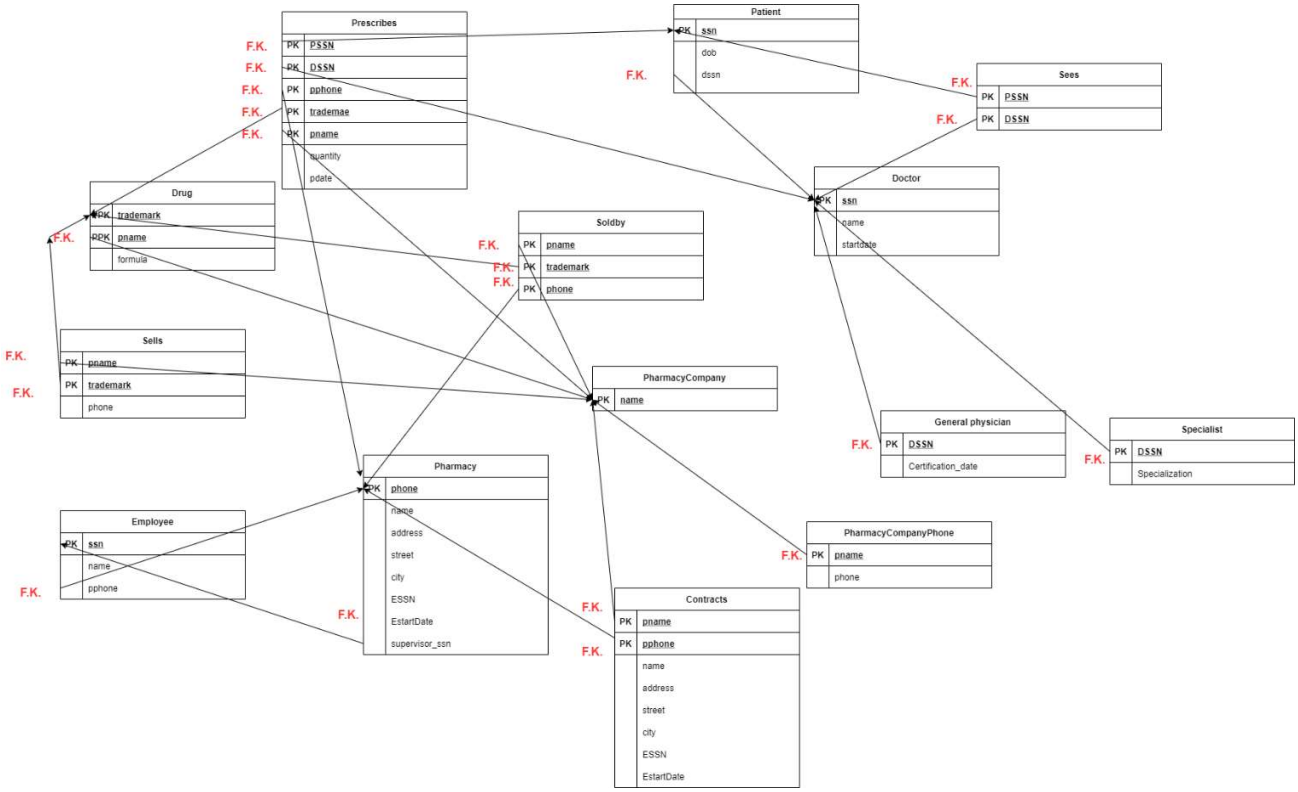
Assignment 4 Suggested Solutions

Task 1

Define relations with associated attributes, primary keys, alternate keys and foreign keys for the complete conceptual model described later in the assignment.



OR

Patient (ssn, name, dob,dssn)

PK ssn

FK dssn REFERENCES Doctor (ssn)

Doctor (ssn, name, startdate)

PK ssn

General_Physician (DSSN, certificate_date)

PK DSSN

FK DSSN REFERENCES Doctor (ssn)

Specialist (DSSN, specialization)

PK DSSN

FK DDSN REFERENCES Doctor(ssn)

Sees (PSSN, DSSN)

PK PSSN, DSSN

FK PSSN REFERENCES Patient (ssn)

FK DSSN REFERENCES Doctor (ssn)

Primary (PSSN, Dssn)

PK PSSN, DSSN

FK PSSN REFERENCES Patient (ssn)

FK DSSN REFERENCES Doctor (ssn)

Pharmacy (phone, name, address, street, city,supervisor_ssn)

PK phone

FK supervisor_ssn REFERENCES Employee (ssn)

PharmacyCompany (name, phone)

PK name

Contracts (pname, pphone, sdate, edate, text)

PK pname, pphone

FK pname REFERENCES PharmacyCompany(name)

FK pphone REFERENCES Pharmacy (phone)

Drug (pname, trademark, formula)

PK pname, trademark

FK pname REFERENCES PharmacyCompany (name)

Soldby (pname, trademark, pphone)

PK pname, trademark, pphone

FK pname, trademark REFERENCES Drug (pname, trademark)

FK pphone REFERENCES Pharmacy(phone)

Sells (pname, trademark, pphone)

PK pname, trademark, pphone

FK pname,trademark REFERENCES Drug (pname, trademark)

FK pphone REFERENCES Pharmacy (phone)

Prescribes(PSSN, DSSN pphone, trademark, pname, quantity, pdate)

PK PSSN, DSSN, pphone, trademark, pname

FK PSSN REFERENCES Patient (ssn)

FK DSSN REFERENCE Doctor (ssn)

FK pphone REFERENCES Pharmacy (phone)

FK trademark REFERENCES Drug (pname, trademark)

Employee (ssn, name, pphone)

PK ssn

FK pphone REFERENCES Pharmacy(phone)

Task 2

In addition to creating a relational schema for the database, you should also create tables resulted from Task 1 into MySQL. Create SQL export file and submit with the report.

```
CREATE DATABASE IF NOT EXISTS hospital COLLATE = utf8_general_ci;
 USE hospital;

CREATE TABLE PharmacyCompany (
name varchar (100) NOT NULL PRIMARY KEY
phone varchar (10)
);

CREATE TABLE Drug (
Pname varchar (255)
trademark varchar (255) NOT NULL PRIMARY KEY
formula varchar(255),
);

CREATE TABLE Pharmacy (
phone varchar (10) NOT NULL PRIMARY KEY,
name varchar (100),
address varchar (255)
street varchar (255),
city varchar (100),
);

CREATE TABLE Contracts(
Company_name varchar (100) NOT NULL,
Pahrmacy_phone varchar (10) NOT NULL,
);
```

```
CREATE TABLE sells (
Company_name varchar(255) NOT NULL,
trademark varchar (255) NOT NULL,
Pharmacy_phone varchar (10) NOT NULL,
);

CREATE TABLE Soldby (
Company_name varchar(255) NOT NULL,
trademark varchar (255) NOT NULL,
Pharmacy_phone varchar (10) NOT NULL,
);

ALTER TABLE Drug
ADD CONSTRANT Drug_PharmacyCompany_Fk FOREIGN KEY (pname) REFERENCES PharmacyCompany (name);

ALTER TABLE Contracts
ADD CONSTRAINT PK_contact PRIMARY KEY (Company_name, Pahrmacy_phone),
ADD CONSTRANT Contracts_PharmacyCompany_Fk FOREIGN KEY (Company_name) REFERENCES PharmacyCompany (name),
ADD CONSTRANT Contracts_Pharmacy_Fk FOREIGN KEY (Pharmacy_phone) REFERENCES Pharmacy (phone);

ALTER TABLE Sells
ADD CONSTRAINT PK_sells PRIMARY KEY (Company_name, trademark, Pharmacy_phone),
ADD CONSTRAINT Sells_Drug_FK FOREIGN KEY (Company_name, trademark) REFERENCES Drug(Company_name, trademark),
ADD CONSTRAINT  Sells_Pharmacy_FK FOREIGN KEY (Pharmacy_phone) REFERENCES Pharmacy (phone);

ALTER TABLE Soldby
ADD CONSTRAINT PK_sells PRIMARY KEY (Company_name, trademark),
```

ADD CONSTRAINT Soldby_Drug_FK FOREIGN KEY (Company_name, trademark) REFERENCES Drug(Company_name, trademark),

ADD CONSTRAINT Soldby_PharmacyCompany_FK FOREIGN KEY (Pharmacy_phone) REFERENCES PharmacyCompany (phone);

Task 3

1. What should be the primary key of the table?

   (limo_id, journey_date, start_time)

2. List the functional dependencies related to the table.

FD_1: limo_id -> limo_registration, class

FD_2: driver_id -> driver_name

FD_3: (limo_id, journey_date, start_time) -> driver_id

FD_4: class -> price, limo_capacity

3. In which normal form is this relation? Explain your answer.

   This is in 1NF since there are no repeated groups in the table. (In 1NF, multi-value attributes are stored in separate tables). It is not in 2NF, due to FD1.

4. Convert the table to 3NF.

   1NF -> 2NF:
   Split the table in two – move the attributes that are not dependent on the complete key to a separate table:

   Journey(limo_id, journey_date, start_time, driver_id, driver_name)

   Limo(limo_id, limo_registration, class, limo_capacity, price(NOK))

   The tables are now in 2NF but not in 3NF, due to FD2 and FD4
   2NF -> 3NF: Split the table such that transitive dependent attributes are moved to separate tables:

   Journey(limo_id, journey_date, start_time, driver_id)
   Limo(limo_id, limo_registration, class)
   Limo_type(class, limo_capacity, price(NOK))

Driver(driver_id, driver_name)

Journey

limo_id	journey_date	start_time	driver_id
L1	20.02.21	10.00	D1
L1	20.02.21	13.00	D1
L1	21.02.21	10.00	D1
L2	20.02.21	10.00	D2
L2	22.02.21	14.00	D2
L2	23.02.21	11.00	D2

Limo

limo_id	limo_registration	class
L1	DN3526	1
L2	CY2534	2

Limo_type

class	limo_capacity	price(NOK)
1	8	400
2	12	600

Driver

driver_id	driver_name
D1	Pete
D2	Jane

5. Are the tables you created in task 4 in BCNF too? Convert the tables to BCNF if not.

The tables are also in BCNF, as:

- the determinant in FD_1 (limo_id) is the PK of Limo,
- the determinant in FD_2 (driver_id) is the PK in Driver
- the determinant in FD_3 (limo_id) is the PK in Journey
- the determinant in FD_4 (limo_id, journey_date, start_time) is the PK in Limo_type