```
create database dentists_polyclinic;

CREATE TABLE Insurance (
  insurance_id integer NOT NULL,
  company_name varchar(50)  NOT NULL,
  start_date   DATE NOT NULL,
  end_date     DATE NOT NULL,
  co_insurance decimal(5,2),
  PRIMARY KEY (insurance_id)
);

SELECT insurance_id
FROM insurance
GROUP BY insurance_id
HAVING COUNT(distinct company_name) > 1;

CREATE INDEX Insurance_Company_Name
ON Insurance (company_name);
desc insurance;
select * from insurance;

insert into insurance values (101,'National Insurance Co.Ltd','2011-03-12','2020-04-10',55);
insert into insurance values (102,'Go digital General Insurance','2011-09-12','2023-04-10',40);
insert into insurance values (103,'HDFC ERGO General insurance','2010-06-01','2024-05-07',60);
insert into insurance values (104,'HDFC ERGO General insurance','2010-06-01','2024-05-07',60);
insert into insurance values (105,'National Insurance Co.Ltd','2008-03-09','2022-09-23',30);
insert into insurance values (106,'National Insurance Co.Ltd','2008-03-09','2022-09-23',30);



desc insurance;
select * from insurance;

create table patient1(patient_id integer NOT NULL , polyclinic_name varchar(20) not
null,
patient_name varchar(20) unique not null, dob date not null,insurance_id integer,
foreign key(insurance_id) references insurance(insurance_id),sex char(4) not null,
Problem_or_Disease varchar(50) not null,Dno integer not null,doc_id integer not
null,
registration_time time not null,registration_date date not null,
primary key(patient_id),foreign key(doc_id) references doctor_info(doc_id));
SET FOREIGN_KEY_CHECKS=0;
desc patient1;

insert into patient1 values
(1,'Dental Polyclinic','Mr.Smith','1967-03-25',101,'M','Soft tissue
Inflammation',1,100,'17:00:00','2022-03-19');
insert into patient1 values
(2,'Dental Polyclinic','Mr.Andrews','1978-02-04',102,'M','Gum
Disease',2,300,'14:00:00','2022-03-20');
insert into patient1 values
(3,'Dental Polyclinic','Mrs.Rodriguez','1987-07-28',103,'F','Deep
Decay',1,200,'17:00:00','2022-03-21');
```

```
insert into patient1 values
(4,'Dental Polyclinic','Mr.Holt','1983-08-
21',104,'M','Cavities',3,400,'21:00:00','2022-03-19');
insert into patient1 values
(5,'Dental Polyclinic','Ms.Ruby','1998-01-16',105,'F','Missing
Teeth',3,400,'17:00:00','2022-03-20');
insert into patient1 values
(6,'Dental Polyclinic','Ms.Franceska','2000-03-19',106,'F','Mobile
Teeth',3,500,'18:00:00','2022-03-19');
select * from patient1;
desc patient1;

create table PATIENT_PHONE (patient_id INTEGER NOT NULL,foreign key(patient_id)
references patient1(patient_id),
Phone_number numeric not null);
insert into patient_phone values(1,9821000690),(1,8999452345),(2,9811223300),
(2,9786577724),
(3,9013211091),(4,9210747010),(5,9900887045),(5,9900889085),(6,9601887095);
desc patient_phone;
select * from patient_phone;



CREATE TABLE VISITS AS (select
patient_name,patient_id,registration_time,registration_date,
CASE
when registration_time <'16:00:00' then 'SORRY ! COME WITHIN THE SPECIFIED TIMINGS'
when registration_time > '20:30:00' then 'SORRY ! COME WITHIN THE SPECIFIED
TIMINGS'
WHEN dayname(registration_date) not in
('MONDAY','TUESDAY','WEDNESDAY','THURSDAY','FRIDAY','SATURDAY') THEN
'SORRY ! WE ARE OPEN ONLY FROM MONDAY-SATURDAY'
ELSE 'REGISTRATION CAN BE DONE'
End as Final_Details
from patient1 );

SELECT patient_name,patient_id
FROM VISITS
GROUP BY patient_name,patient_id
HAVING COUNT(distinct Final_Details) > 1;

SELECT * FROM VISITS;
alter table VISITS ADD CONSTRAINT foreign key (patient_id)
references patient1 (patient_id);
alter table VISITS add constraint primary key(patient_id);
alter table VISITS add constraint foreign key (patient_name) references
patient1(patient_name);
desc visits;


create table previous_visits (patient_id Integer not null,foreign key(patient_id)
references patient1(patient_id),visits date not
null,prev_treatment_taken_from_this_clinic
 varchar(50) not null);

SELECT patient_id,visits
FROM previous_visits
GROUP BY patient_id,visits
HAVING COUNT(distinct prev_treatment_taken_from_this_clinic) > 1;
```

```
insert into previous_visits values
(1,'2017-08-11','Root Canal'),
(1,'2019-02-07','Root Canal'),
(2,'2016-12-11','Gums'),
(2,'2018-12-11','Gums'),
(3,'2018-03-10','Cavities'),
(4,'2018-03-23','Missing Teeth'),
(5,'2017-10-25','Mobile Teeth');
select * from previous_visits;
desc previous_visits;

CREATE TABLE new_patients
AS (SELECT patient_id,patient_name,insurance_id from patient1 where not exists
( select patient_id FROM previous_visits
WHERE patient1.patient_id = previous_visits.patient_id));

Alter table new_patients add New_patient Varchar(50) default('WELCOME!YOUR FIRST
CHECKUP IS FREE') not null;
Alter table new_patients add discount_given decimal(5,2) default(100);

alter table new_patients add primary key(patient_id);
alter table new_patients add foreign key (patient_id) references
patient1(patient_id);
alter table new_patients add foreign key (patient_name) references
patient1(patient_name);
alter table new_patients add foreign key (insurance_id) references
patient1(insurance_id);

select * from new_patients;
desc new_patients;

create table regular_patients AS (select patient_id,patient_name,insurance_id
 from patient1 where patient_id in (select patient_id
from previous_visits group by patient_id having
count(previous_visits.patient_id)>=2));

Alter table regular_patients add discount_given decimal(5,2) default(10);

alter table regular_patients add primary key(patient_id);
alter table regular_patients add foreign key (patient_id) references
patient1(patient_id);
alter table regular_patients add foreign key (patient_name) references
patient1(patient_name);
alter table regular_patients add foreign key (insurance_id) references
patient1(insurance_id);

select * from regular_patients;
desc regular_patients;

create table doctor_info( doc_id INTEGER NOT NULL primary key, salary_slipno
INTEGER NOT NULL
 unique, doc_name varchar(20) not null,Dep_no integer not null,Dep_name varchar(20)
not null,
 foreign key(Dep_no) references department(Dep_no),foreign key
 (dep_name) references department(dep_name)) ;
insert into doctor_info values(100,100,'Dr. Ray',1,'Endodontist');
insert into doctor_info values(200,101,'Dr. Bing',1,'Endodontist');
insert into doctor_info values(300,102,'Dr. Stromberg',2,'Periodontist');
insert into doctor_info values(400,103,'Dr. David',3,'General Dentist');
insert into doctor_info values(500,104,'Dr. James',3,'General Dentist');
Alter table doctor_info add constraint foreign key (salary_slipno) references
doc_salary (salary_slipno);
desc doctor_info;
select * from doctor_info;

create table doctor_phone(doc_id INTEGER NOT NULL,foreign key(doc_id) references
doctor_info(doc_id),
Phone_number numeric not null);
insert into doctor_phone values(100,9821054690),(100,8976452345),(200,9811223344),
(200,9786574624),
(300,9143211091),(400,9213447010),(500,9900887755);
desc doctor_phone;
select * from doctor_phone;

create table doc_salary(salary_slipno INTEGER primary key, foreign
key(salary_slipno)
references doctor_info(salary_slipno), salary Numeric not null,
Number_of_years_working integer not null);
insert into doc_salary values(100,500000,4);
insert into doc_salary values(101,250200,1);
insert into doc_salary values(102,512200,5);
insert into doc_salary values(103,700000,8);
insert into doc_salary values(104,656666,6);
desc doc_salary;
select * from doc_salary;

create table department(dep_no INTEGER not null, dep_name varchar(20) unique not
null,primary key(dep_no) );
insert into department values(1,'Endodontist');
insert into department values(2,'Periodontist');
insert into department values(3,'General Dentist');
desc department;
select * from department;

create table endodontist(doc_id INTEGER not null, foreign key(doc_id) references
doctor_info(doc_id),
root_canal varchar(100) not null,charges integer not null);
insert into endodontist values(100,'Soft tissue inflammation',4000);
insert into endodontist values(200,'Deep decay',7000);
desc endodontist;

create table periodontist(doc_id INTEGER not null primary key, foreign key(doc_id)
references doctor_info(doc_id),
gums varchar(100) not null,price integer not null);
insert into periodontist values(300,'Gum Disease',6000);
select * from periodontist;
desc periodontist;

create table gen_dentist(doc_id INTEGER not null,
foreign key(doc_id) references
doctor_info(doc_id),cavities_OR_missing_teeth_OR_mobile_teeth
varchar(20) not null,PRICE integer not null);
insert into gen_dentist values(400,'Cavities',2000);
insert into gen_dentist values(400,'Missing Teeth',2500);
insert into gen_dentist values(400,'Mobile Teeth',2700);
insert into gen_dentist values(500,'Cavities',3000);
insert into gen_dentist values(500,'Missing Teeth',3500);
insert into gen_dentist values(500,'Mobile Teeth',3700);
select * from gen_dentist;
desc gen_dentist;

create table TOTAL_BILL AS (select Patient_id,insurance_id,patient_name,
case
when (patient1.doc_id=100 and Dno=1) then (select charges from endodontist where
doc_id=100)
when (patient1.doc_id=200 and Dno=1) then (select charges from endodontist where
doc_id=200)
when (patient1.doc_id=300 and Dno=2) then (select price from periodontist where
doc_id=300)
when (patient1.doc_id=400 and Dno=3 and Problem_or_Disease like
'Cavities') then (select price from gen_dentist where (doc_id=400 and
cavities_OR_missing_teeth_OR_mobile_teeth like
'Cavities'))
when (patient1.doc_id=400 and Dno=3 and Problem_or_Disease like
'Missing Teeth') then (select price from gen_dentist where (doc_id=400 and
cavities_OR_missing_teeth_OR_mobile_teeth
like 'Missing Teeth'))
when (doc_id=400 and Dno=3 and Problem_or_Disease like
'Mobile Teeth' ) then (select price from gen_dentist where (doc_id=400 and
cavities_OR_missing_teeth_OR_mobile_teeth like
'Mobile Teeth'))
when (doc_id=500 and Dno=3 and Problem_or_Disease like
'Cavities') then (select price from gen_dentist where (doc_id=500 and
cavities_OR_missing_teeth_OR_mobile_teeth like
'Cavities'))
when (doc_id=500 and Dno=3 and Problem_or_Disease like
'Missing Teeth') then (select price from gen_dentist where (doc_id=500 and
cavities_OR_missing_teeth_OR_mobile_teeth
like 'Missing Teeth'))
when (doc_id=500 and Dno=3 and Problem_or_Disease like
'Mobile Teeth') then (select price from gen_dentist where (doc_id=500 and
cavities_OR_missing_teeth_OR_mobile_teeth like
'Mobile Teeth'))
else 0
end as charges
from patient1 );

select * from total_bill;
desc total_bill;
SET SQL_SAFE_UPDATES = 0;

alter table total_bill add discount_given integer not null;

update total_bill e
INNER JOIN regular_patients r
ON e.patient_id = r.patient_id
SET e.discount_given = (charges * r.discount_given/100.00);
select * from total_bill;

update total_bill e
INNER JOIN new_patients n
ON e.patient_id = n.patient_id
SET e.discount_given = (charges * n.discount_given/100.00);
select * from total_bill;

alter table total_bill add column charge_after_discount integer not null;
```

```
update total_bill set charge_after_discount=charges-discount_given;

alter table total_bill add Money_Insurance integer not null ;

update total_bill e
inner join insurance i
on e.insurance_id=i.insurance_id
SET Money_insurance= ((charge_after_discount) * co_insurance/100.00);
select * from total_bill;

alter table total_bill add Patient_Pay integer not null ;

update total_bill e
inner join insurance i
on e.insurance_id=i.insurance_id
set patient_pay=(charge_after_discount)-money_insurance;

select * from total_bill;
desc total_bill;

alter table total_bill add bill_no integer not null;
alter table total_bill add primary key(bill_no,patient_id);
alter table total_bill modify column bill_no integer NOT NULL AUTO_INCREMENT;
alter table total_bill add cashier_id integer not null;
alter table total_bill add foreign key(cashier_id) references cashier(cashier_id);
alter table total_bill add foreign key(Insurance_id) references
patient1(insurance_id);
alter table total_bill add foreign key(patient_id) references patient1(patient_id);
alter table total_bill add foreign key(patient_name) references
patient1(patient_name);
update total_bill set cashier_id=301 where (bill_no%2=0);
update total_bill set cashier_id=302 where (bill_no%2 != 0);


desc total_bill;

create table dependents(depen_name varchar(100),
phone_no numeric not null,patient_id INTEGER, foreign key(patient_id) references
patient1(patient_id),
primary key(patient_id,depen_name) );
insert into dependents values('Roger',9165625400,1);
insert into dependents values('Fin',9165623880,2);
insert into dependents values('Thomas',9789879765,3);
insert into dependents values('Alfie',9914323523,4);
insert into dependents values('Arthur',9678229119,5);
insert into dependents values('Anjali',9678229119,5);
select * from dependents;
desc dependents;

create table medic_hist(patient_id INTEGER,
foreign key(patient_id) references patient1(patient_id), past_treatment
varchar(50),
allergies varchar(50), pain_tooth varchar(50), heart_probs varchar (50),
other_illness varchar(50),
primary key(patient_id,past_treatment));
insert into medic_hist values(1,'Root Canal','Penicillin',null,'High
BP','Diabetes');
insert into medic_hist values(2,'Root Canal',null,'Upper Left
Tooth',null,'Rhinitis');
```

```
insert into medic_hist values(3,'Loose Teeth','Pollen',null,'High BP','Arthritis');
insert into medic_hist values(4,'Decay','Pollen','Lower Left Side',null,null);
insert into medic_hist values(5,'Gingivitis','Lignocaine','Lower Right Side','High
BP','Cardiac Problem');
insert into medic_hist values(1,'Loose teeth',null,'Upper Left
Tooth',null,'Rhinitis');
select * from medic_hist;
desc medic_hist;

create table cashier(Name varchar(20) not null,cashier_id integer not null primary
key,
salary integer not null);
insert into cashier values('Amit',301,3000);
insert into cashier values ('Rohit',302,400);
select count(*) from cashier;
select * from cashier;

create table cashier_PHONE (cashier_id INTEGER NOT NULL,foreign key(cashier_id)
references cashier(cashier_id),
Phone_number numeric not null);
insert into cashier_phone values(301,9999765642),(301,6542758545),(302,8645324455),
(302,7689000678);
alter table cashier_phone add constraint primary key(phone_number);

desc cashier;
desc cashier_phone;
select distinct cashier_id from cashier_phone;
select count(distinct(cashier_id)) from cashier_phone;
select * from cashier_phone;
```