Table of Contents

Login/Logout

Table 1

```
-- Login
SELECT * FROM PriviledgedUsers WHERE BINARY Username = %s AND BINARY Password = %s

--Get login information
WITH ManagersRole AS
(SELECT Username, 'Manager'  AS Role FROM Managers),
InventoryClkerksRole AS
(SELECT Username, 'Inventory Clerk'  AS Role FROM InventoryClerks),
SalespeopleRole AS
(SELECT Username, 'Salesperson' AS Role FROM Salespeople),
ServiceWritersRole AS
(SELECT Username, 'Service Writer' AS Role FROM ServiceWriters)
SELECT PU.Username, R.Role FROM PriviledgedUsers AS PU
JOIN (SELECT * FROM ManagersRole UNION SELECT * FROM  InventoryClkerksRole
UNION SELECT * FROM  SalespeopleRole UNION SELECT * FROM ServiceWritersRole) AS R
ON PU.Username = R.Username
WHERE PU.Username = '{a}';
```

Search Vehicle

Table 2

```
--Get Number of unsold vehicles
SELECT ((SELECT COUNT(*) FROM Vehicles)-(SELECT COUNT(*) FROM Salesevents)) AS NumUnsold;

--Get Model Year dropdown
SELECT DISTINCT ModelYear FROM Vehicles ORDER BY ModelYear DESC;

--Get Manufacturer dropdown
SELECT ManufacturerName FROM Manufacturer;

--Get Type dropdown
SELECT 'Car' AS Type UNION SELECT 'SUV' UNION SELECT 'Truck'
UNION SELECT 'Convertible' UNION SELECT 'Van MiniVan';

--Get Color dropdown
SELECT Color FROM Colors;

--Search Vehicles
With TypeInfo AS (
SELECT Vin, 'Car' AS Type FROM Cars
UNION SELECT Vin, 'SUV' AS Type FROM SUVs
UNION SELECT Vin, 'Van MiniVan' AS Type FROM VanMiniVans
UNION   SELECT Vin, 'Truck' AS Type FROM Trucks
UNION SELECT Vin, 'Convertible' AS Type FROM Convertibles
),
SearchResult AS (
SELECT Vehicles.Vin, T.Type, ModelYear,  Manufacturer, ModelName AS Model,
C.VColors AS Colors, InvoicePrice * 1.25 AS ListPrice, S.SaleDate, Description
FROM Vehicles
LEFT OUTER JOIN
(SELECT Vin, GROUP_CONCAT(Colors SEPARATOR ' ') AS VColors
FROM VehicleColors GROUP BY Vin
) AS C ON C.Vin = Vehicles.Vin
LEFT OUTER JOIN SalesEvents AS S ON S.Vin = Vehicles.Vin
JOIN TypeInfo AS T ON T.Vin =Vehicles.Vin
WHERE TRUE AND
(CASE WHEN %s IS NOT NULL
THEN Type = '{t}' ELSE TRUE END)
AND
(CASE WHEN %s IS NOT NULL
THEN Manufacturer = '{m}' ELSE TRUE END)
AND
(CASE WHEN %s IS NOT NULL
 THEN ModelYear = '{y}' ELSE TRUE END)
AND
(CASE WHEN %s IS NOT NULL
```

```
THEN InvoicePrice * 1.25  <=  '{maxp}'  ELSE TRUE END)
AND
(CASE WHEN %s IS NOT NULL
THEN InvoicePrice * 1.25 >=  '{minp}' ELSE TRUE END)
AND
(CASE WHEN %s IS NOT NULL
THEN C.VColors = '{c}' OR C.VColors LIKE BINARY %s
ELSE TRUE END)
AND
(CASE WHEN %s IS NOT NULL
THEN  Manufacturer LIKE BINARY %s OR ModelName LIKE BINARY %s
OR ModelYear LIKE BINARY %s OR Description LIKE BINARY %s
ELSE TRUE END))
SELECT Vin, Type, ModelYear,  Manufacturer, Model, Colors,
CAST(ListPrice as DECIMAL(10,2)),
(CASE WHEN %s IS NOT NULL AND Description LIKE BINARY %s
THEN "X" ELSE " " END) AS MatchDescription
FROM SearchResult WHERE TRUE AND
(CASE WHEN %s IS NOT NULL THEN Vin = '{v}' ELSE TRUE END)
AND
(CASE WHEN '{IsM}' IS FALSE THEN SaleDate IS NULL
ELSE (CASE WHEN '{f}' = 'sold' THEN SaleDate IS NOT NULL
WHEN '{f}' = 'unsold' THEN SaleDate IS NULL
ELSE TRUE END) END)
ORDER BY Vin;
```

Vehicle Detail

Table 3

```
-- General Section
With TypeInfo AS (
        SELECT Vin, 'Car' AS Type FROM Cars
    UNION SELECT Vin, 'SUV' AS Type FROM SUVs
    UNION SELECT Vin, 'Van MiniVan' AS Type FROM VanMiniVans
    UNION       SELECT Vin, 'Truck' AS Type FROM Trucks
    UNION SELECT Vin, 'Convertible' AS Type FROM Convertibles
    )
    SELECT V.Vin, Type, Manufacturer, ModelName AS Model, ModelYear,
    CAST(1.25*InvoicePrice AS DECIMAL(10,2)) AS ListPrice, C.VColors AS Color,
    IFNULL(Description, ''), InvoicePrice, DateAdded,
        CONCAT(FirstName, ' ', LastName) AS ClerkName
    FROM  Vehicles AS V
    LEFT OUTER JOIN
    (SELECT Vin, GROUP_CONCAT(Colors SEPARATOR ' ') AS VColors
    FROM VehicleColors GROUP BY Vin
    ) AS C ON C.Vin = V.Vin
    LEFT OUTER JOIN TypeInfo ON V.Vin = TypeInfo.Vin
        LEFT JOIN PriviledgedUsers AS P ON V.ClerkUsername = P.Username
    WHERE V.Vin = '{a}';

-- Type attribute
SELECT NumOfDoors FROM Cars WHERE VIN = '{a}';

SELECT DrivetrainType, NumberOfCupholders FROM SUVs WHERE VIN = '{a}';

SELECT HasDriverSideBackDoor FROM VanMiniVans WHERE VIN = '{a}';

SELECT CargoCoverType, NumberOfRearAxies, CargoCapacity Trucks WHERE VIN = '{a}';

SELECT BackSeatCount, RoofType FROM Convertibles WHERE VIN = '{a}';

-- Sale Section
With CustomerInfo AS
(
SELECT  CONCAT(P.FirstName, ' ', P.LastName) AS Name, NULL AS ContactName,
NULL AS ContactTitle,C.PhoneNum AS Phone,
IFNULL(C.Email, '') AS Email, P.CustomerID,
CONCAT(C.Street, ', ', C.City, ', ', C.State, ', ', C.Zipcode) AS Address
FROM Persons AS P
INNER JOIN Customers AS C ON P.CustomerID = C.CustomerID
UNION ALL
SELECT Name, CONCAT(ContactFName, ' ', ContactLName) AS ContactName,
ContactTitle, C.PhoneNum AS Phone, IFNULL(C.Email, '') AS Email, B.CustomerID,
CONCAT(C.Street, ', ', C.City, ', ', C.State, ', ', C.Zipcode) AS Address
```

```
FROM Business AS B
INNER JOIN Customers AS C ON B.CustomerID = C.CustomerID
)
SELECT SalePrice, SaleDate, CONCAT(SP.FirstName, ' ', SP.LastName) AS SalespersonName,
CustomerInfo.Name, CustomerInfo.ContactName, CustomerInfo.ContactTitle,
CustomerInfo.Phone, CustomerInfo.Email, CustomerInfo.Address
FROM SalesEvents
INNER JOIN PrIviledgedUsers AS SP ON SP.Username = SalesEvents.Username
INNER JOIN CustomerInfo ON SalesEvents.CustomerID = CustomerInfo.CustomerID
WHERE SalesEvents.Vin = '{a}';

-- Repair Section
With CustomerInfo AS
(
SELECT  CONCAT(P.FirstName, ' ', P.LastName) AS Name , P.CustomerID
FROM Persons AS P
UNION ALL
SELECT Name , B.CustomerID
FROM Business AS B
)
SELECT CustomerInfo.Name, CONCAT(SW.FirstName, ' ', SW.LastName) AS ServiceWriterName,
R.StartDate, R.EndDate, R.Laborcharge, IFNULL(PA.PartCost, 0) AS PartCost,
(IFNULL(PA.PartCost, 0) + R.LaborCharge) AS TotalCostTotalCost
FROM RepairEvents AS R
LEFT JOIN
(SELECT Vin, StartDate, SUM(QuantityUsed * Price) AS PartCost
FROM Parts GROUP BY Vin, StartDate )  AS PA
ON R.Vin = PA.Vin AND R. StartDate = PA.StartDate
INNER JOIN PrIviledgedUsers AS SW ON SW.Username = R.Username
INNER JOIN CustomerInfo ON R.CustomerID = CustomerInfo.CustomerID
WHERE R.Vin = '{a}';
```

Add Vehicle

Table 4

```
-- Get manufacturer dropdown and color dropdown are the same as search

-- Add vehicle
INSERT INTO Vehicles (Vin, Manufacturer, ModelName, ModelYear, DateAdded, InvoicePrice,
Description,
ClerkUsername) VALUES ('{a}', '{b}', '{c}', '{d}', (SELECT CURDATE()),'{e}', %s, '{f}');

--Add color
INSERT INTO VehicleColors (Vin, Colors) VALUES (%s, %s);

--Add car
INSERT INTO Cars (Vin, NumOfDoors) VALUES (%s, %s);

--Add SUV
INSERT INTO SUVs (Vin, DrivetrainType, NumberOfCupHolders) VALUES (%s, %s, %s);

---Add truck
INSERT INTO Trucks (Vin, CargoCoverType, NumberOfRearAxies, CargoCapacity) VALUES
(%s, %s, %s, %s);

--Add van
INSERT INTO VanMiniVans (Vin, HasDriverSideBackDoor) VALUES (%s, %s);

--Add convertible
INSERT INTO Convertibles (Vin, RoofType, BackSeatCount) VALUES (%s, %s, %s);

--Delete vehicle tuple when type attributes get error
delete from vehicles where vin = %s;

--Delete vehicle color tuple when type attributes get error
delete from vehiclecolors where vin = %s;
```

Search/Add Customer

Table 5

```
-- Search Customer
SELECT Name, Id, ContactName, ContactTitle, Address, PhoneNum, Email, CustomerID
            FROM
            (
            SELECT  CONCAT(P.FirstName, ' ', P.LastName) AS Name, P.License AS Id, C.CustomerID,
            '' AS ContactName, '' AS ContactTitle,C.PhoneNum, IFNULL(C.Email, '') AS EMAIL,
            CONCAT(C.Street, ', ', C.City, ', ', C.State, ', ', C.Zipcode) AS Address,
            'Person' AS CustomerType
            FROM Persons AS P
            INNER JOIN Customers AS C ON P.CustomerID = C.CustomerID
            UNION ALL
            SELECT Name, TaxNum AS Id, C.CustomerID,
            CONCAT(ContactFName, ' ', ContactLName) AS ContactName, ContactTitle, C.PhoneNum,
                    IFNULL(C.Email, '') AS EMAIL,
            CONCAT(C.Street, ', ', C.City, ', ', C.State, ', ', C.Zipcode) AS Address,
            'Business' AS CustomerType
            FROM Business AS B
            INNER JOIN Customers AS C ON B.CustomerID = C.CustomerID
            ) AS U
            WHERE CustomerType =  %s AND Id = %s;

--Add customer, general
INSERT INTO Customers (Street, City, State,
ZipCode, Email, PhoneNum) VALUES (%s,
%s, %s, %s, %s, %s);

--Add business
INSERT INTO Business (TaxNum, Name, ContactFName,
ContactLName, ContactTitle, CustomerID) VALUES
(%s, %s, %s, %s, %s, %s);

--Add person
INSERT INTO Persons (License, FirstName,
LastName, CustomerID) VALUES (%s, %s, %s, %s);


--Delete customer if person/business information gets error
DELETE FROM Customers WHERE CustomerID = '{a}';
```

Add Sales

Table 6

```
-- Check price
ELECT 0.95 * %s > InvoicePrice FROM Vehicles WHERE Vin = %s;

-- Add sale
INSERT INTO SalesEvents (Vin, SaleDate, SalePrice,
CustomerID, Username) VALUES (%s, %s,
%s, %s, %s);
```

View/Search/Add Repair

Table 7

```
--Check if the vehicle is a sold vehicle
SELECT (%s IN (SELECT Vin FROM SalesEvents) );

-- View vehicle information
With TypeInfo AS (
SELECT Vin, 'Car' AS Type FROM Cars
UNION SELECT Vin, 'SUV' AS Type FROM SUVs
UNION SELECT Vin, 'Van MiniVan' AS Type FROM VanMiniVans
UNION   SELECT Vin, 'Truck' AS Type FROM Trucks
UNION SELECT Vin, 'Convertible' AS Type FROM Convertibles
)
SELECT Vehicles.Vin, T.Type, ModelYear,  Manufacturer, ModelName AS Model,
C.VColors AS Colors
FROM Vehicles
JOIN (SELECT Vin, GROUP_CONCAT(Colors SEPARATOR ' ') AS VColors
FROM VehicleColors GROUP BY Vin
) AS C ON C.Vin = Vehicles.Vin
JOIN TypeInfo AS T ON T.Vin =Vehicles.Vin
WHERE  Vehicles.Vin = %s

-- Check if there is an open repair
SELECT %s IN (SELECT Vin FROM RepairEvents WHERE EndDate IS NULL);

-- View open repair
SELECT R.Vin, R.StartDate, R.Odometer, R.LaborCharge, IFNULL(PA.PartCost, 0) AS PartCost,
(IFNULL(PA.PartCost, 0) + LaborCharge) AS TotalCost, Description
FROM RepairEvents AS R
LEFT JOIN (SELECT Vin, StartDate, SUM(QuantityUsed * Price) AS PartCost
FROM Parts GROUP BY Vin, StartDate)  AS PA
ON R.Vin = PA.Vin AND R.StartDate = PA.StartDate
WHERE R.Vin = %s AND EndDate IS NULL;

-- Add new repair
INSERT INTO RepairEvents (Vin, StartDate, EndDate, LaborCharge,
Odometer, Description, CustomerID, Username) VALUES
(%s, (SELECT CURDATE()), NULL, '0.00', %s, NULL, %s, %s);

-- Update description
UPDATE RepairEvents SET Description = %s WHERE Vin = %s AND StartDate = %s;

-- Update labor charge
UPDATE RepairEvents SET LaborCharge =  %s WHERE Vin = %s AND StartDate = %s;

-- Add parts
INSERT INTO Parts (Vin, StartDate, Price, Number, QuantityUsed,
```

```
VendorName) VALUES (%s, %s, %s, %s, %s, %s);

-- Update parts
UPDATE Parts SET QuantityUsed = %s
WHERE Vin = %s AND StartDate = %s AND Number = %s;

-- Close repair
UPDATE RepairEvents SET EndDate = (SELECT CURDATE())
WHERE Vin = %s AND StartDate = %s;
```

Sales by Color Report

Table 8

```
With SingleColorSoldVehicles AS
        (
        SELECT SalesEvents.Vin, SaleDate, VehicleColors.Colors AS Color
        FROM SalesEvents
        INNER JOIN VehicleColors ON VehicleColors.Vin = SalesEvents.Vin
        WHERE SalesEvents.Vin IN
        (SELECT VIN FROM
        (SELECT Vin, Count(*) AS ColorCount
        FROM VehicleColors
        GROUP BY Vin HAVING ColorCount = 1) AS SingleColorVin
        )
        ),
        MultiColorSoldVehicles AS
        (
        SELECT Vin, SaleDate FROM SalesEvents
        WHERE Vin NOT IN
        (SELECT VIN FROM SingleColorSoldVehicles)
        ),
        SingleColorVehicleMonthSalesByColor AS
        (
                SELECT Color, Count(*) AS SaleCount FROM SingleColorSoldVehicles
                WHERE  DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents) , SaleDate)  <= 30
                GROUP BY Color
        ),
        SingleColorVehicleYearSalesByColor AS
        (
        SELECT Color, Count(*) AS SaleCount FROM SingleColorSoldVehicles
                WHERE DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents)  , SaleDate)  <= 365
                GROUP BY Color
        ),
        SingleColorVehicleAllTimeSalesByColor AS
        (
        SELECT Color, Count(*) AS SaleCount FROM SingleColorSoldVehicles
                GROUP BY Color
        )
        SELECT DISTINCT C.Color,
        IFNULL (M.SaleCount,0) AS MonthlySales,
        IFNULL (Y.SaleCount,0) AS YearSales,
        IFNULL (A.SaleCount,0) AS AllTimeSales
        FROM Colors AS C
        LEFT OUTER JOIN SingleColorVehicleMonthSalesByColor AS M
        ON C.Color = M.Color
        LEFT OUTER JOIN SingleColorVehicleYearSalesByColor AS Y
        ON C.Color = Y.Color
        LEFT OUTER JOIN SingleColorVehicleAllTimeSalesByColor AS A
```

```
        ON C.Color = A.Color
        UNION ALL
        SELECT 'Multiple' AS Color,
        (SELECT COUNT(*) FROM MultiColorSoldVehicles
        WHERE  DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents) , SaleDate)  <= 30)  AS
MonthlySales,
        (SELECT COUNT(*) FROM MultiColorSoldVehicles
        WHERE DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents) , SaleDate)  <= 365)  AS
YearSales,
        (SELECT COUNT(*) FROM MultiColorSoldVehicles)  AS AllTimeSales
        ORDER BY Color;
```

Sales by Type Report

Table 9

```
With TypeInfo AS (
        SELECT Vin, 'Car' AS Type FROM Cars
UNION SELECT Vin, 'SUV' AS Type FROM SUVs
UNION SELECT Vin, 'Van MiniVan' AS Type FROM VanMiniVans
UNION   SELECT Vin, 'Truck' AS Type FROM Trucks
UNION SELECT Vin, 'Convertible' AS Type FROM Convertibles
),
VehiclesWithType AS
(
SELECT Vs.Vin, T.Type FROM Vehicles AS Vs
JOIN TypeInfo AS T ON T.Vin =Vs.Vin
),
TypeSoldVehicles AS
(
SELECT SalesEvents.Vin, SaleDate, VT.Type FROM SalesEvents
INNER JOIN VehiclesWithType AS VT ON SalesEvents .Vin = VT.Vin
),
VehicleMonthSalesByType AS
(
        SELECT Type, Count(*) AS SaleCount FROM TypeSoldVehicles
        WHERE DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents) , SaleDate)  <= 30
        GROUP BY Type
),
VehicleYearSalesByType AS
(
SELECT Type, Count(*) AS SaleCount FROM TypeSoldVehicles
        WHERE DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents) , SaleDate)  <= 365
        GROUP BY Type
),
VehicleAllTimeSalesByType AS
(
SELECT Type, Count(*) AS SaleCount FROM TypeSoldVehicles
        GROUP BY Type
),
AllTypes AS (
SELECT 'Car' AS Type UNION SELECT 'SUV' UNION SELECT 'Truck'
UNION SELECT 'Convertible' UNION SELECT 'Van MiniVan'
)
SELECT T.Type,
  IFNULL (M.SaleCount,0) AS MonthlySales,
  IFNULL (Y.SaleCount,0) AS YearSales,
  IFNULL (A.SaleCount,0) AS AllTimeSales
FROM AllTypes AS T
LEFT OUTER JOIN VehicleMonthSalesByType AS M ON T.Type = M.Type
LEFT OUTER JOIN VehicleYearSalesByType AS Y ON T.Type = Y.Type
```

```
LEFT OUTER JOIN VehicleAllTimeSalesByType AS A ON T.Type = A.Type
ORDER BY Type;
```

Sales by Manufacturer Report

Table 10

```
With ManuSoldVehicles AS
(
SELECT SalesEvents.Vin, SaleDate, Vehicles.Manufacturer FROM SalesEvents
INNER JOIN Vehicles ON Vehicles.Vin = SalesEvents.Vin
),
VehicleMonthSalesByManu AS
(
        SELECT Manufacturer, Count(*) AS SaleCount FROM ManuSoldVehicles
        WHERE DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents) , SaleDate)  <= 30
        GROUP BY Manufacturer
),
VehicleYearSalesByManu AS
(
SELECT Manufacturer, Count(*) AS SaleCount FROM ManuSoldVehicles
        WHERE DATEDIFF((SELECT MAX(SaleDate) FROM SalesEvents) , SaleDate)  <= 365
        GROUP BY Manufacturer
),
VehicleAllTimeSalesByManu AS
(
SELECT Manufacturer, Count(*) AS SaleCount FROM ManuSoldVehicles
        GROUP BY Manufacturer
)
SELECT ManufacturerName,
IFNULL (M.SaleCount,0) AS MonthlySales,
IFNULL (Y.SaleCount,0) AS YearSales,
IFNULL (A.SaleCount,0) AS AllTimeSales
FROM Manufacturer AS Manu
LEFT OUTER JOIN VehicleMonthSalesByManu AS M
ON Manu.ManufacturerName = M.Manufacturer
LEFT OUTER JOIN VehicleYearSalesByManu AS Y
ON Manu.ManufacturerName = Y.Manufacturer
LEFT OUTER JOIN VehicleAllTimeSalesByManu AS A
 ON Manu.ManufacturerName = A.Manufacturer
WHERE M.SaleCount IS NOT NULL  OR A.SaleCount IS NOT NULL   OR Y.SaleCount IS NOT NULL
ORDER BY ManufacturerName;
```

Gross Customer Income Report

Table 11

```
-- Main report
WITH CustomerInfo AS
(
SELECT CustomerID, CONCAT(FirstName, ' ', LastName) AS Name FROM Persons AS P
UNION ALL
SELECT CustomerID, Name FROM Business
),
SalesInfo AS
(
SELECT CustomerID, SUM(SalePrice) AS SalesIncome, Count(*) AS SalesNumber,
MAX(SaleDate) AS LastSalesDate, MIN(SaleDate) AS FirstSalesDate
FROM SalesEvents
GROUP BY CustomerID
),
RepairInfo AS
(
SELECT CustomerID, SUM(LaborCharge) + SUM(IFNULL(PA.PartCost,0) ) AS RepairIncome,
Count(*) AS RepairNumber, MAX(RepairEvents.StartDate) AS LastRepairDate,
MIN(RepairEvents.StartDate) AS FirstRepairDate
FROM RepairEvents
LEFT JOIN
(SELECT Vin, StartDate, SUM(QuantityUsed * Price) AS PartCost
FROM Parts GROUP BY Vin, StartDate)  AS PA
ON RepairEvents.Vin = PA.Vin AND RepairEvents. StartDate = PA.StartDate
GROUP BY CustomerID
)
SELECT C.CustomerID, C.Name,
CASE WHEN R.FirstRepairDate < S.FirstSalesDate THEN R.FirstRepairDate ELSE S.FirstSalesDate END
AS FirstDate, CASE WHEN R.LastRepairDate >  S.LastSalesDate THEN R.LastRepairDate ELSE
S.LastSalesDate END AS LastDate,
IFNULL(S.SalesNumber,0) AS SalesNumber, IFNULL(R.RepairNumber,0) AS RepairNumber,
IFNULL(R.RepairIncome,0) + IFNULL(S.SalesIncome,0) AS TotalIncome
FROM CustomerInfo AS C
LEFT OUTER JOIN SalesInfo AS S ON C.CustomerID = S.CustomerID
LEFT OUTER JOIN RepairInfo AS R ON C.CustomerID =   R.CustomerID
ORDER BY TotalIncome DESC, LastDate DESC
LIMIT 15;

-- More detail, sale section
SELECT  SaleDate, SalePrice, V.Vin, V.ModelYear AS YEAR, V.Manufacturer,
V.ModelName AS Model,  CONCAT(PU.FirstName, ' ', PU.LastName) AS SalespersonName
FROM SalesEvents
LEFT JOIN Vehicles AS V ON V.Vin = SalesEvents.Vin
LEFT JOIN PriviledgedUsers AS PU ON PU.Username = SalesEvents.Username
WHERE SalesEvents.CustomerID = %s
```

```
ORDER BY SaleDate DESC, Vin ASC;

-- More detail, repair section
SELECT RepairEvents.StartDate, IFNULL(EndDate, '') AS EndDate1, RepairEvents.Vin,
Odometer, LaborCharge, IFNULL(PA.PartCost,0) AS PartCost,
(IFNULL(PA.PartCost,0) + LaborCharge) AS TotalCharge,
CONCAT(PU.FirstName, ' ', PU.LastName) AS ServiceWriterName
FROM RepairEvents
LEFT JOIN
(SELECT Vin, StartDate, SUM(QuantityUsed * Price) AS PartCost
FROM Parts GROUP BY Vin, StartDate)  AS PA
ON RepairEvents.Vin = PA.Vin AND RepairEvents. StartDate = PA.StartDate
LEFT JOIN PriviledgedUsers AS PU ON PU.Username = RepairEvents.Username
WHERE RepairEvents.CustomerID = %s
ORDER BY StartDate DESC, EndDate1 != '', EndDate DESC, PA.Vin ASC;
```

Repairs by Manufacturer/Type/Model Report

Table 12

```
-- Main report
WITH RepairInfo AS
(
SELECT RepairEvents.Vin, RepairEvents.StartDate, V.Manufacturer, LaborCharge,
IFNULL(PA.PartCost, 0 ) AS PartCost,
(IFNULL(PA.PartCost, 0) + LaborCharge ) AS PartLaborCost
FROM RepairEvents
LEFT JOIN
(SELECT Vin, StartDate, SUM(QuantityUsed * Price) AS PartCost
FROM Parts GROUP BY Vin, StartDate)  AS PA
ON RepairEvents.Vin = PA.Vin AND RepairEvents. StartDate = PA.StartDate
LEFT JOIN Vehicles AS V ON RepairEvents.Vin = V.Vin
)
SELECT M.ManufacturerName, IFNULL(RR.TotalRepairCount, 0),
IFNULL(RR.TotalLaborCost, 0), IFNULL(RR.TotalPartCost, 0),
IFNULL(RR.TotalLaborPartCost, 0)
FROM Manufacturer AS M
LEFT OUTER JOIN
(SELECT Manufacturer, Count(*) AS TotalRepairCount,
SUM(LaborCharge) AS TotalLaborCost, SUM(PartCost) AS TotalPartCost,
SUM(PartLaborCost) AS TotalLaborPartCost
FROM RepairInfo
GROUP BY Manufacturer) AS RR
ON M.ManufacturerName =  RR.Manufacturer
ORDER BY ManufacturerName;

-- More detial
With TypeInfo AS (
          SELECT Vin, 'Car' AS Type FROM Cars
UNION SELECT Vin, 'SUV' AS Type FROM SUVs
UNION SELECT Vin, 'Van MiniVan' AS Type FROM VanMiniVans
UNION   SELECT Vin, 'Truck' AS Type FROM Trucks
UNION SELECT Vin, 'Convertible' AS Type FROM Convertibles
),
VehiclesWithType AS
(
SELECT Vs.Vin, ModelName, ModelYear, DateAdded, InvoicePrice,Manufacturer,
ClerkUsername, Description , T.Type FROM Vehicles AS Vs
JOIN TypeInfo AS T ON T.Vin =Vs.Vin
),
RepairInfo AS
(
SELECT RepairEvents.Vin, RepairEvents.StartDate, V.Manufacturer, LaborCharge,
IFNULL(PA.PartCost, 0) AS PartCost,
(IFNULL(PA.PartCost, 0) + LaborCharge) AS PartLaborCost, ModelName AS Model, Type
```

```
FROM RepairEvents
LEFT JOIN
(SELECT Vin, StartDate, SUM(QuantityUsed * Price) AS PartCost
FROM Parts GROUP BY Vin, StartDate)  AS PA
ON RepairEvents.Vin = PA.Vin AND RepairEvents. StartDate = PA.StartDate
LEFT JOIN VehiclesWithType  AS V ON RepairEvents.Vin = V.Vin
),
TypeCount AS(
SELECT Type, '' AS Model, Count(*) AS TotalRepairCount,
SUM(LaborCharge) AS TotalLaborCost, SUM(PartCost) AS TotalPartCost,
SUM(PartLaborCost) AS TotalPartLaborCost, Count(*) AS TypeRepairCount
FROM RepairInfo
WHERE Manufacturer = %s
GROUP BY Type),
ModelCount AS(
SELECT Type, Model, Count(*) AS TotalRepairCount,
SUM(LaborCharge) AS TotalLaborCost, SUM(PartCost) AS TotalPartCost,
SUM(PartLaborCost) AS TotalPartLaborCost
FROM RepairInfo
WHERE Manufacturer = %s
GROUP BY Type, Model
),
ModelTypeCount AS (
SELECT M.Type, M.Model, M.TotalRepairCount, M.TotalLaborCost, M.TotalPartCost,
M.TotalPartLaborCost, T.TypeRepairCount
FROM ModelCount AS M
LEFT JOIN TypeCount AS T ON T.Type = M.Type
),
SortTable AS (
SELECT Type, Model, TotalRepairCount,TotalLaborCost,
TotalPartCost, TotalPartLaborCost, TypeRepairCount FROM ModelTypeCount
UNION ALL
SELECT Type, Model, TotalRepairCount,TotalLaborCost,
TotalPartCost, TotalPartLaborCost, TypeRepairCount FROM TypeCount
ORDER BY TypeRepairCount DESC, Type, Model !='', TotalRepairCount DESC
)
SELECT Type, Model, TotalRepairCount,TotalLaborCost,
TotalPartCost, TotalPartLaborCost FROM SortTable;
```

Below Cost Sales Report

Table 13

```
SELECT S.SaleDate, S.SalePrice, V.InvoicePrice,
	CONCAT(ROUND((S.SalePrice / V.InvoicePrice )*100,2), '%') AS PriceRatio,
	C.Name AS CustomerName , CONCAT(P.FirstName, ' ', P.LastName) AS SalespersonName
FROM SalesEvents AS S
INNER JOIN Vehicles AS V ON V.Vin = S.Vin
INNER JOIN
(
	SELECT CustomerID, CONCAT(FirstName, ' ', LastName) AS Name FROM Persons
UNION ALL
SELECT CustomerID, Name FROM Business
) AS C ON C.CustomerID = S.CustomerID
INNER JOIN PriviledgedUsers AS P ON P.Username = S.Username
WHERE S.SalePrice < V.InvoicePrice
ORDER BY S.SaleDate DESC, PriceRatio DESC;
```

Average Time in Inventory Report

Table 14

```
With AllTypes AS (
SELECT 'Car' AS Type UNION SELECT 'SUV' UNION SELECT 'Truck'
UNION SELECT 'Convertible' UNION SELECT 'Van MiniVan'
),
TypeInfo AS (
            SELECT Vin, 'Car' AS Type FROM Cars
UNION SELECT Vin, 'SUV' AS Type FROM SUVs
UNION SELECT Vin, 'Van MiniVan' AS Type FROM VanMiniVans
UNION   SELECT Vin, 'Truck' AS Type FROM Trucks
UNION SELECT Vin, 'Convertible' AS Type FROM Convertibles
)
SELECT AllTypes.Type, IFNULL(A.AverageTime, 'N/A') AS AverageTimeInInventory
FROM AllTypes
LEFT OUTER JOIN
(SELECT T.Type, ROUND(AVG(DATEDIFF( DATE_ADD(S.SaleDate,INTERVAL 1 DAY), DateAdded) ),2) AS
AverageTime
FROM Vehicles
INNER JOIN SalesEvents AS S ON Vehicles.Vin = S.Vin
INNER JOIN TypeInfo AS T ON T.Vin =Vehicles.Vin
GROUP BY Type) AS A
ON AllTypes.Type = A.Type
ORDER BY Type ASC;
```

Parts Statistics Report

Table 15

```
SELECT VendorName,  SUM(QuantityUsed * Price) AS TotalCost,
SUM(QuantityUsed) AS NumberOfPart
FROM Parts GROUP BY VendorName
ORDER BY TotalCost DESC;
```

Monthly Sales Report

Table 16

```
-- Main Report
ELECT  Date_FORMAT(SaleDate, '%Y-%m') AS SaleYearMonth, COUNT(*) AS SaleCount,
SUM(S.SalePrice) AS SaleIncome, (SUM(S.SalePrice) - SUM(V.InvoicePrice) ) AS SaleNetIncome,
CONCAT(ROUND((SUM(S.SalePrice) / SUM(V.InvoicePrice) )*100,2), '%') AS SaleRatio
FROM SalesEvents AS S
INNER JOIN Vehicles AS V
ON V.Vin = S.Vin
GROUP BY SaleYearMonth
ORDER BY SaleYearMonth DESC;

-- More Detail
SELECT CONCAT(P.FirstName, ' ',P.LastName) AS SalespersonName, COUNT(*) AS SaleCount,
SUM(S.SalePrice) AS SalespersonIncome
FROM SalesEvents AS S
INNER JOIN PriviledgedUsers AS P
ON P.Username = S.Username
WHERE YEAR( S.SaleDate) = YEAR( CONCAT(%s,  '-01'))
AND MONTH( S.SaleDate) = MONTH(CONCAT(%s,  '-01'))
GROUP BY P.Username
ORDER BY SaleCount DESC, SalespersonIncome DESC;
```