Queries Output

**/\* 1. List the customers. For each customer, indicate which category he or she fall into, and his or her**

**contact information \*/**

select privateIndividual.cFirstName, privateIndividual.cLastName,privateIndividual.mailingAddress, 'Prospective Customer' as "Prospective Customer"

from Customer natural join privateIndividual

left outer join prospectiveCustomer using(customerID)

where prospectiveCustomer.pEMAIL is NOT NULL

union

select privateIndividual.cFirstName, privateIndividual.cLastName,privateIndividual.mailingAddress, 'Premier Customer' as "Premier Customer"

from Customer natural join privateIndividual

left outer join premierCustomer using(customerID)

where premierCustomer.moneySpent is NOT NULL

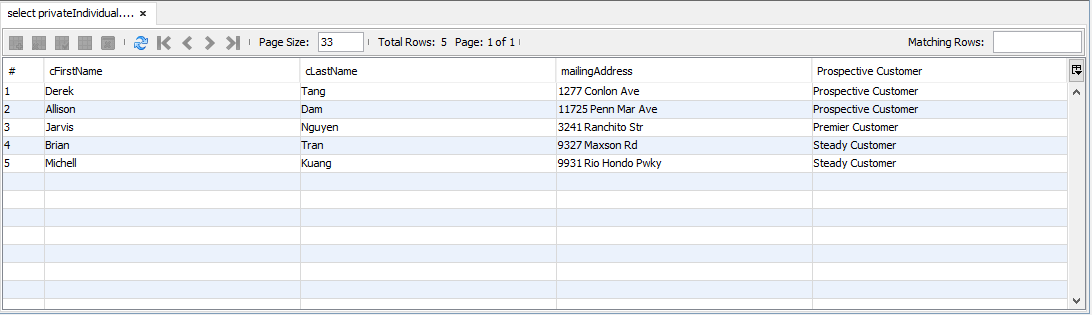
union

select privateIndividual.cFirstName, privateIndividual.cLastName,privateIndividual.mailingAddress, 'Steady Customer' as "Steady Customer"

from Customer natural join privateIndividual

left outer join steadyCustomer using(customerID)

where steadyCustomer.sEMAIL is NOT NULL;

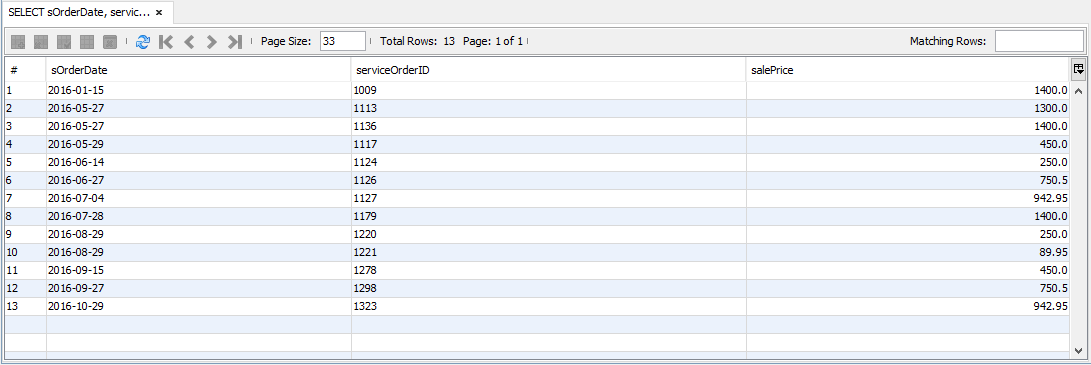


**/\* 2. For each service visit, list the total cost to the customer for that visit. \*/**

SELECT sOrderDate, serviceOrderID, salePrice FROM serviceOrder

INNER JOIN orderedService USING (serviceOrderID)

GROUP BY sOrderDate, serviceOrderID, salePrice;



**/\* 3. List the top three customers in terms of their net spending for the past two years, and the total that they have spent in that period. \*/**

**/\* 4. Find all of the mechanics who have three or more skills. \*/**

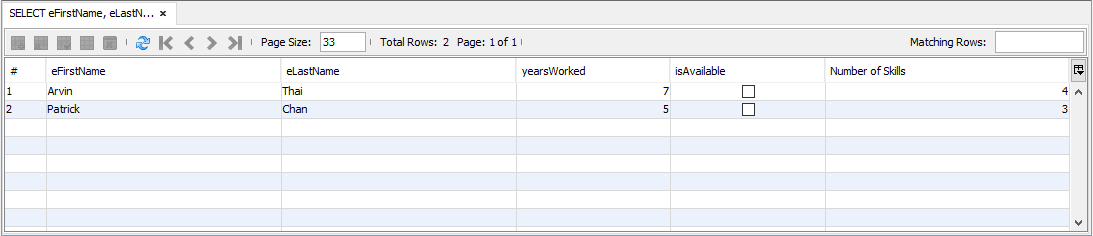
SELECT eFirstName, eLastName, yearsWorked, isAvailable, COUNT(certification.mechanicID) AS "Number of Skills" FROM mechanic

INNER JOIN certification ON mechanic.eID = certification.mechanicID

INNER JOIN employee ON employee.eID = mechanic.eID

GROUP BY eFirstName, eLastName, yearsWorked, isAvailable

HAVING COUNT(certification.mechanicID) > 2;



**/\* 5. Find all of the mechanics who have three or more skills in common. \*/**

SELECT DISTINCT mech.eID, mech.eFirstName, mech.eLastName FROM (

SELECT eID, eFirstName, eLastName, skillName, COUNT(certification.skillName) AS skillCount FROM mechanic

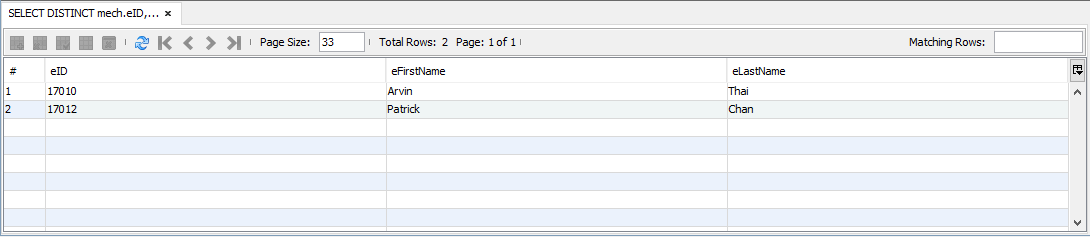
INNER JOIN employee USING (eID)

INNER JOIN certification ON certification.mechanicID = employee.eID

GROUP BY eID, eFirstName, eLastName

HAVING skillCount > 2) AS mech

INNER JOIN certification ON certification.mechanicID = mech.eID;



**/\* 6. For each maintenance package, list the total cost of the maintenance package, as well as a list of all of the maintenance items within that package. \*/**

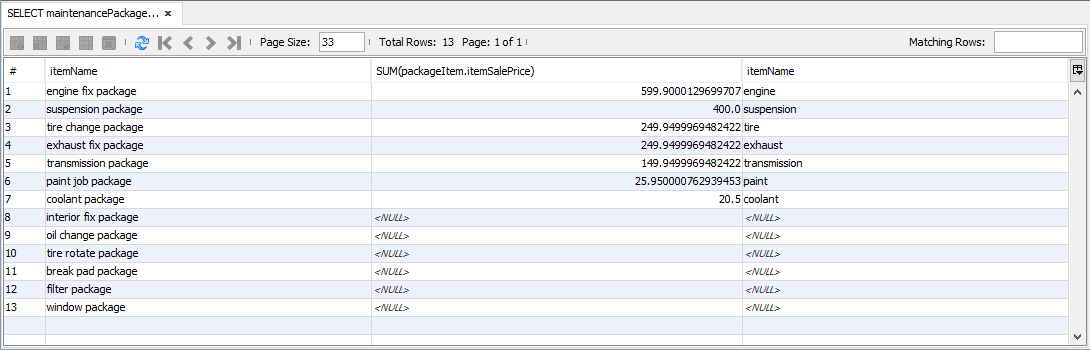
SELECT maintenancePackage.itemName, SUM(packageItem.itemSalePrice), packageItem.itemName FROM maintenancePackage

LEFT JOIN PackageItem

ON maintenancePackage.itemName = packageItem.packageName

GROUP BY maintenancePackage.itemName

ORDER BY packageItem.itemSalePrice DESC;



**/\* 7. Find all of those mechanics who have one or more maintenance items that they lacked one or**

**more of the necessary skills. \*/**

Select distinct eFirstName, eLastName

From employee

Inner Join mechanic

ON employee.eID = mechanic.eID

Inner Join orderedService

ON employee.eID = orderedService.mechanicID

Inner Join maintenanceItem

ON orderedService.itemName = maintenanceItem.itemName

Inner Join certification

ON certification.mechanicID = employee.eID

WHERE employee.eID NOT IN (Select distinct employee.eID

From employee

Inner Join mechanic

ON employee.eID = mechanic.eID

Inner Join orderedService

ON employee.eID = orderedService.mechanicID

Inner Join maintenanceItem

ON orderedService.itemName = maintenanceItem.itemName

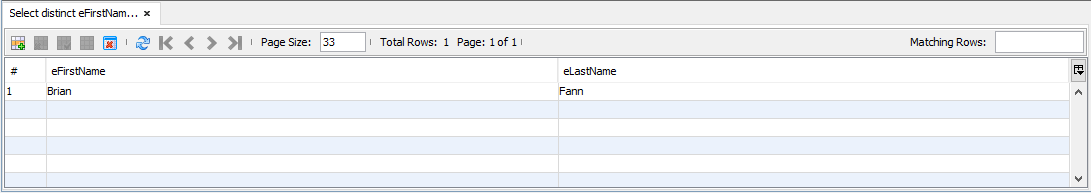
Inner Join certification

ON certification.mechanicID = employee.eID

Inner join maintenanceSkill

ON certification.skillName = maintenanceSkill.skillName

)



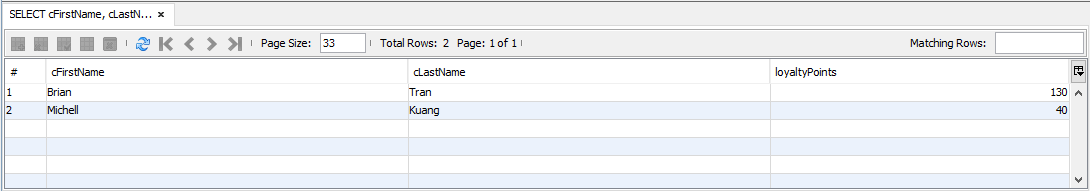
**/\* 8. List the customers, sorted by the number of loyalty points that they have, from largest to**

**smallest. \*/**

SELECT cFirstName, cLastName, loyaltyPoints FROM privateIndividual

INNER JOIN steadyCustomer USING (customerID)

ORDER BY loyaltyPoints DESC;



**/\* 9. The premier customers and the difference between what they have paid in the past year, versus the services that they actually used during that same time. List from the customers with then largest difference to the smallest. \*/**

**/\* 10. Report on the steady customers based on the net profit that we have made from them over the past year, and the dollar amount of that profit, in order from the greatest to the least. \*/**

select privateIndividual.cFirstName, privateIndividual.cLastName, sum(salePrice)

from steadyCustomer

inner join privateIndividual

ON steadyCustomer.customerID = privateIndividual.customerID

inner join vehicle

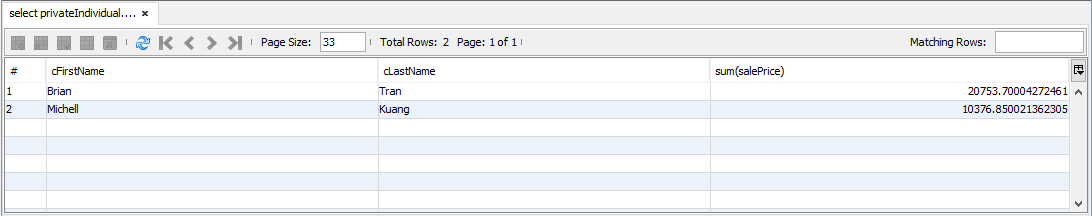
ON steadyCustomer.customerID = vehicle.customerID

inner join serviceOrder

ON vehicle.VIN = serviceOrder.VIN

inner join orderedService

group by steadyCustomer.customerID



**/\* 11. List the three suppliers who have supplied us the largest number of parts (not total quantity of parts, but the largest number of distinct parts) over the past year.**

Select supplier.sName, Count(partOrderLine.UPC) FROM supplier

Left Join partOrder

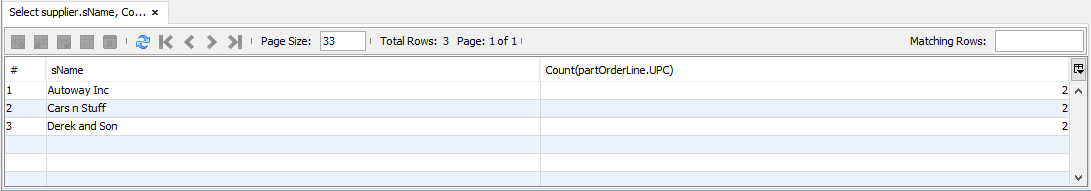
on supplier.sName = partOrder.supplierName

Left Join partOrderLine

on partOrder.orderID = partOrderLine.orderID

Group by supplier.sName

Limit 3;



/\* 12. List the five suppliers who have supplied us the largest dollar value of parts in the past year.\*/

Select sName, Sum(partPrice) FROM supplier

INNER JOIN partOrder

ON supplier.sName = partOrder.supplierName

INNER JOIN partOrderLine

ON partOrder.orderID = partOrderLine.orderID

INNER JOIN part

ON partOrderLine.UPeC = part.UPC

WHERE (Date(GETDATE() - partOrder.pOrderDate) < 365)

GROUP BY sName

**/\* 13. Find the mechanic who is mentoring the most other mechanics. List the skills that the mechanic is passing along to the other mechanics. \*/**

SELECT temp.eFirstName, temp.eLastName, GROUP\_CONCAT(skillName SEPARATOR ', ') AS "Skills being passed" FROM (

SELECT DISTINCT menteeCount.eID, menteeCount.eFirstName, menteeCount.eLastName, skillName FROM (

SELECT eID, eFirstName, eLastName, COUNT(mentorID) AS menteeCount FROM employee

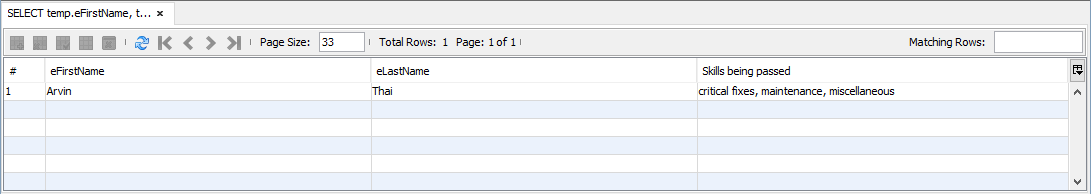
INNER JOIN mentorship ON employee.eID = mentorship.mentorID

GROUP BY eFirstName, eLastName

ORDER BY COUNT(mentorID) DESC

LIMIT 1) AS menteeCount

INNER JOIN mentorship ON mentorship.mentorID = menteeCount.eID) AS temp;



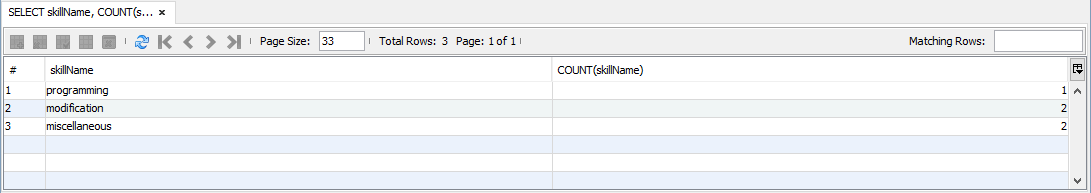
**/\* 14. Find the three skills that have the fewest mechanics who have those skills. \*/**

SELECT skillName, COUNT(skillName) FROM certification

GROUP BY skillName

ORDER BY COUNT(skillName) ASC

LIMIT 3;



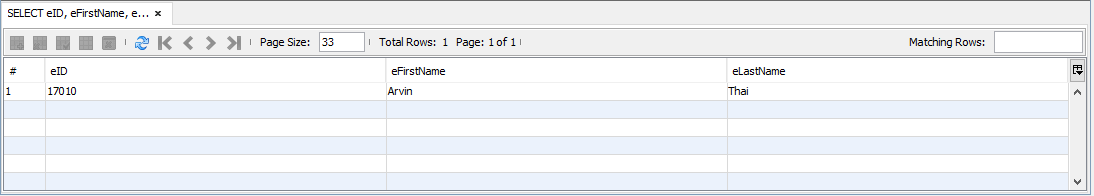
**/\* 15. List the employees who are both service technicians as well as mechanics. \*/**

SELECT eID, eFirstName, eLastName FROM mechanic

INNER JOIN serviceTechnician USING (eID)

INNER JOIN employee USING (eID)

GROUP BY eID, eFirstName, eLastName;



/\* 16. Three additional queries that demonstrate the five additional business rules. Feel free to create

additional views to support these queries if you so desire. \*/