Team Members

_____

_____

_____

_____

_____

The following is a check-list for all the required documentation

1. A revised design, based on feedback from the first part. This includes class diagrams and English descriptions of classes and associations.
2. English description of all attributes.
3. The relation scheme, based on the design.
4. DDL used to create all the tables
5. DML used to insert the data
6. Queries to produce the reports, as described below
7. Sample output of each of the queries

Views

1. Customer_v – for each customer, indicate his or her name as well as the customer type (prospect, steady or premier) as well as the number of years that customer has been with us.

2. Customer_addresses_v – for each customer, indicate whether they are an individual or a corporate account, and display all of the addresses that we are managing for that customer.

3. Mechanic_mentor_v – reports all of the mentor/mentee relationships at Dave's, sorted by the name of the mentor, then the name of the mentee.

4. Premier_profits_v – On a year by year basis, show the premier customer's outlay versus what they would have been charged for the services which they received had they merely been steady customers.

5. Prospective_resurrection_v – List all of the prospective customers who have had three or more contacts, and for whom the most recent contact was more than a year ago. They might be ripe for another attempt.

Queries

1. List the customers. For each customer, indicate which category he or she fall into, and his or her contact information. If you have more than one independent categorization of customers, please indicate which category the customer falls into for all of the categorizations.

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

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.

5. Find all of the mechanics who have three or more skills **in common**.
    a. Please give the name of each of the two mechanics sharing 3 or more skills.
    b. Please make sure that any given pair of mechanics only shows up once.

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.

7. Find all of those mechanics who have one or more maintenance items that they lacked one or more of the necessary skills.

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

9. List 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 the 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.

11. List the three premier customers who have paid Dave's Automotive the greatest amount in the past year, and the sum of their payments over that period. Be sure to take into account any discounts that they have earned by referring prospective customers.

12. List the five model, make, and year that have caused the most visits on average to Dave's automotive **per vehicle** in the past three years, along with the average number of visits per vehicle.

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

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

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

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.

    a.

    b.

    c.

CECS 323 Final Project Part 1

I. Business Rule

- A current customer referral benefits will be applied to his or her account continuously until those benefits run out.
- An employee can either be a technician or a mechanics but not both.
- Technician must have knowledge of all services provided at Dave's Automotive.
- Technician only need to write services which are not part of vehicle routine services.
- A mastery level must be an integer between 1 to 10 inclusive where 1 is the minimum mastery level and 10 is the maximum mastery level of the mechanic's skill.

II. Class Definition in English

- **Customer**: people who require or have a possible of requiring service at Dave's Automotive.
- **Current**: customers who come to Dave's Automotive for vehicle service.
- **Prospective**: customers who have potential of requiring vehicle service from Dave's Automotive.
- **PromotionContact**: information associated with date, time, and mode of communication in which Dave's Automotive attempt to entice a prospective customer into a current customer.
- **Premier**: current customers who pay monthly (buy insurance) for service from Dave's Automotive.
- **Steady**: current customers who pay per visit for service from Dave's Automotive.
- **Address**: location or locations in which current customers associate with.
- **PriceIncrease**: the amount and cause of an increase in monthly payment of premier customers.
- **Vehicle**: a machine who purpose to carry people around from place to place at high speed.
- **VehicleFamily**: contains general information about a vehicle.
- **MaintainOrder**: services that require to be perform on a specific vehicle.
- **Employee**: people who work at Dave's Automotive.
- **Technician**: employees of Dave's Automotive who purpose is to diagnose and write extra services required by a vehicle.
- **Mechanics**: employees of Dave's Automotive who purpose are to do maintain or repair of a vehicle.
- **MaintainPackage**: a collection of service.
- **MaintainPackageLine**: shows relationship between MaintainPackage and MaintainOrder.
- **Skillset**: abilities of mechanics to do a specific job.
- **SkillsetLine**: keep track of mystery level of a specific skill for a specific customer.
- **MaintainItem**: a specific service that has been assigned to a specific mechanic.

- **TraniningSkill**: show a history of training relationship between mechanics.
- **ZipLocation**: a place in United States where postal service has assigned a specific identification code.
- **JobQueLine:** show a relationship between maintain item and mechanics and it keeps track of date in which a mechanic work on a specific item.
- **ItemWork:** shows relationship between MaintainOrder and MaintainItem.
- **ReferralBenefitHistory:** Keep tracks of referral benefits of a current customer.
- **Appointment:** tracks an appointment of a vehicle
- **AppointmentStatus:** shows the available status for each appointment.
- **CommunicationType:** shows the available communication type for each promotion contact.
- **MasteryLevel:** shows the available mastery level for each skillsetline.

III. Association:

- **Customers:**
  - A customer is a current customer, a perspective customer, or others but he or she can only be one.
  - A perspective customer is a customer.
  - A current customer is a customer.
- **Prospective:**
  - A perspective customer received one to three promotion contact.
  - A promotion contact was received by one and only one perspective customer.
- **Current**:
  - A current customer related to one to many address.
  - An address is related to one and only one current customer.
  - A current customer is either a premier customer or a steady customer but not both.
  - A premier customer is a current customer.
  - A steady customer is a current customer.
  - A current customer owned one to many vehicles.
  - A vehicle is owned by one and only one current customer.
  - A current customer has zero to many referral benefit histories.
  - A referral benefit history is belonged to one and only one current customer.
- **Premier:**
  - A premier customer associate with zero to many price increases.
  - A price increase is associated with one and only one premier customer.
- **Vehicle:**
  - A vehicle is part of one and only one vehicle family.
  - A vehicle family is made up of zero to many vehicles.

- A vehicle requires one to many maintain orders.
- A maintain order is required by one and only one vehicle.
- A vehicle has zero to many appointments.
- An appointment is belonged to one and only one vehicles

- **Maintain Order:**
  - A maintain order link to zero to many maintain package lines. (1)
  - A maintain package line linked to one and only one maintain order. (1)
  - A maintain order contains one to many item works. (2)
  - An item work contains within one and only one maintain order. (2)
- **Maintain Package:**
  - A maintain package link to zero to many maintain package lines. (1)
  - A maintain package line linked to one and only one maintain order. (1)
  - A maintain package composed of one to many maintain item.
  - A maintain item is composed within one and only one maintain package.
- **Employee:**
  - An employee is a technician, a mechanics, or others but he or she can only be one.
  - A technician is an employee.
  - A mechanics is an employee.
- **Technician:**
  - A technician writes one to many maintain orders.
  - A maintain order was written by one and only one technician.
- **Mechanic:**
  - A mechanic trained zero to many other mechanic.
  - A mechanic is trained by one and only one other mechanic.
  - A mechanic is associated with one to many skillset lines. (4)
  - A skillset line is associated with one and only one mechanic. (4)
  - A mechanic is responsible for zero to many job que line. (3)
  - A job que line is responsible by one and only one mechanic. (3)
- **Skillset:**
  - A skillset is associated with one to many skillset line. (4)
  - A skillset line is associated with one and only one skillset. (4)
- **Maintain Item:**
  - A maintain item is contain within zero to many job que line. (3)
  - A job que line contains one and only one maintain item. (3)
  - A maintain item contain within one to many item work. (2)
  - An item work contains one and only one maintain item. (2)
- **AppointmentStatus:**
  - An appointment status is associated with one to many appointments.
  - An appointment is associated with one and only one appointment status.

- **CommunicationType**
  - A communication type is associated with one to many promotion contacts.
  - A promotion contact is associated with one and only one communication type.
- **MasteryLevel**
  - A mastery level is associated with zero to many skillsetlines.
  - A skillsetline is associated with one and only one mastery level.

IV. Normalization:

- All classes are in third normalization form because we eliminated multivalve and repeated values. In addition, we also able to eliminate sub key through the use of lossless join decomposition especially for class **Address** and **Vehicle**.

V. Attribute Definition:

a. **Customer**:
   i. firstName: customer first name.
   ii. lastName: customer last name.
   iii. dateOfBirth: customer date of birth.
   iv. phoneNumber: customer phone number.
   v. email: customer email.

b. **Current**:
   i. JoinedDate: date in which a current customer became a member.

c. **Prospective**:
   i. ReferralName: name of a customer who refers this prospective customer.
   ii. DeadFlag: shows if Dave's automobile should try to contact them again for promotional purpose.

d. **PromotionContact**:
   i. Date: date of contact attempt.
   ii. Time: time of contact attempt.

e. **Premier**:
   i. AnnualFee: yearly membership fee.
   ii. DueDate: the next due date for monthly membership fee.
   iii. OriginalPrice: original membership fee before price increase.

f. **Steady**:
   i. LoyaltyPoint: loyalty point of a steady customer.

g. **Address**:
   i. Type: address type.
   ii. Address: street address.

h. **PriceIncrease**:
   i. PercentIncreased: the percent of price increase to membership fee of premier customers.

    ii. Reason: reason for price increase.

i. **Vehicle**:
    i. Vin: vehicle identification number.
    ii. Mileage: vehicle current mileage.
    iii. ExpectedMileageThisYear: mileage expect to put on vehicle this year.
    iv. MaintainInterval: the maintain mileage interval
    v. RoutineService: mandatory service packages that performed on vehicle every visit.
j. **VehicleFamily**:
    i. Model: model of a vehicle.
    ii. Year: year of a vehicle.
    iii. Maker: manufacturer of a vehicle.
k. **MaintainOrder**:
l. **Employee:**
    i. Name: name of employee.
    ii. Salary: hour rate paid.
    iii. HiredDate: hired date.
m. **Technician**:
n. **Mechanics**:
o. **MaintainPackage**:
    i. Name: name of maintain package.
    ii. Description: description of maintain package.
p. **MaintainPackageLine**:
q. **Skillset**:
    i. Name: name of skill set.
    ii. Description: description of skill set.
r. **SkillsetLine**:
s. **MaintainItem**:
    i. Name: name of maintain item.
    ii. Skill: skill name required to do the maintain item.
    iii. Cost: price we charged customer for performing the maintain item.
    iv. BuyInPrice: cost of performing the maintain item.
t. **TraniningSkill**:
    i. StartDate: start of mentoring relationship.
    ii. EndDate: end of mentoring relationship.
    iii. SkillTrained: skillset trained during the mentoring relationship.
u. **ZipLocation**:
    i. Zipcode: zip code of a zip location.
    ii. City: city of that zip location.
    iii. State: state of that zip location.
v. **JobQueLine:**

    i. DateOfWork: date in which a mechanic assigned to that maintain item.

w. **ItemWork:**
    i. Date: date in which the relationship between MaintainOrder and MaintainItem was established.

x. **ReferralBenefitHistory:**
    i. Benefit: benefit received
    ii. Date: date that benefit was granted.
    iii. Flag: if benefit already applied to account or not.

y. **Appointment:**
    i. Date: date of the appointment.
    ii. Time: time of the appointment.
    iii. ExpectedTime: expected time taken for that appointment.

z. **AppointmentStatus:**
    i. Status: the status of an appointment.

aa. **CommunicationType:**
    i. Type: type of communication.

bb. **MasteryLevel:** shows the available mastery level for each skillsetline.
    i. MasteryLevel: level available to a skillset.

```sql
--DDL Used to great table--
CREATE TABLE CommunicationType (
    cmType          VARCHAR(42),
    PRIMARY KEY (cmType)
);

CREATE TABLE Customer (
    cID             INT NOT NULL,
    cFirstName      VARCHAR(42),
    cLastName       VARCHAR(100),
    cDateOfBirth    DATE,
    cPhoneNumber    VARCHAR(15),
    cEmail          VARCHAR(47),
    PRIMARY KEY (cID),
    INDEX (cFirstName),
    INDEX (cLastName),
    INDEX (cEmail)
);

CREATE TABLE ReferralBenefitHistory (
    cID         INT NOT NULL,
    rBenefit    VARCHAR (50) NOT NULL,
    rDate       DATE NOT NULL,
    rFlag       VARCHAR (50),
    PRIMARY KEY (cID, rBenefit, rDate),
    FOREIGN KEY (cID) REFERENCES Customer (cID)
);
CREATE TABLE Prospective (
    cID             INT NOT NULL,
    pReferralName   VARCHAR(100),
    pDeadFlag       TINYINT,
    PRIMARY KEY (cID),
    FOREIGN KEY (cID) REFERENCES Customer (cID)
);

CREATE TABLE PromotionContact (
    cID     INT NOT NULL,
    pcDate  DATE NOT NULL,
    pcTime  TIME,
    cmType  VARCHAR(42),
    PRIMARY KEY (cID, pcDate),
    FOREIGN KEY (cID) REFERENCES Prospective (cID),
    FOREIGN KEY (cmType) REFERENCES CommunicationType (cmType)
);

CREATE TABLE Current (
    cID            INT NOT NULL,
    cJoinedDate    DATE,
    PRIMARY KEY (cID),
    FOREIGN KEY (cID) REFERENCES Customer (cID)
);

CREATE TABLE ZIPLocation (
    zZIPCode            VARCHAR(10) NOT NULL,
    zCity               VARCHAR(100) NOT NULL,
    zState              VARCHAR(42) NOT NULL,
    PRIMARY KEY (zZIPCode)
);

CREATE TABLE Address (
    cID         INT NOT NULL,
    aType       VARCHAR(42),
    zZIPCode    VARCHAR(10),
    aAddress    VARCHAR(100),
    PRIMARY KEY (cID, aType),
    FOREIGN KEY (cID) REFERENCES Current (cID),
```

```
    FOREIGN KEY (zZIPCode) REFERENCES ZIPLocation(zZIPCode)
);

CREATE TABLE Steady (
    cID             INT NOT NULL,
    sLoyaltyPoints  INT,
    PRIMARY KEY (cID),
    FOREIGN KEY (cID) REFERENCES Current (cID)
);

CREATE TABLE Premier (
    cID         INT NOT NULL,
    pAnnualFee  DECIMAL(10,2) UNSIGNED,
    PRIMARY KEY (cID),
    FOREIGN KEY (cID) REFERENCES Current (cID)
);

CREATE TABLE PriceIncrease (
    cID                 INT NOT NULL,
    piPercentIncrease   DOUBLE,
    piReason            VARCHAR(1701),
    PRIMARY KEY (cID),
    FOREIGN KEY (cID) REFERENCES Premier (cID)
);

CREATE TABLE VehicleFamily (
    vfID    INT NOT NULL,
    vfMake  VARCHAR(42),
    vfModel VARCHAR(42),
    vfYear  YEAR,
    PRIMARY KEY (vfID),
    INDEX (vfMake),
    INDEX (vfModel)
);
CREATE TABLE MaintainPackage (
    mpID INT NOT NULL,
    mpName VARCHAR(128) NOT NULL,
    mpDescription VARCHAR(1337),
    PRIMARY KEY (mpID)
);
CREATE TABLE Vehicle (
    vVIN                        CHAR(17) NOT NULL,
    vMileage                    MEDIUMINT UNSIGNED NOT NULL,
    cID                         INT NOT NULL,
    vExpectedMileageThisYear    MEDIUMINT UNSIGNED,
    vRoutineServices            INT(100) NOT NULL,
    vfID                        INT NOT NULL,
    PRIMARY KEY (vVIN),
    FOREIGN KEY (cID) REFERENCES Current (cID),
    FOREIGN KEY (vfID) REFERENCES VehicleFamily (vfID),
    FOREIGN KEY (vRoutineServices) REFERENCES MaintainPackage (mpID)
);

CREATE TABLE Employee (
    eID         INT NOT NULL,
    eName       VARCHAR(420),
    eSalary     DECIMAL(10,2) UNSIGNED,
    eHiredDate  DATE,
    PRIMARY KEY (eID)
);

CREATE TABLE Technician (
    eID INT NOT NULL,
    PRIMARY KEY (eID),
    FOREIGN KEY (eID) REFERENCES Employee (eID)
);
```

```sql

CREATE TABLE Mechanic (
    eID INT NOT NULL,
    PRIMARY KEY (eID),
    FOREIGN KEY (eID) REFERENCES Employee (eID)
);

CREATE TABLE TrainingSkill (
    tsTrainerID     INT NOT NULL,
    tsTraineeID     INT NOT NULL,
    tsStartDate     DATE,
    tsEndDate       DATE,
    tsSkillTrained  VARCHAR(42),
    PRIMARY KEY (tsTrainerID, tsTraineeID, tsStartDate, tsEndDate),
    FOREIGN KEY (tsTrainerID) REFERENCES Mechanic (eID),
    FOREIGN KEY (tsTraineeID) REFERENCES Mechanic (eID),
    FOREIGN KEY (tsSkillTrained) REFERENCES Skillset (ssName)
);

CREATE TABLE MasteryLevel (
    mlLevel INT UNSIGNED NOT NULL,
    PRIMARY KEY (mlLevel),
    CONSTRAINT CHK_mlLevel CHECK (mlLevel>=1 AND mlLevel<=10)
);

CREATE TABLE Skillset (
    ssName          VARCHAR(42),
    ssDescription   VARCHAR(1701),
    PRIMARY KEY (ssName)
);

CREATE TABLE SkillsetLine (
    eID             INT NOT NULL,
    ssName          VARCHAR(42),
    slMasteryLevel  INT UNSIGNED NOT NULL,
    PRIMARY KEY (eID, ssName),
    FOREIGN KEY (slMasteryLevel) REFERENCES MasteryLevel (mlLevel),
    FOREIGN KEY (eID) REFERENCES Mechanic (eID),
    FOREIGN KEY (ssName) REFERENCES Skillset (ssName)
);

CREATE TABLE AppointmentStatus (
    aStatus VARCHAR(42),
    PRIMARY KEY (aStatus)
);

CREATE TABLE Appointment (
    vVIN    CHAR(17) NOT NULL,
    aDate   DATE,
    aTime   TIME,
    aExpectedTime TIME,
    aStatus VARCHAR(42),
    PRIMARY KEY (vVIN, aDate),
    FOREIGN KEY (aStatus) REFERENCES AppointmentStatus (aStatus),
    FOREIGN KEY (vVIN) REFERENCES Vehicle(vVIN)
);

CREATE TABLE MaintainOrder (
    moID INT NOT NULL,
    vVIN CHAR(17),
    moWrittenBy INT NOT NULL,
    PRIMARY KEY (moID),
    FOREIGN KEY (vVIN) REFERENCES Vehicle (vVIN),
    FOREIGN KEY (moWrittenBy) REFERENCES Technician(eID),
);

```

```
CREATE TABLE MaintainItem(
    miID INT NOT NULL,
    miSkill VARCHAR (50) NOT NULL,
    miName VARCHAR (50),
    miBuyInPrice DECIMAL (10,2) UNSIGNED,
    miCost  DECIMAL(10,2) UNSIGNED,
    mpID INT NOT NULL,
    PRIMARY KEY (miID),
    FOREIGN KEY (mpID) REFERENCES MaintainPackage (mpID),
    FOREIGN KEY (miSkill) REFERENCES Skillset (ssName)
);

CREATE TABLE ItemWork(
    miID INT NOT NULL,
    moID INT NOT NULL,
        iwDate DATE,
    PRIMARY KEY (miID, moID),
    FOREIGN KEY (miID) REFERENCES MaintainItem(miID),
    FOREIGN KEY (moID) REFERENCES MaintainOrder (moID)
);

CREATE TABLE JobQueueLine(
    miID INT NOT NULL,
    eID INT NOT NULL,
    jqlDateOfWork DATE,
    PRIMARY KEY (miID, eID, jqlDateOfWork),
    FOREIGN KEY (miID) REFERENCES MaintainItem (miID),
    FOREIGN KEY (eID) REFERENCES Mechanic (eID)
);

CREATE TABLE MaintainPackageLine (
    moID    INT NOT NULL,
    mpID    INT NOT NULL,
    PRIMARY KEY (moID, mpID),
    FOREIGN KEY (moID) REFERENCES MaintainOrder (moID),
    FOREIGN KEY (mpID) REFERENCES MaintainPackage (mpID)
);
```

```sql
--DML used to insert data--

INSERT INTO MaintainPackage
    VALUES
    (1, 'Engine Check Package', 'Investigate engine issues'),
    (2, 'ECU Check', 'Investigate ECU corruptions or faults'),
    (3, 'Transmission Check Package', 'Investigate transmission issues'),
    (4, 'Oil Check Package', 'Investigate oil related issues'),
    (5, 'Air Conditioner Check Package', 'Investigate air conditioner issues'),
    (6, 'Wheel Check Package', 'Investigate tire issues'),
    (7, 'Basic Maintainence Package','Rountine Service such as normal oil change, tire alignments, and fluid check.'),
    (8, 'Brake Check Package', 'Investigate brake issues')
;

INSERT INTO AppointmentStatus
    VALUES
    ('FUTURE'),
    ('APPOINTMENT KEPT'),
    ('CANCELLED')
;

INSERT INTO CommunicationType
    VALUES
    ('Email'),
    ('Phone Call')
;

INSERT INTO Customer
    VALUES
    (50, 'Timmy', 'Carder', '1982-09-22', '329-323-4512', 'nottheprez@gmail.com'),
    (42, 'Haddam', 'Sussein', '1967-11-01', '323-315-9695', 'bestkingevar@gmail.com'),
    (24, 'Matoi', 'Ryuko', '1997-01-02', '928-888-8251', 'sailorclothes@gmail.com'),
    (57, 'Celino', 'Barnes', '1983-08-15', '800-888-8888', 'injuryatturnyfam@hotmail.com'),
    (94, 'Imi', 'Mopkins', '1965-03-02', '123-456-7890', 'reelprufessurlelz@gmail.com'),
    (17, 'Rolph', 'Ormond', '1898-04-10', '202-555-0123', 'RolphOrmond@gmail.com'),
    (5, 'Moïse', 'Gundahar', '1975-08-25', '202-555-0110', 'MoïseGundahar@gmail.com'),
    (55, 'Arminius', 'Donat', '1982-03-21', '803-261-8908', 'ArminiusDonat@gmail.com'),
    (1, 'Sukarno', 'Nadir', '1957-12-12', '513-455-3891', 'SukarnoNadir@gmail.com'),
    (100, 'J\'onn', 'J\'onzz', '1955-11-15', '619-555-4156', 'jjonzz@deo.gov'),
    (101, 'Tony', 'Stark', '1970-05-29', '212-555-4164', 'tony@stark.com'),
    (102, 'Ami', 'Mizuno', '1978-09-10', '657-555-5156', 'amimizuno@geocities.com'),
    (103, 'Selina', 'Kyle', '1986-03-15', '212-555-4235', 'selinakyle@yahoo.com'),
    (104, 'Bruce', 'Wayne', '1975-04-29', '212-555-2346', 'bwayne@wayneenterprises.com')
;

INSERT INTO Prospective
    VALUES
    (1, 'Timmy', TRUE),
    (5, 'Haddam', FALSE),
    (55, 'Matoi', TRUE),
    (101, 'Steve Rogers', TRUE),
    (100, 'Kara Danvers', FALSE)
;

INSERT INTO PromotionContact
    VALUES
    (1, '2010-12-01', '10:12:00', 'Email'),
    (1, '2005-01-10', '1:15:00', 'Phone Call'),
    (1, '2007-08-12', '5:00:00', 'Email'),
    (5, '2014-05-15', '1:00:00', 'Email'),
    (5, '2012-07-17', '3:30:00', 'Phone Call'),
    (55, '2017-05-9', '10:12:00', 'Email'),
    (101, '2008-05-02', '13:37:00', 'Email'),
    (101, '2010-05-07', '17:01:00', 'Phone Call'),
```

```
    (101, '2013-05-03', '12:00:00', 'Email')
;

INSERT INTO Current
    VALUES
    (17, '2000-01-01'),
    (94, '2010-05-05'),
    (57, '2005-07-20'),
    (24, '2007-08-15'),
    (42, '2009-10-25'),
    (50, '2013-12-12')
;

INSERT INTO ReferralBenefitHistory VALUES
    (17, '50$ off Next Monthly Payment', '2012-05-10', 'Yes'),
    (17, '50$ off Next Monthly Payment', '2010-05-20', 'Yes'),
    (17, '50$ off Next Monthly Payment', '2016-07-24', 'Yes'),
    (17, '50$ off Next Monthly Payment', '2007-10-25', 'Yes'),
    (17, '50$ off Next Monthly Payment', '2003-12-10', 'Yes'),
    (17, '50$ off Next Monthly Payment','2017-05-12', 'No'),
    (94, 'Free Oil Change', '2012-06-13', 'Yes'),
    (94, 'Free Oil Change', '2017-05-10', 'No')
;

INSERT INTO Steady
    VALUES
    (94, 106),
    (57, 50)
;

INSERT INTO Premier
    VALUES
    (17, 198, '2017-06-12', 198.00),
    (24, 500, '2017-12-01', 50.00),
    (42, 100,'2017-07-01', 100.00),
    (50, 268,'2017-09-01', 268.00)
;

INSERT INTO PriceIncrease
    VALUES
    (24,900, 'Total Lost Collision')
;

INSERT INTO VehicleFamily
    VALUES
    (398, 'Toyota', 'Camery', '2015'),
    (378, 'Toyota', '4Runner', '2007'),
    (488, 'Toyota', 'Avalon', '1995'),
    (118, 'Nissan', 'Altima', '2017'),
    (770, 'BMW', '318', '2002'),
    (906, 'Nissan', 'Cube', '2006'),
    (540, 'Nissan', 'Pickup', '2004'),
    (920, 'Lexus', 'GX470', '2005')
;

INSERT INTO ZIPLocation
    VALUES
        ('17042', 'Lebanon', 'PA'),
    ('16001', 'Butler', 'PA'),
    ('07093', 'West New York', 'NJ'),
    ('07501', 'Paterson', 'NJ'),
    ('44256', 'Medina', 'OH'),
    ('92683', 'Westminster', 'CA'),
    ('90840', 'Long Beach', 'CA'),
    ('92620', 'Irvine', 'CA'),
    ('95209', 'Stockton', 'CA'),
```

```
    ('90210', 'Beverly Hills', 'CA'),
    ('92840', 'Garden Grove', 'CA'),
    ('91950', 'National City', 'CA'),
    ('90265', 'Malibu', 'CA')
;

INSERT INTO Address
    VALUES
    (17,'Home', '17042', '123 6th St'),
    (94,'Home', '16001', '4 Goldfield Rd.'),
    (57,'Home', '07093', '71 Pilgrim Avenue'),
    (24,'Home', '44256', '44 Shirley Ave.'),
    (42,'Home', '07501', '70 Bowman St.'),
    (50,'Home', '07093', '514 S. Magnolia St.'),
    (17,'Workplace', '17042', '65 Bayberry Street'),
    (17,'Warehouse', '17042', '8320 West Border Ave.'),
    (94,'Workplace', '16001', '225 Pawnee Ave.')
;

INSERT INTO Vehicle
    VALUES
    ('4T1BF3EK3AU552262', '2695616', 17, '6732', 7 ,378),
    ('1FTHF25H1PNA40810', '14840', 50, '11528', 7, 488),
    ('1GNSC5EC8FR580741', '102606', 57, '7698', 7, 906),
    ('2B4GP44R3XR290659', '493285', 17, '12512', 7, 920),
    ('1FMYU04161KF01101', '476524', 24, '11164', 7, 118),
    ('1FTSF31F22E716686', '33290', 42, '9073', 7, 488),
    ('2FZACFCT44AN00194', '118918', 17, '13031', 7, 378),
    ('1FMCU0G97DUC71489', '193113', 94, '9427', 7, 118),
    ('WA1VFAFL1DA140855', '230357', 57, '7934', 7, 378),
    ('1GC2CVCG1CZ106012', '228538', 42, '14124', 7, 378)
;
INSERT INTO Appointment
    VALUES
    ('4T1BF3EK3AU552262', '2017-06-5', '08:30:00', '0:30:00', 'FUTURE'),
    ('4T1BF3EK3AU552262', '2017-04-5', '09:00:00', '0:45:00', 'APPOINTMENT KEPT'),
    ('4T1BF3EK3AU552262', '2017-03-5', '10:25:00', '0:30:00', 'APPOINTMENT KEPT'),
    ('1FTHF25H1PNA40810', '2017-06-7', '11:24:00', '0:30:00', 'CANCELLED'),
    ('1GNSC5EC8FR580741', '2017-06-8', '12:45:00', '0:45:00', 'FUTURE'),
    ('2B4GP44R3XR290659', '2017-06-9', '13:30:00', '0:30:00', 'FUTURE'),
    ('1FMYU04161KF01101', '2017-06-15', '14:45:00', '0:45:00', 'FUTURE'),
    ('1GC2CVCG1CZ106012', '2017-06-20', '15:15:00', '0:30:00', 'FUTURE'),
    ('4T1BF3EK3AU552262', '2017-06-24', '16:00:00', '0:45:00', 'FUTURE'),
    ('1FTSF31F22E716686', '2017-06-27', '8:45:00', '0:30:00', 'FUTURE'),
    ('4T1BF3EK3AU552262', '2017-06-30', '9:15:00', '1:00:00', 'FUTURE'),
    ('2FZACFCT44AN00194', '2017-06-21', '10:00:00', '0:15:00', 'FUTURE'),
    ('1FMCU0G97DUC71489', '2017-06-17', '11:30:00', '0:30:00', 'FUTURE')
;

INSERT INTO Employee
    VALUES
        (1,'Seyyed Ortzi', 17, '2001-05-4'),
        (2,'Sead Huey', 12, '2002-12-24'),
        (3,'Clive Finnagán', 12.57, '2010-07-21'),
        (4,'Nerses Teobaldo', 18.42, '2004-1-15'),
        (5,'Stuart Evaristo', 20.47, '2001-7-2'),
        (6,'Filip Ermanno', 19.77, '2009-8-2'),
        (7,'Hüseyn Kay', 30.78, '2001-5-27'),
        (8,'Andro Pere', 25.97, '2003-7-26'),
        (9,'Evgeny Octave', 31.44, '2001-2-28'),
        (10,'Aeron Amariah', 22.00, '2003-3-25')
;

INSERT INTO Technician
    VALUES
        (1),
```

```
        (2),
        (7),
        (9),
        (8)
;
INSERT INTO Mechanic
    VALUES
        (1),
        (2),
        (3),
        (4),
        (5),
        (6),
        (10)
;
INSERT INTO MaintainOrder values
    (1, '4T1BF3EK3AU552262', 7),
    (2, '4T1BF3EK3AU552262', 7),
    (3, '4T1BF3EK3AU552262', 8),
    (4, '1FTHF25H1PNA40810', 8),
    (5, '1GNSC5EC8FR580741', 7),
    (6, '2B4GP44R3XR290659', 9),
    (7, '1FMYU04161KF01101', 9),
    (8, '1FTSF31F22E716686', 9),
    (9, '2FZACFCT44AN00194', 7),
    (10, '1FMCU0G97DUC71489', 8),
    (11, 'WA1VFAFL1DA140855', 7),
    (12, '1GC2CVCG1CZ106012', 9)
;

INSERT INTO MaintainPackageLine values
    (1, 2),
    (2, 3),
    (3, 2),
    (4, 3),
    (5, 4),
    (6, 5),
    (7, 5),
    (8, 6),
    (9, 1),
    (10, 3),
    (11, 4),
    (12, 6)
;

INSERT INTO MasteryLevel
    VALUES
        (01),
        (02),
        (03),
        (04),
        (05),
        (06),
        (07),
        (08),
        (09),
        (10)
;

INSERT INTO Skillset
    VALUES
    ('Oil & filter', 'Oil and filter change'),
    ('Tire rotation', 'Tire rotation'),
    ('Tire balancing', 'Tire balancing'),
    ('Freon exchange', 'Flush freon from air conditioning system and replace'),
    ('Brake rotor resurfacing', 'Resurfacing the brake rotors'),
```

```
    ('Brake fluid flush', 'Flush brake fluid from system and replace'),
    ('Engine fixer', 'Fix the engine in cars'),
    ('Transmission fixer', 'Fix tranmission related issue'),
    ('Computer Technician', 'Fix computer in car')
;

INSERT INTO SkillsetLine
    VALUES
    (1, 'Oil & filter', 3),
    (1, 'Tire rotation', 10),
    (1, 'Tire balancing',2),
    (2, 'Brake fluid flush', 7),
    (3, 'Brake rotor resurfacing', 2),
    (3, 'Computer Technician', 2),
    (4, 'Tire balancing', 4),
    (4, 'Computer Technician', 10),
    (5, 'Freon exchange', 10),
    (6, 'Engine fixer', 1),
    (6, 'Oil & filter', 5),
    (6, 'Tire rotation', 1),
    (6, 'Tire balancing',3),
    (10, 'Oil & filter', 10),
    (10,'Engine fixer',5),
    (10,'Transmission fixer',3)
;

INSERT INTO TrainingSkill
    VALUES
        (1, 2, '2015-05-04', '2015-05-30', 'Oil & filter'),
        (5, 3, '2001-09-11', '2001-09-11', 'Freon exchange'),
        (6, 10, '2014-05-05', '2014-06-30', 'Tire rotation'),
        (4, 10, '2005-04-23', '2005-07-04', 'Tire balancing'),
        (4, 2, '2005-04-23', '2005-07-04', 'Freon exchange')

;

INSERT INTO MaintainItem values
    (1, 'Engine fixer', 'Repair engine block', 0.00, 300.00, 1),
    (2, 'Engine fixer', 'Repair manifolds', 0.00, 100.00, 1),
    (3, 'Computer Technician', 'Inspect wiring and profile in ECU', 0.00, 10.00, 2),
    (4, 'Transmission fixer', 'Repair AT system', 400.00, 600.00, 3),
    (5, 'Transmission fixer', 'Repair MT system', 200.00, 300.00, 3),
    (6, 'Transmission fixer', 'Inspect shift mechanisms', 0.00, 50.00, 3),
    (7, 'Oil & filter', 'Replace engine oil and filter', 30.00, 65.00, 4),
    (8, 'Transmission fixer', 'Replace transmission oil (AT)', 40.00, 200.00, 4),
    (9, 'Freon exchange', 'Refill Freon in AC system', 40.00, 50.00, 5),
    (10, 'Tire rotation', 'Rotate tires', 0.00, 10.00, 6),
    (11, 'Tire balancing', 'Balance tires', 0.00, 5.00, 6),
    (12, 'Tire balancing', 'Replace tires', 0.00, 200.00, 6),
    (13, 'Oil & filter', 'Basic oil change', 30.00, 60.00, 7),
    (14, 'Tire rotation', 'Basic tire rotation', 0.00, 10.00, 7),
    (15, 'Brake fluid flush', 'Replace brake fluid', 30.00, 40.00, 8),
    (16, 'Brake rotor resurfacing', 'Resurface rotors', 20.00, 50.00, 8)
;

INSERT into ItemWork values
    (3, 1 ,'2017-5-12'),
    (4, 2 ,'2017-5-11'),
    (5, 2 ,'2011-5-13'),
    (6, 2 ,'2012-5-14'),
    (3, 3 ,'2013-5-15'),
    (4, 4 ,'2014-5-16'),
    (5, 4 ,'2015-5-17'),
    (6, 4 ,'2016-5-18'),
    (7, 5 ,'2017-5-19'),
    (8, 5 ,'2017-5-11'),
```

```
      (9, 6 ,'2016-5-12'),
      (9, 7, '2015-5-13'),
      (10, 8 ,'2014-5-14'),
      (11, 8 ,'2013-5-15'),
      (12, 8, '2012-5-16'),
      (1, 9 ,'2011-5-17'),
      (2, 9, '2010-5-18'),
      (4, 10 ,'2017-5-19'),
      (5, 10 ,'2016-5-11'),
      (6, 10 ,'2015-5-12'),
      (7, 11 ,'2014-5-13'),
      (8, 11 ,'2013-5-14'),
      (10, 12 ,'2012-5-15'),
      (11, 12 ,'2011-5-16'),
      (12, 12, '2010-5-17');
      (13, 1 ,'2017-5-12'),
      (14, 1 ,'2017-5-11'),
      (13, 2 ,'2011-5-13'),
      (14, 2 ,'2012-5-14'),
      (13, 3 ,'2013-5-15'),
      (14, 3 ,'2014-5-16'),
      (13, 4 ,'2015-5-17'),
      (14, 4 ,'2016-5-18'),
      (13, 5 ,'2017-5-19'),
      (14, 5 ,'2017-5-11'),
      (13, 6 ,'2016-5-12'),
      (14, 6, '2015-5-13'),
      (13, 7 ,'2014-5-14'),
      (14, 7 ,'2013-5-15'),
      (13, 8, '2012-5-16'),
      (14, 8 ,'2011-5-17'),
      (13, 9, '2010-5-18'),
      (14, 9 ,'2017-5-19'),
      (13, 10 ,'2016-5-11'),
      (14, 10 ,'2015-5-12'),
      (13, 11 ,'2014-5-13'),
      (14, 11 ,'2013-5-14'),
      (13, 12 ,'2012-5-15'),
      (14, 12 ,'2011-5-16')
;
INSERT into JobQueueLine values
      (3, 1 ,'2017-5-12'),
      (4, 2 ,'2017-5-11'),
      (5, 2 ,'2011-5-13'),
      (6, 2 ,'2012-5-14'),
      (3, 3 ,'2013-5-15'),
      (4, 4 ,'2014-5-16'),
      (5, 4 ,'2015-5-17'),
      (6, 4 ,'2016-5-18'),
      (7, 5 ,'2017-5-19'),
      (8, 5 ,'2017-5-11'),
      (9, 6 ,'2016-5-12'),
      (9, 1, '2015-5-13'),
      (10, 2 ,'2014-5-14'),
      (11, 3 ,'2013-5-15'),
      (12, 4, '2012-5-16'),
      (1, 5 ,'2011-5-17'),
      (2, 6, '2010-5-18'),
      (4, 10 ,'2017-5-19'),
      (5, 10 ,'2016-5-11'),
      (6, 10 ,'2015-5-12'),
      (7, 1 ,'2014-5-13'),
      (8, 2 ,'2013-5-14'),
      (10, 2 ,'2012-5-15'),
      (11, 3 ,'2011-5-16'),
      (12, 4, '2010-5-17');
```

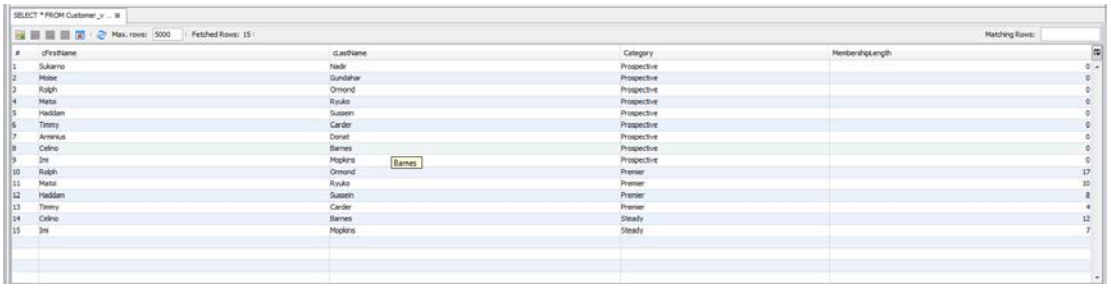
```
395       (13, 1 ,'2017-5-12'),
396       (14, 2 ,'2017-5-11'),
397       (13, 3 ,'2011-5-13'),
398       (14, 4 ,'2012-5-14'),
399       (13, 5 ,'2013-5-15'),
400       (14, 6 ,'2014-5-16'),
401       (13, 10,'2015-5-17'),
402       (14, 1 ,'2016-5-18'),
403       (13, 2 ,'2017-5-19'),
404       (14, 3 ,'2017-5-11'),
405       (13, 4 ,'2016-5-12'),
406       (14, 5, '2015-5-13'),
407       (13, 6 ,'2014-5-14'),
408       (14, 10,'2013-5-15'),
409       (13, 1, '2012-5-16'),
410       (14, 2 ,'2011-5-17'),
411       (13, 3, '2010-5-18'),
412       (14, 4 ,'2017-5-19'),
413       (13, 05 ,'2016-5-11'),
414       (14, 06 ,'2015-5-12'),
415       (13, 10 ,'2014-5-13'),
416       (14, 01 ,'2013-5-14'),
417       (13, 02 ,'2012-5-15'),
418       (14, 03 ,'2011-5-16')
419   ;
420
```

VIEW

1. Customer_v – for each customer, indicate his or her name as well as the customer type (prospect, steady or premier) as well as the number of years that customer has been with us.

    a. create view Customer_v as select cFirstName, cLastName,'Prospective'as Category, 0 as MembershipLength from Prospective p inner join Customer c1 UNION select cFirstName, cLastName,'Premier'as Category, (2017-YEAR(c2.cJoinedDate)) as MembershipLength from Premier p inner join Current c2 on p.cID=c2.cID inner join Customer c3 on p.cID=c3.cID UNION select cFirstName, cLastName,'Steady'as Category, (2017-YEAR(c4.cJoinedDate)) as MembershipLength from Steady s inner join Current c4 on s.cID=c4.cID inner join Customer c5 on s.cID=c5.cID;

    b.

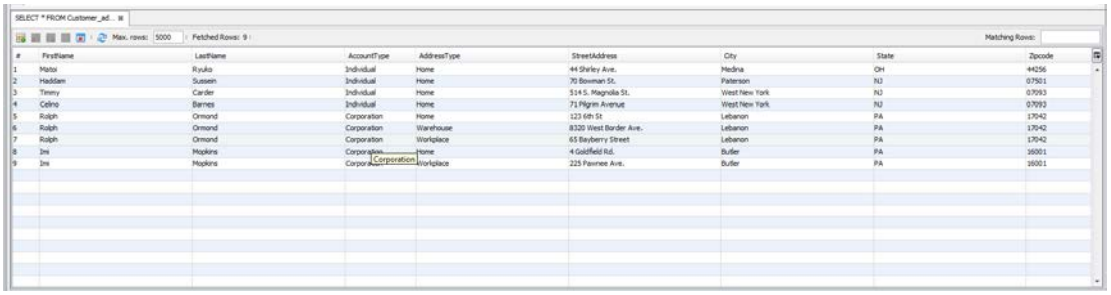
    c. 

#	cFirstName	cLastName	Category	MembershipLength
1	Sukarno	Nadir	Prospective	0
2	Moise	Gundahar	Prospective	0
3	Rolph	Ormond	Prospective	0
4	Matoi	Ryuko	Prospective	0
5	Haddam	Sussein	Prospective	0
6	Timmy	Carder	Prospective	0
7	Arminius	Donet	Prospective	0
8	Celino	Barnes	Prospective	0
9	Imi	Mopkins	Prospective	0
10	Rolph	Ormond	Premier	17
11	Matoi	Ryuko	Premier	10
12	Haddam	Sussein	Premier	8
13	Timmy	Carder	Premier	4
14	Celino	Barnes	Steady	12
15	Imi	Mopkins	Steady	7

2. Customer_addresses_v – for each customer, indicate whether they are an individual or a corporate account, and display all of the addresses that we are managing for that customer.

    a. create view Customer_addresses_v as select cFirstName as FirstName,cLastName as LastName,'Individual' as AccountType, aType as AddressType, aAddress as StreetAddress,zip.zCity as City, zip.zState as State, zip.zZIPCode as Zipcode from Address left outer join Customer on Address.cID=Customer.cID left outer join ZIPLocation zip on Address.zZIPCode=zip.zZIPCode GROUP by Address.cID Having count(Address.cID)=1 UNION select c.cFirstName as FirstName, c.cLastName as LastName,'Corporation' as AccountType, a.aType as AddressType, a.aAddress as StreetAddress,zip1.zCity as City, zip1.zState as State, zip1.zZIPCode as Zipcode from Address a left outer join Customer c on a.cID=c.cID left outer join ZIPLocation zip1 on zip1.zZIPCode=a.zZIPCode where a.cID in (select c1.cID from Address a1 left outer join Customer c1 on a1.cID=c1.cID GROUP BY c1.cFirstName HAVING COUNT(a1.cID)>1);

    b.

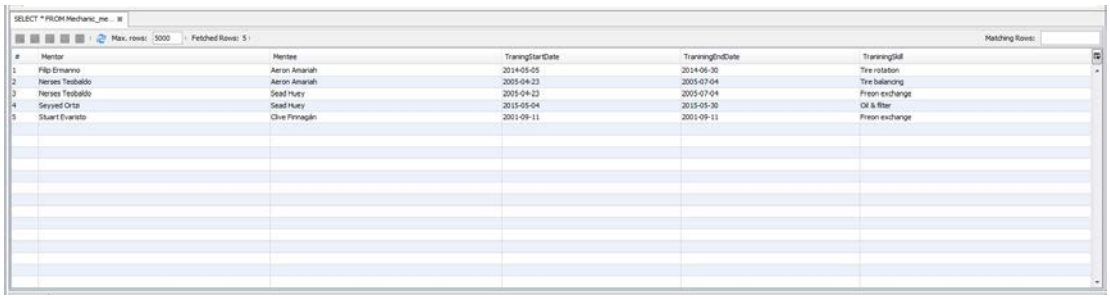
    c. 

#	FirstName	LastName	AccountType	AddressType	StreetAddress	City	State	Zipcode
1	Matoi	Ryuko	Individual	Home	44 Shirley Ave.	Medina	OH	44256
2	Haddam	Sussein	Individual	Home	70 Bowman St.	Paterson	NJ	07501
3	Timmy	Carder	Individual	Home	514 S. Magnolia St.	West New York	NJ	07093
4	Celino	Barnes	Individual	Home	71 Pilgrim Avenue	West New York	NJ	07093
5	Rolph	Ormond	Corporation	Home	123 6th St	Lebanon	PA	17042
6	Rolph	Ormond	Corporation	Warehouse	8320 West Border Ave.	Lebanon	PA	17042
7	Rolph	Ormond	Corporation	Workplace	65 Bayberry Street	Lebanon	PA	17042
8	Imi	Mopkins	Corporation	Home	4 Goldfield Rd.	Butler	PA	16001
9	Imi	Mopkins	Corporation	Workplace	225 Pawnee Ave.	Butler	PA	16001

3. Mechanic_mentor_v – reports all of the mentor/mentee relationships at Dave's, sorted by the name of the mentor, then the name of the mentee.

    a. create view Mechanic_metor_v as select e1.eName as Mentor, e2.eName as Mentee, t.tsStartDate as TraningStartDate,t.tsEndDate as TraniningEndDate ,t.tsSkillTrained as TraniningSkill from Mechanic m1 inner join Employee e1 on m1.eID=e1.eID right outer join TrainingSkill t on t.tsTrainerID=m1.eID left outer join Mechanic m2 on m2.eID=t.tsTraineeID inner join Employee e2 on m2.eID=e2.eID order by e1.eName, e2.eName;

    b.

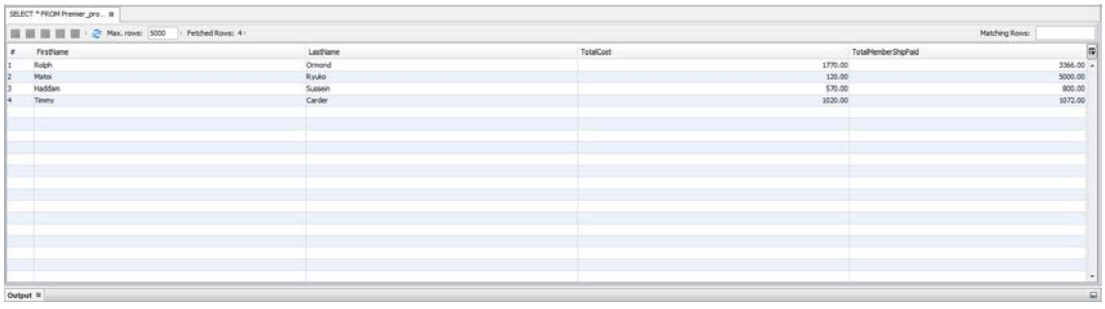
    c. 

#	Mentor	Mentee	TraningStartDate	TraningEndDate	TraningSkill
1	Filip Ermanno	Aeron Amariah	2014-05-05	2014-06-30	Tire rotation
2	Nerses Teobaldo	Aeron Amariah	2003-04-23	2003-07-04	Tire balancing
3	Nerses Teobaldo	Sead Huey	2003-04-23	2003-07-04	Freon exchange
4	Seyyed Orta	Sead Huey	2015-05-04	2015-05-30	Oil & filter
5	Stuart Evaristo	Clive Finnagin	2001-09-11	2001-09-11	Freon exchange

4. Premier_profits_v – On a year by year basis, show the premier customer's outlay versus what they would have been charged for the services which they received had they merely been steady customers.

    a. create view Premier_profits_v as select c.cFirstName as FirstName, c.cLastName as LastName, sum(m.miCost) as TotalCost, (p.pAnnualFee*(2017-YEAR(c2.cJoinedDate))) as TotalMemberShipPaid from ItemWork i left outer join MaintainItem m on i.miID=m.miID left outer join MaintainOrder mo on i.moID=mo.moID left outer join Vehicle v on mo.vVin=v.vVin left outer join Customer c on c.cID=v.cID left outer join Premier p on c.cID=p.cID left outer join Current c2 on c2.cID=p.cID where c.cID in (select p1.cID from Premier p1) group by c.cID ;

    b.

    c. 

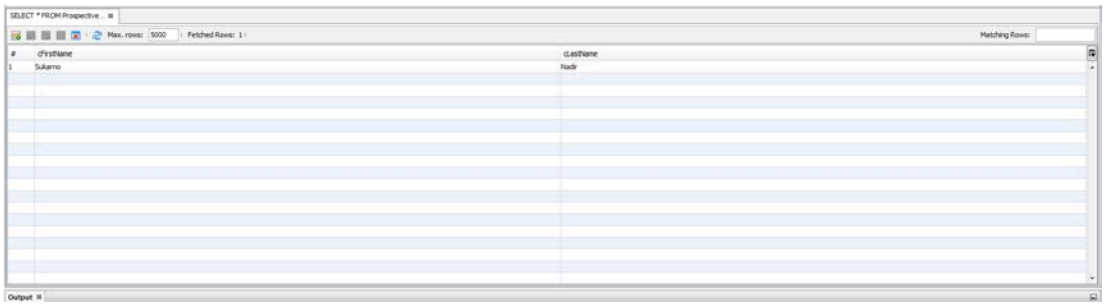
#	FirstName	LastName	TotalCost	TotalMemberShipPaid
1	Ralph	Ormond	1770.00	3366.00
2	Matoi	Ryuko	120.00	5000.00
3	Haddam	Sussein	570.00	800.00
4	Timmy	Carder	1020.00	1072.00

5. Prospective_resurrection_v – List all of the prospective customers who have had three or more contacts, and for whom the most recent contact was more than a year ago. They might be ripe for another attempt.

    a. create view Prospective_resurrection_v as select c1.cFirstName, c1.cLastName from Customer c1 inner join Prospective p1 on c1.cID=p1.cID where c1.cID NOT IN (select

p.cID from PromotionContact pc left outer join Prospective p on pc.cID=p.cID where pc.pcDate>'2013-12-31') AND p1.pDeadFlag=1;

b.

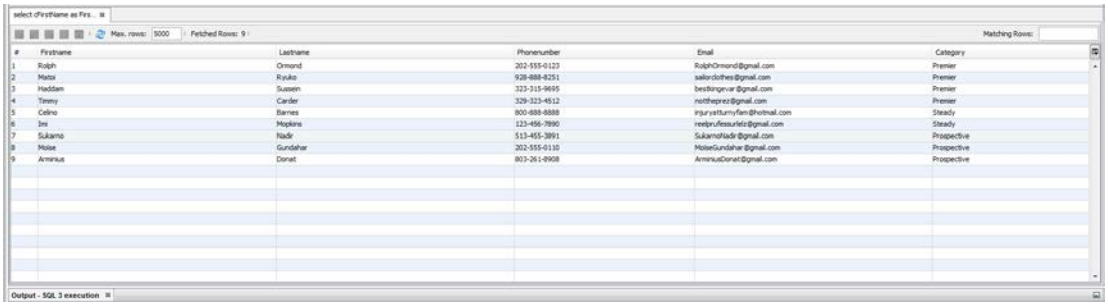
c. 

Queries

1. List the customers. For each customer, indicate which category he or she fall into, and his or her contact information. If you have more than one independent categorization of customers, please indicate which category the customer falls into for all of the categorizations.

    a. select cFirstName as Firstname, cLastName as Lastname, cPhoneNumber as Phonenumber, cEmail as Email, 'Premier' as Category from Premier natural join Customer UNION select cFirstName as Firstname, cLastName as Lastname, cPhoneNumber as Phonenumber, cEmail as Email, 'Steady' as Category from Steady natural join Customer UNION select cFirstName as Firstname, cLastName as Lastname, cPhoneNumber as Phonenumber, cEmail as Email, 'Prospective' as Category from Prospective natural join Customer;

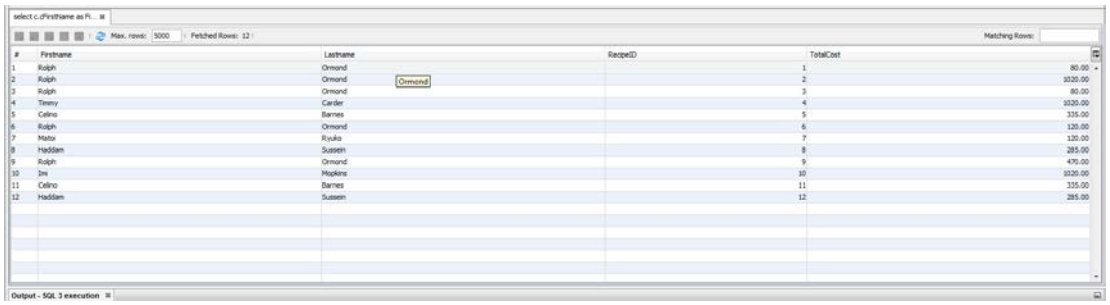
    b.

    c. 

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

    a. select c.cFirstName as Firstname,c.cLastName as Lastname, mo.moID as RecipeID, sum(mi.miCost) as TotalCost from Customer c right outer join Vehicle v on c.cID=v.cID right outer join MaintainOrder mo on v.vVIN=mo.vVIN right outer join ItemWork iw on mo.moID=iw.moID left outer join MaintainItem mi on iw.miID=mi.miID group by mo.moID;

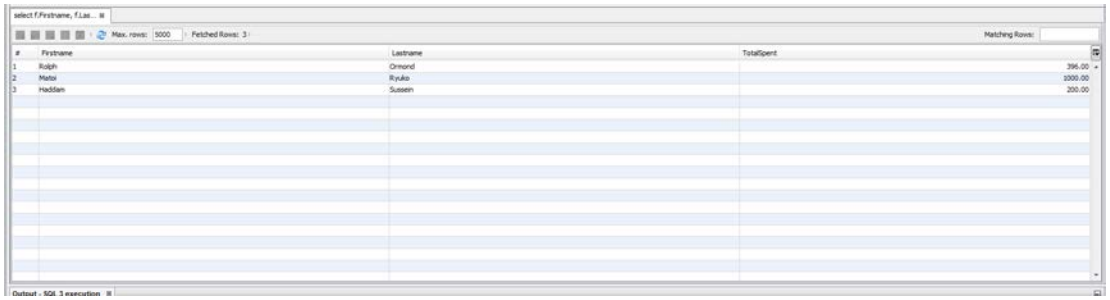
    b.

c. 

#	Firstname	Lastname	RecipeID	TotalCost
1	Rolph	Ormond	1	80.00
2	Rolph	Ormond	2	1020.00
3	Rolph	Ormond	3	80.00
4	Timmy	Carder	4	1020.00
5	Celino	Barnes	5	335.00
6	Rolph	Ormond	6	120.00
7	Matoi	Ryuko	7	120.00
8	Haddam	Sussein	8	285.00
9	Rolph	Ormond	9	470.00
10	Imi	Mopkins	10	1020.00
11	Celino	Barnes	11	335.00
12	Haddam	Sussein	12	285.00

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.

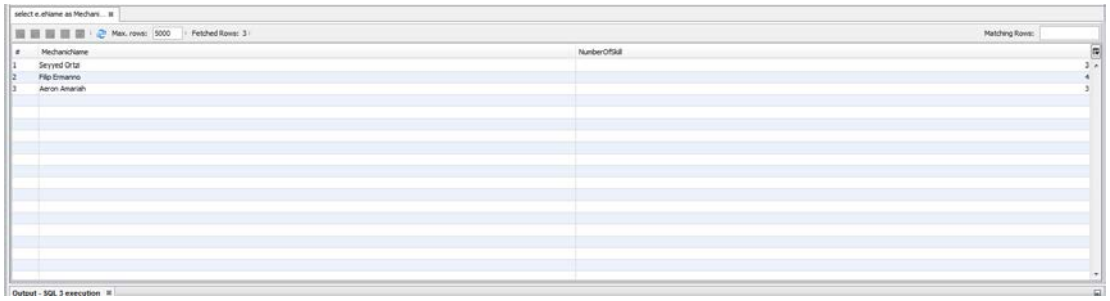
    a. select f.Firstname, f.Lastname, f.TotalSpent from(select c.cFirstName as Firstname, c.cLastName as Lastname,(p.pAnnualFee*2) as TotalSpent from Customer c inner join Premier p on c.cID=p.cID inner join Current cu on p.cID=cu.cID UNION select c1.cFirstName as Firstname, c1.cLastName as Lastname, sum(mi.miCost) as TotalSpent from Customer c1 inner join Steady s on c1.cID=s.cID right outer join Vehicle v on s.cID=v.cID right outer join MaintainOrder mo on v.vVIN=mo.vVIN right outer join ItemWork iw on mo.moID=iw.moID left outer join MaintainItem mi on iw.miID=mi.miID where iw.iwDate>'2015-01-01' group by (s.cID)) AS f LIMIT 3;

    b. c. 

#	Firstname	Lastname	TotalSpent
1	Rolph	Ormond	396.00
2	Matoi	Ryuko	1000.00
3	Haddam	Sussein	200.00

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

    a. select e.eName as MechanicName, COUNT(sk.eID) as NumberOfSkill from Employee e inner join Mechanic m on e.eID=m.eID right outer join SkillsetLine sk on m.eID=sk.eID GROUP By m.eID HAVING COUNT(sk.eID)>2;

    b. 

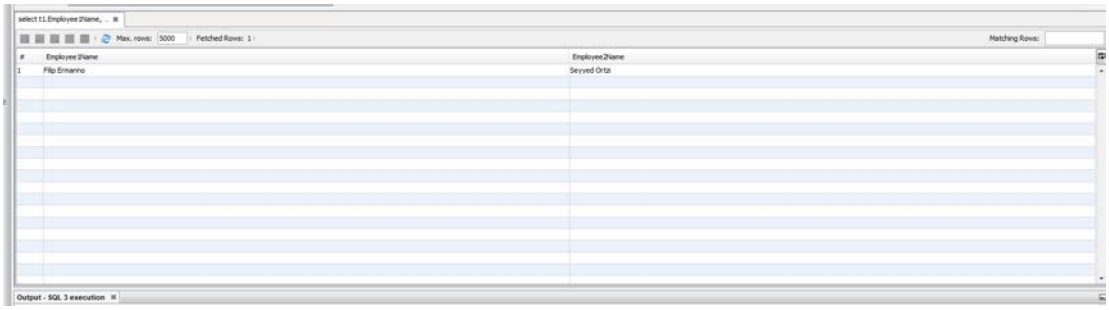
#	MechanicName	NumberOfSkill
1	Seyyed Ortal	3
2	Filip Ermanno	4
3	Aeron Amariah	3

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

    a. select t1.Employee1Name, t2.Employee2Name from (select e1.eName as Employee1Name, sk1.ssName as Employee1Skill from SkillsetLine sk1 left outer join

Employee e1 on sk1.eID=e1.eID) as t1 inner join (select e2.eName as Employee2Name, sk2.ssName as Employee2Skill from SkillsetLine sk2 left outer join Employee e2 on sk2.eID=e2.eID) as t2 on t1.Employee1Skill=t2.Employee2Skill where t1.Employee1Name <t2.Employee2Name group by CONCAT(t1.Employee1Name, t2.Employee2Name) HAVING Count(CONCAT(t1.Employee1Name, t2.Employee2Name))>2;

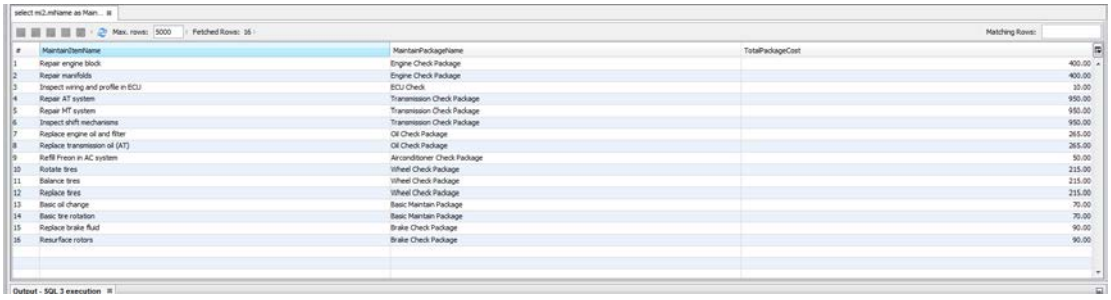
b.

c. 

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.

    a. select mi2.miName as MaintainItemName, t.MaintainPackageName as MaintainPackageName, t.TotalCostForThisPackage as TotalPackageCost from (select mi.mpID as MaintainPackageID, mp.mpName as MaintainPackageName, sum(mi.miCost) as TotalCostForThisPackage from MaintainItem mi left outer join MaintainPackage mp on mi.mpID=mp.mpID GROUP by (mp.mpID)) as t right outer join MaintainItem mi2 on t.MaintainPackageID=mi2.mpID;

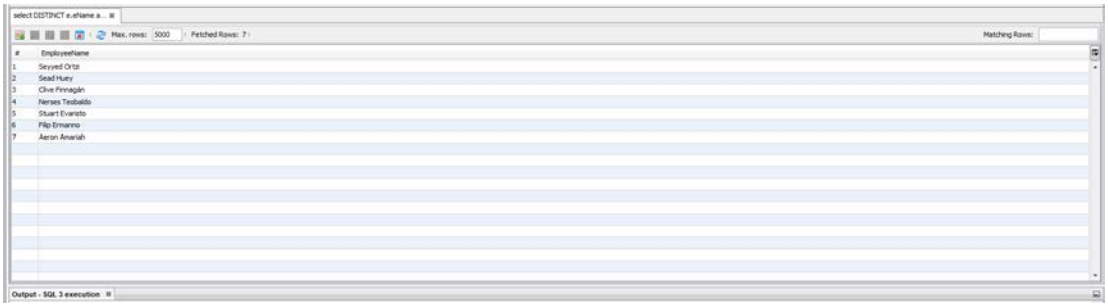
    b.

    c. 

7. Find all of those mechanics who have one or more maintenance items that they lacked one or more of the necessary skills.

    a. select DISTINCT e.eName as EmployeeName from JobQueueLine jql left outer join Employee e on jql.eID=e.eID left outer join MaintainItem mi on mi.miID=jql.miID where mi.miSkill not in (select sl.ssName from SkillsetLine sl where e.eID=sl.eID);

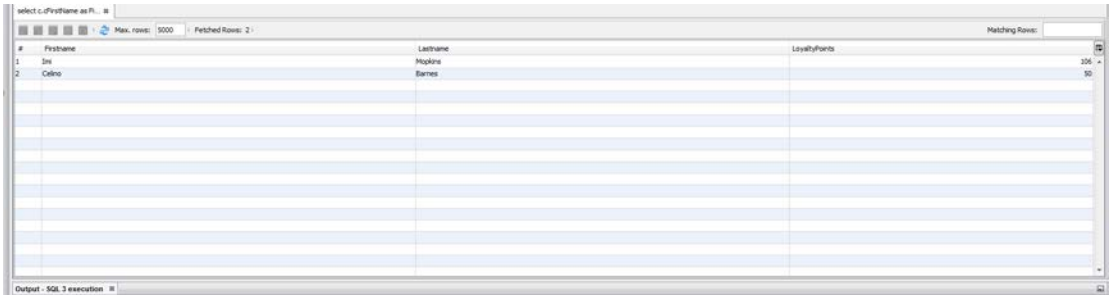
    b.

c. 

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

    a. select c.cFirstName as Firstname, c.cLastName as Lastname, s.sLoyaltyPoints as LoyaltyPoints from Customer c inner join Steady s on c.cID=s.cID Order by s.sLoyaltyPoints DESC;

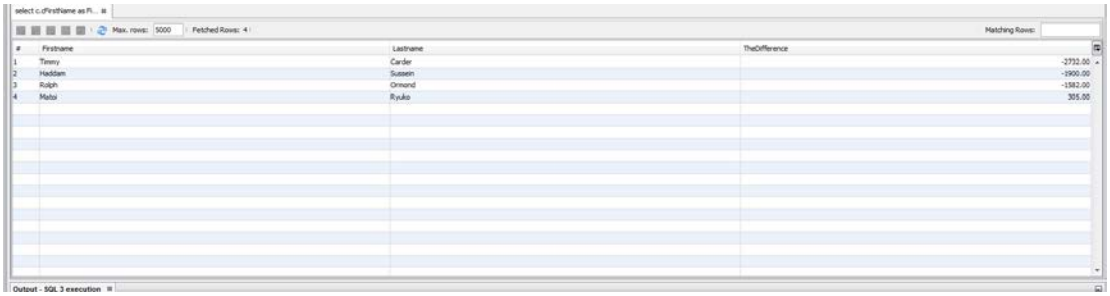
    b.

c. 

9. List 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 the largest difference to the smallest.

    a. select c.cFirstName as Firstname, c.cLastName as Lastname, (p.pAnnualFee-SUM(mi.miCost)) as TheDifference from Customer c inner join Premier p on c.cID=p.cID right outer join Vehicle v on p.cID=v.cID right outer join MaintainOrder mo on v.vVIN=mo.vVIN right outer join ItemWork iw on mo.moID=iw.moID left outer join MaintainItem mi on mi.miID=iw.moID where c.cID in (select p1.cID from Premier p1) GROUP by p.cID ORDER BY TheDifference;

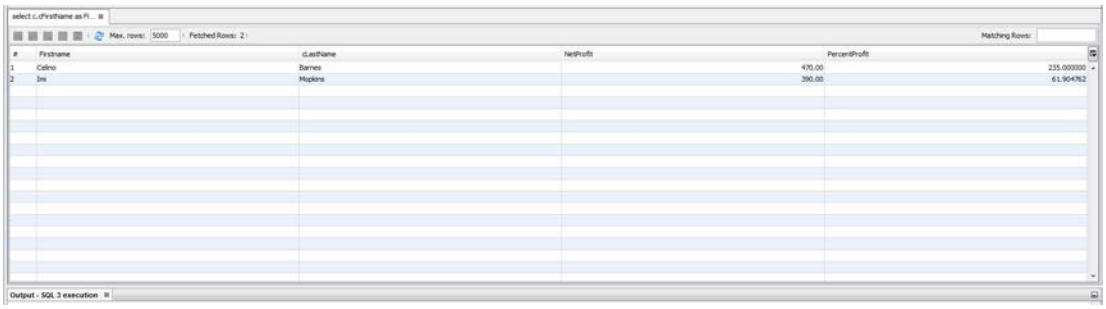
    b.

c. 

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.

a. select c.cFirstName as Firstname, c.cLastName, (SUM(mi.miCost)-SUM(mi.miBuyInPrice)) as NetProfit, ((SUM(mi.miCost)-SUM(mi.miBuyInPrice))/SUM(mi.miBuyInPrice)*100) as PercentProfit from Customer c inner join Steady s on c.cID=s.cID right outer join Vehicle v on s.cID=v.cID right outer join MaintainOrder mo on mo.vVIN=v.vVIN right outer join ItemWork iw on iw.moID=mo.moID left outer join MaintainItem mi on iw.miID=mi.miID where c.cID in (select s1.cID from Steady s1) GROUP by s.cID;

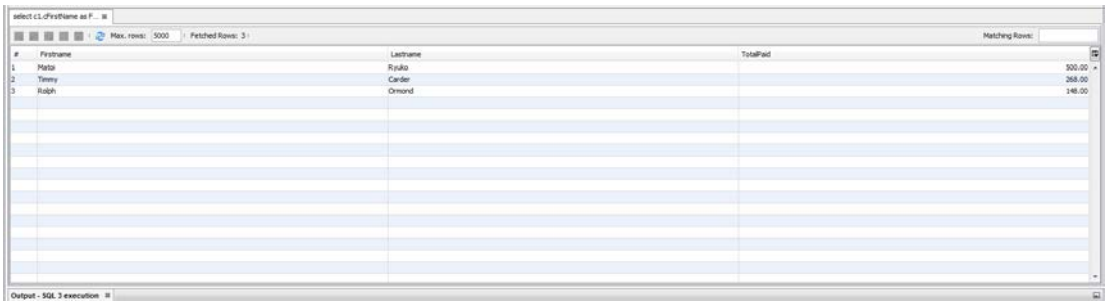
b.

c. 

11. List the three premier customers who have paid Dave's Automotive the greatest amount in the past year, and the sum of their payments over that period. Be sure to take into account any discounts that they have earned by referring prospective customers.

a. select c1.cFirstName as Firstname, c1.cLastName as Lastname, (p1.pAnnualFee-(IFNULL(t.DiscountAmmount,0))) as TotalPaid from Customer c1 inner join Premier p1 on c1.cID=p1.cID left outer join (select p.cID as PremierCustomerID,count(p.cID)*50 as DiscountAmmount from Customer c inner join Premier p on c.cID=p.cID right outer join ReferralBenefitHistory rbh on rbh.cID=p.cID where rbh.rDate between '2015-12-31' AND '2016-12-31' group by p.cID) t on t.PremierCustomerID=p1.cID order by (p1.pAnnualFee*12-(IFNULL(t.DiscountAmmount,0)))Desc limit 3;

b.

c. 

12. List the five model, make, and year that have caused the most visits on average to Dave's automotive **per vehicle** in the past three years, along with the average number of visits per vehicle.

a. select vf.vfModel as Model, vf.vfYear as Year, vf.vfMake as Maker, count(CONCAT(vf.vfModel,vf.vfMake, vf.vfYear)) as NumberVisited from MaintainOrder mo left outer join Vehicle v on mo.vVIN=v.vVIN left outer join VehicleFamily vf on vf.vfID=v.vfID where mo.moID in (select i.moID from ItemWork i

where i.iwDate>'2013-12-31') group by CONCAT(vf.vfModel,vf.vfMake, vf.vfYear) order by count(CONCAT(vf.vfModel,vf.vfMake, vf.vfYear)) DESC limit 5;

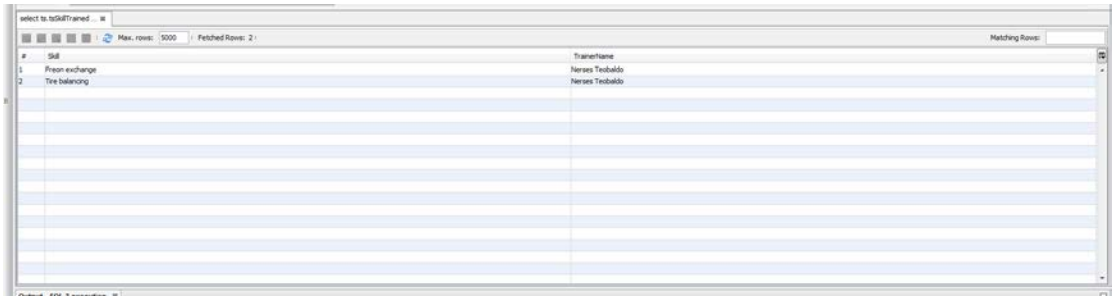
b.

c. 

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

    a. select ts.tsSkillTrained as Skill, e.eName as TrainerName from TrainingSkill ts left outer join Employee e on ts.tsTrainerID=e.eID where ts.tsTrainerID=(select t.ID from (select ts1.tsTrainerID as ID, count(ts1.tsTrainerID) as Total from TrainingSkill ts1 group by ts1.tsTrainerID)as t order by t.Total desc limit 1);

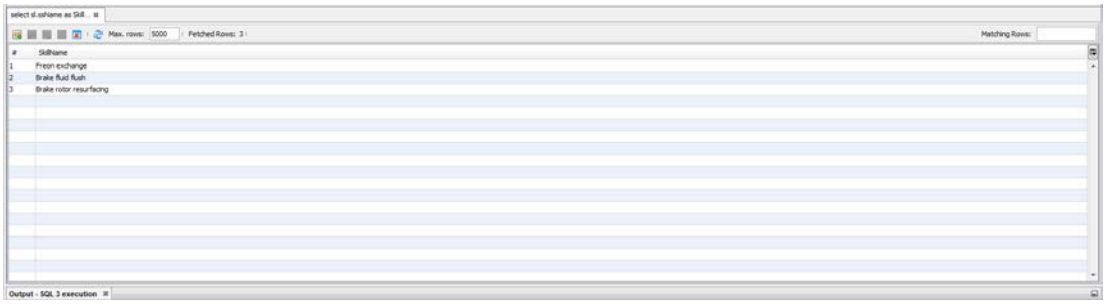
    b.

    c. 

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

    a. select sl.ssName as SkillName from SkillsetLine sl left outer join Skillset s on sl.ssName=s.ssName group by sl.ssName order by count(sl.ssName) limit 3;

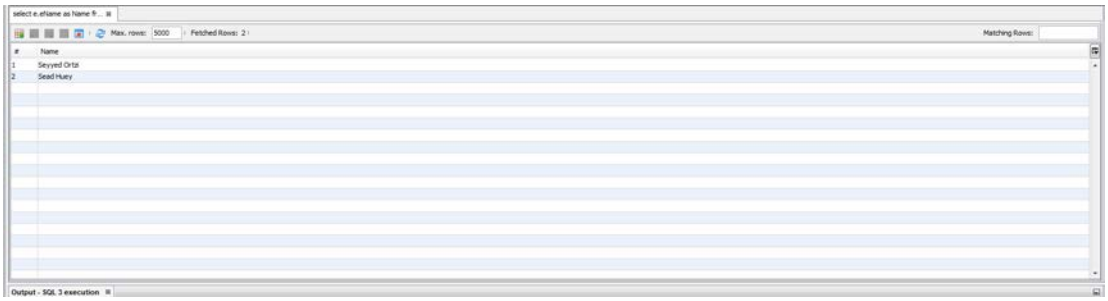
    b.

    c. 

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

    a. select e.eName as Name from Employee e inner join Technician t on e.eID=t.eID inner join Mechanic m on e.eID=m.eID;

b.

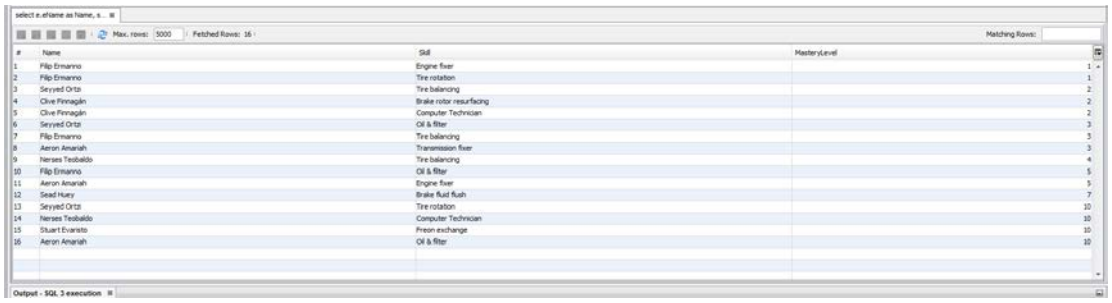
c. 

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.

16.1 Show that all mastery level is beween 1 and 10 inclusive.

a. select e.eName as Name, sl.ssName as Skill, sl.slMasteryLevel as MasteryLevel from SkillsetLine sl left outer join Employee e on sl.eID=e.eID left outer join Skillset s on s.ssName=sl.ssName;

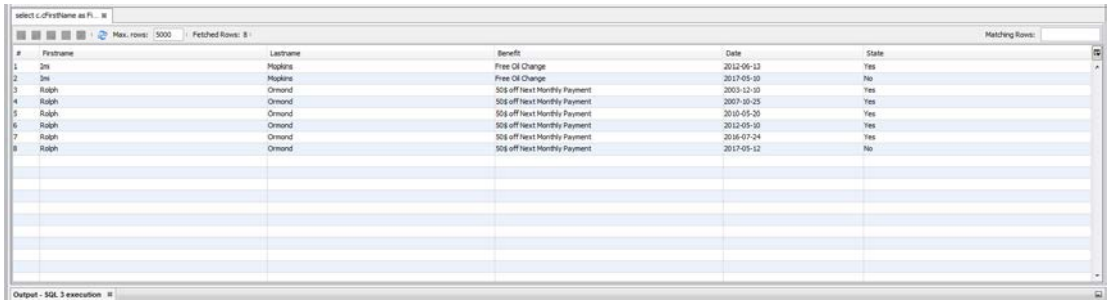
b.

c. 

16.2 Referral Benefit History

d. select c.cFirstName as Firstname, c.cLastName as Lastname, rbh.rBenefit as Benefit, rbh.rDate as Date, rbh.rFlag as State from ReferralBenefitHistory rbh left outer join Customer c on rbh.cID=c.cID order by c.cFirstName, c.cLastName;

e.

f. 

16.3 Show the TechicianMaintainPackage service and required MaintainItemPackage

g. select mo.vVIN as CarVIN, mo.moID as MaintainOrderID, v.vRoutineServices as RoutineMaintainPackageID, m1.mpName as AdditionalServicePackageID from MaintainOrder mo left outer join Vehicle v on mo.vVIN=v.vVIN right outer join MaintainPackageLine mp on mp.moID=mo.moID left outer join MaintainPackage m1 on m1.mpID=mp.mpID;

h.

i.