Assignment 1 – RDBMS

Jing Hua Ye, CIT 10/02/2019

1 Conceptual Design and ERD

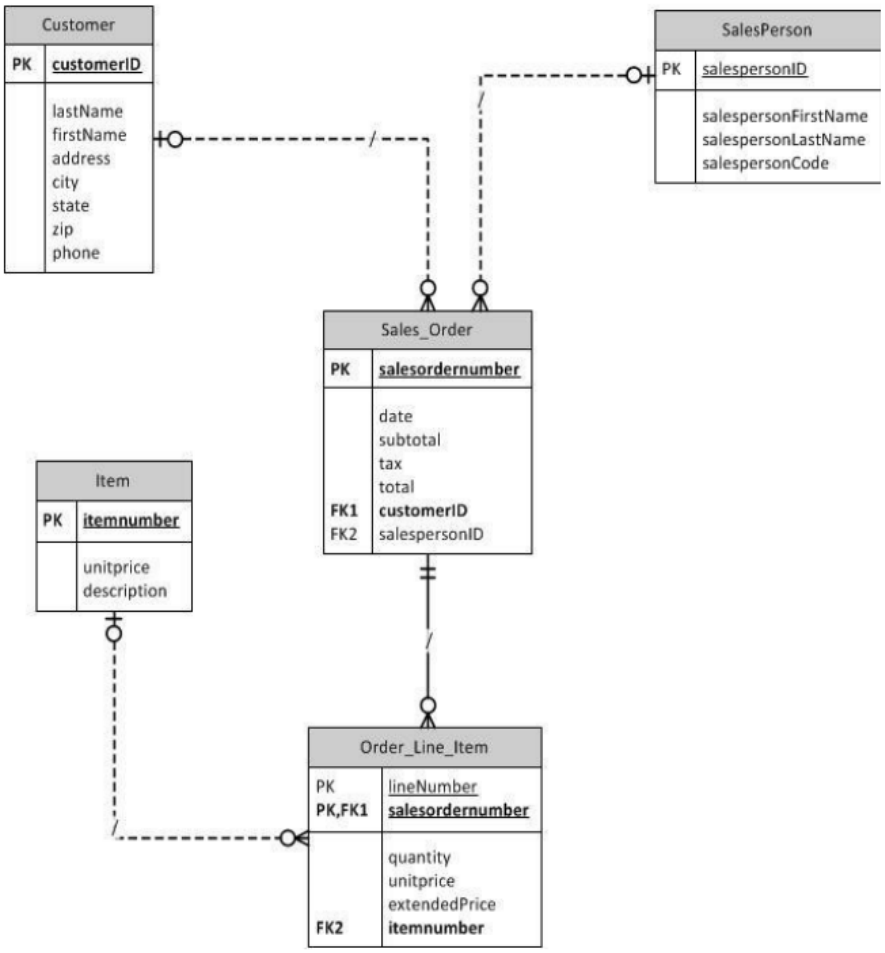
The Prescriptions-R-X chain of pharmacies has offered to give you a free lifetime supply of medicine if you design its database. Given the rising cost of health care, you agree. Here's the information that you gather:

- Patients are identified by an SSN, and their names, addresses, and ages must be recorded.
- Doctors are identified by an SSN. For each doctor, the name, speciality, and years of experience must be recorded.
- Each pharmaceutical company is identified by name and has a phone number.
- For each drug, the trade name and formula must be recorded. Each drug is sold by a given pharmaceutical company, and the trade name identifies a drug uniquely from among the products of that company. If a pharmaceutical company is deleted, you need not keep track of its products any longer.
- Each pharmacy has a name, address, and phone number.
- Every patient has a primary physician. Every doctor has at least one patient.
- Each pharmacy sells several drugs and has a price for each. A drug could be sold at several pharmacies, and the price could vary from one pharmacy to another.
- Doctors prescribe drugs for patients. A doctor could prescribe one or more drugs for several patients, and a patient could obtain prescriptions from several doctors. Each prescription has a date and a quantity associated with it. You can assume that, if a doctor prescribes the same drug for the same patient more than once, only the last such prescription needs to be stored.
- Pharmaceutical companies have long-term contracts with pharmacies. A pharmaceutical company can contract with several pharmacies, and a pharmacy can contract with several pharmaceutical companies. For each contract, you have to store a start date, an end date, and the text of the contract.
- Pharmacies appoint a supervisor for each contract. There must always be a supervisor for each contract, but the contract supervisor can change over the lifetime of the contract.

Use Visual Paradigm to draw a conceptual diagram and a ERD.

2 Logical Design

Translate the following ER diagram into a relational database schema.



3 SQL

Screenshot for the answer and the output of each query is required. The bold underline font represents a PK and red bold font represents a FK.

The database 'HotelBooking' holds the room booking information for a hotel. It contains the following tables:

- hotel(**hotelNo**, hotelName, city)
- room(**roomNo, hotelNo**, type, price)
- guest(**guestNo**, guestName, guestAddress)
- booking(**bookingNo**, dateFrom, dateTo, **guestNo**, **hotelNo**, **roomNo**)

The file setup_hotel.sql is a script which creates the database, each table, inserts some entries on each table and commits the changes to disk. Open MySQL and run the script to setup and start working with the database.

Develop the following SQL queries:

1. Update the price of all rooms by 5 percent.
2. List all double or family rooms with a price below $40 per night, in ascending order of price.

3. What is the average price of a room?
4. List the rooms that are currently unoccupied at the Hilton Hotel.
5. List the number of rooms in each hotel in Washington.
6. List the bookings for which no dateTo has been specified.
7. List full details of all hotels in London
8. How many hotels are there?
9. What is the total revenue per night from all double rooms?
10. Suppose Mary wants to cancel all her hotel bookings, create a command to delete all Mary's hotel bookings.

4 Transaction

Screenshot for the answer and the output is required. We will now create a new table named "T", having three columns: id (of type integer, the primary key), s (of type character string with a length varying from 1 to 40 characters), and si (of type small integer):

```
CREATE TABLE T (id INT NOT NULL PRIMARY KEY, s VARCHAR(40), si SMALLINT);
```

It is now time to append/insert some rows to the newly created table:

```
INSERT INTO T (id, s) VALUES (1, 'first');
INSERT INTO T (id, s) VALUES (2, 'second');
INSERT INTO T (id, s) VALUES (3, 'third');
SELECT * FROM T ;
```

An attempt is made to cancel/rollback the current transaction, by issuing the following command:

```
ROLLBACK;
SELECT * FROM T ;
```

It appears to have worked, however: after issuing a new "SELECT * FROM T" command, the table is seen to continue registering the three rows. A surprise ...

The source of the surprise has a name: "AUTOCOMMIT". MySQL starts in the AUTOCOMMIT mode in which every transaction need to be started by "START TRANSACTION" command, and after end of the transaction MySQL returns to the AUTOCOMMIT mode again. To verify the above, the following set of SQL commands is executed:

```
START TRANSACTION;
INSERT INTO T (id, s) VALUES (4, 'fourth');
SELECT * FROM T ;
ROLLBACK;
SELECT * FROM T;
```

The following statements are executed:

```
INSERT INTO T (id, s) VALUES (5, 'fifth');
ROLLBACK;
SELECT * FROM T;
```

What is the result set obtained by executing the above SELECT * FROM T statement?