dim_employee` – Sales representative information.

Fact Table:

`fact_sales` – Central sales table referencing all dimensions.

Attribute Specification

Table: dim_vehicle

Field	Type	Constraints
vehicle_id	INT	PRIMARY KEY
model_name	VARCHAR (50)	NOT NULL
vehicle_type	ENUM ('BEV', 'PHEV', 'Hybrid')	NOT NULL
battery_capacity	INT	NOT NULL, CHECK (battery_capacity > 0)
range_miles	INT	NOT NULL
base_price	DECIMAL(10,2)	NOT NULL
launch_year	YEAR	NOT NULL
drive_type	VARCHAR (20)	NOT NULL
charge_time	DECIMAL (4,2)	(Hours for a full charge; NULL allowed if not applicable)
colour_variety	VARCHAR (100)	Each Available colour as a separate record
VIN_number	VARCHAR(17)	NOT NULL
Engine_number	VARCHAR(17)	NOT NULL

The dim_vehicle table serves as a foundational dimension within the EV database, capturing comprehensive details about each electric vehicle model offered by Tee's Electrified Vehicles Division. Its design is both logically structured and aligned with best practices for data integrity and analytical use.

Field Descriptions and Constraints:

- **vehicle_id (INT, PRIMARY KEY)**: A unique identifier for each vehicle record, ensuring entity-level uniqueness and supporting efficient indexing.
- **model_name (VARCHAR(50), NOT NULL):** Records the official name of the vehicle model. The NOT NULL constraint ensures that all entries have an identifiable name for traceability.
- **vehicle_type (ENUM: 'BEV', 'PHEV', 'Hybrid'):** Defines the propulsion system type. Using an ENUM constraint limits values to a controlled set, reducing data entry errors and enhancing filtering capabilities during analysis
- **battery_capacity (INT, NOT NULL, CHECK > 0):** Specifies the battery capacity in kWh. The CHECK constraint enforces logical data entry by disallowing zero or negative values.
- **range_miles (INT, NOT NULL):** Indicates the driving range on a full charge. Critical for performance benchmarking and customer comparisons.
- **base_price (DECIMAL(10,2), NOT NULL):** Stores the starting price of the vehicle, formatted for precision in currency. This field supports financial reporting and pricing strategies.
- **launch_year (YEAR, NOT NULL):** Captures the market release year of the model, useful for trend analysis and lifecycle tracking.
- **drive_type (VARCHAR(20), NOT NULL):** Describes the vehicle's drivetrain (e.g., FWD, RWD, AWD), essential for understanding performance attributes.

- **charge_time (DECIMAL(4,2), NULLABLE):** Represents the time (in hours) required for a full battery charge. This field allows NULL for models where this is not applicable, such as hybrids.
- **colour_variety (VARCHAR(100)):** Reflects available color options. This design assumes each color is stored as a separate record or string (though normalization into a separate color dimension may be advisable for scalability).
- **VIN_number (VARCHAR(17), NOT NULL):** The unique Vehicle Identification Number, adhering to industry standards for traceability and regulatory compliance.
- **Engine_number (VARCHAR(17), NOT NULL):** A unique identifier for the vehicle's engine, further supporting traceability and manufacturing controls.

Table: dim_dealership

Field	Type	Constraints
dealership_id	INT	PRIMARY KEY
dealership_name	VARCHAR (100)	NOT NULL
contact_email	VARCHAR (100)	UNIQUE
Phone	INT	NOT NULL
city	VARCHAR (50)	NOT NULL
province	VARCHAR (2)	NOT NULL
region	VARCHAR (20)	NOT NULL
has_fast_charging	BIT	DEFAULT FALSE
total_ev_inventory	INT	NOT NULL
display_capacity	INT	NOT NULL

The dim_dealership table is designed to store key attributes of each dealership within Tee's Electrified Vehicles Division network. This dimension supports strategic decisions related to dealership performance, geographic distribution, infrastructure capabilities, and inventory management.

Field Descriptions and Constraints:

- **dealership_id (INT, PRIMARY KEY):** A unique identifier for each dealership. This primary key ensures the integrity of records and facilitates efficient joins with fact tables.
- **dealership_name (VARCHAR(100), NOT NULL):** Captures the full name of the dealership. The NOT NULL constraint ensures every entry is associated with a known entity, essential for accurate reporting and customer interaction.
- **contact_email (VARCHAR(100), UNIQUE):** Stores a unique email address for dealership communication. The UNIQUE constraint prevents duplication and helps maintain reliable contact data.
- **Phone (INT, NOT NULL):** Holds the dealership's phone number. While stored as an integer, best practice may suggest using VARCHAR to preserve formatting (e.g., dashes or parentheses).
- **city (VARCHAR(50), NOT NULL):** Indicates the city location of the dealership, useful for regional market analysis and customer segmentation.

- **province (VARCHAR(2), NOT NULL):** Stores the two-character province or state abbreviation (e.g., "AB" for Alberta), standardizing geographic reporting.
- **region (VARCHAR(20), NOT NULL):** Identifies the broader operational or sales region (e.g., West, Central), which can aid in territory-based performance assessments.
- **has_fast_charging (BIT, DEFAULT FALSE):** A Boolean flag indicating whether the dealership is equipped with fast-charging infrastructure. This is useful for service capability assessments and customer experience enhancements.
- **total_ev_inventory (INT, NOT NULL):** Represents the total number of electric vehicles currently held in stock at the dealership. This field is critical for demand planning and inventory turnover analysis.
- **display_capacity (INT, NOT NULL):** Reflects the physical or showroom capacity available for displaying EV models, relevant to merchandising and layout planning.

Table: dim_time

Field	Type	Constraints
time_id	INT	PRIMARY KEY
sale_date	DATE	NOT NULL
month	INT	NOT NULL, CHECK (month BETWEEN 1 AND 12)
quarter	VARCHAR (5)	NOT NULL
year	INT	NOT NULL
is_holiday	BIT	DEFAULT FALSE

The dim_time table functions as a core time dimension in the EV data warehouse, enabling temporal analysis across sales, inventory, and performance metrics. This dimension supports slicing and filtering data by date, month, quarter, year, and holiday status, which are essential for trend analysis and time-based reporting.

Field Descriptions and Constraints:

- **time_id (INT, PRIMARY KEY):** A unique identifier for each time entry, typically mapped to individual dates. It serves as the primary key and facilitates efficient joins with fact tables for time-based analysis.
- **sale_date (DATE, NOT NULL):** Stores the actual calendar date of each transaction or record. This field is essential for daily-level tracking and aligns with standard SQL date operations.
- **month (INT, NOT NULL, CHECK BETWEEN 1 AND 12):** Represents the numeric month of the year (1–12). The CHECK constraint ensures valid values, preventing data quality issues in time-based aggregations.

- **quarter (VARCHAR(5), NOT NULL):** Indicates the fiscal or calendar quarter (e.g., "Q1", "Q2"). This field allows for grouping and comparison across quarterly performance metrics.
- **year (INT, NOT NULL):** Captures the year associated with each record, supporting historical analysis and year-over-year trend tracking.
- **is_holiday (BIT, DEFAULT FALSE):** A Boolean flag indicating whether the given date is a holiday. This is useful for analyzing the impact of holidays on sales and operational performance.

Table: dim_employee

Field	Type	Constraints
employee_id	INT	PRIMARY KEY, AUTO_INCREMENT
first_name	VARCHAR (50)	NOT NULL
last_name	VARCHAR (50)	NOT NULL
position	VARCHAR (50)	DEFAULT 'Sales Associate'
dealership_id	INT	FOREIGN KEY REFERENCES dim_dealership(dealership_id)

The dim_employee table is designed to capture essential information about staff members employed within Tee’s Electrified Vehicles Division. It establishes a clear connection between employees and the dealerships they are assigned to, supporting both operational and analytical needs such as workforce distribution, performance tracking, and organizational structure.

Field Descriptions and Constraints:

- **employee_id (INT, PRIMARY KEY, AUTO_INCREMENT):** A unique, system-generated identifier for each employee. The AUTO_INCREMENT feature ensures seamless record creation while maintaining uniqueness.
- **first_name (VARCHAR(50), NOT NULL):** Stores the employee’s first name. The NOT NULL constraint ensures that this critical identifier is always present for communication and recordkeeping.
- **last_name (VARCHAR(50), NOT NULL):** Captures the employee’s last name. Combined with the first name, this enables proper identification and reporting.
- **position (VARCHAR(50), DEFAULT 'Sales Associate'):** Indicates the employee’s job title or role within the dealership. The default value assumes the most common position, simplifying data entry while allowing flexibility for other roles (e.g., Manager, Technician).
- **dealership_id (INT, FOREIGN KEY REFERENCES dim_dealership(dealership_id)):** Establishes a relational link to the dim_dealership table, associating each employee with a specific dealership. This foreign key ensures referential integrity and supports location-based workforce analysis.

Table: fact_sales

Field	Type	Constraints
sale_id	INT	PRIMARY KEY, AUTO_INCREMENT
customer_id	INT	FOREIGN KEY REFERENCES dim_customer(customer_id)
vehicle_id	INT	FOREIGN KEY REFERENCES dim_vehicle(vehicle_id)
dealership_id	INT	FOREIGN KEY REFERENCES dim_dealership(dealership_id)
Time_period_id	INT	FOREIGN KEY REFERENCES dim_time(time_id)
employee_id	INT	(OPTIONAL—if tracking which sales rep made the sale) FOREIGN KEY REFERENCES dim_employee(employee_id)
sale_price	DECIMAL (10,2)	NOT NULL, CHECK (sale_price > 0)
tax_amount	DECIMAL (10,2)	NOT NULL
financing_used	BIT	(Indicates whether the customer used Tee financing)

The fact_sales table is the central fact table in Tee's Electrified Vehicles Division data warehouse. It captures each individual sales transaction and serves as the core for quantitative analysis of vehicle sales, revenue, dealership performance, customer behavior, and employee contribution. This table connects various dimension tables to form a fully functional star schema, supporting comprehensive business intelligence and reporting capabilities.

Field Descriptions and Constraints:

- **sale_id (INT, PRIMARY KEY, AUTO_INCREMENT):** A unique, system-generated identifier for each sale. It acts as the primary key and ensures the uniqueness of each transaction record.
- **customer_id (INT, FOREIGN KEY REFERENCES dim_customer(customer_id)):** Links the transaction to a specific customer, enabling customer-centric analysis such as loyalty tracking and segmentation.
- **vehicle_id (INT, FOREIGN KEY REFERENCES dim_vehicle(vehicle_id)):** Associates the sale with a specific electric vehicle. This relationship supports product-level sales tracking and inventory planning.
- **dealership_id (INT, FOREIGN KEY REFERENCES dim_dealership(dealership_id)):** Connects the sale to the dealership where it occurred. This is essential for evaluating dealership performance and regional sales trends.
- **Time_period_id (INT, FOREIGN KEY REFERENCES dim_time(time_id)):** Establishes the time context for the sale, supporting time-series analysis (e.g., sales by month, quarter, or year).
- **employee_id (INT, OPTIONAL, FOREIGN KEY REFERENCES dim_employee(employee_id)):** Optionally tracks which sales representative completed the transaction. This enables employee-level performance reporting and incentive tracking.

- **sale_price (DECIMAL(10,2), NOT NULL, CHECK > 0):** Records the final sale price of the vehicle. The CHECK constraint ensures only positive values are accepted, supporting revenue and profitability analysis.
- **tax_amount (DECIMAL(10,2), NOT NULL):** Captures the tax collected on the sale. This is important for financial reporting and compliance with regulatory requirements.
- **financing_used (BIT):** A Boolean indicator showing whether the customer utilized Tee's internal financing options. This field supports credit usage analysis and financing program effectiveness.

Primary and Foreign Keys

Table Name	Primary Key
dim_vehicle	vehicle_id
dim_dealership	dealership_id
dim_time	time_id
dim_employee	employee_id
fact_sales	sale_id

Each dimension table (dim_vehicle, dim_dealership, dim_time, dim_employee) contains a uniquely identifying primary key that is referenced in the fact_sales table through foreign key relationships. This design facilitates robust analytical capabilities and ensures data consistency across the system.

Foreign Key in fact_sales	Reference Table	References Column
vehicle_id	dim_vehicle	vehicle_id
dealership_id	dim_dealership	dealership_id
time_id	dim_time	time_id
employee_id	dim_employee	employee_id
Foreign Key in fact_sales	Reference Table	References Column

These foreign key relationships reinforce the star schema structure by linking the fact_sales table to key dimension tables. This design ensures data consistency, enables complex join operations, and supports multidimensional analytics across vehicles, dealerships, employees, time periods, and customers.

Establishing Entity Relationships

In our star schema design, the fact_sales table sits at the center and links directly to all

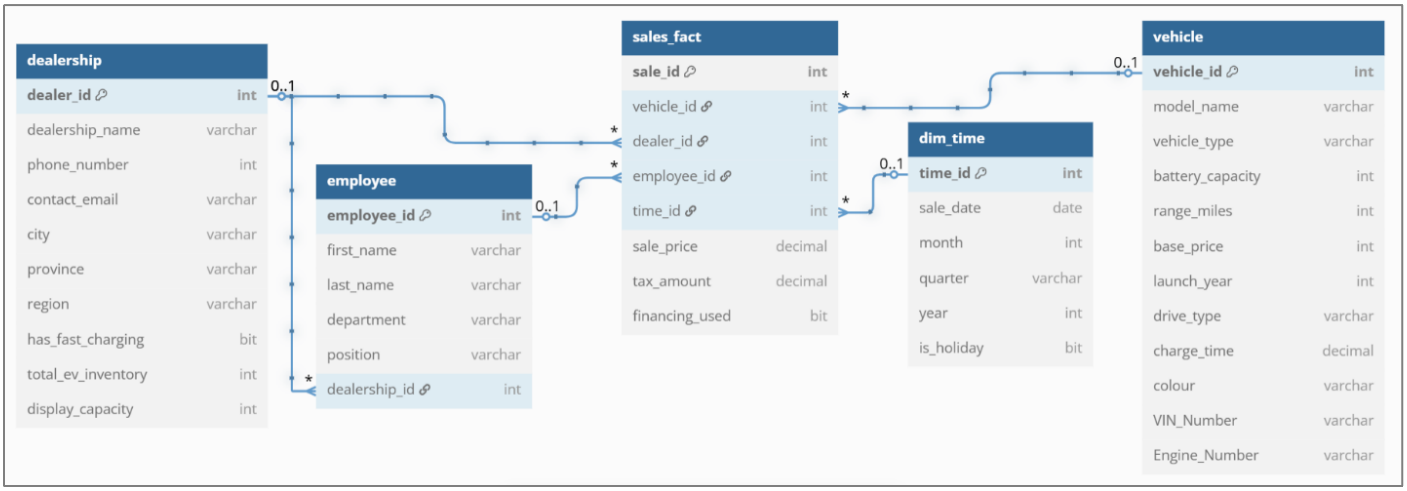
dimension tables. The relationships are one-to-many from each dimension to the fact table:

- **dim_vehicle → fact_sales:** (One to many) - Each vehicle may be associated with many sales.
- **dim_dealership → fact_sales**: (One to many) - A dealership appears in multiple sales.
- **dim_time_period → fact_sales**: (One to many) - Every sale is associated with a specific sale date (and related time attributes).
- **dim_employee → fact_sales:** (One to many) - Each sales representative may be involved in numerous sales.

Data Integrity Constraints

- Unique and non-null constraints on critical fields.
- ENUM and CHECK constraints on type, range, and price.
- Foreign key references to maintain relational consistency.

Entity-Relationship Diagram



dealership	
dealer_id	int
dealership_name	varchar
phone_number	int
contact_email	varchar
city	varchar
province	varchar
region	varchar
has_fast_charging	bit
total_ev_inventory	int
display_capacity	int

employee	
employee_id	int
first_name	varchar
last_name	varchar
department	varchar
position	varchar
dealership_id	int

sales_fact	
sale_id	int
vehicle_id	int
dealer_id	int
employee_id	int
time_id	int
sale_price	decimal
tax_amount	decimal
financing_used	bit

dim_time	
time_id	int
sale_date	date
month	int
quarter	varchar
year	int
is_holiday	bit

vehicle	
vehicle_id	int
model_name	varchar
vehicle_type	varchar
battery_capacity	int
range_miles	int
base_price	int
launch_year	int
drive_type	varchar
charge_time	decimal
colour	varchar
VIN_Number	varchar
Engine_Number	varchar

This ERD effectively visualizes a **star schema** suitable for business intelligence and reporting. The design supports:

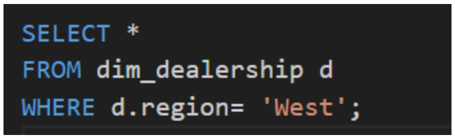
- Fast querying for metrics like total sales, average sale price, etc.
- Filtering and grouping across vehicle specs, dealership locations, employee performance, and time periods
- Optional employee tracking for sales, enabling flexible analysis

SQL Statements

MySQL syntax is used to define all five tables with proper constraints and key definitions. (Refer to the full SQL code section.)

Sample Queries and Views

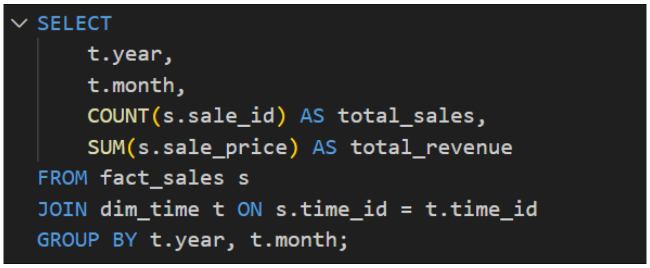
- **CSELECT with Filter**



```
SELECT *
FROM dim_dealership d
WHERE d.region= 'West';
```

The objective of this SQL query is to retrieve all records from the dim_dealership dimension table where the dealership is located in the 'West' region.

- **SELECT with 1 JOIN and GROUP BY**



```
SELECT
    t.year,
    t.month,
    COUNT(s.sale_id) AS total_sales,
    SUM(s.sale_price) AS total_revenue
FROM fact_sales s
JOIN dim_time t ON s.time_id = t.time_id
GROUP BY t.year, t.month;
```

This view is particularly useful for:

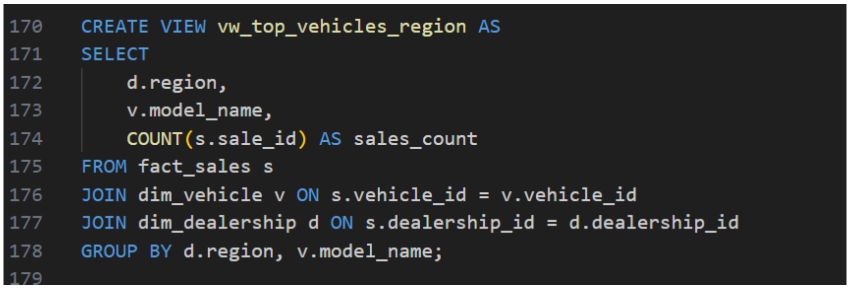
- Monthly sales performance dashboards
- Revenue forecasting and trend analysis
- Executive-level reporting on EV market performance
- Identifying seasonal patterns in EV sales

- SELECT with 2 JOINS and GROUP BY

The view vw_top_vehicles_region provides a summary of vehicle sales by model across different geographic regions.

This view is valuable for:

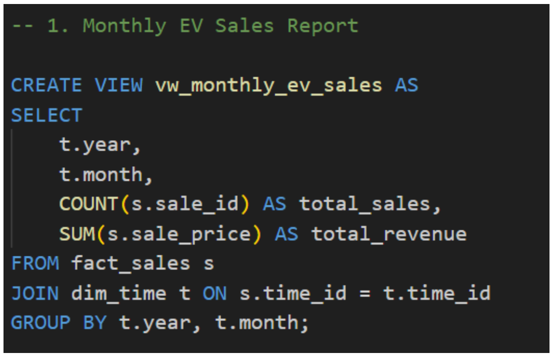
- Identifying top-selling models by region
- Regional marketing strategy and inventory planning
- Dealer performance comparisons
- Supporting visual dashboards for sales distribution analysis



```sql
CREATE VIEW vw_top_vehicles_region AS
SELECT
    d.region,
    v.model_name,
    COUNT(s.sale_id) AS sales_count
FROM fact_sales s
JOIN dim_vehicle v ON s.vehicle_id = v.vehicle_id
JOIN dim_dealership d ON s.dealership_id = d.dealership_id
GROUP BY d.region, v.model_name;
```

Analytical views

- ### Monthly EV Sales



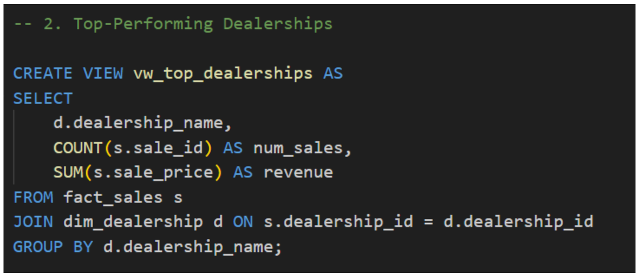
```sql
-- 1. Monthly EV Sales Report

CREATE VIEW vw_monthly_ev_sales AS
SELECT
    t.year,
    t.month,
    COUNT(s.sale_id) AS total_sales,
    SUM(s.sale_price) AS total_revenue
FROM fact_sales s
JOIN dim_time t ON s.time_id = t.time_id
GROUP BY t.year, t.month;
```

This view, vw_monthly_ev_sales, is designed to track and summarize electric vehicle (EV) sales performance on a **monthly basis**. It provides two key performance metrics for each month: the total number of EVs sold and the total revenue generated from those sales.

- Monthly sales reporting and executive summaries
- Revenue trend analysis and forecasting
- Identifying seasonal sales patterns
- Dashboard integration for performance tracking
- Evaluating the impact of promotions or incentives by month

- Top Performing Dealerships



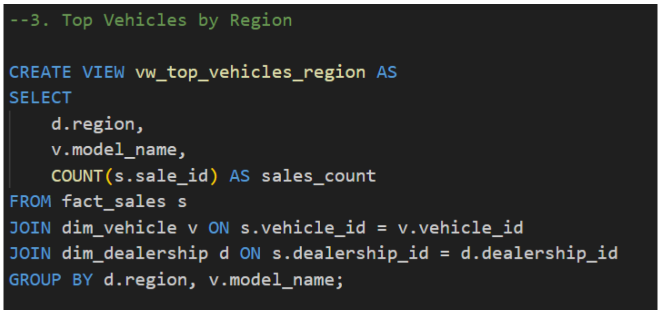
```
-- 2. Top-Performing Dealerships

CREATE VIEW vw_top_dealerships AS
SELECT
    d.dealership_name,
    COUNT(s.sale_id) AS num_sales,
    SUM(s.sale_price) AS revenue
FROM fact_sales s
JOIN dim_dealership d ON s.dealership_id = d.dealership_id
GROUP BY d.dealership_name;
```

This view, vw_top_dealerships, is designed to provide a performance summary for each dealership based on sales activity. It displays how many sales were made and the total revenue generated by each dealership, enabling direct comparison of dealership performance across the network.

- Ranking top-performing dealerships by volume or revenue
- Supporting incentive programs and performance bonuses
- Regional or corporate sales analysis
- Identifying underperforming dealerships for strategic support
- Integrating into executive dashboards or BI reports

- Regional vehicle trends



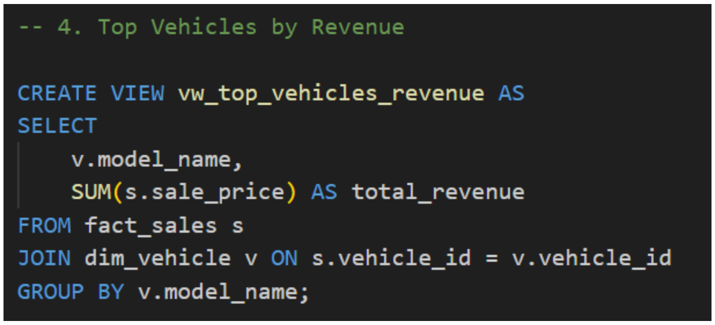
```
--3. Top Vehicles by Region

CREATE VIEW vw_top_vehicles_region AS
SELECT
    d.region,
    v.model_name,
    COUNT(s.sale_id) AS sales_count
FROM fact_sales s
JOIN dim_vehicle v ON s.vehicle_id = v.vehicle_id
JOIN dim_dealership d ON s.dealership_id = d.dealership_id
GROUP BY d.region, v.model_name;
```

The vw_top_vehicles_region view provides a summary of vehicle sales across different geographic regions. It groups sales by region and vehicle model, giving insight into which vehicle models perform best in which areas.

- Identifying regional vehicle model preferences
- Supporting targeted marketing or regional inventory planning
- Benchmarking model performance across regions
- Data input for regional sales dashboards and visualizations

- Revenue by vehicle



```
-- 4. Top Vehicles by Revenue

CREATE VIEW vw_top_vehicles_revenue AS
SELECT
    v.model_name,
    SUM(s.sale_price) AS total_revenue
FROM fact_sales s
JOIN dim_vehicle v ON s.vehicle_id = v.vehicle_id
GROUP BY v.model_name;
```

The view vw_top_vehicles_revenue is created to identify the **highest-earning vehicle models** based on total sales revenue. It helps decision-makers understand which models contribute most to overall sales income, regardless of how many units were sold.

- Ranking vehicle models by financial performance
- Supporting pricing and product strategy decisions
- Evaluating profitability and market demand
- Sales dashboard integration for model-level revenue analysis

Conclusion:

The finalized database design for Tee's Electrified Vehicles Division successfully fulfills the project's objectives by offering a well-structured and analytics-ready solution. With clearly defined entities and relationships, robust data integrity constraints, and a star schema tailored for efficient reporting, the system is well-positioned to support key business functions. The inclusion of sample SQL queries, view definitions, and a representative dataset further demonstrates the design's practical utility. Overall, this comprehensive approach ensures reliable support for EV sales tracking, customer insights, and dealership performance evaluation.