Stage IV – Elaboration: Database Design

Adeena Ahmed, Hafsah Shaik, Justin Wain, Christopher Eng, David Orpen, Ryan Arnold, Philip Caggiano

1. Demonstrate that all the relations in the relational schema are normalized to Boyce–Codd normal form (BCNF).
• For each table, specify whether it is in BCNF or not, and explain why.
• For each table that is not in BCNF, show the complete process that normalizes it to BCNF.

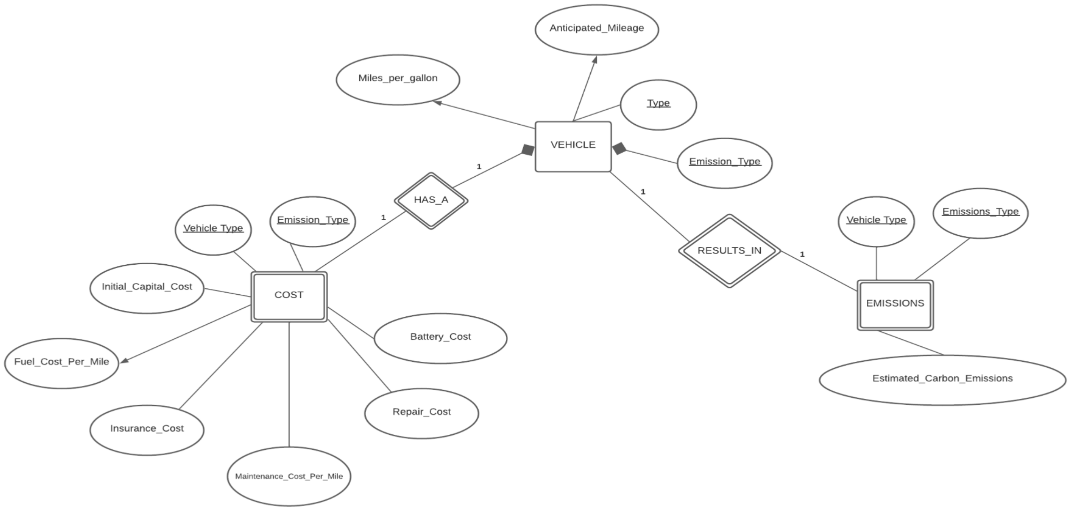
1. **<u>COST</u>**

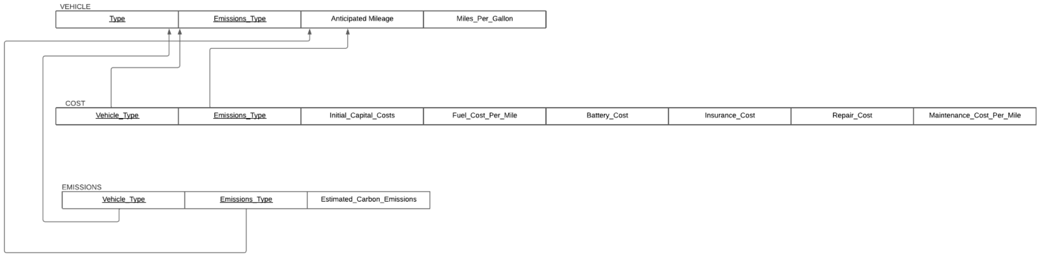
   COST is in BCNF. COST is in 1NF since it does not have any composite attributes, multivalued attributes, or nested relations. COST is in 2NF since every non prime attribute in COST depends on all prime attributes {Vehicle_Type, Emission_Type}. COST is in 3NF since it is in 2NF and no non-prime attribute is transitively dependent on the primary key {Vehicle_Type, Emission_Type}. COST is in BCNF since every functional dependency in COST X→A, implies that X is a superkey. No prime attributes in COST are dependent on another non-prime attribute. Hence, COST is in BCNF.

2. **<u>VEHICLE</u>**

   VEHICLE is in BCNF. VEHICLE is in 1NF since it does not have any composite attributes, multivalued attributes, or nested relations. VEHICLE is in 2NF since every non prime attribute in VEHICLE depends on all prime attributes {Vehicle_Type, Emission_Type}. VEHICLE is in 3NF since it is in 2NF and no non-prime attribute is transitively dependent on the primary key {Vehicle_Type, Emission_Type}. VEHICLE is in BCNF since every functional dependency in VEHICLE X→A, implies that X is a superkey. No prime attributes in VEHICLE are dependent on another non-prime attribute. Hence, VEHICLE is in BCNF.

3. Define the different views (virtual tables) required. For each view, list the data and transaction requirements. Give a few examples of queries, in English, to illustrate.





VEHICLE

Type	Emissions_Type	Anticipated Mileage	Miles_Per_Gallon

COST

Vehicle_Type	Emissions_Type	Initial_Capital_Costs	Fuel_Cost_Per_Mile	Battery_Cost	Insurance_Cost	Repair_Cost	Maintenance_Cost_Per_Mile

EMISSIONS

Vehicle_Type	Emissions_Type	Estimated_Carbon_Emissions

Views:

1. **User**
   - A user can come to this website and view costs and emissions associated with different vehicles as well as test different fleet vehicle transition strategies.

   - <u>Transactions:</u>
     - View categories of vehicles
       - Data requirements:
         - We will need to select Type from VEHICLE in order to display the different categories of vehicles
     - View costs associated with a vehicle
       - Data requirements:

- From VEHICLE: Type and Emissions_Type, inputted by user
- To display:
  - VEHICLE and COST must first be joined in order to find costs associated with this vehicle.
  - Anticipated_Mileage (Used to calculate Fuel Cost, Tire Costs, and Maintenance Costs)
  - Initial_Capital_Cost
  - Fuel_Cost_Per_Mile
  - Insurance_Cost
  - Maintenance_Cost_Per_Mile
  - Repair_Cost
  - Battery_Cost

- View emissions associated with a vehicle
  - Data requirements:
    - From VEHICLE: Type and Emissions_Type, inputted by user
    - To display:
      - VEHICLE and EMISSIONS must first be joined in order to find costs associated with this vehicle.
      - Anticipated_Mileage
      - Miles_Per_Gallon
      - Estimated_Carbon_Emissions
      - These are used to calculate the annual emissions
- View costs associated with a vehicle fleet
  - Similar to “View costs associated with a vehicle,” however, it must be performed with each vehicle group (Type, Emissions_Type) added to the fleet.
- View emissions associated with a vehicle fleet
  - Similar to “View emissions associated with a vehicle,” however, it must be performed with each vehicle group (Type, Emissions_Type) added to the fleet.

Example Queries:

1. View costs associated with Passenger Vehicle with an Internal Combustion Engine
2. View emissions associated with a Passenger Van with Zero emissions

2. **Admin**
   - An admin can perform all of the same functions as a user. They also can insert, update, and delete from any table.
   - Transactions:
     i. Add a vehicle type and its associated costs and emissions.

- Data requirements: We require the following data points to add a vehicle to our database.
- VEHICLE:
    a. Type, Emissions_Type, Anticipated_Milege, Miles_Per_Gallon
- COST
    a. Initial_Capital_Costs
    b. Fuel_Cost_Per_Mile
    c. Maintenance_Cost_Per_Mile
    d. Repair_Cost
    e. Battery_Cost
    f. Insurance_Cost
- EMISSIONS
    a. Estimated_Carbon_Emissions

ii. Delete a vehicle type and its associated costs and emissions

- Data Requirements:
    a. To delete an an entire vehicle:
        i. For VEHICLE: Type, Emissions_Type
        ii. Tuples in EMISSIONS, COST, and VEHICLE associated with {Type, Emissions_Type} must be deleted.

iii. Update a vehicle's costs

- Data Requirements:
    a. We require Vehicle_Type and Emission_Type in order to access the tuple with the vehicle's costs in COST

iv. Update a vehicle's emissions related information

- Data Requirements:
    a. We require Vehicle_Type and Emission_Type in order to access the tuple with the vehicle's emissions in EMISSIONS

Example Queries:

1. Calculate the annual costs of owning three Low Speed Utility vehicles with an Internal Combustion Engine.
2. Update a Public Safety Vehicle's Hybrid Engine Repair_Cost cost.

4. Design a complete set of SQL queries to satisfy the transaction requirements identified in the previous stages, using the relational schema and views defined in tasks 2 and 3 above.

- View categories of vehicles

    -
      ```
      SELECT Type
      FROM VEHICLE;
      ```
- View costs associated with a vehicle

```
CREATE VIEW VEHICLE_COSTS AS
SELECT *
FROM COST
INNER JOIN VEHICLE
ON VEHICLE.TYPE = COST.VEHICLE_TYPE AND
VEHICLE.EMISSIONS_TYPE = COST.EMISSIONS_TYPE;

SELECT *
FROM VEHICLE_COSTS
WHERE TYPE = INPUT_TYPE AND EMISSIONS_TYPE =
INPUT_EMISSIONS_TYPE;
```

- View emissions associated with a vehicle

```
CREATE TABLE VEHICLE_EMISSIONS AS
SELECT *
FROM COST
INNER JOIN VEHICLE
ON VEHICLE.TYPE = EMISSIONS.VEHICLE_TYPE AND
VEHICLE.EMISSIONS_TYPE = EMISSIONS.EMISSIONS_TYPE;

SELECT *
FROM VEHICLE_EMISSIONS
WHERE TYPE = INPUT_TYPE AND EMISSIONS_TYPE =
INPUT_EMISSIONS_TYPE;
```

- View costs associated with a vehicle fleet

```
CREATE VIEW VEHICLE_COSTS AS
SELECT *
FROM COST
INNER JOIN VEHICLE
ON VEHICLE.TYPE = COST.VEHICLE_TYPE AND
VEHICLE.EMISSIONS_TYPE = COST.EMISSIONS_TYPE;
```

- View emissions associated with a vehicle fleet

```
CREATE TABLE VEHICLE_EMISSIONS AS
SELECT *
FROM COST
INNER JOIN VEHICLE
ON VEHICLE.TYPE = EMISSIONS.VEHICLE_TYPE AND
VEHICLE.EMISSIONS_TYPE = EMISSIONS.EMISSIONS_TYPE;
```

- Add a vehicle type and its associated costs and emissions.

```
INSERT INTO VEHICLE (Type, Emissions_Type, Miles_per_gallon,
Anticipated_Mileage)
Input;
```

```
INSERT INTO COSTS (Type, Emissions_Type,Initial_Capital_Costs,
Fuel_Cost_Per_Mile, Maintenance_Cost_Per_Mile, Repair_Cost, Battery_Cost,
Insurance_Cost)
Input;

INSERT INTO EMISSIONS (Type, Emissions_Type,
Estimated_Carbon_Emissions)
```

- Delete a vehicle type and its associated costs and emissions

```
DELETE FROM COSTS
WHERE VEHICLE_TYPE=type AND EMISSIONS_TYPE=emissions_type;
DELETE FROM EMISSIONS
WHERE VEHICLE_TYPE=type AND EMISSIONS_TYPE=emissions_type;
DELETE FROM VEHICLE
WHERE TYPE=type AND EMISSIONS_TYPE=emissions_type;
```

- Update a vehicle’s costs

```
        UPDATE COSTS
        SET Vehicle_Type=x, Emissions_Type=y,Initial_Capital_Costs=z,
Fuel_Cost_Per_Mile=a, Maintenance_Cost_Per_Mile=b, Repair_Cost=c,
Battery_Cost=d, Insurance_Cost=e
        WHERE VEHICLE_TYPE=v_type AND EMISSIONS_TYPE=e_type;
```

- Update a vehicle’s emissions related information

```
UPDATE EMISSIONS
SET Vehicle_Type=x, Emissions_Type=y,Estimated_Carbon_Emissions=e
WHERE VEHICLE_TYPE=v_type AND EMISSIONS_TYPE=e_type;
```