#### **Lambton College - Term 1**

#### **CSD-2204: Database Design and SQL**

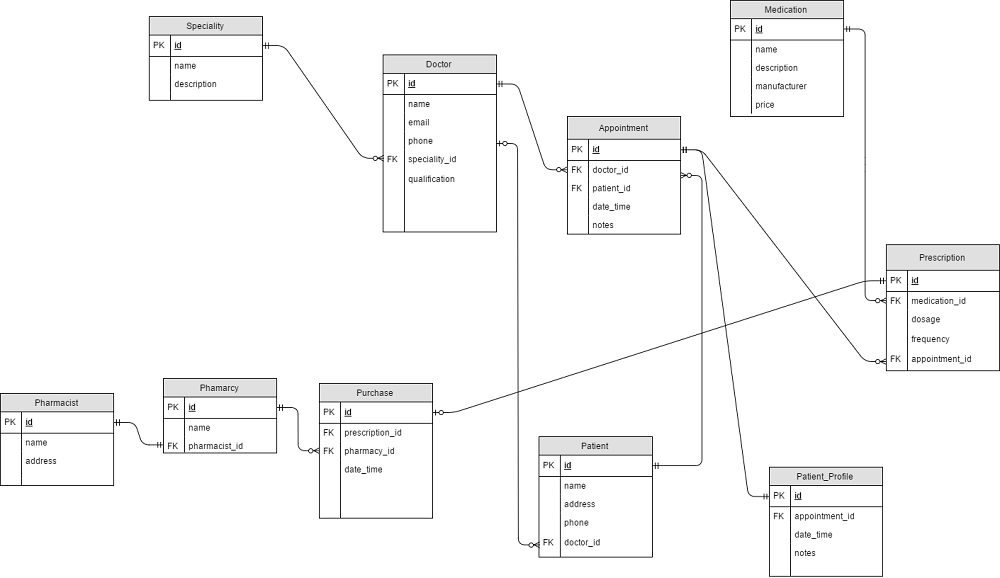
#### 

#### **C0702741**

#### **C0676975**

#### **C0704250**

E-R diagram clinic database



Description of relations E-R diagram:

**Business Rules**:

1. The clinic has several doctors

* The table doctor makes possible to the clinic register many doctors.

2. Each doctor has a speciality (GP, Cardiology, Pediatrics, Children, etc...)

* 1 doctor has 1 speciality and 1 speciality can be in zero or many doctors. To fit this requirement speciality id goes to the Doctor table.

3. Each patient has a primary doctor

* One patient can have only one main doctor but one doctor can have zero or many patients. The Doctor attribute in patient entity shows the patient’s main doctor (family doctor).

4. Each patient may have several prescriptions (may take several medications)

* The appointment entity is working as associate entity between patient and prescription entity where the patient can have zero or many appointment and each appointment may have one or many prescriptions.

5. If the patient's primary is not available, then any available doctor might write the prescription to this patient.

* Either the primary doctor is not available yet the patient will be treated by only one doctor. The table appointment makes it possible, any doctor can treat any patient. To know who is the patient’s primary doctor we use the doctor\_id inside Patient table.

6. For the Term Project, add the pharmacy's functionality as follow:

a. The clinic has only one pharmacy

* 1 clinic can have only one pharmacy and one pharmacy can work under one and only one clinic.

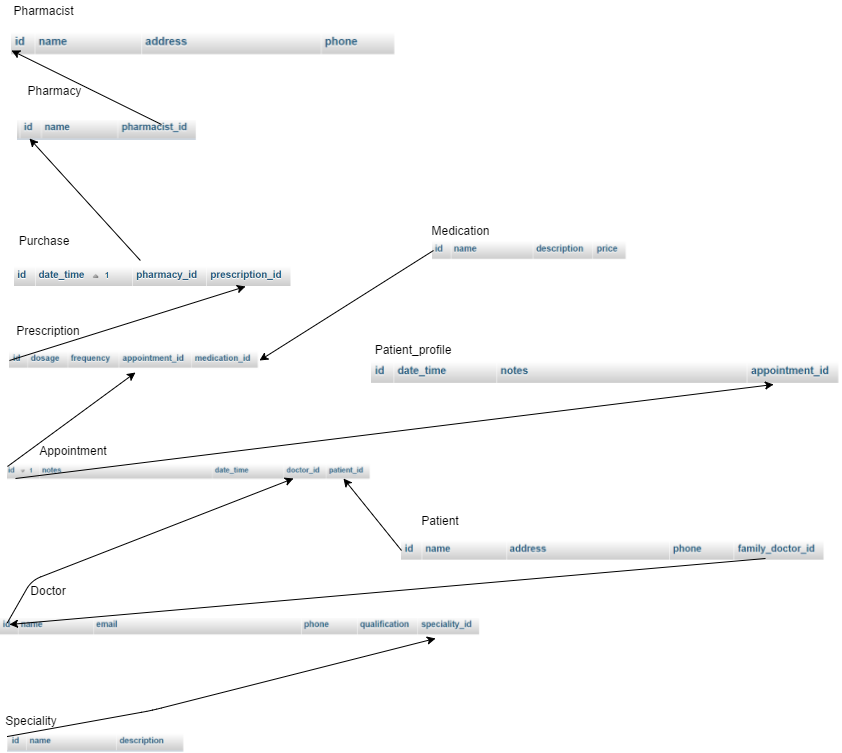
b. The pharmacy may has one and only one pharmacist

* The pharmacist\_id attribute in the pharmacy entity shows that one Pharmacy can have only one pharmacist.

c. The patient may purchase the medication(s) given in the doctor's prescription(s) from the clinic's pharmacy.

* The prescription entity is working as associative entity between purchase and medication entities, each medication is listed on one prescription. The patient can purchase medication from one and only one pharmacy and, the pharmacy can sell zero or many medications to the patient through the prescription.

Relational Model:



Documentation of clinic database

1. **Doctor\_entity:** It contains attributes as: id,name,email,phone,speciality\_id,qualification

**Id :** is a **primary key**  attribute .

**Name** : Represents name of the doctor.

**Email :** Doctor’s email address.

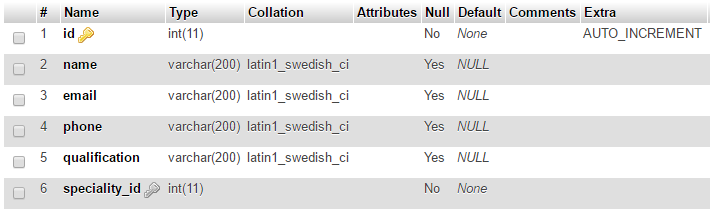
**Phone:** Phone number of the doctor.

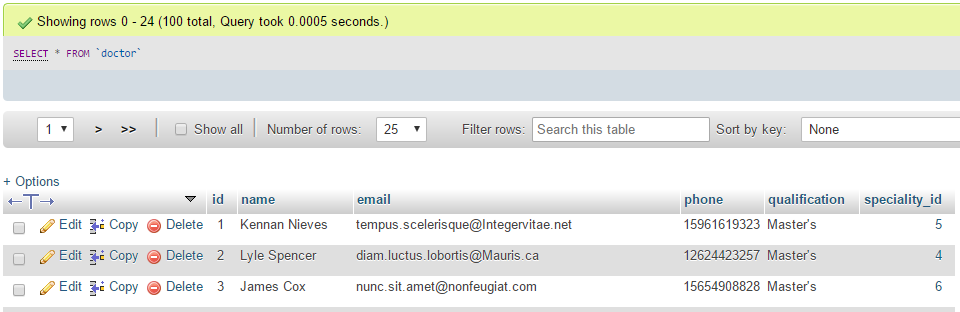
**Speciality\_id :** It is a foreign key attribute used to contact both **Doctor**  and **Speciality** entities. Which is important to get the speciality

of the doctor.

**Qualification:** It contains data about doctor’s qualification.

**For example:** **1, Dr. Kamal,** [Dr.Kamal,abc@jhs.com](about:blank),234-846-8478,**45**, Gynecologists





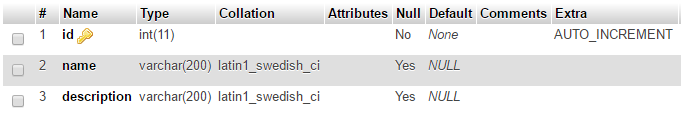
**2) Speciality\_entity:** It contains attributes as:id,name,description

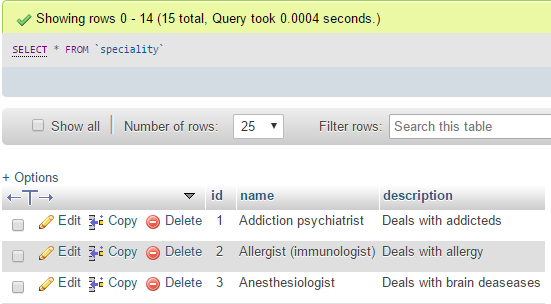
**Id:** is a **primary key**  attribute.

**Name:**Represents name of the speciality.

**Description:**It tells that what is the working in real life of speciality.

**For example:45,** Gynecologists,Gynecologists (OBGYNs) are medical professionals who specialize in pregnancy, childbirth and the female reproductive system. These physicians can **work** in clinics, hospitals and other medical facilities





**3) Appointment\_entity:**It contains attributes as:id,**doctor\_id**,**patient\_id**,date/time,notes

**Id:** is a **primary key**  attribute.

**Doctor\_id:**It is a foreign key attribute used to contact both **Doctor**  and **Appointment** entities. It tells that which doctor set

appointment for which patient.

**Patient\_id:**It is a foreign key attribute used to contact both **patient**  and **Appointment** entities. It tells that which patient get

appointment from which doctor .

**Date/Time:**It’s for knowing the exact date as well as time of the

appointment for both patient and doctor.

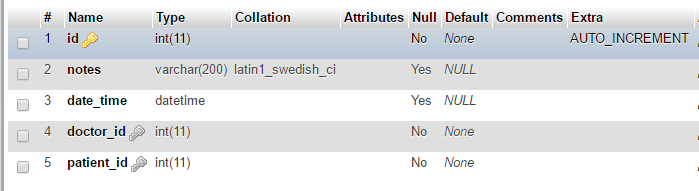
**Notes:** It haves each and every information about appointment.

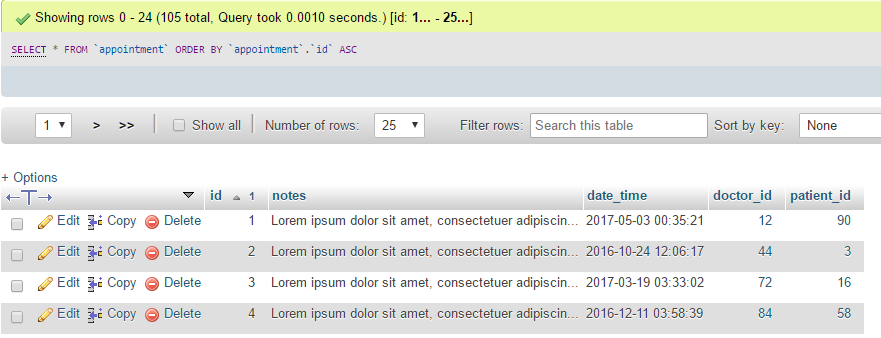
**Example:524, 23 , 73,** 12/05/2017-5:30 pm, this appointment is

issued by Mr. A on phone.Patient name Jack has

appointment with doctor Mr. Duston regarding stomach

Pain.





**4) Patient\_entity:** It contains attributes as: id,name,address,phone,**doctor-id**

**Id:** is a **primary key**  attribute.

**Doctor\_id:**It is a foreign key attribute used to contact both **Doctor**  and **Patient** entities. It tells that which doctor handles

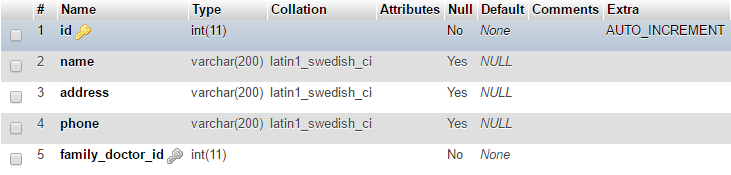
that patient.

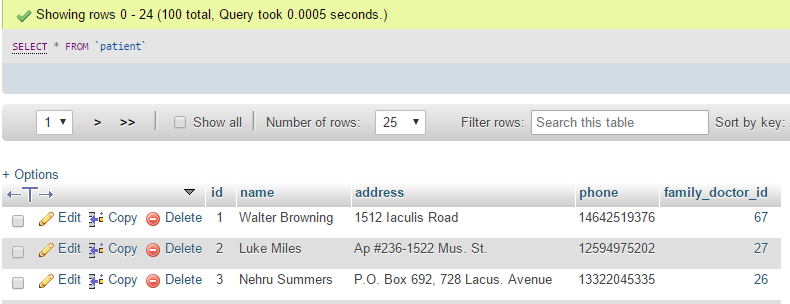
**Name** : Represents name of the patient.

**Address :** Patient's address ,its help full in case emergency.

**Phone:** Phone number of the patient.

**For example:** **1,** Mr.Karam,656 vankirk Brampton ,234-846-8478,**45**





**5)Medication\_entity:**It contains attributes as: id,name,description,manufacturer,price

**Id:** is a **primary key**  attribute.

**Name** : Represents name of the medication.

**Description:** Explains for what the medication is useful.

**Manufacturer:** The Brand who make the medication.

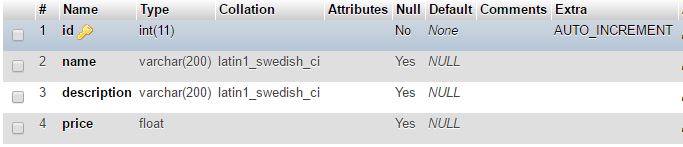
**Price:** For knowing the cost of medication.

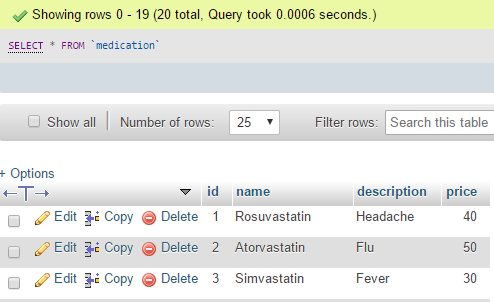
**For example:468,** tylenol, ‘headache, fever’, ‘tylenol inc’, $10.

**Insert Table Example:**

Insert into medication (id, name, description, manufacturer, price) values

(tylenol, ‘headache, fever’, ‘tylenol inc’, $10).





**6)Prescription\_entity:**It contains attributes as: id,**Medication\_id, Appointment\_id,**

frequency,dosage

**Id:** is a **primary key**  attribute.

**Medication\_id:**It is a foreign key attribute used to contact both

**Medication** and **Prescription** entities. Which shows that

which medication is prescribed to the patient by doctor.

**Appointment\_id:**It is a foreign key attribute used to contact both

**Appointment** and **Prescription** entities. Which shows

that when patent can visit to doctor according to his

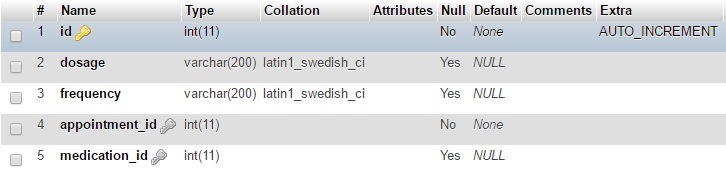
prescription.

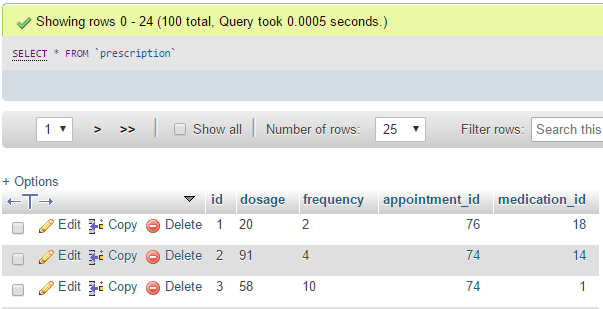
**Frequency:** It contain the information that which medicine you have to

take and how many times a day.

**Dosage:** It tells that which power of the medicine is needed.

**For example: 56,43,675,**3 times a day, 500mg





**7)Patient\_Profile:**It contains attributes as:id,**appointment\_id**,date/time,notes.

**Id:** is a **primary key**  attribute.

**Appointment\_id:**It is a foreign key attribute used to contact both

**Appointment** and **Patient\_profile** entities. Which shows

that when patent can visit to doctor according to his prescription.

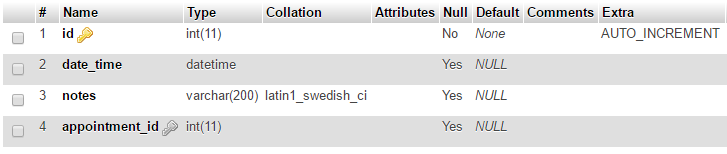
**Date/Time:**It’s for knowing the exact date as well as time of the appointment

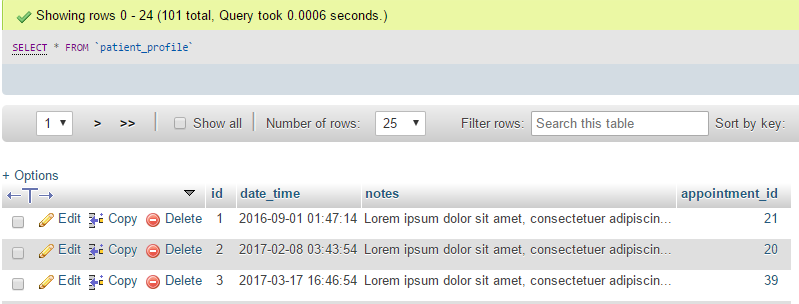
for both patient and doctor.

**Notes:** It haves each and every information about patient.

**For example:57,465,**21-05-2017/02:27 pm, patient is recovering with the

Process and medicine.





**8)Pharmacist\_entity:** It contains attributes as: id,name,address

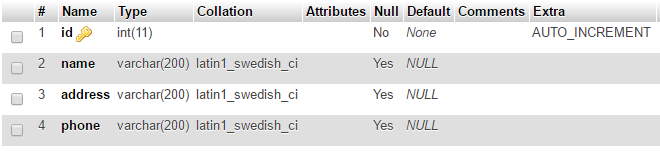
**Id:** is a **primary key**  attribute.

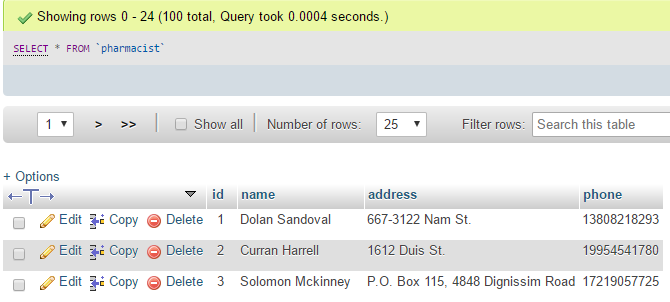
**Name** : Represents name of the pharmacist.

**Address :** Pharmacist’s address to contact or you can say to meet him

regarding any medical information.

**For example:57,**Mr. Joraz,9875 Hellroad, Toronto





**9)Pharmacy\_entity:** It contains attributes as: id,name,**Pharmacist\_Id**

**Id:** is a **primary key**  attribute.

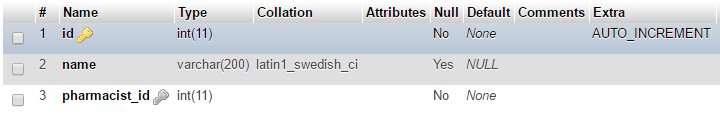
**Name** : Represents name of the pharmacy.

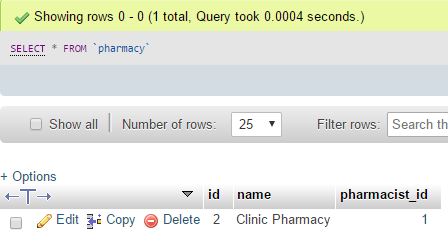
**Pharmacist\_Id :**It is a foreign key attribute used to contact both

**Pharmacy** and **Pharmacist** entities. It tells that which

pharmacist is working at which pharmacy.

**For example:57,**Durage Mart,**768**





**10) Purchase\_entity:**It contains attributes as:id, **Pharmacy\_id,Pharmacist\_id**,date/time,notes

**Id:** is a **primary key**  attribute.

**Pharmacy\_id:**It is a foreign key attribute used to contact both

**pharmacy** and **purchase** entities. It tells that which

pharmacy is providing the drug.

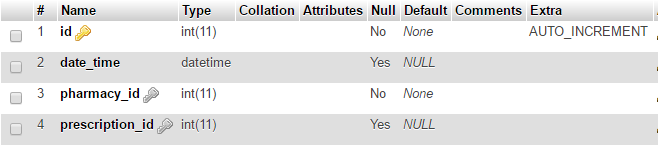
**Pharmacist\_id:**It is a foreign key attribute used to contact both

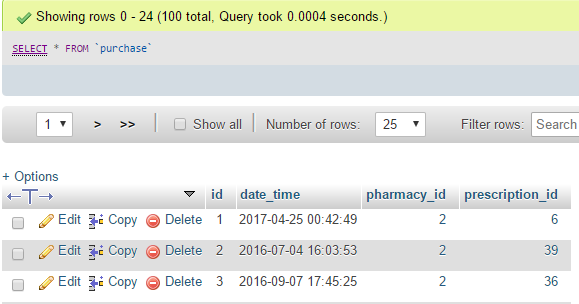
**pharmacy** and **purchase** entities. It tells that

Which pharmacist is purchasing the drug .

**Date/Time:**It’s for knowing the exact date as well as purchased

time of the medicine.





Queries:

1. Using SQL build the following views (or reports):

a. The clinic prints and distributes newsletters for its patients who are actively visiting the clinic. Active patients are identified by those who visit the clinic at least twice in the past year

SELECT p.id, p.name, p.phone, p.address, Count(p.id) as counter\_visits

FROM patient\_profile as pp INNER JOIN appointment as ap ON pp.appointment\_id = ap.id INNER JOIN patient as p ON ap.patient\_id = p.id

where pp.date\_time >= DATE\_SUB(NOW(), INTERVAL 1 YEAR) GROUP BY p.id HAVING counter\_visits >= 2

b. List the number of patients in each medical speciality

SELECT spec.name, COUNT(pat.id) as patient\_counter from speciality as spec

INNER JOIN doctor as doc ON doc.speciality\_id = spec.id

INNER JOIN patient as pat ON pat.family\_doctor\_id = doc.id

WHERE pat.family\_doctor\_id = doc.id

GROUP BY spec.id

c. List the doctors who work in the clinic sorted in descending order by the number of visits from patients in each month

SELECT doc.id as doctor\_id, doc.name as doctor\_name, COUNT(app.doctor\_id) as patient\_counter, MONTH(app.date\_time) as month, MONTHNAME(app.date\_time) as month\_name FROM appointment as app INNER JOIN doctor as doc ON doc.id = app.doctor\_id GROUP BY app.doctor\_id, month ORDER BY month ASC, patient\_counter DESC

d. In a monthly basis, the clinic needs to provide a list of doctors showing how many times a doctor examined a patient during the last month, sorted in descending order by the number of exams

SELECT doc.id as doctor\_id, doc.name as doctor\_name, COUNT(app.doctor\_id) as patient\_counter, MONTH(app.date\_time) as month, MONTHNAME(app.date\_time) as month\_name

FROM appointment as app INNER JOIN doctor as doc ON doc.id = app.doctor\_id

where MONTH(app.date\_time) = MONTH(CURDATE())

GROUP BY app.doctor\_id

ORDER BY patient\_counter DESC

e. In an annual basis, the clinic needs to provide a list of doctors showing how many times a doctor examined a patient during the last year, sorted in descending order by the number of exams

SELECT doc.id as doctor\_id, doc.name as doctor\_name, COUNT(app.doctor\_id) as appointment\_counter, YEAR(app.date\_time) as year

FROM appointment as app INNER JOIN doctor as doc ON doc.id = app.doctor\_id

where YEAR(app.date\_time) = YEAR(CURDATE())

GROUP BY app.doctor\_id

ORDER BY appointment\_counter DESC

f. In a monthly basis, the clinic needs to provide a list of doctors showing how many times a doctor worked in behalf of another doctor during the last month. Sort the list in descending order by the number of times

SELECT doc.id as doctor\_id, doc.name as doctor\_name, COUNT(app.doctor\_id) as patient\_counter, MONTH(app.date\_time) as month, MONTHNAME(app.date\_time) as month\_name

FROM appointment as app

INNER JOIN doctor as doc ON doc.id = app.doctor\_id

INNER JOIN patient as pat ON pat.id = app.patient\_id

where MONTH(app.date\_time) = MONTH(CURDATE()) AND pat.family\_doctor\_id != doc.id

GROUP BY app.doctor\_id

ORDER BY patient\_counter DESC

g. In an annual basis, the clinic needs to provide a list of doctors showing how many times a doctor worked in behalf of another doctor during the last year. Sort the list in descending order by the number of times

select [doc.name](http://doc.name/) as Doctor, count(app.doctor\_id) as no\_of\_times, year(app.date\_time) as Year

from appointment app

left join doctor doc on [doc.id](http://doc.id/) = app.doctor\_id

where app.doctor\_id <> (select family\_doctor\_id from patient p where [p.id](http://p.id/) = app.patient\_id )

and year(app.date\_time) = year(DATE\_SUB(CURDATE(), INTERVAL 1 YEAR))

group by Doctor\_id, Year

order by no\_of\_times DESC;

h. In a monthly basis, the pharmacy needs to get a list of the Medication names sorted in descending order by the number of times it is prescribed by a doctor during the last month

SELECT [med.name](http://med.name/) as Medicine, count(pres.medication\_id) as No\_of\_times\_prescribed

from purchase pur

left join prescription pres on pur.prescription\_id = [pres.id](http://pres.id/)

left join medication med on pres.medication\_id = [med.id](http://med.id/)

where month(pur.date\_time) = month(DATE\_SUB(curdate(), INTERVAL 1 MONTH)) and year(pur.date\_time) = year(DATE\_SUB(curdate(), INTERVAL 1 MONTH))

group by 1

order by 2 DESC;

i. In an annual basis, the pharmacy needs to get a list of the Medication names sorted in descending order by the number of times it is prescribed by a doctor during the last year

select [med.name](http://med.name/) as Medicine\_Name , count(pres.medication\_id) as No\_of\_times

from purchase pur

left join prescription pres on pur.prescription\_id = [pres.id](http://pres.id/)

left join medication med on pres.medication\_id = [med.id](http://med.id/)

where year(pur.date\_time) = year(DATE\_SUB(curdate(), INTERVAL 1 Year))

group by 1

order by 2 DESC;

j. In a monthly basis the pharmacy needs to get a list of all medications that are not prescribed during the last month

select DISTINCT [med.name](http://med.name/) as Medicine\_Name\_Not\_Ordered\_In\_Last\_Month

from medication med

where [med.id](http://med.id/) not in(

select pres.medication\_id

from prescription pres

right join purchase pur on pur.prescription\_id = [pres.id](http://pres.id/)

where month(pur.date\_time) = month(DATE\_SUB(curdate(), INTERVAL 1 MONTH)) and year(pur.date\_time) = year(DATE\_SUB(curdate(), INTERVAL 1 MONTH)))

k. In an annual basis the pharmacy needs to get a list of all medications that are not prescribed during the last year

select DISTINCT [med.name](http://med.name/) as Medicine\_Name\_Not\_Ordered\_In\_Last\_Year

from medication med

where [med.id](http://med.id/) not in(

select pres.medication\_id

from prescription pres

right join purchase pur on pur.prescription\_id = [pres.id](http://pres.id/)

where year(pur.date\_time) = year(DATE\_SUB(curdate(), INTERVAL 1 YEAR)))

l. Build a list showing how many prescriptions received by the pharmacy for each month starting with the current month and back to the same month last year. i.e. 13 months history

select count(pur.prescription\_id), month(pur.date\_time) as Month, year(pur.date\_time) as Year

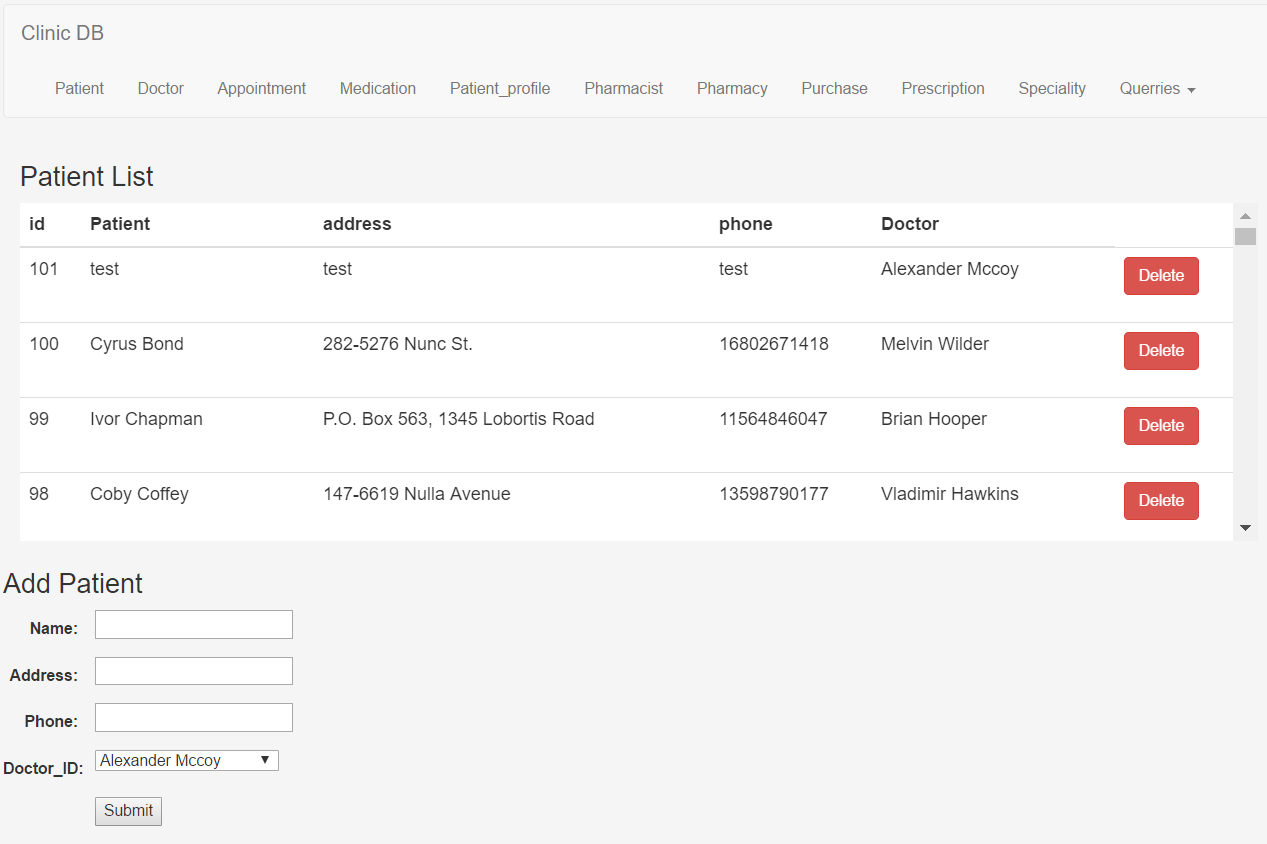
from purchase pur

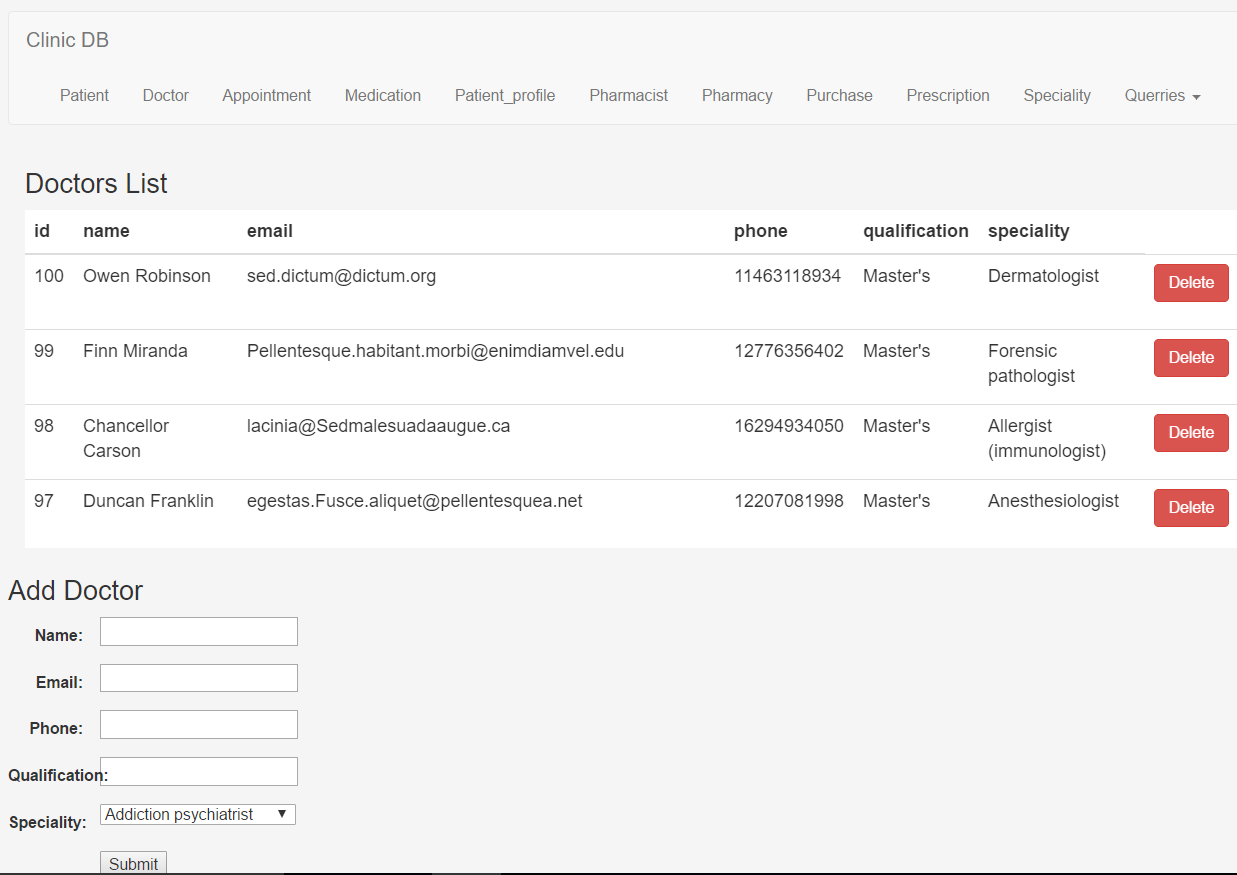
where pur.date\_time > DATE\_SUB(curdate(),INTERVAL 13 MONTH)

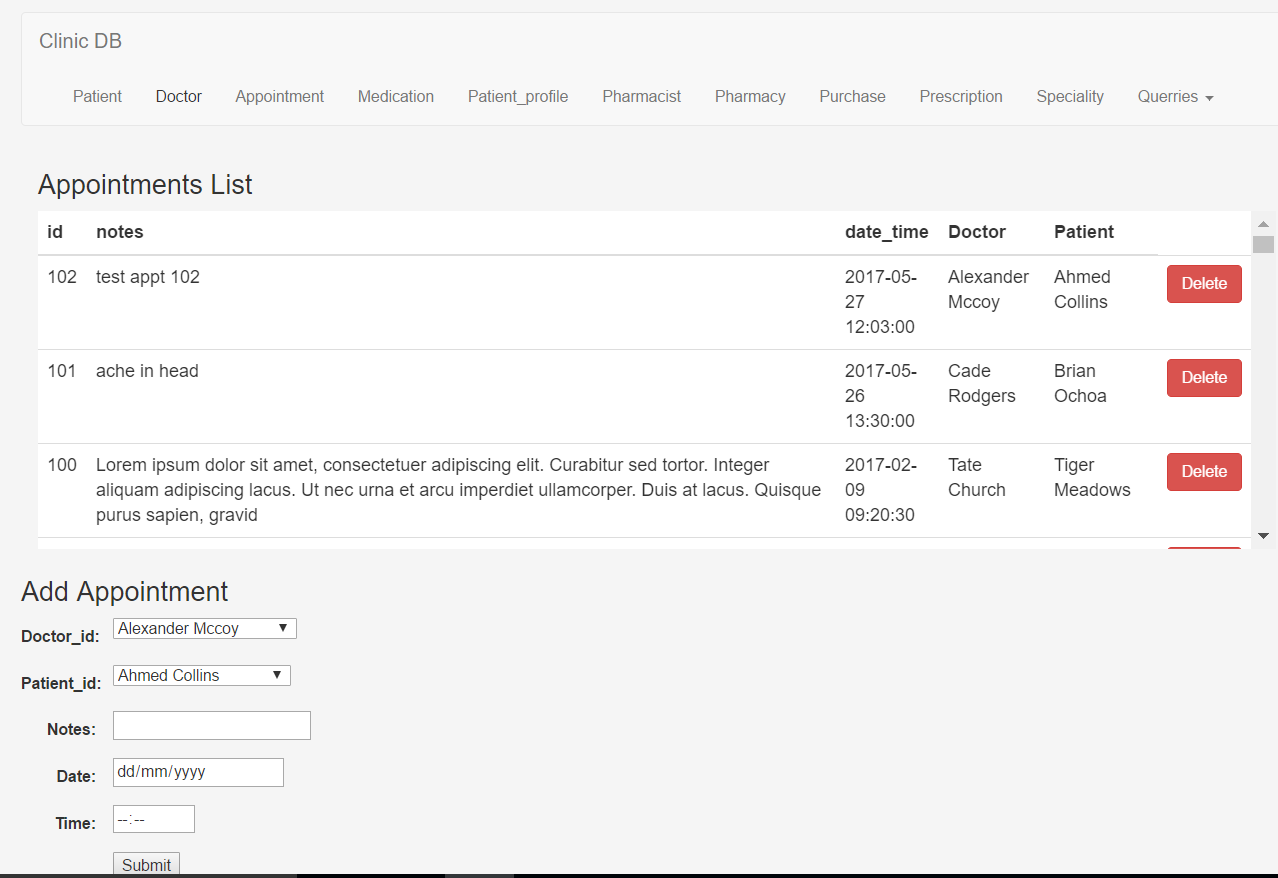
group by 2,3

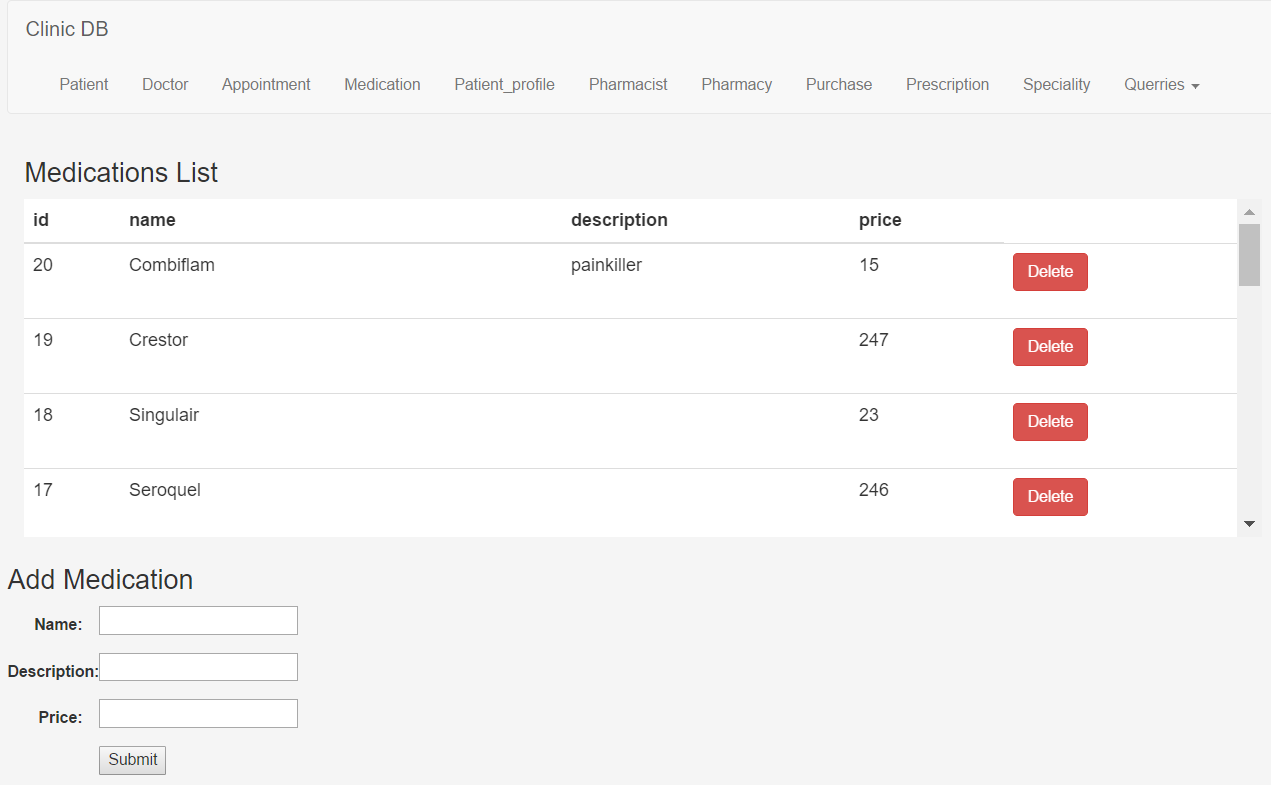
order by 3 DESC, 2 DESC;

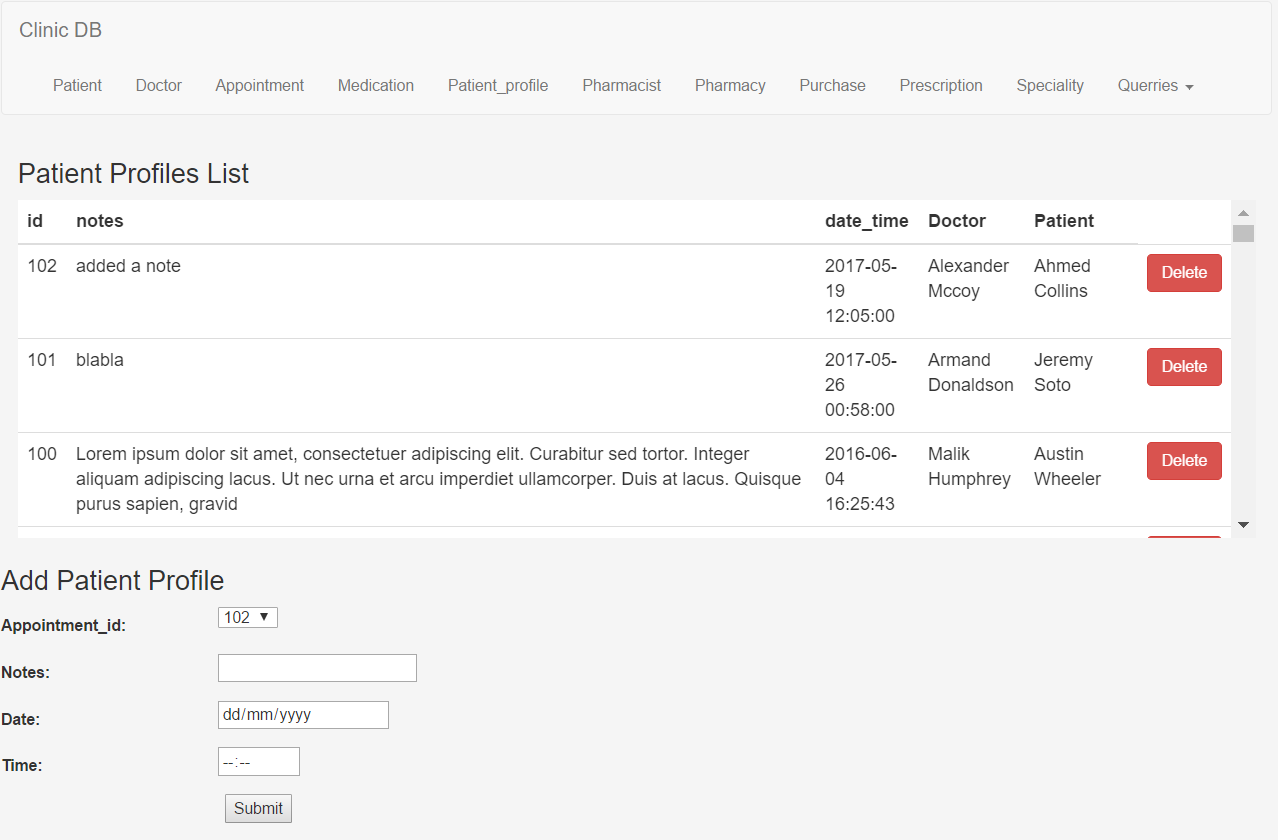
ENTRY FORMS:

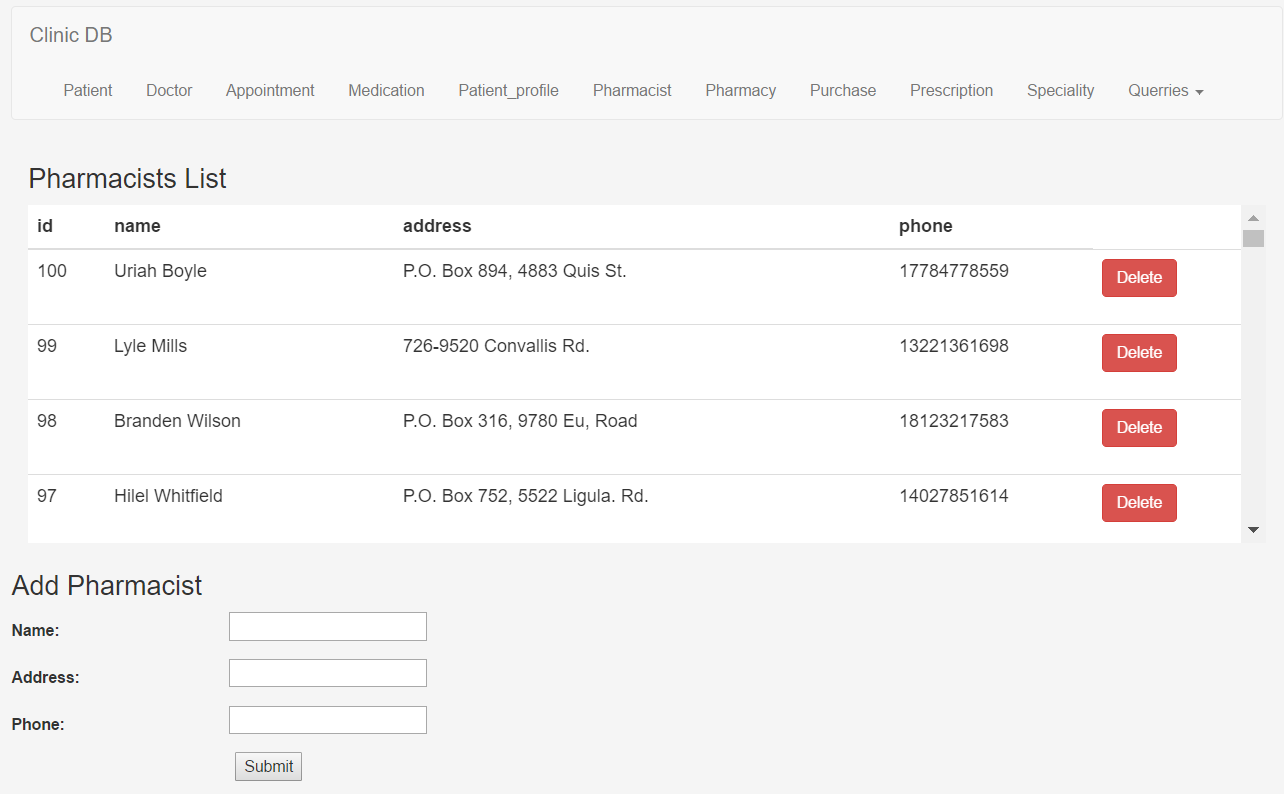


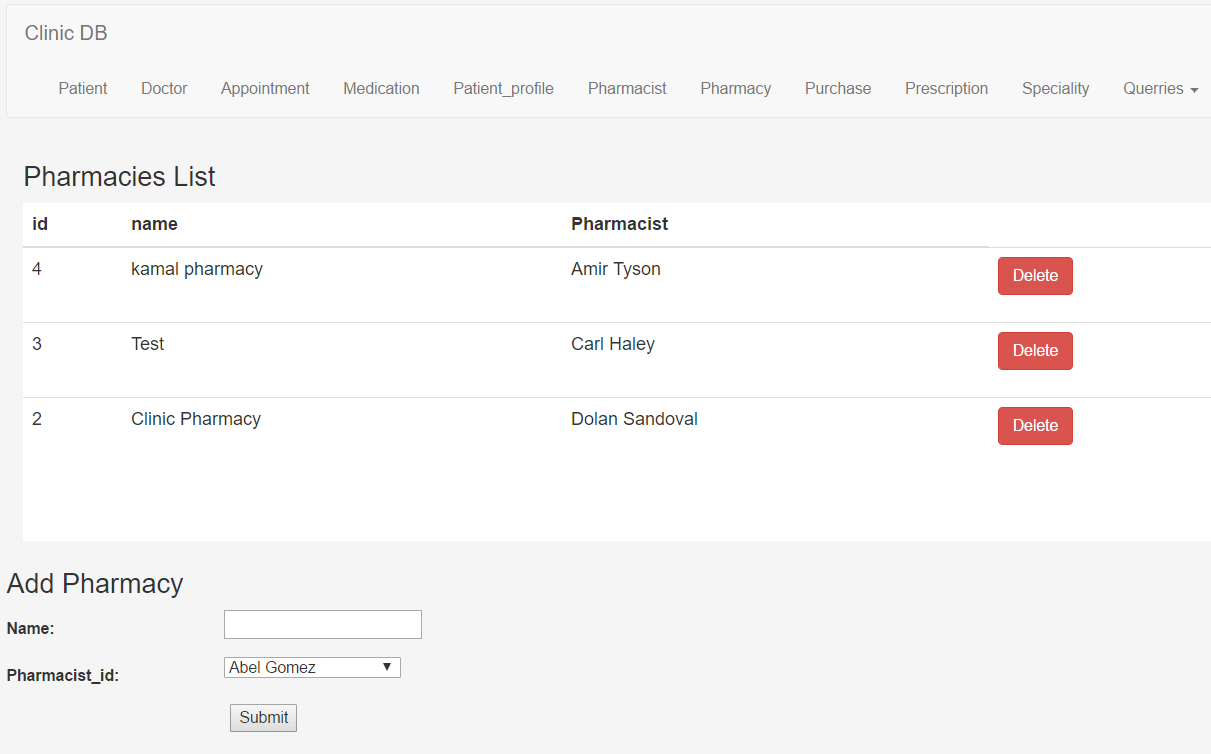


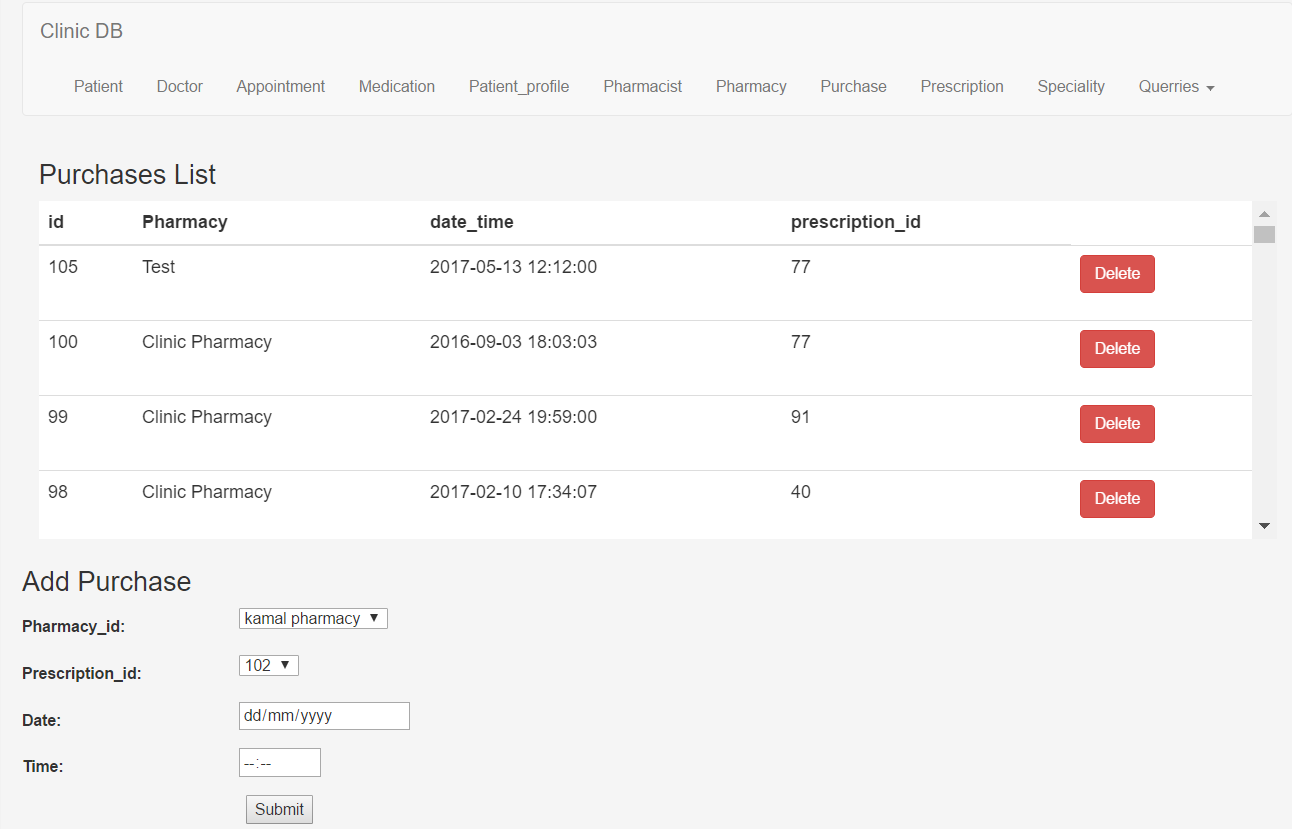


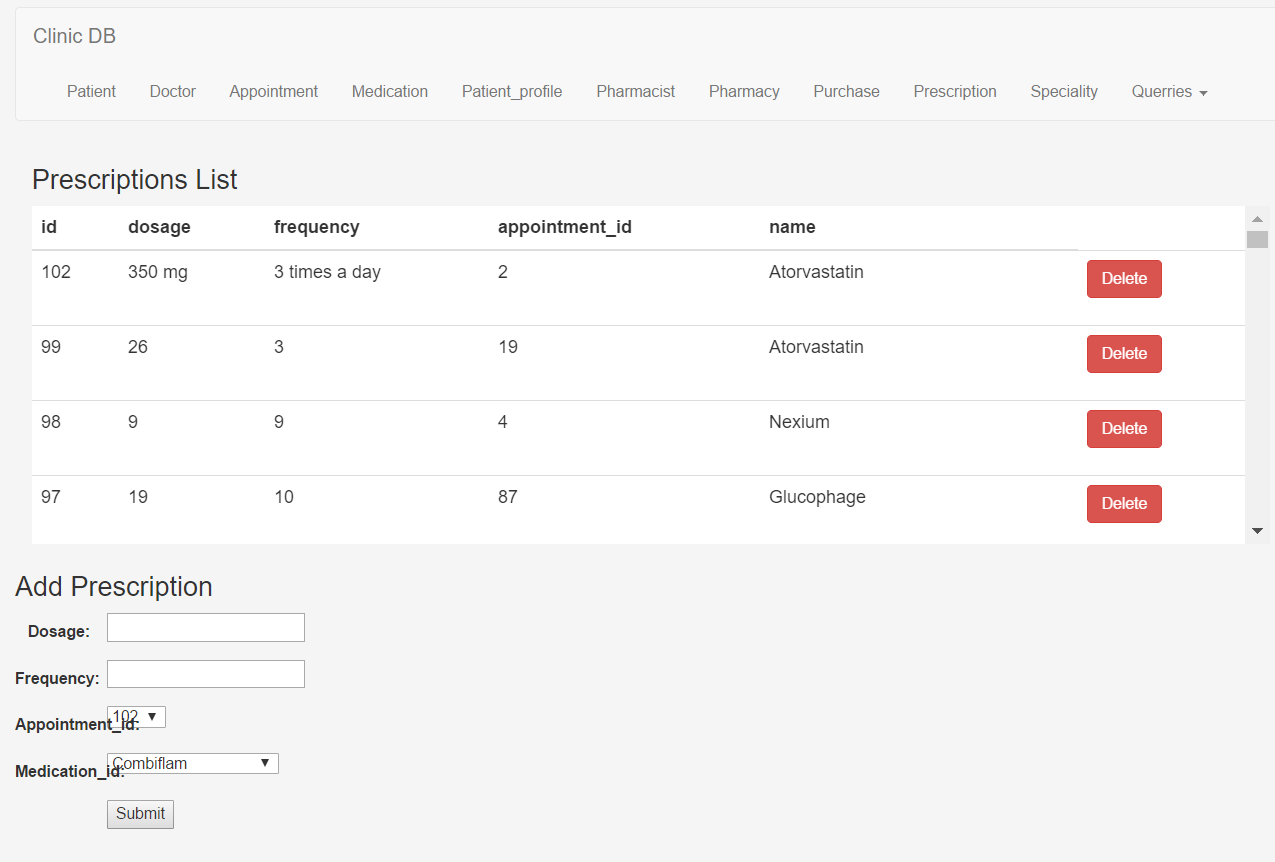


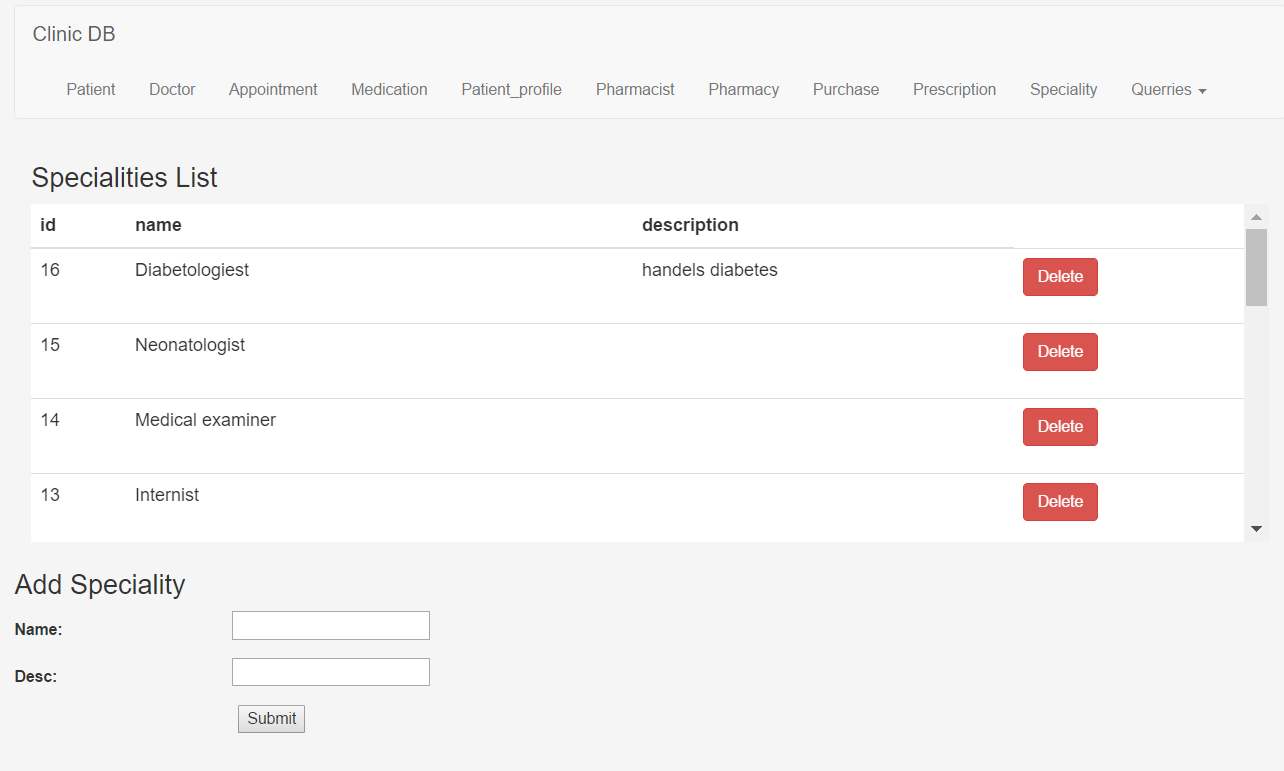






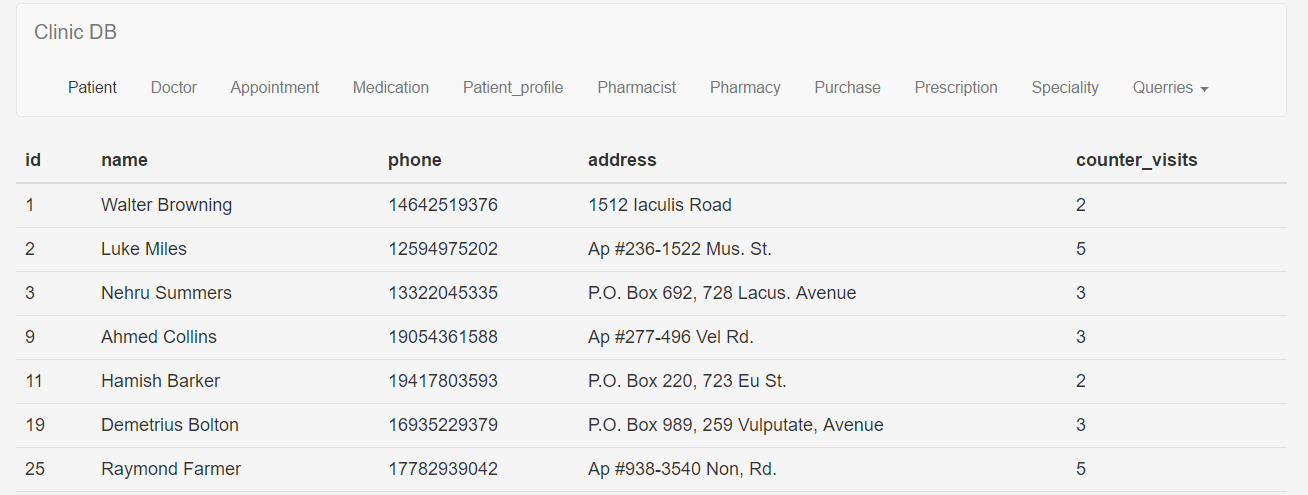




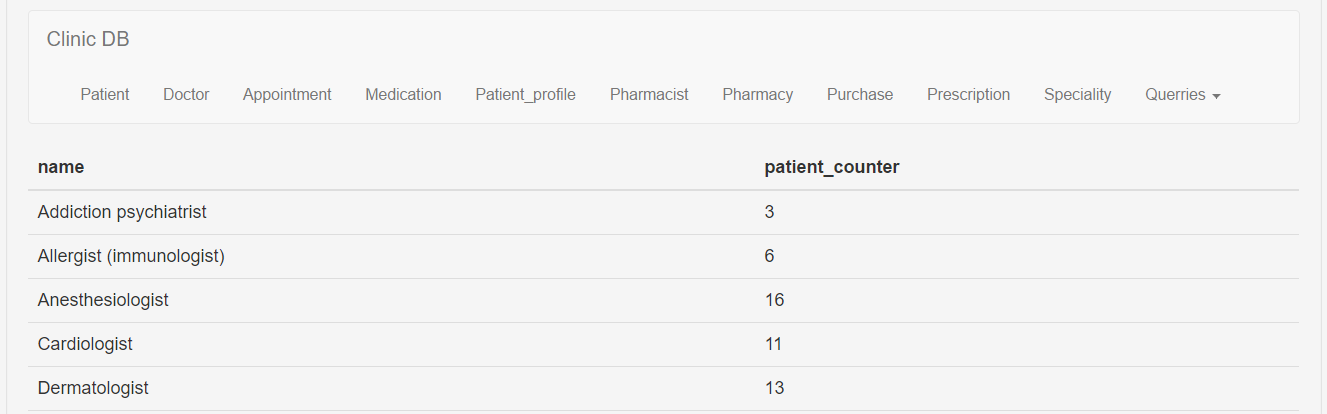


QUERIES:

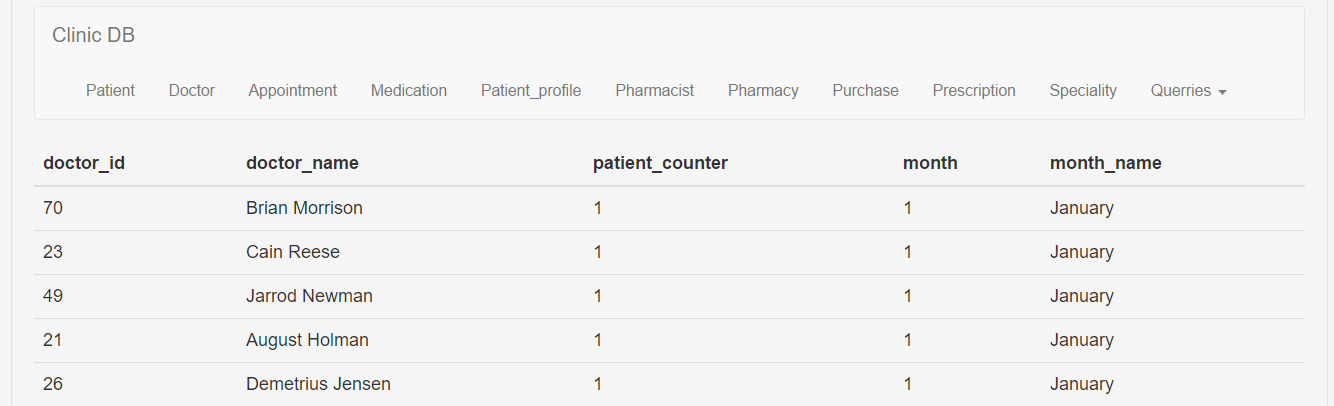
A)



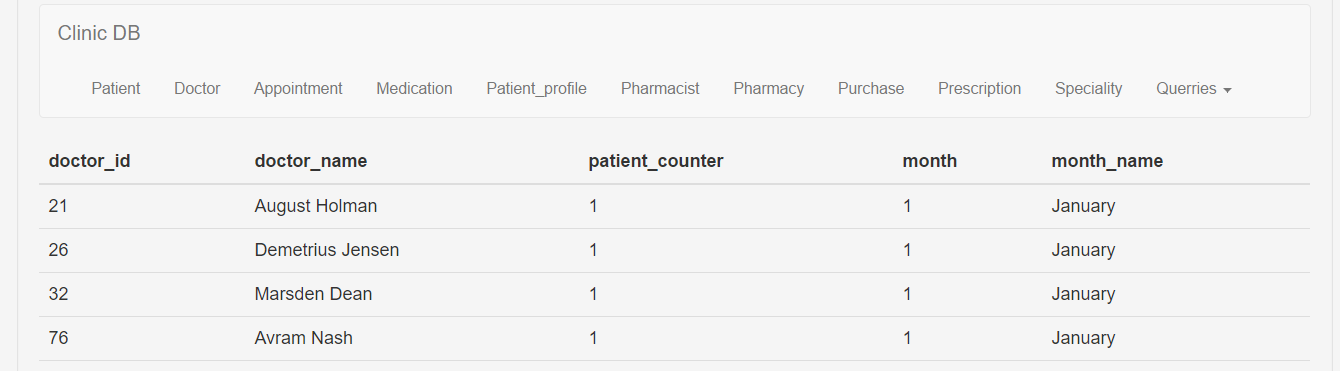
B)



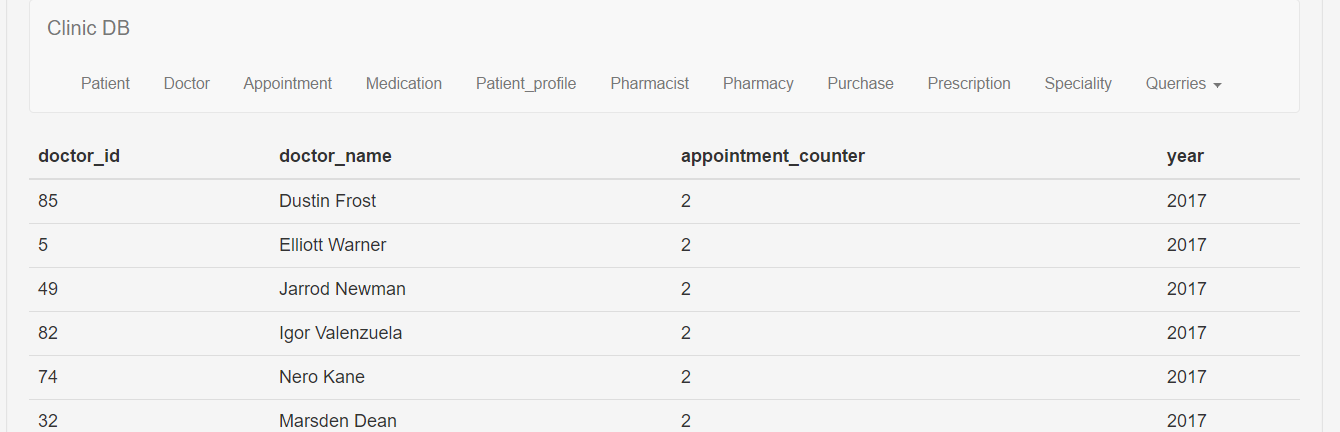
C)



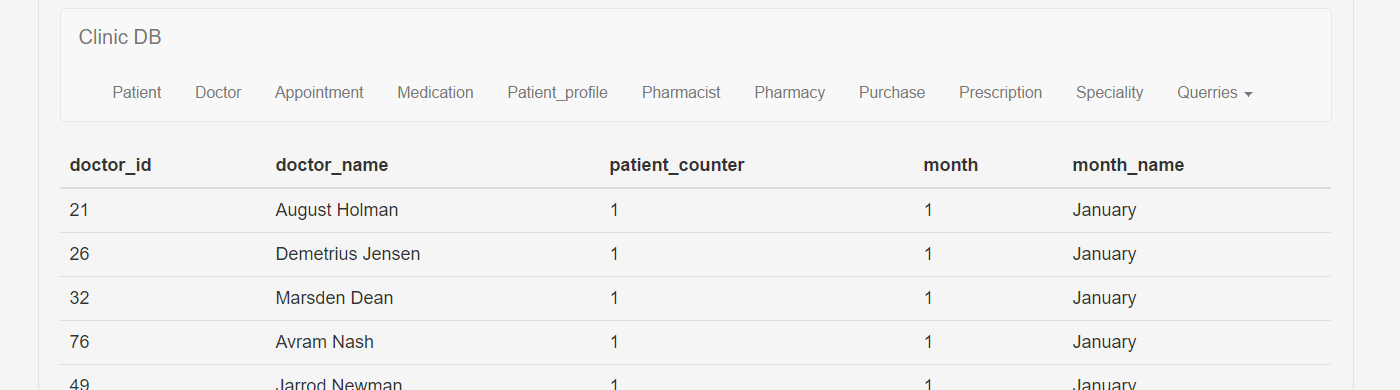
D)



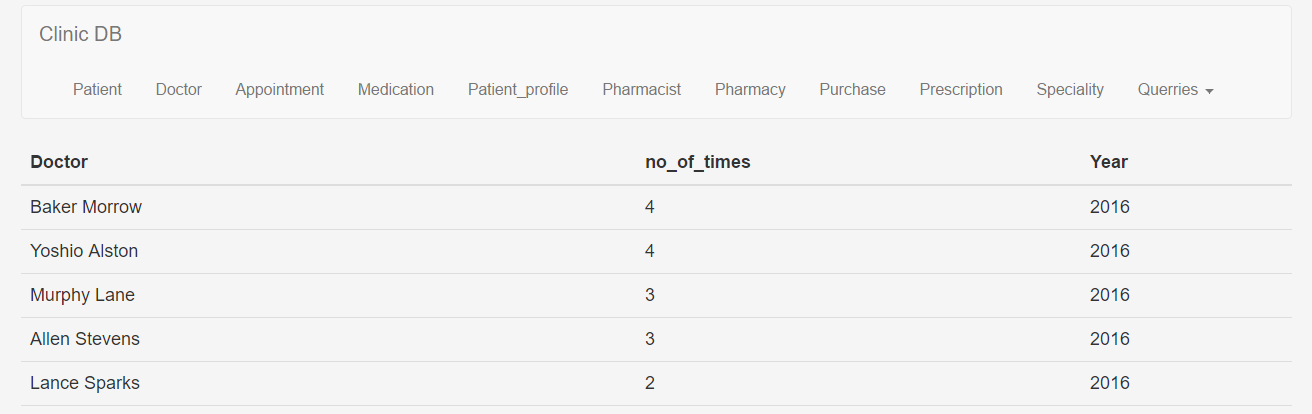
E)



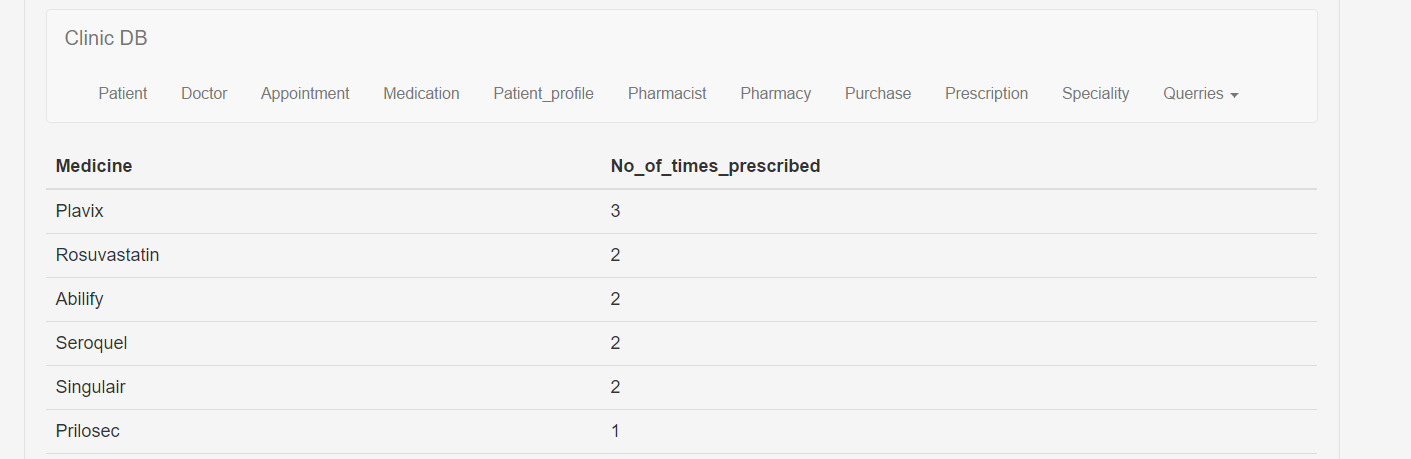
F)



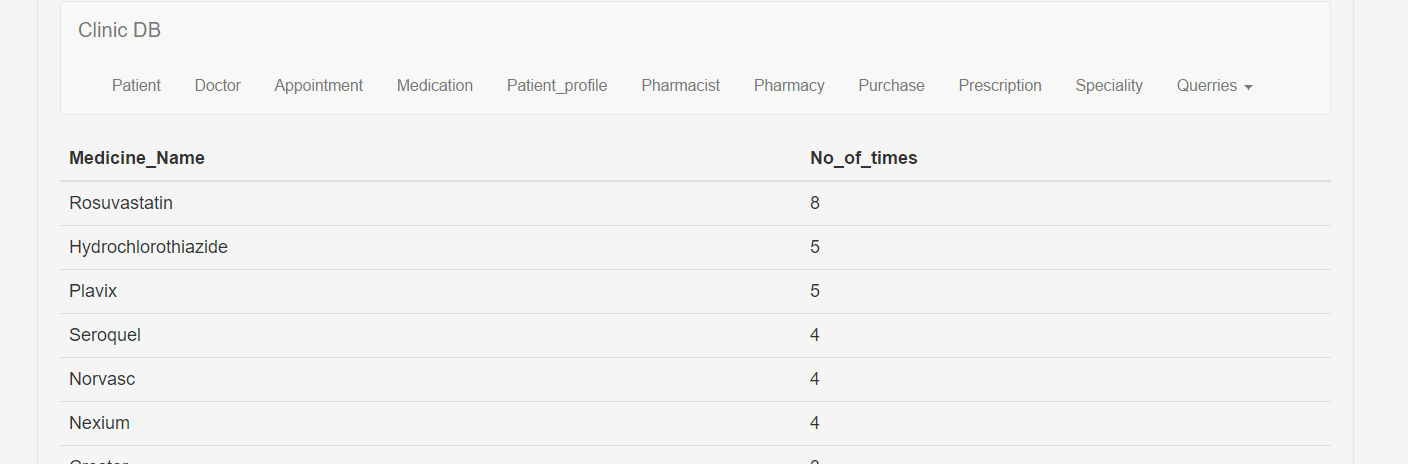
G)



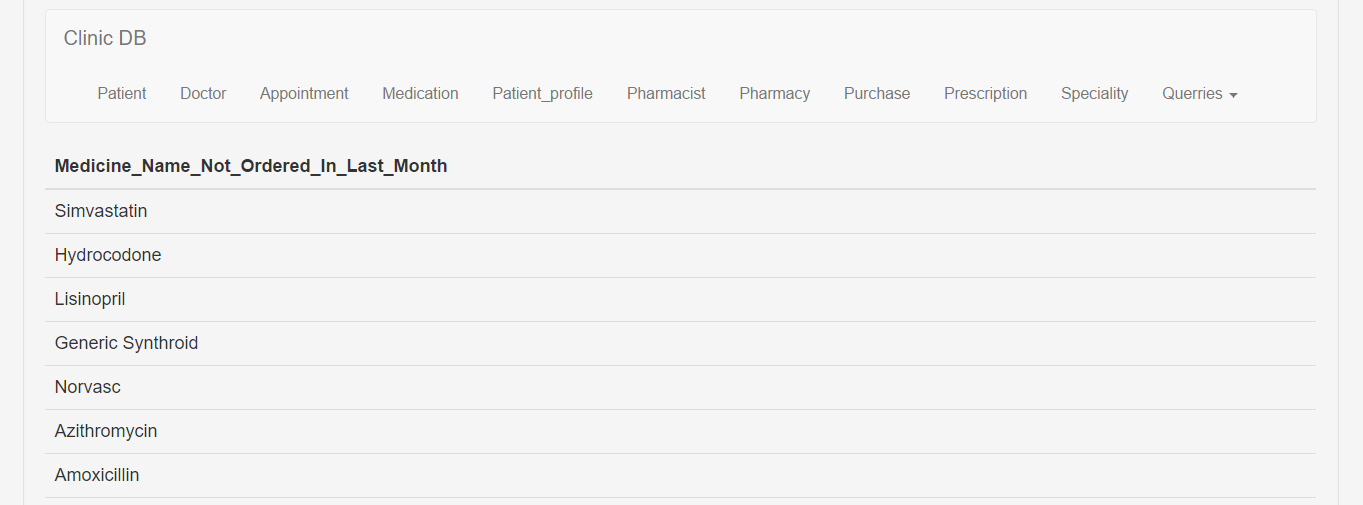
H)



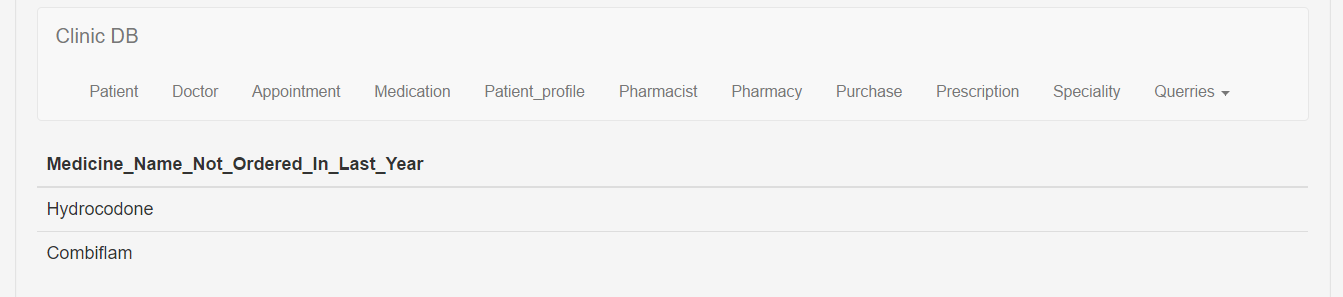
I)



J)



K)



L)

