

```

/*ViewQuery1*/
CREATE VIEW TotalVehiclesByLocation AS
SELECT l.location_id, COUNT(lv.VIN) AS TotalVehicles
FROM location l
JOIN location_vehicle_electric_link lv ON l.location_id = lv.location_id
WHERE l.county = 'king'
GROUP BY l.location_id;

```

-- This SQL statement retrieves the total count of electric vehicles associated with each location in King County from the 'location' and 'location_vehicle_electric_link' tables.

```

/*ViewQuery2*/
CREATE VIEW VehicleSpecsInLocation AS
SELECT vs.vin, vs.electric_vehicle_type, lm.location_id
FROM vehicle_specs vs
JOIN location_mapping lm ON vs.vin = lm.dol_vehicle_id
WHERE vs.base_msrp > 30000;

```

-- This SQL query selects the VIN, electric vehicle type, and location ID from the vehicle_specs table, joining it with the location_mapping table on VIN, and filters out records where the base MSRP is greater than or equal to zero.

```

/*ViewQuery3*/
CREATE VIEW AverageMSRPByModel AS
SELECT m.VIN , AVG(vs.base_msrp) as AverageMSRP
FROM model m
JOIN vehicle_specs vs ON m.VIN = vs.VIN AND m.model_year > (
SELECT AVG(m2.model_year)
FROM model m2)
GROUP BY m.VIN;

```

-- This query calculates the average MSRP for each model where the model year is above the average model year.

```

/*ViewQuery4*/
CREATE VIEW ElectricUtilityDetails AS
SELECT e.electric_utility ,lv.vin ,l.city,l.state
FROM electric_utility e
JOIN location_vehicle_electric_link lv ON
e.electric_utility_id=lv.electric_utility_id
JOIN location l on lv.location_id=l.location_id
WHERE e.electric_utility LIKE '%Power%';

```

-- This query joins the electric_utility, location_vehicle_electric_link, and location tables to list all electric utilities that contain the word 'Power' and their associated vehicles and locations.

```

/*ViewQuery5*/
CREATE VIEW MostCommonVehicleInLocation AS
SELECT l.Location_ID, m.Model, COUNT(vs.VIN) as VehicleCount
FROM location l
JOIN location_vehicle_electric_link lv ON l.Location_ID = lv.Location_ID

```

```
JOIN vehicle_specs vs ON lv.VIN = vs.VIN
JOIN model m ON vs.VIN = m.VIN
WHERE m.Model IN (
SELECT Model
FROM model
GROUP BY Model
HAVING COUNT(m.VIN) > 1)
GROUP BY l.Location_ID, m.Model
ORDER BY VehicleCount DESC;
```

-- This SQL query retrieves the location ID, model, and the count of vehicles associated with each unique model at each location, filtering only those models with more than one occurrence across all locations, and orders the results by vehicle count in descending order.