C02013 Assignment 2 – B

Notes

- Students should read *everything* presented below carefully.
- This assignment is worth **20%** of the overall grade.
- Assignment 2 is group work.

Submission

You should submit your coursework to your ***lab*** tutor on Wednesday 30th November 2016.

Each group has to submit the following items:

- A paper report.
- A SQL script file.

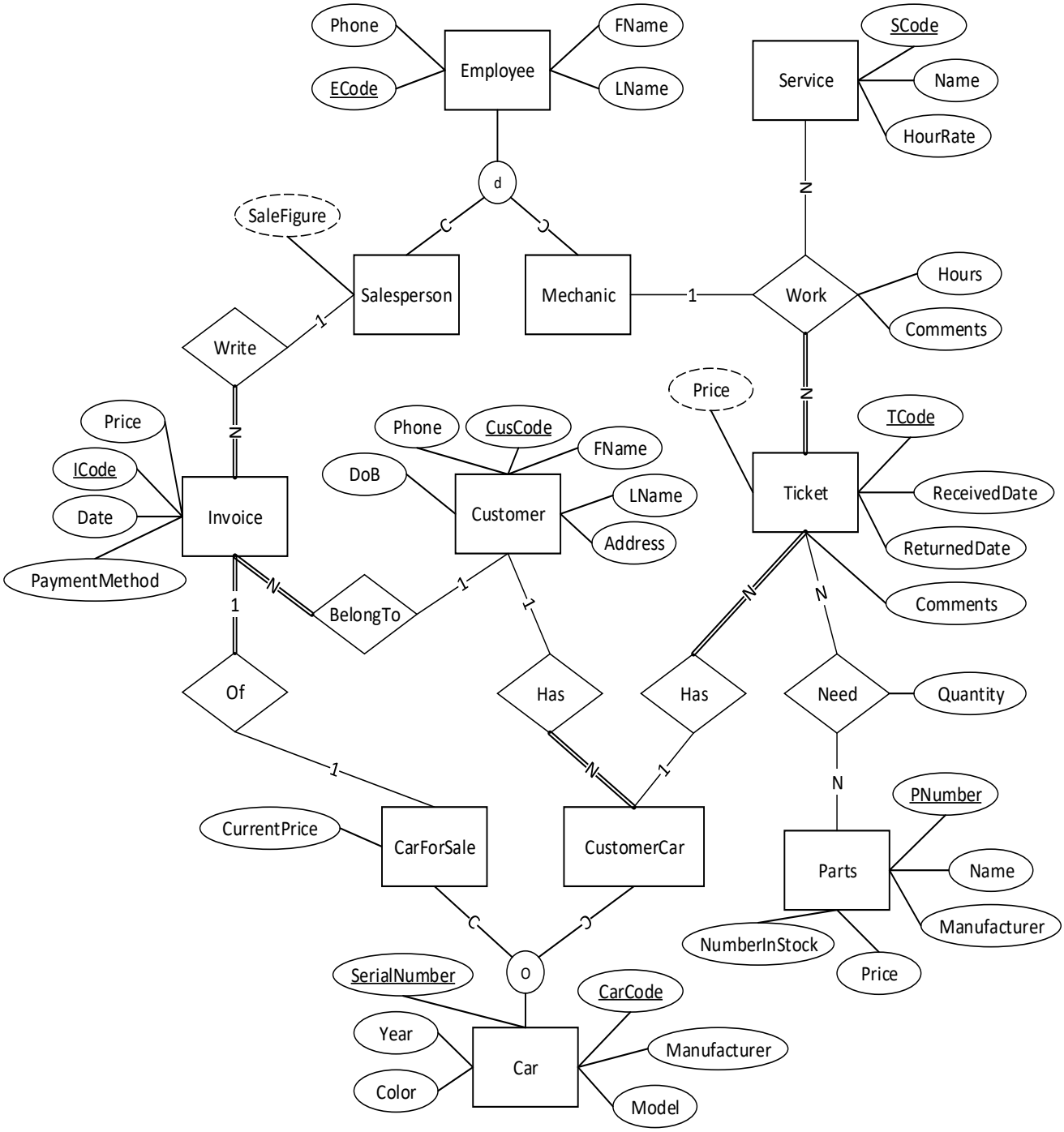
Requirements

In the assignment 2, you are expected to implement and query the database as shown in following pages, using Oracle:

Part 1: (5p)

A- Implementing the database (3p)

Given the following EERD and its relational database schema, students are required to implement it using ORACLE DBMS.



EMPLOYEE			
Column name	***PK***	***FK***	***Constraints - Notes***
ECode	x		If an employee is a salesperson, his/her ECode must obey the pattern "S[0-9]^3". The number following "S" is automatically increased. Example: S001, S002,… If an employee is a mechanic, his/her ECode must obey the pattern: "M[0-9]^3". The number following "SP" is automatically increased. Example: M001, M002,…
FName			Not Null
LName			Not Null
Phone			It is a string of 10 or 11 digits and starts with the digit '0' Not Null
IsSalesperson			0: No 1: Yes Not Null
SaleFigure			(*Reference: Part 1 – B – 1*) If IsSalesperson = 0, SaleFigure is Null. If IsSaleperson = 1, SaleFigure is Not Null. And SaleFigure >= 0 Default = 0
IsMechanic			0: No 1: Yes Not Null

CUSTOMER			
Column name	***PK***	***FK***	***Constraints - Notes***
CusCode	x		Auto increment ID
FName			Not Null
LName			Not Null
Phone			It is a string of 10 or 11 digits and starts with the digit '0' Not Null
Address			Not Null
DoB			
Email			A valid email format. *(Simple email format: a string of any number of [0-9], [a-z], [A-Z] , '.' , '_' and a single '@')*
CarCode		x	CAR.CarCode CarCode must belong to a car whose IsCustomerCar is 1.

CAR			
Column name	***PK***	***FK***	***Constraints - Notes***
CarCode	x		
SerialNumber			Unique Not Null
Manufacturer			Not Null
Model			Not Null
Year			Not Null
Color			Not Null
IsCarForSale			0: No 1: Yes Not Null
CurrentPrice			If IsCarForSale = 1, CurrentPrice must be Not Null If IsCarForSale = 0, CurrentPrice must be Null

IsCustomerCar			0: No 1: Yes Not Null

INVOICE			
Column name	***PK***	***FK***	***Constraints - Notes***
ICode	x		
Date			Not Null
PaymentMethod			{Cash, Credit Card, Bank transfer}
SalesPersonCode		x	EMPLOYEE.ECode SalesPersonCode must belong to a salesperson Not Null
CusCode		x	CUSTOMER.CusCode Not Null
CarCode		x	CAR.CarCode CarCode must belong to a car for sale. Not Null
Price			(*Reference: Part 1 – B – 2*) Not Null

SERVICE			
Column name	***PK***	***FK***	***Constraints - Notes***
SCode	x		
Name			Not Null
HourRate			Price for a working hour HourRate >= 0 Not Null

TICKET			
Column name	***PK***	***FK***	***Constraints - Notes***
TCode	x		
ReceivedDate			Not Null
ReturnedDate			ReturnedDate >= ReceivedDate Not Null
Comments			
Price			(*Reference: Part 1 – B – 3*)

TICKET-SERVICE			
Column name	***PK***	***FK***	***Constraints - Notes***
TCode	x	x	TICKET.TCode
SCode	x	x	SERVICE.SCode
MechanicCode		x	EMPLOYEE.ECode MechanicCode must belong to a mechanic. Not Null
Hours			Hours = 0.5 x N and N is an integer and N > = 1 (Example: 0.5, 1, 1.5 …) Not Null
Comments			

PART			
Column name	***PK***	***FK***	***Constraints - Notes***
PNumber	x		
Name			Not Null

Manufacturer			Not Null
Price			Price >= 0 Default = 0 Not Null
NumberInStock			NumberInStock >= 0 Not Null

TICKET-PART			
Column name	***PK***	***FK***	***Constraints - Notes***
TCode	x	x	TICKET.TCode
PNumber	x	x	PART.PNumber
Quantity			Quantity >= 0 Default = 1 (*Reference: Part 1 – B – 4*) Not Null

B- Constraints (1.5p)

Write triggers to ensure the following constraints or business rules:

1. In table EMPLOYEE, Sale figure is the sum of prices of all invoices which are written by him/her.
2. Selling price in an invoice must be equal or less than the *current price* in the CAR table. However, the maximum discount value cannot exceed 10% of the *current price*.
3. Price of a ticket is the sum of prices of services and parts.
4. A. When INSERT a record to table TICKET-PART, quantity must be smaller than the NumberInStock in table PART at that time (for a same part number).
   B. If INSERT is successful, the NumberInStock must be updated accordingly.
5. In a period of time from ReceivedDate to ReturnedDate, a car cannot be issued two or more tickets.

(*) If you have any questions, please consult your lab tutor.

C- Index (0.5p)

Database users often query the following information:

1. Get a customer's information by his/her name.
2. Get a car's information by its serial number.

Based on the common queries, you should create the index on the proper fields in order to increase the system performance.

Part 2: INSERT, UPDATE, DELETE, SELECT (2p)

1. You are required to insert valid and meaningful data into the database. Each table has at least 4 rows. (1p)
2. Change the HourRate of the service whose name is "Car Wash" to 50 000. (0.25p)
3. Delete a mechanic whose name is "Nguyễn Văn Ân". (0.25p)
4. Retrieve all salesperson who did not sell any cars in October 2016. (0.25p)
5. Count the number of cars TOYOTA (Manufacturer) INNOVA (Model) 2015 (Year) which were replaced the part number *01234*. (0.25p)

Part 3: STORE PROCEDURE, FUNCTION (2p)

1. Create a procedure/function to insert into table CAR.
   Input: SerialNumber, Manufacturer, Model, Year, Color, Type ("Car for sale" or "Customer's car"), Customer code (if its type is "Customer's car")
   Output: SUCCESSFUL or FAIL. If FAIL, show the error message. (1p)

2. Create a procedure/function to find mechanics who works less than 25 hours per month for two consecutive months in a provided year.
   Input: Year
   Output: List of mechanics' code, name, the two consecutive months, and two number of working hours in these months. (1p)

Part 4: ACCESS CONTROL (1p)

1. Login to the database with a DBA user (SYS, SYSTEM, SYSMAN).
2. Create 2 users:
   - User *ServiceUser*.
   - User *SalesUser*.
3. Grant the 2 users the privilege/role to connect to the database.
4. *ServiceUser* is responsible for all services, parts and customers and cars coming for services. *ServiceUser* can assign a mechanic for a service. However, he/she cannot modify information of a mechanic, and see any information of salepersons. Grant proper privileges to *ServiceUser*.
5. *SalesUser* is responsible for cars for sale, customers, and invoices. Grant proper privileges to *SalesUser*.
6. Disallow *SalesUser* to DELETE on CUSTOMER.