

95 – 703 E: Database Management

Implementation Project

This project is to be completed in Oracle Database 11g Express by groups of two students. Pairing will be provided by the instructor. This document is divided into the following parts: description of the model (Part I), the specific tasks to do (Part II), and what is required for submission (Part III). You have five weeks to complete the project. Submit hardcopy of your project no later than 3 p.m. on December 3rd.

Part I. Description of the model

A car rental company has several outlets (i.e., locations) from which clients pick up cars. Each outlet has a number of vehicles that may be rented for various periods of time. For each car/vehicle we record the car ID, year, model and the daily rate.

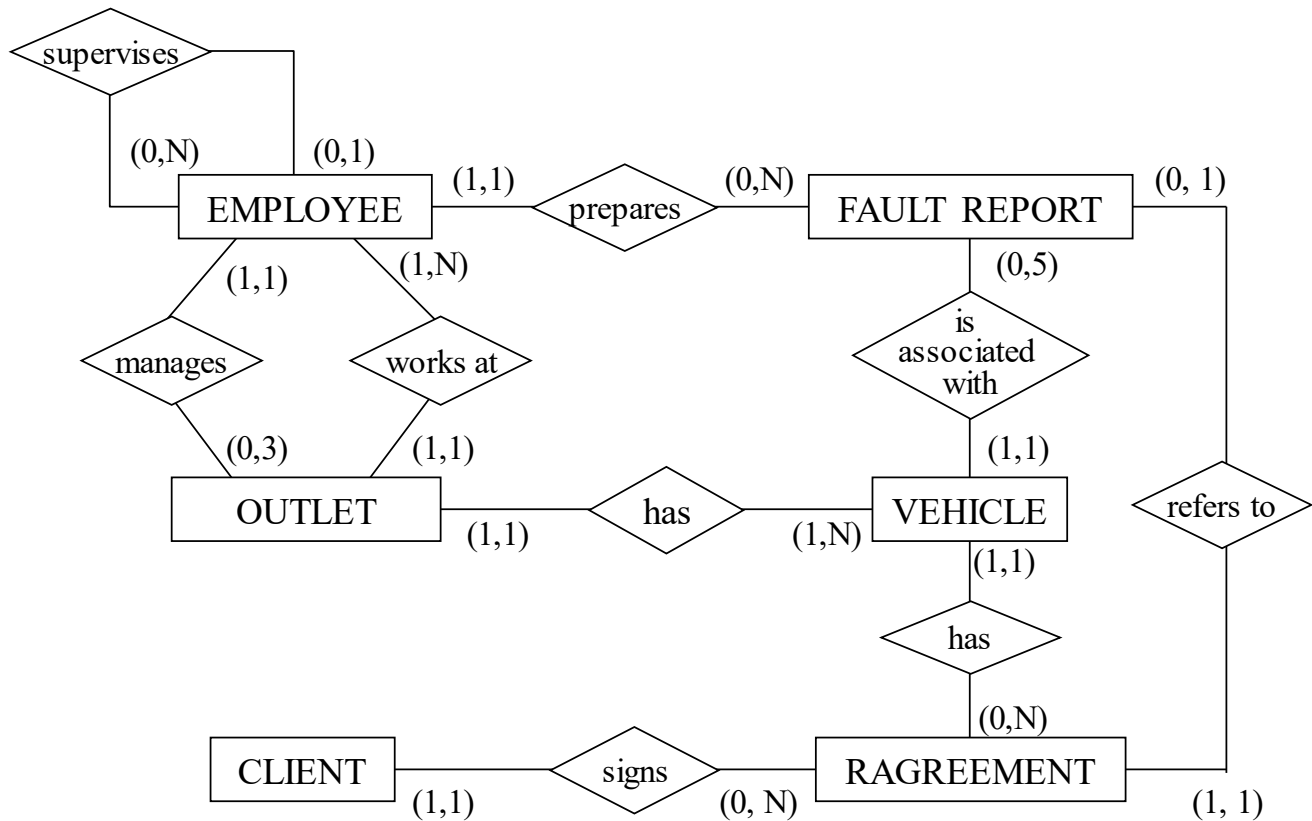
Each outlet has several employees working at the outlet including mechanics, sales reps and/or administrative assistants. We need to keep only some basic information about the employees. A few of the employees supervise other employees (e.g., one sales rep would supervise other sales reps). For each supervised employee, we need to identify which other employee supervises them. Some employees serve as “area managers.” Each outlet is managed by one area manager. The area managers may manage up to 3 outlets.

For each rental agreement¹ (agreement with client about renting a car), we need to record rental date, rental time, and the number of miles driven before rental. When the car is returned the date and time is also recorded, as well as the mileage (i.e., odometer reading after car is returned). The rental agreement must also include the insurance type used.

When a car is returned, it is checked by one of the employees (typically a mechanic on duty at the time) for any “faults”. A fault report is created only when there are “issues” with the returned car. When needed, the employee prepares a fault report for the vehicle. The fault report is very simple and includes only date & time and comments. Once a vehicle has received 5 fault reports, that vehicle is put out of service and no longer available for rentals. We also need to link the fault report to individual Rental Agreement, in case a follow up is needed. The ERD and schema for the model is shown below.

¹ The entity name is shortened to “RAGREEMENT” in the ERD and schema.

ER Diagram:



Schema:

EMPLOYEE (EmpNo, FName, Lname, Position, Phone, Email, DOB, Gender, Salary, HireDate, OutNo@, SupervisorNo@)

FAULTREPORT (ReportNum, DateChecked, Comments, EmpNo@, LicenseNo@, RentalNo@)²

OUTLET (outNo, Street, City, State, ZipCode, Phone, ManagerNo@)

VEHICLE (LicenseNo, Make, Model, Color, Year, NoDoors, Capacity, DailyRate, InspectionDate, outNo@)

CLIENT (clientNo, ClientName, Street, City, State, ZipCode, WebAddress, Contact_FName, Contact_LName, Phone, Email)

RAGREEMENT (RentalNo, StartDate, ReturnDate, MileageBefore, MileageAfter, InsuranceType, ClientNo@, LicenseNo@)³

² *DateChecked* attribute stores both the date and the time when the report was created

³ Both *StartDate* & *ReturnDate* attributes store date and time information of each transaction

Part II.

A. Create two separate SQL scripts for creating all necessary tables, data integrity constraints, and for inserting data into the tables to implement the database as defined above.

1. Script #1: Write all CREATE TABLE statements that have the necessary integrity constraints including primary keys, foreign keys, etc. Start the script with a set of “Drop Table” statements, that will allow you to clean up the database before creating it (very useful when you recreate the database using the script). Use the names of tables as well as the names of attributes exactly as provided in the conceptual model above. *All constraints, except one, must be created and enforced within the “Create Table” statements and must be named using the convention discussed in class (lecture 12).* Run the script to create the DB tables and to create the missing constraint (as a last statement in your script). Your CREATE TABLE statements should also include the following:
 - a. The *HireDate* should have a default value that is current date.
 - b. The age of the vehicle must be less than 10 years, and
 - c. Three more appropriate and useful check constraints of your choice (but not the “Not Null” or “Unique” constraints).
2. Script #2: After the DB tables and all data integrity constraints are created, insert about ten rows of data into each table. Provide data that would allow you to test all queries as defined in section B below (each query must give you results). It may be required to add more than 10 rows in some tables in order to show that your queries work as intended. *Outlet* table can have few rows. Run the script to perform the task of populating the tables. (*Note: All constraints must be enabled and enforced – not deferred – before you populate your tables.*)
3. After all tables are populated, list the complete content of each table. Format each table’s contents to make sure it is readable.

B. Create SQL queries to answer the following questions

1. For each rental agreement, the company needs a detailed report that includes the following information: Rental number, start date and time, return date and time, mileage information (before and after), vehicle information (license number, outlet number, make, model, and year). If the rental required a fault report, also include the date the vehicle was checked.
2. For each outlet, provide number of vehicles at the outlet, number of rentals from the outlet started within last year, average distance driven per rental for last year, number of employees working at the outlet, and the number of rentals in the last year per employee. As the last row provide the grand total (i.e., the measures aggregated for the whole rental company).
3. The company needs to keep track of how well each outlet is performing each month of past year. Create a cross-tab query that lists revenue generated in each outlet and each month of past year. The last three columns should provide the total revenue for the month, number of rentals for the month, and the revenue per rental generated within the month. As last row provide total revenue in each outlet for the year, overall revenue for the year, the total number of rentals, and revenue per rental generated within the year. Sort results in chronological order of the months. Include zero in each cell where we do not have any values.

(Note: In this organization, revenue is defined as money that is collected when the vehicle is returned. As a result, the date the vehicle is returned will define when the revenue is generated.)

4. The car rental company has been operating for 7 days per week for the past six months. Based on the rentals from the past six months, the company wants to determine which day of the week could be a day off for its employees in order to have minimum impact on its business. To do that, we need a cross-tab query that lists the number of rentals and number of fault reports initiated within the six month period for each outlet – day of the week combination. The last column should give the outlet totals and last row should give the weekday totals.
5. For every outlet-manager combination, list number of rentals, the revenue per rental, and the number of fault reports per rental. Also include grand totals for each manager. Identify each manager by their employee number as well as their names.
6. Identify outlet(s) that generated the maximum revenue in the period of two quarters: last quarter of last year and first quarter of this year. Append to the result the outlet that generated minimum revenue. In each of the cases provide the revenue.
7. For each car make and model combination that is represented at any outlet, list the number of cars we have, average age of those cars, the number of rentals within this year, total number of days cars were rented this year, and number of fault reports that been generated this year. For each of the metrics, provide subtotals for “make” and the grand total. If we have cars of any make/model combination that have not been rented, then list 0 for the metrics that relates to rentals.
8. We would like to assess the likelihood of getting a fault report for each of the vehicle “makes” we have. Calculate the needed measure for each quarter of last year and for each “make”. Rank each make within each quarter when the vehicles were rented. The higher the chance of getting a fault report, the higher the rank. Also include the number of rentals of each make within quarter.
9. To understand the customers’ behavior we want to evaluate vehicle rentals in the context of locations. For each outlet, list the number of customers (who rented cars from the outlet) who live in the same state as the location of the outlet. Also include the number of rentals by those customers. Also include for each outlet, the proportion of the customers from within the state comparing with all customers in the outlet and the proportion of rentals from those customers compared with all rentals in the outlet.
10. List employee ID and name, his/her title, the ID and street address of the outlet where he or she works, and the number of fault reports prepared by the employee in the past 90 days. Show zero for the employ(s) who did not prepare any reports during the 90 days period. For each employee show the “level” in the “supervising hierarchy” and indent the name accordingly to the level so that the name of a person who is lower in the hierarchy is indented more than their supervisor.

Part III.

Create a title page that includes your names, email addresses, and the group number as it was assigned to your group. Clearly identify each part of your submitted project and each answer. Submit the following:

1. Printout of the two separate SQL script files (*Script#1 & Script#2*) as you created in Part II.A. as well as the content of the tables. (*Note: Do not print the “spool file” of executing the scripts*)
2. The queries created in Part II.B. Include both the query statements as you execute them and the results of executing the queries.

Grading

Projects will be graded based on the following criteria:

- Correctness of SQL statements based on requirements provided. Although there may be more than one way to approach each question, the SQL results must match the requirements as defined in each question.
- Readability of your SQL statements and results.
- Punctuality. The deadline for the submission is 3 p.m. on December 3rd.
 - Projects submitted on December 3rd but later than 3 p.m. will receive a 25% penalty
 - Projects will not be accepted after December 3rd.