**Main Landing**

**Number of Total Vehicles Available**

| SELECT count(Vehicle.VIN) AS number\_of\_tot\_vehicles  FROM Vehicle  LEFT OUTER JOIN Sale  ON Vehicle.VIN=Sale.VIN  WHERE Sale.Sale\_price IS NULL; |
| --- |

### **LOGIN**

| SELECT password FROM **EmployeeUser** WHERE username = $username; |
| --- |

### **Get Number Of Total Vehicles Available**

| SELECT count(Vehicle.VIN) AS [Number Of Total Vehicles ]  FROM Vehicle LEFT OUTER JOIN Sale ON Vehicle.VIN=Sale.VIN  WHERE Sale.Sale\_price IS NULL; |
| --- |

**Get Options for Search Dropdowns**

| SELECT Manufacturer\_name FROM Manufacturer;  --DropDown for Vehicle Type  SELECT DISTINCT Vehicle\_type  FROM (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType  SELECT DISTINCT (Year) from Vehicle;  SELECT DISTINCT (Color) from Color; |
| --- |

### **Search Vehicle by VIN**

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND VIN='$VIN'  ORDER BY V.VIN ASC; |
| --- |

### **Search Vehicles**

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  ORDER BY V.VIN ASC; |
| --- |

* Searching can be done on the following criteria

Vehicle type : user will select an option from the **dropdown** the value will be stored in the variable $vehicleType

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND VehicleType.Vehicle\_type = '$VehicleType'  ORDER BY V.VIN ASC; |
| --- |

• Manufacturer: user will select an option from the **dropdown** the value will be stored in the variable $Manufacturer

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.Manufacturer\_name = '$Manufacturer'  ORDER BY V.VIN ASC; |
| --- |

• Model year: user will select an option from the **dropdown** the value will be stored in the variable $Model\_Year

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.Year = '$Model\_Year'  ORDER BY V.VIN ASC; |
| --- |

• Color: user will select an option from the **dropdown** the value will be stored in the variable $Color

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN) like '%$Color%'  ORDER BY V.VIN ASC; |
| --- |

• List price (either greater than and/or less than an entered value)

* Greater Than

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.List\_price > '$List\_Price'  ORDER BY V.VIN ASC; |
| --- |

* user search List Price less than

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s) AND  v.List\_price < '$List\_Price'  ORDER BY V.VIN ASC, |
| --- |

* user search List Price by Entered Value

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s) AND  v.List\_price = '$List\_Price'  ORDER BY V.VIN ASC; |
| --- |

• User enters a keyword and it's stored in the variable $keyword, the system will search on specific columns (either entirely or as a substring).

Keyword, which searches the manufacturer, model year, model name and description fields.

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND (v.Manufacturer\_name like '%$keyword%'  OR v.Year like '%$keyword%'  OR v.Model\_name like '%$keyword%'  OR v.Description like '%$keyword%')  ORDER BY V.VIN ASC; |
| --- |

### **Filter By Sold/Unsold/All Vehicles**

### **Unsold Vehicles**:

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  ORDER BY V.VIN ASC; |
| --- |

**Sold Vehicles:**

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  WHERE v.VIN IN( SELECT s.VIN FROM Sale s)  ORDER BY V.VIN ASC; |
| --- |

**All Vehicles:**

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name,V.Description  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  ORDER BY V.VIN ASC; |
| --- |

### 

### **Add Vehicle**

| INSERT INTO Vehicle  (VIN,Year,Model\_name,Description,Invoice\_price,List\_price,Inventory\_date  ,Manufacturer\_name,Username)  VALUES  ('$VIN', '$year', '$model\_name', '$description', '$invoice\_price', '$list\_price', GETDATE(), '$manufacturer\_name', '$username') |
| --- |

Insert Vehicle type in the corresponding table.

* if Inventory clerk select Car insert Attributes for car:

| INSERT INTO Car(VIN,Doors\_count) VALUES ($VIN,$Doors\_count) |
| --- |

* if Inventory clerk select Truck insert Attributes for Truck:

| INSERT INTO Truck(VIN,Cargo\_capacity,Cargo\_cover\_type,Axle\_count) VALUES ('$VIN','$Cargo\_capacity','$Cargo\_cover\_type','$Axle\_count') |
| --- |

* if Inventory clerk select Convertible insert Attributes for Convertible:

| INSERT INTO Convertible (VIN,Roof\_type,Back\_seat\_count) VALUES ('$VIN','$Roof\_type','$Back\_seat\_count') |
| --- |

* if Inventory clerk select SUV insert Attributes for SUV

| INSERT INTO SUV(VIN,Drivetrain\_type,Cupholder\_count) VALUES ('$VIN','$Drivetrain\_type','$Cupholder\_count') |
| --- |

* if Inventory clerk select VanMinivan insert Attributes for VanMinivan:

| INSERT INTO VanMinivan(VIN,Has\_driver\_back\_door) VALUES ('$VIN','$Has\_driver\_back\_door') |
| --- |

### **View Vehicle**

* + If the role is anonymous, sales person, service writer:

If the user select a Vehicle type Car from the main search display details of the vehicle

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description  ,c.Doors\_count  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN Car c ON v.VIN=C.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type truck display detail and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description  ,t.Cargo\_capacity,t.Cargo\_cover\_type,t.Axle\_count  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN Truck t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type SUV display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description  ,t.Cupholder\_count,t.Drivetrain\_type  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **SUV** t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type Convertible display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description  ,t.Back\_seat\_count,t.Roof\_type  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **Convertible** t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type Convertible display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description  ,t.Has\_driver\_back\_door  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **VanMinivan** t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

* + If the role is Inventory Clerk :

If the user select a Vehicle type Car display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description,v.Invoice\_price  ,c.Doors\_count  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **Car** c ON v.VIN=C.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type Truck display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description,v.Invoice\_price  ,t.Cargo\_capacity,t.Cargo\_cover\_type,t.Axle\_count  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **Truck** t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type SUV display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN)  ,v.List\_price,V.Description,v.Invoice\_price  ,t.Cupholder\_count,t.Drivetrain\_type  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **SUV** t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type Convertible display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN),v.List\_price,V.Description,v.Invoice\_price  ,t.Back\_seat\_count,t.Roof\_type  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **Convertible** t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

If the user select a Vehicle type VanMinivan display the details and the corresponding vehicle Attributes

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Manufacturer\_name,v.Model\_name  , color = ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN),v.List\_price,V.Description,v.Invoice\_price  ,t.Has\_driver\_back\_door  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN **VanMinivan** t ON v.VIN=t.VIN  WHERE v.VIN NOT IN( SELECT s.VIN FROM Sale s)  AND v.VIN='$VIN'; |
| --- |

* + If the role is Managers or Roland Around:
    - If the vehicle has been sold:

| SELECT v.VIN  ,VehicleType.Vehicle\_type  ,v.Year,v.Model\_name,v.Manufacturer\_name  , ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=v.VIN) AS COLOR  ,v.List\_price, v.Description,Inventory\_date,(ic.First\_name + ' ' + ic.Last\_name ) AS Inventory\_Clerk,v.Invoice\_price  ,s.Sale\_date ,s.Sale\_price , (e.First\_name + ' ' + e.Last\_name) as SalesPersonName  ,cp.CustomerName as BuyerName  ,c.Phone\_number as BuyerPhone,c.Email as BuyerEmail,c.Street\_address as BuyerAdress,c.State,c.Postal\_code  FROM **Vehicle** v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN Sale s ON v.VIN=s.VIN  LEFT JOIN **EmployeeUser** e ON s.Username=e.Username  LEFT JOIN **EmployeeUser** ic ON v.Username=ic.Username  LEFT JOIN **Customer** c ON s.Customer\_id=c.Customer\_id  LEFT JOIN (SELECT p.Customer\_id, (p.First\_name + ' ' + p.Last\_name)as CustomerName FROM **Person** p  UNION  SELECT b.Customer\_id, b.Business\_name as CustomerName FROM Business b) as CP  ON CP.customer\_id=c.customer\_id  WHERE v.VIN='$VIN' |
| --- |

* + - If the vehicle has repairs ( **Repairs Section** )

| SELECT  cp.CustomerName  ,(e.First\_name + ' ' + e.Last\_name) as ServiceWriter  ,r.Start\_date, r.Completion\_date, r.Labor\_charges, r.Total\_cost  FROM Repair r  LEFT JOIN EmployeeUser e ON r.Username=e.Username  LEFT JOIN Customer c ON r.Customer\_id=c.Customer\_id  LEFT JOIN (SELECT p.Customer\_id, (p.First\_name + ' ' + p.Last\_name)as CustomerName FROM Person p  UNION  SELECT b.Customer\_id, b.Business\_name as CustomerName FROM Business b) as CP  ON CP.customer\_id=c.customer\_id  WHERE r.VIN='$VIN' |
| --- |

### **Lookup Customer**

* + If *drivers\_license* is found

| SELECT First\_name, Last\_name, Phone\_number, Email, Street\_address, City, State, Postal\_code FROM Customer INNER JOIN Person  ON Customer.customer\_id = Person.customer\_id |
| --- |

* + If *TIN* is found

| SELECT Contact\_name, Contact\_title, Business\_name, Phone\_number, Email, Street\_address, City, State, Postal\_code FROM Customer INNER JOIN Business  ON Customer.customer\_id = Business.customer\_id |
| --- |

### **Add Customer**

| INSERT INTO Customer (phone\_number, email, street\_address, city, state, postal\_code) VALUES ( '$phone\_numer', '$email', '$street\_address', '$city', '$state', '$postal\_code') |
| --- |

* If **CUSTOMER** is Person,then

| INSERT INTO Person (driver\_license, customer\_id, first\_name, last\_name) VALUES ('$driver\_license','$customer\_id','$first\_name', '$last\_name'); |
| --- |

* If **CUSTOMER** is Business, then

| INSERT INTO Business(TIN, customer\_id, contact\_name, contact\_title, Business\_name)  VALUES ('$TIN', '$customer\_id', '$contact\_name', '$contact\_title', '$Business\_name'); |
| --- |

### **Enter Sale**

| INSERT INTO Sale (VIN,Username,Customer\_id,Sale\_price,Sale\_date)  VALUES('$VIN','$Username','$Customer\_id','$Sale\_price','$Sale\_date') |
| --- |

### **Create Repair**

| INSERT INTO Repair (VIN,Customer\_id,Start\_date,Description, Odometer\_reading,Username) VALUES('$VIN','$Customer\_id','$Start\_date','$Description','$Odometer\_reading','$Username'); |
| --- |

### **Add Part**

| INSERT INTO Part(VIN ,Customer\_id ,Start\_date ,Part\_number ,Vendor\_name ,Quantity ,Price)  VALUES('$VIN' ,'$Customer\_id' ,'$Start\_date' ,'$Part\_number' ,'$Vendor\_name' ,'$Quantity' ,'$Price' ); |
| --- |

### **Complete Repair**

| UPDATE Repair SET Labor\_charges = '$labor\_charges' ,Total\_cost = '$total\_cost',  Description = '$description' ,Completion\_date = '$completion\_date' ,Username = '$username'  WHERE VIN='$VIN' AND start\_date='$start\_date' AND Customer\_id='$customer\_id' |
| --- |

### **Update Labor Charge**

| SELECT Labor\_charges  FROM Repair  WHERE VIN = '$currentVin' AND Start\_date = '$currentRepairStartDate' |
| --- |

| UPDATE Repair  SET Labor\_charges = '$newLaborCharges'  WHERE VIN = $currentVin AND Start\_date = '$currentRepairStartDate' |
| --- |

### **View Repair** l

| SELECT *Start\_date, Completion\_date, Customer\_id,* *Username, Description*, *Odometer\_reading Labor\_charges, Total\_cost, (Total\_cost-Labor\_charges) AS Part\_cost*  FROM Repair  WHERE VIN = '$currentVin' AND  Completion\_date IS NOT NULL |
| --- |

* Get customer *First\_name* and *Last\_name*

| SELECT *First\_name, Last\_name,*  FROM Individual Person  WHERE *Customer\_id* = '$*Customer\_id*' |
| --- |

Get customer *Business\_name*

| SELECT *Business\_name*  FROM Business  WHERE *Customer\_id* = '$*Customer\_id*' |
| --- |

| SELECT *Quantity*, *Vendor\_name*, *Part\_number*, *Price*  FROM Repair  WHERE VIN = '$currentVin' AND  Start\_date = '$currentRepairStartDate' |
| --- |

| SELECT Display *Description*, *Odometer\_reading*, and *Start\_date*  WHERE VIN = '$currentVin' AND  Start\_date = '$currentRepairStartDate' AND  Completion\_date IS NULL |
| --- |

### 

### 

### **View/Generate Reports**

### **Sales By Color:**

| SELECT MAX(Sale\_date) from Sale |
| --- |

| SELECT CarColor  ,count(CASE WHEN s.Sale\_date > $MaxSaleDate-30 THEN s.Sale\_date ELSE NULL END) LastMonth  ,count(CASE WHEN s.Sale\_date > $MaxSaleDate-365 THEN s.Sale\_date ELSE NULL END) LastYear  ,count(s.Sale\_date) Alltime  FROM (  SELECT 'Aluminum' AS CarColor  UNION SELECT 'Beige' AS CarColor UNION SELECT 'Black' AS CarColor  UNION SELECT 'Blue'AS CarColor UNION SELECT 'Brown' AS CarColor  UNION SELECT 'Bronze' AS CarColor UNION SELECT 'Claret' AS CarColor  UNION SELECT 'Copper' AS CarColor UNION SELECT 'Cream' AS CarColor  UNION SELECT 'Gold' AS CarColor UNION SELECT 'Gray' AS CarColor  UNION SELECT 'Green' AS CarColor UNION SELECT 'Maroon' AS CarColor  UNION SELECT 'Metallic' AS CarColor UNION SELECT 'Navy' AS CarColor  UNION SELECT 'Orange' AS CarColor UNION SELECT 'Pink' AS CarColor  UNION SELECT 'Purple' AS CarColor  UNION SELECT 'Red' AS CarColor  UNION SELECT 'Rose' AS CarColor  UNION SELECT 'Rust' AS CarColor  UNION SELECT 'Silver' AS CarColor  UNION SELECT 'Tan' AS CarColor  UNION SELECT 'Turquoise' AS CarColor  UNION SELECT 'White' AS CarColor  UNION SELECT 'Yellow' AS CarColor  UNION SELECT 'Multiple' AS CarColor) AS Colors  LEFT JOIN (select  CASE  WHEN ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=s.VIN) LIKE '%|%' THEN 'Multiple'  ELSE ( SELECT DISTINCT STRING\_AGG(c.Color,' | ') FROM Color c WHERE c.VIN=s.VIN) END AS ColorCase ,s.Sale\_date  from Sale s ) as s ON CarColor = ColorCase GROUP BY CarColor |
| --- |

### 

### **Sales By Type:**

| SELECT MAX(Sale\_date) from Sale |
| --- |

| SELECT VT  ,count(CASE WHEN sales.Sale\_date > $MaxSaleDate -30 THEN sales.Sale\_date ELSE NULL END) lastMonth  ,count(CASE WHEN sales.Sale\_date > $MaxSaleDate -365 THEN sales.Sale\_date ELSE NULL END) LastYear  ,count(sales.Sale\_date) Alltime  from (  select 'Car' as VT  UNION select 'SUV' as VT  UNION select 'Truck' as VT  UNION select 'Convertible' as VT  UNION select 'VanMinivan' as VT  ) as UnionVt  LEFT JOIN (  SELECT v.VIN,VehicleType.Vehicle\_type,s.Sale\_date  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN Sale s ON V.VIN=s.VIN  WHERE s.Sale\_date is not null  ) AS sales ON sales.Vehicle\_type = UnionVt.VT  GROUP BY UnionVt.VT  ORDER BY VT ASC |
| --- |

### **Sales By Manufacturer:**

| SELECT MAX(Sale\_date) from Sale |
| --- |

| SELECT (Manufacturer\_name)  ,count(CASE WHEN Sale.Sale\_date > $MaxSaleDate -30 THEN Sale.Sale\_date ELSE NULL END) LastMonth  ,count(CASE WHEN Sale.Sale\_date >$MaxSaleDate -365 THEN Sale.Sale\_date ELSE NULL END) LastYear  ,count(Vehicle.VIN) Alltime  FROM Vehicle LEFT OUTER JOIN Sale ON Vehicle.VIN = Sale.VIN  WHERE Sale.Sale\_price IS NOT NULL  GROUP BY Manufacturer\_name  ORDER BY Manufacturer\_name ASC |
| --- |

### **Gross Customer Income:**

| SELECT TOP 15  c.Customer\_id,CP.CustomerName  ,CASE  WHEN MIN(s.Sale\_date) IS NULL AND MIN(r.Start\_date) IS NULL THEN NULL  WHEN MIN(s.Sale\_date) IS NULL AND MIN(r.Start\_date) IS NOT NULL THEN MIN(r.Start\_date)  WHEN MIN(s.Sale\_date) IS NOT NULL AND MIN(r.Start\_date) IS NULL THEN MIN(s.Sale\_date)  WHEN MIN(s.Sale\_date) > MIN(r.Start\_date) THEN MIN(r.Start\_date)  ELSE MIN(s.Sale\_date)  END AS FirstService  ,CASE  WHEN MAX(s.Sale\_date) IS NULL AND MAX(r.Start\_date) IS NULL THEN NULL  WHEN MAX(s.Sale\_date) IS NULL AND MAX(r.Start\_date) IS NOT NULL THEN MAX(r.Start\_date)  WHEN MAX(s.Sale\_date) IS NOT NULL AND MAX(r.Start\_date) IS NULL THEN MAX(s.Sale\_date)  WHEN MAX(s.Sale\_date) > MAX(r.Start\_date) THEN MAX(s.Sale\_date)  ELSE MAX(r.Start\_date)  END AS MostRecentService  ,COUNT(s.Sale\_date) AS NumberOfSales  ,COUNT(r.Start\_date) AS NumberOfRepairs  ,ISNULL(SUM(s.Sale\_price),0) + ISNULL(SUM(r.Total\_cost),0) AS GrossIncome  FROM Customer c  LEFT JOIN (SELECT p.Customer\_id, (p.First\_name + ' ' + p.Last\_name)as CustomerName FROM Person p  UNION  SELECT b.Customer\_id, b.Business\_name as CustomerName FROM Business b) AS CP  ON c.Customer\_id = CP.Customer\_id  LEFT JOIN Sale s ON c.Customer\_id = s.Customer\_id  LEFT JOIN Repair r ON c.Customer\_id = r.Customer\_id AND s.VIN=r.VIN  GROUP BY C.Customer\_id,CP.CustomerName  ORDER BY GrossIncome DESC, MostRecentService DESC |
| --- |

**Customer Drill-Down**

**Sales Section**

| SELECT  CP.CustomerName,s.Sale\_date  ,s.Sale\_price  ,s.VIN  ,v.Year  ,v.Manufacturer\_name  ,v.Model\_name  ,eu.First\_name + ' ' + eu.Last\_name AS SalesPersonName  FROM Customer c  LEFT JOIN (SELECT p.Customer\_id, (p.First\_name + ' ' + p.Last\_name)as CustomerName FROM Person p  UNION  SELECT b.Customer\_id, b.Business\_name as CustomerName FROM Business b) AS CP  ON c.Customer\_id = CP.Customer\_id  LEFT JOIN Sale s ON c.Customer\_id = s.Customer\_id  LEFT JOIN Vehicle v ON s.VIN = v.VIN  LEFT JOIN EmployeeUser eu ON s.Username = eu.Username  WHERE c.Customer\_id = '$Customer\_id'  ORDER BY s.Sale\_date DESC, s.VIN DESC |
| --- |

**Repairs Section**

| SELECT  CP.CustomerName  ,r.Start\_date  ,r.Completion\_date  ,r.VIN  ,r.Odometer\_reading  ,r.Labor\_charges  ,r.Total\_cost  ,eu.First\_name + ' ' + eu.Last\_name AS SalesPersonName  FROM Customer c  LEFT JOIN (SELECT p.Customer\_id, (p.First\_name + ' ' + p.Last\_name)as CustomerName FROM Person p  UNION  SELECT b.Customer\_id, b.Business\_name as CustomerName FROM Business b) AS CP  ON c.Customer\_id = CP.Customer\_id  LEFT JOIN Repair r ON c.Customer\_id = r.Customer\_id  LEFT JOIN EmployeeUser eu ON r.Username = eu.Username  WHERE c.Customer\_id = '$Customer\_id'  ORDER BY r.Start\_date DESC, r.VIN DESC, r.Completion\_date ASC |
| --- |

### **Average Time In Inventory:**

| SELECT VT  ,ISNULL(CAST(AVG(DATEDIFF(DAY,sales.Inventory\_date,sales.Sale\_date)+1 )AS varchar),'N/A') AS AVERAGE  from (  select 'Car' as VT  UNION select 'SUV' as VT  UNION select 'Truck' as VT  UNION select 'Convertible' as VT  UNION select 'VanMinivan' as VT  ) as UnionVt  LEFT JOIN (  SELECT v.VIN,VehicleType.Vehicle\_type,s.Sale\_date,v.Inventory\_date  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN Sale s ON V.VIN=s.VIN  WHERE s.Sale\_date is not null  ) AS sales ON sales.Vehicle\_type = UnionVt.VT  GROUP BY UnionVt.VT  ORDER BY VT ASC |
| --- |

### **Part Statistics:**

| SELECT Vendor\_name,  SUM (Part.price \* Part.Quantity) AS [Total\_Dollar\_Amount], SUM (Quantity) AS [Total\_Number\_Parts]  FROM Part  GROUP BY Vendor\_name  ORDER BY [Total\_Dollar\_Amount] DESC; |
| --- |

### 

### **Below Cost Sales:**

| SELECT  FORMAT (s.Sale\_date, 'MM-dd-yyyy') Sale\_Date  ,v.Invoice\_price  ,s.Sale\_price  ,(100\*(Sale\_price)/(Invoice\_price)) AS Profit\_ratio  ,CASE  WHEN (100\*(Sale\_price)/(Invoice\_price)) <= 95 THEN 'Red'  END AS Background  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN Sale s ON v.VIN=s.VIN  LEFT JOIN EmployeeUser e ON s.Username=e.Username  LEFT JOIN EmployeeUser ic ON v.Username=ic.Username  LEFT JOIN Customer c ON s.Customer\_id=c.Customer\_id  LEFT JOIN (SELECT p.Customer\_id, (p.First\_name + ' ' + p.Last\_name)as CustomerName FROM Person p  UNION  SELECT b.Customer\_id, b.Business\_name as CustomerName FROM Business b) as CP  ON CP.customer\_id=c.customer\_id  WHERE s.Sale\_date IS NOT NULL  ORDER BY  s.Sale\_date DESC,  (100\*(Sale\_price)/(Invoice\_price)) DESC |
| --- |

### 

### 

### 

### 

### 

### 

### 

### **Repairs by Manufacturer/Type/Model**

| SELECT  m.Manufacturer\_name  ,COUNT(r.Start\_date) Repairs  ,SUM(p.Quantity \* p.Price) AS PartsCost  ,SUM(r.Labor\_charges) LaborCost  ,SUM(r.Total\_cost) TotalRepairCost  FROM Manufacturer m  LEFT JOIN Vehicle v ON m.Manufacturer\_name = v.Manufacturer\_name  LEFT JOIN Repair r ON v.VIN = r.VIN  LEFT JOIN Part p ON r.VIN = p.VIN AND r.Start\_date = p.Start\_date AND r.Customer\_id = p.Customer\_id  GROUP BY m.Manufacturer\_name  ORDER BY m.Manufacturer\_name ASC |
| --- |

* If user clicks on row (manufacturer drilldown)

**Vehicle Type Drilldown**: System is going to get $selectedManufacturer from the previous drilldown selection

| SELECT VT AS [Vehicle Type],  SUM(Labor\_charges) AS All\_labor\_Costs,  SUM(Total\_cost) AS Total\_Repair\_cost,  (SUM(Total\_cost) - SUM(Labor\_charges)) AS All\_Parts\_Costs,  COUNT(Start\_date) AS Count\_Repairs  from (  select 'Car' as VT  UNION select 'SUV' as VT  UNION select 'Truck' as VT  UNION select 'Convertible' as VT  UNION select 'VanMinivan' as VT  ) as UnionVt  JOIN (  SELECT v.VIN,VehicleType.Vehicle\_type,v.Model\_name,r.Labor\_charges,r.Total\_cost,r.Start\_date  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  JOIN Repair r ON V.VIN=r.VIN  WHERE Manufacturer\_name = ' $selectedManufacturer'  ) AS repairs ON Repairs.Vehicle\_type = UnionVt.VT  GROUP BY UnionVt.VT  ORDER BY Count\_Repairs DESC; |
| --- |

**Model Drilldown**: System is going to get $selectedManufacturer AND $VehicleType from the previous drilldown select

| SELECT Vehicle\_type, Model\_name,  SUM(Labor\_charges) AS All\_labor\_Costs,  SUM(Total\_cost) AS Total\_Repair\_cost,  (SUM(Labor\_charges) - SUM(Total\_cost)) AS All\_Parts\_Costs,  COUNT(Start\_date) AS Count\_Repairs  from (  select 'Car' as VT  UNION select 'SUV' as VT  UNION select 'Truck' as VT  UNION select 'Convertible' as VT  UNION select 'VanMinivan' as VT  ) as UnionVt  JOIN (  SELECT v.VIN,VehicleType.Vehicle\_type,v.Model\_name,r.Labor\_charges,r.Total\_cost,r.Start\_date  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  JOIN Repair r ON V.VIN=r.VIN  WHERE Manufacturer\_name = '$selectedManufacturer'  ) AS repairs ON repairs.Vehicle\_type = UnionVt.VT  GROUP BY UnionVt.VT,Vehicle\_type,Model\_name  ORDER BY Vehicle\_type,Count\_Repairs DESC,Model\_name; |
| --- |

### 

### **Monthly Sales**

| SELECT  YEAR(s.Sale\_date) SaleYear  ,MONTH(S.SALE\_DATE) SaleMonth  ,COUNT(v.VIN) Total\_Number\_of\_Vehicles\_Sold  ,SUM(s.sale\_price) Total\_Sales\_Income  ,SUM(s.Sale\_price - v.Invoice\_price) total\_net\_income  ,(100\*SUM(Sale\_price)/SUM(Invoice\_price)) AS Profit\_ratio  ,CASE  WHEN (100\*SUM(Sale\_price)/SUM(Invoice\_price)) >= 125 THEN 'Green'  WHEN (100\*SUM(Sale\_price)/SUM(Invoice\_price)) <= 110 THEN 'Yellow'  END AS Background  FROM Vehicle v  LEFT JOIN (  SELECT Car.VIN, 'Car' AS Vehicle\_type FROM Car  UNION SELECT SUV.VIN, 'SUV' AS Vehicle\_type FROM SUV  UNION SELECT Truck.VIN, 'Truck' AS Vehicle\_type FROM Truck  UNION SELECT Convertible.VIN, 'Convertible' AS Vehicle\_type FROM Convertible  UNION SELECT VanMinivan.VIN, 'VanMinivan' AS Vehicle\_type FROM VanMinivan  ) AS VehicleType ON v.VIN= vehicleType.VIN  LEFT JOIN Sale s ON v.VIN=s.VIN  LEFT JOIN EmployeeUser e ON s.Username=e.Username  LEFT JOIN EmployeeUser ic ON v.Username=ic.Username  LEFT JOIN Customer c ON s.Customer\_id=c.Customer\_id  LEFT JOIN (SELECT p.Customer\_id, (p.First\_name + ' ' + p.Last\_name)as CustomerName FROM Person p  UNION  SELECT b.Customer\_id, b.Business\_name as CustomerName FROM Business b) as CP  ON CP.customer\_id=c.customer\_id  WHERE s.Sale\_date IS NOT NULL  GROUP BY  YEAR(s.Sale\_date)  ,MONTH(S.SALE\_DATE)  ORDER BY  YEAR(s.Sale\_date) DESC  ,MONTH(S.SALE\_DATE) DESC |
| --- |

**Monthly Sales Drill Down:**

| SELECT TOP 1  eu.First\_name + ' ' + eu.Last\_name AS SalesPersonName  ,COUNT(s.Username) NumberVehiclesSold  ,YEAR(s.Sale\_date) AS SaleYear  ,MONTH(s.Sale\_date) AS SaleYear  ,SUM(s.Sale\_price) TotalSales  FROM Sale s  LEFT JOIN EmployeeUser eu ON s.Username = eu.Username  WHERE YEAR(s.Sale\_date) = '$YearfromDrillDown' AND MONTH(s.Sale\_date) = '$MonthfromDrillDown'  GROUP BY eu.First\_name + ' ' + eu.Last\_name, YEAR(s.Sale\_date) ,MONTH(s.Sale\_date)  ORDER BY NumberVehiclesSold DESC, TotalSales DESC |
| --- |