SQL QUERIES:

Members :

1. create table Department(department\_name varchar(20) NOT NULL,Department\_code varchar(20) NOT NULL,Primary key(Department\_code));
2. create table medicine(medicine\_name varchar(20) NOT NULL,medicine\_cost double,medicine\_code varchar(20) NOT NULL,PRIMARY KEY(medicine\_code));
3. create table person(first\_name varchar(20) NOT NULL,middle\_name varchar(20) ,sur\_name varchar(20) NOT NULL,person\_id int NOT NULL,date\_of\_birth date,status varchar(20),gender varchar(20) NOT NULL,contact varchar(20),PRIMARY KEY(person\_id));
4. alter table medicine modify medicine\_cost decimal(18,2) default 0.00;
5. create table disease(disease\_name varchar(20) NOT NULL,disease\_code varchar(20) NOT NULL,Department\_code varchar(20) NOT NULL references Department(Department\_code));
6. create table doctor(doctor\_code varchar(20) not null,disease\_code varchar(20) NOT NULL references disease(disease\_code),person\_id int not null UNIQUE REFERENCES person(person\_id));
7. create table patient(patient\_id varchar(20) not null ,person\_id int not null UNIQUE REFERENCES person(person\_id),date\_of\_admission datetime,disease\_code varchar(20) NOT NULL references disease(disease\_code),date\_of\_discharge datetime,inpatient varchar(5),PRIMARY KEY(patient\_id));
8. create table transaction(transaction\_code varchar(20) not null,payment\_mode varchar(20) not null, amount decimal(18,2) default 0.00,patient\_id varchar(20) not null references patient(patient\_id),

PRIMARY KEY(transaction\_code));

1. create table prescription(patient\_id varchar(20) not null references patient(patient\_id),medicine\_code varchar(20) REFERENCES medicine(medicine\_code))
2. ALTER TABLE patient ADD FOREIGN KEY (person\_id) REFERENCES person (person\_id);
3. ALTER TABLE disease ADD FOREIGN KEY (department\_code) REFERENCES department(department\_code);
4. ALTER TABLE doctor ADD FOREIGN KEY (disease\_code) REFERENCES disease (disease\_code);
5. ALTER TABLE doctor ADD FOREIGN KEY (person\_id) REFERENCES person(person\_id);
6. ALTER TABLE patient ADD FOREIGN KEY (disease\_code) REFERENCES disease(disease\_code);
7. ALTER TABLE transaction ADD FOREIGN KEY (patient\_id) REFERENCES patient (patient\_id);
8. ALTER TABLE prescription ADD FOREIGN KEY (patient\_id) REFERENCES patient (patient\_id);
9. ALTER TABLE prescription ADD FOREIGN KEY (medicane\_code) REFERENCES medicane (medicane\_code);

1. Insert query

INSERT INTO prescription( patient\_id,medicane\_code,quantity)

VALUES(‘pt\_10’,’104’,’20’);

INSERT INTO prescription

VALUES('pt\_113','199','8');

INSERT INTO transaction

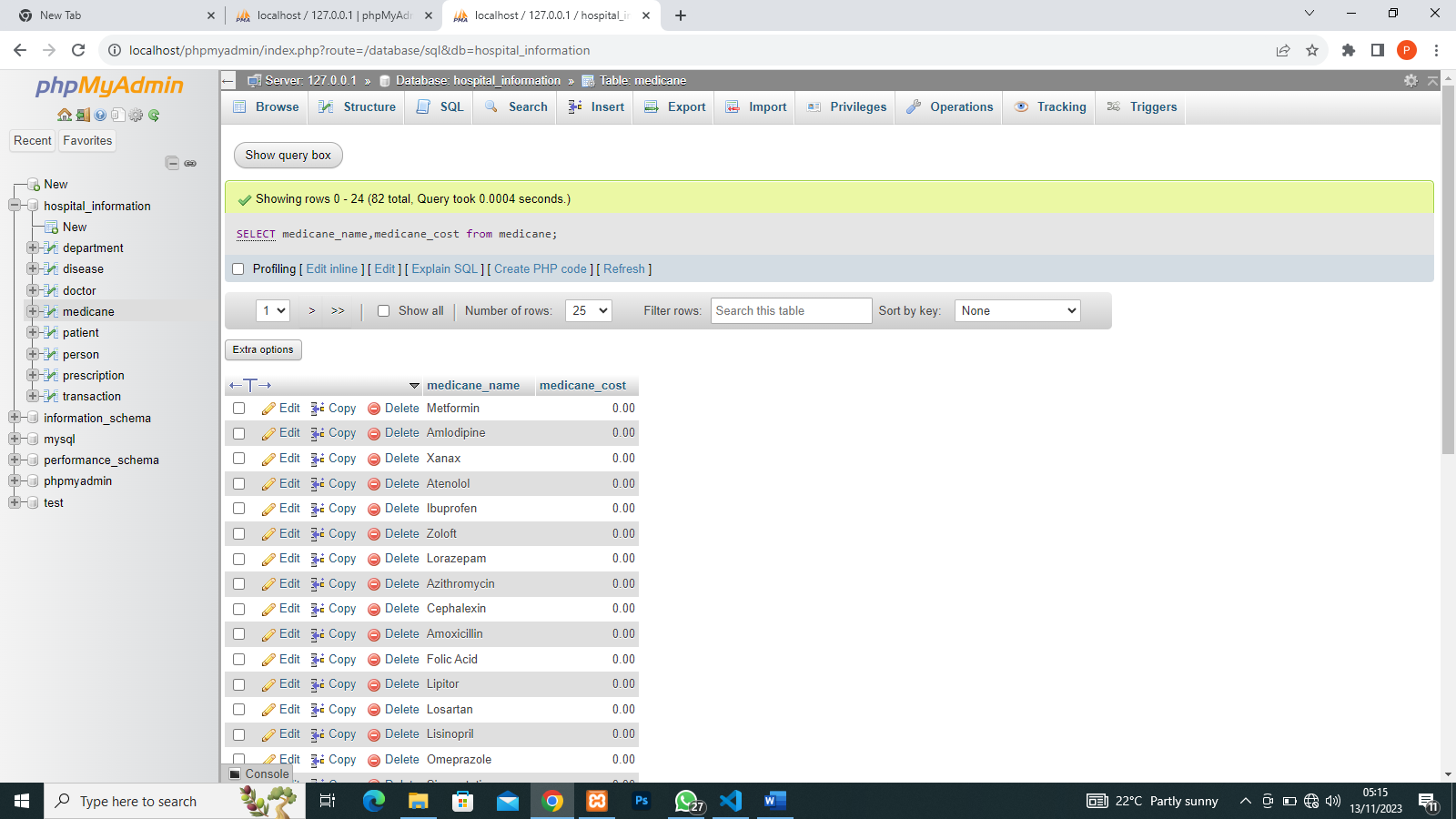
VALUES('xyz123','m-pesa','200','pt\_113');

INSERT INTO transaction

VALUES('uuynbsye','m-pesa','1000','pt\_113'),('abcdzyxw','m-pesa','1200','pt\_151');

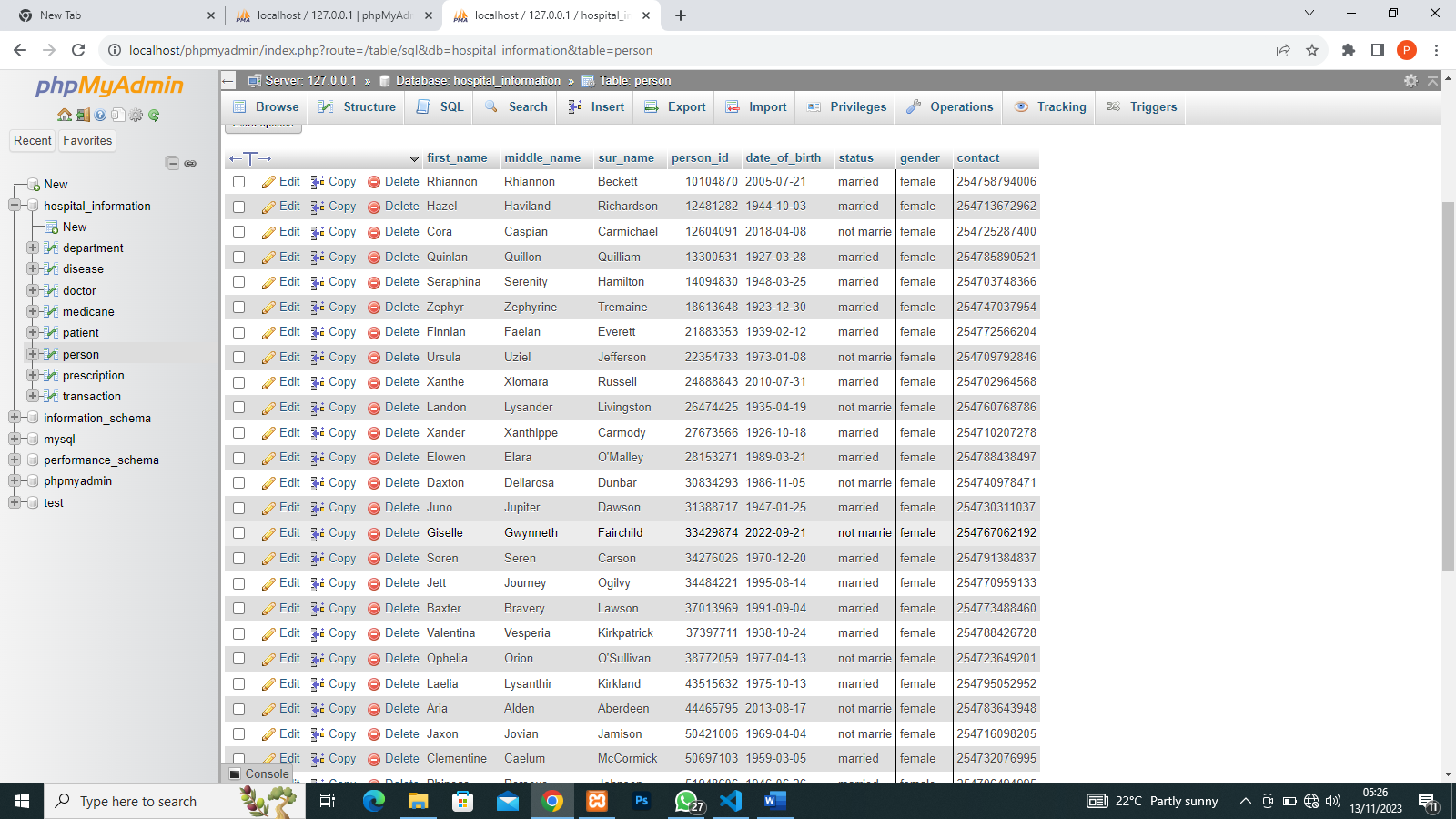
1. .select query

SELECT \* FROM `person`

SELECT medicane\_name,medicane\_cost FROM medicane

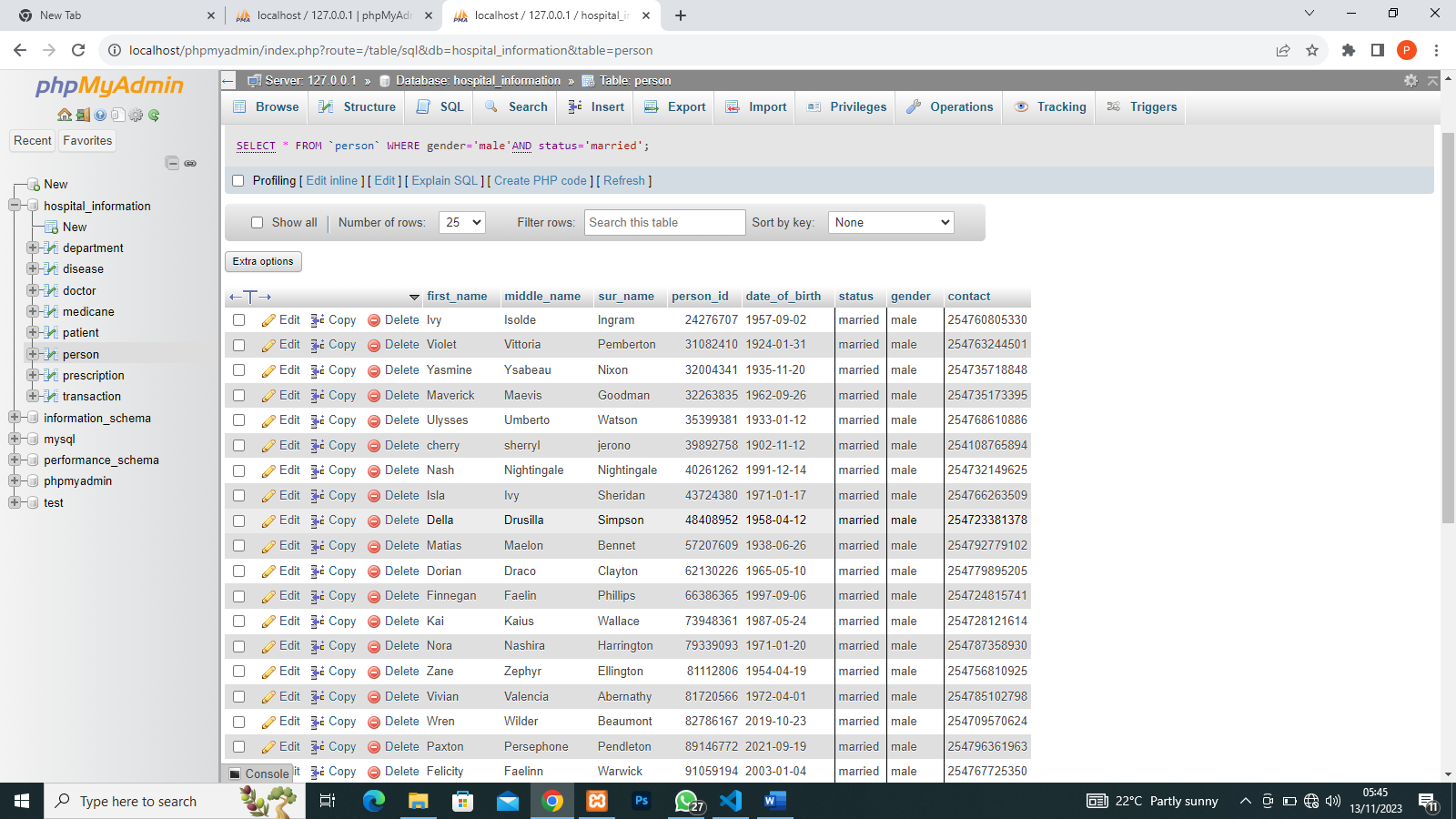
1. SQL WHERE CLAUSE

SELECT \* FROM `person` WHERE gender='female'

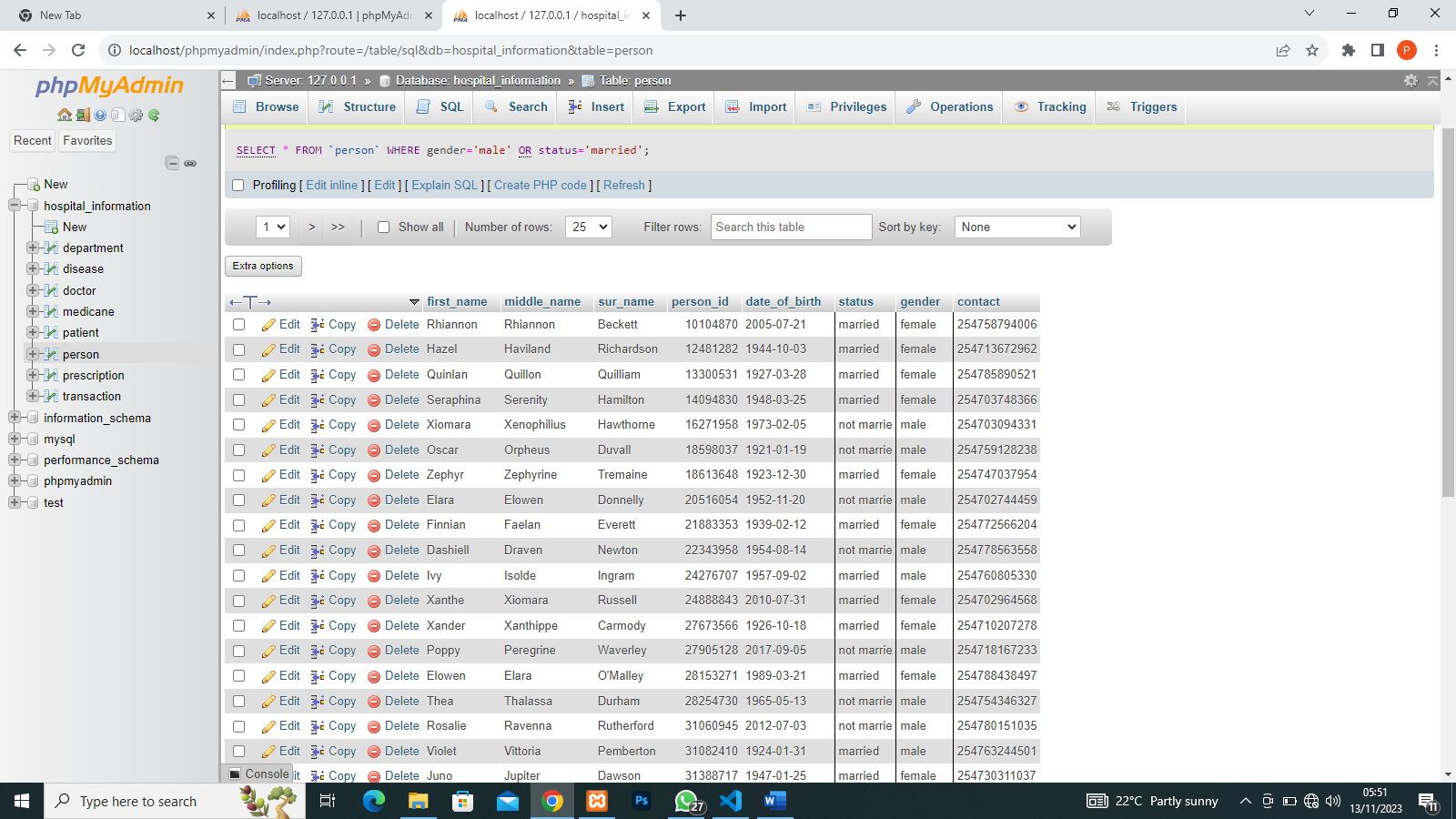


1. SQL AND and OR operators

SELECT \* FROM `person` WHERE gender='male' AND status='married'

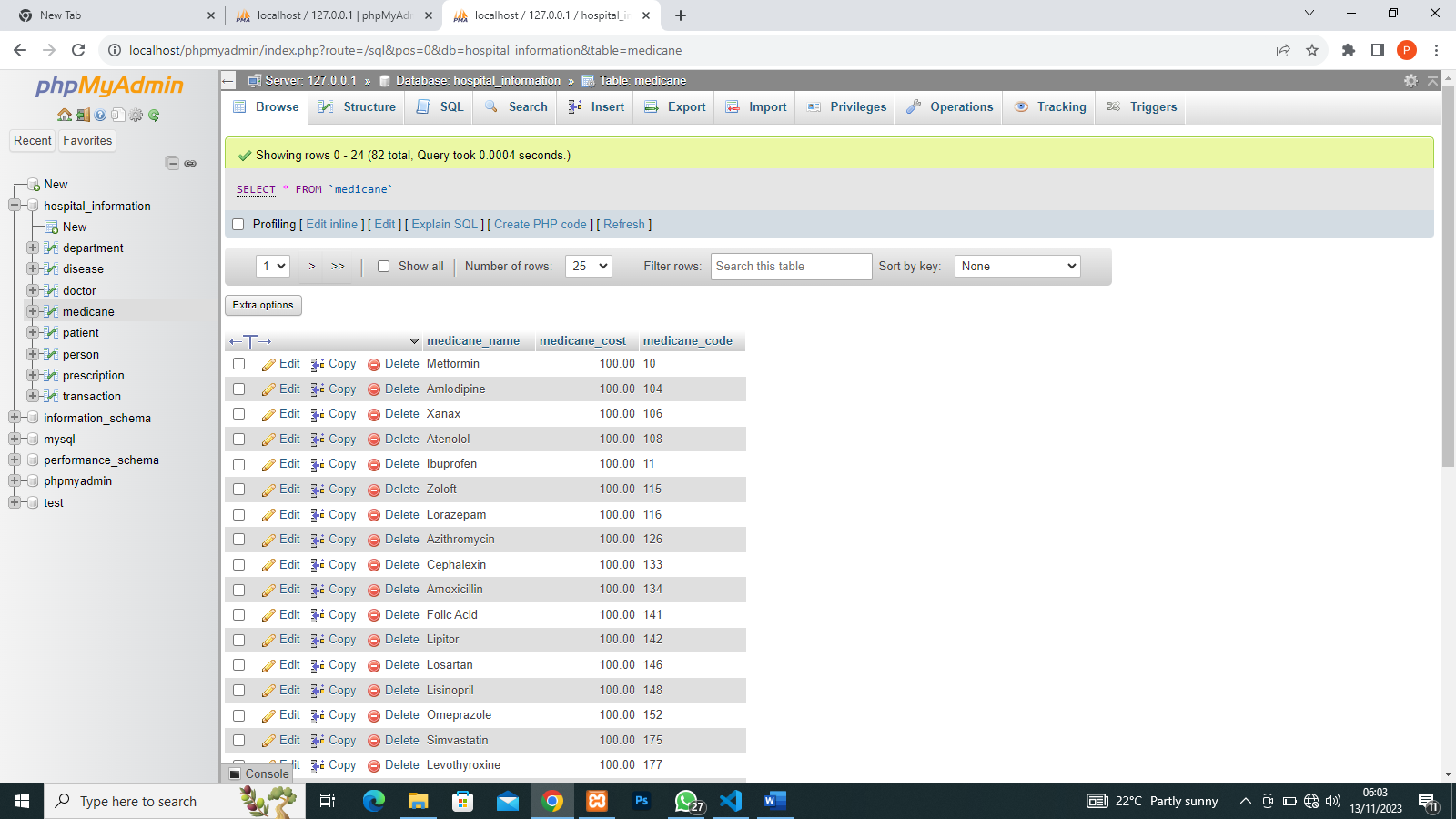


SELECT \* FROM `person` WHERE gender='male' OR status='married'

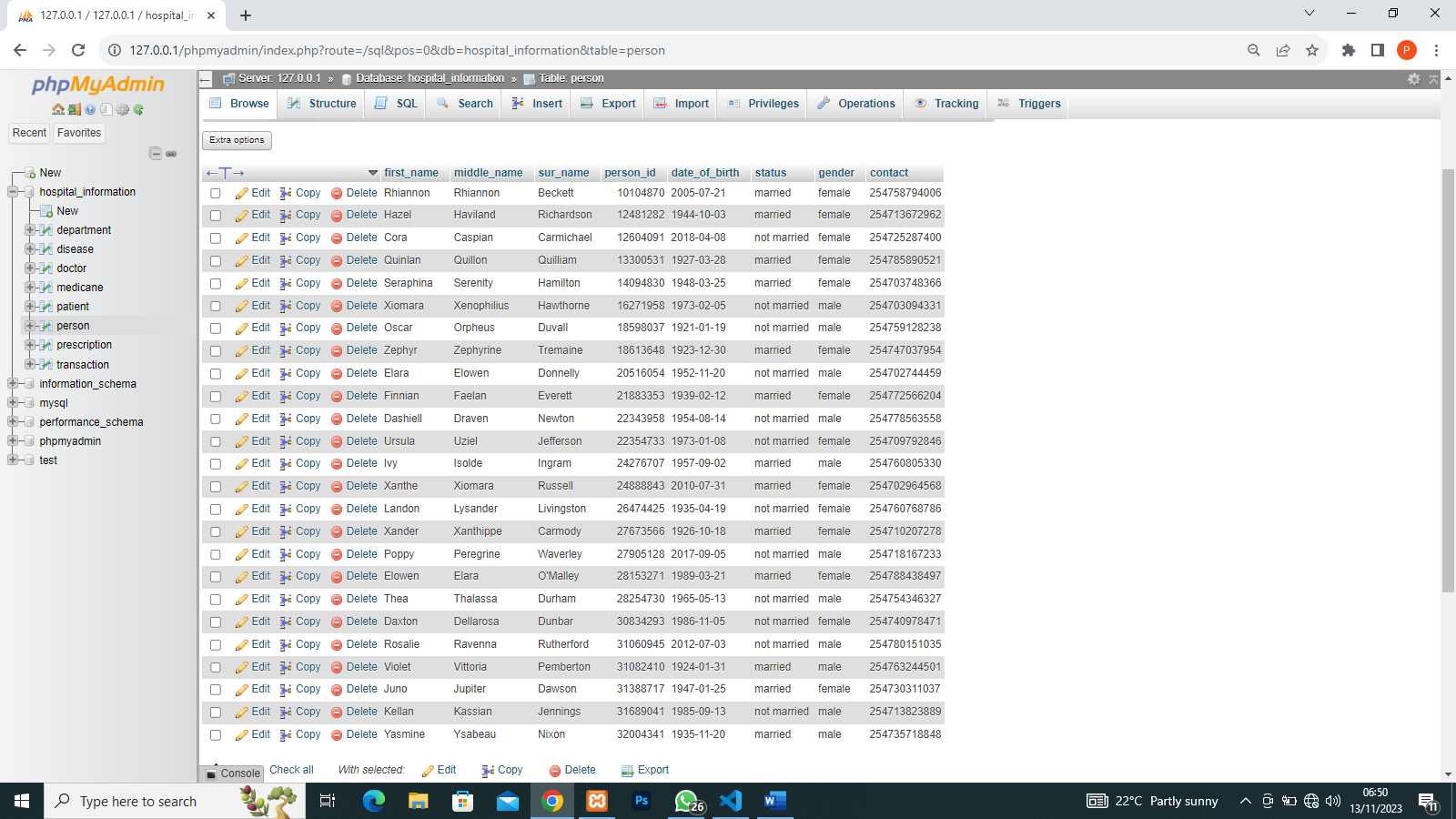


1. UPDATE QUERY

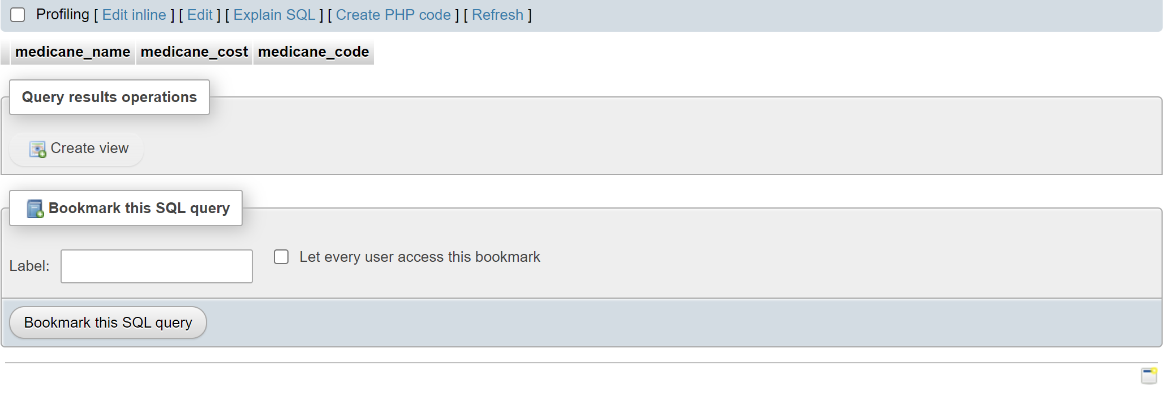
UPDATE `medicane` SET medicane\_cost='100'



UPDATE `person`SET status='not married' WHERE status='not marrie'

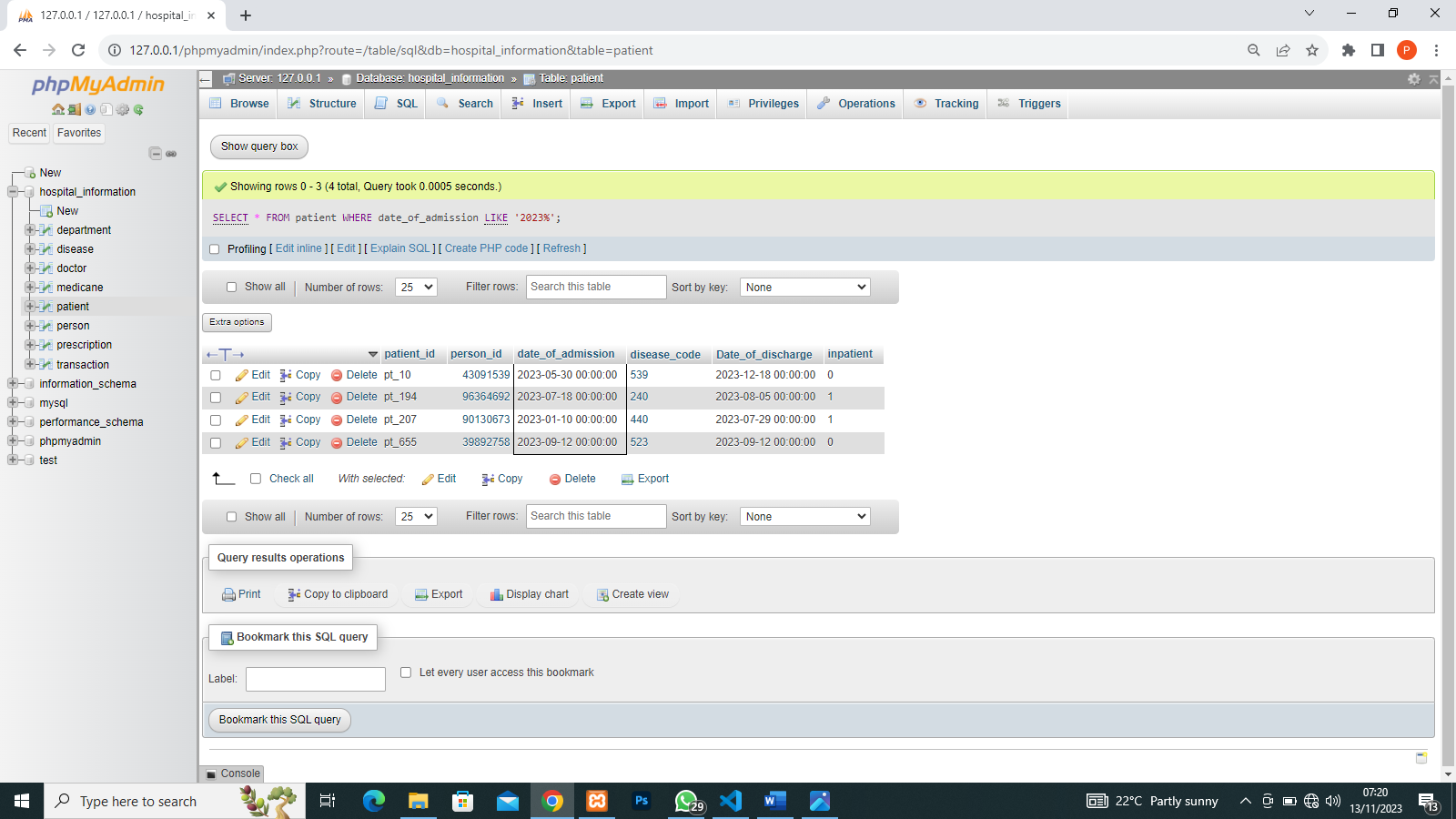


1. DELETE QUERY

DELETE FROM medicane WHERE medicane\_name=’100’

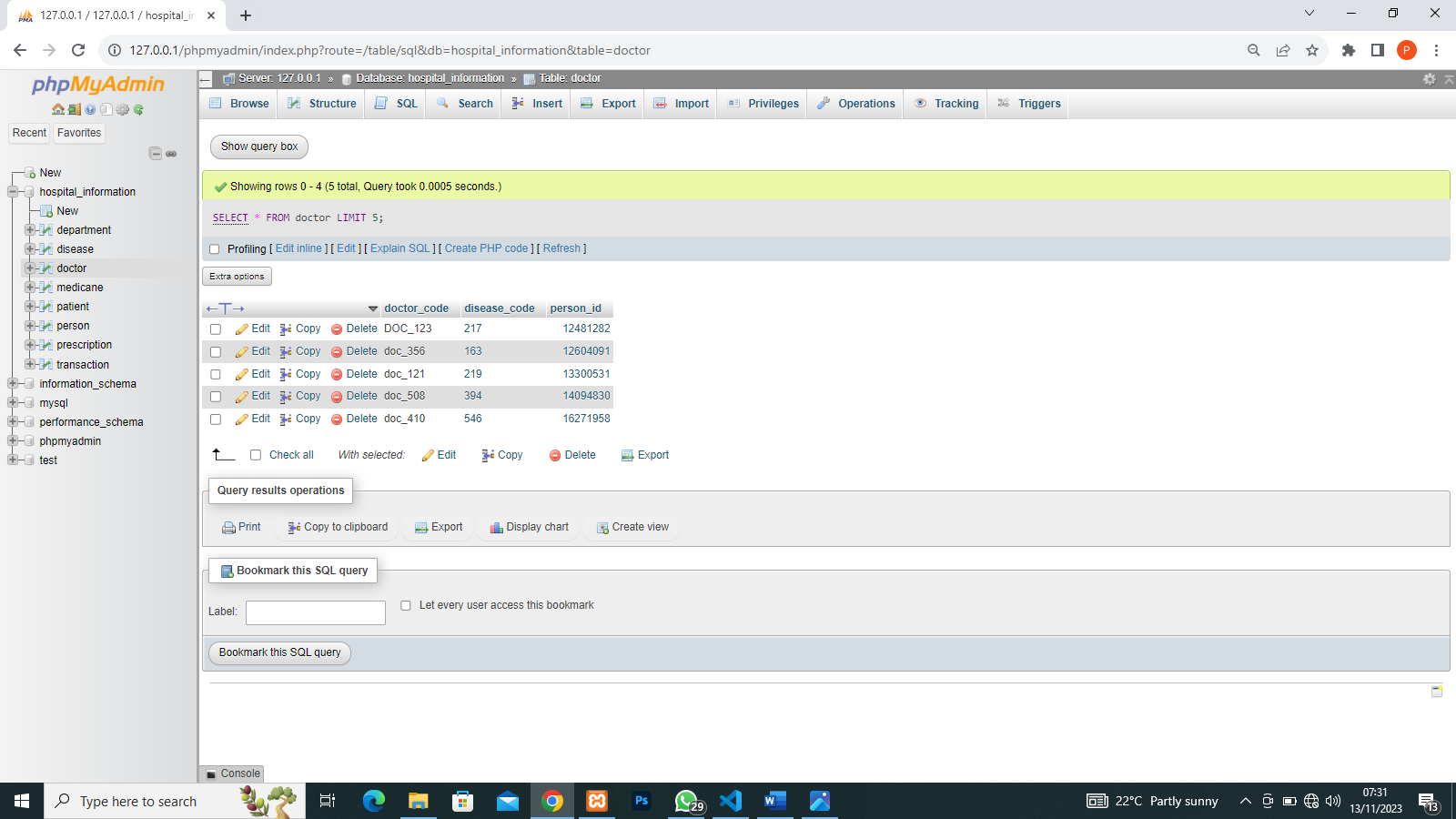
1. SQL LIKE QUERY

SELECT \* FROM ‘patient’ WHERE date\_Of\_admission LIKE ‘2023%’



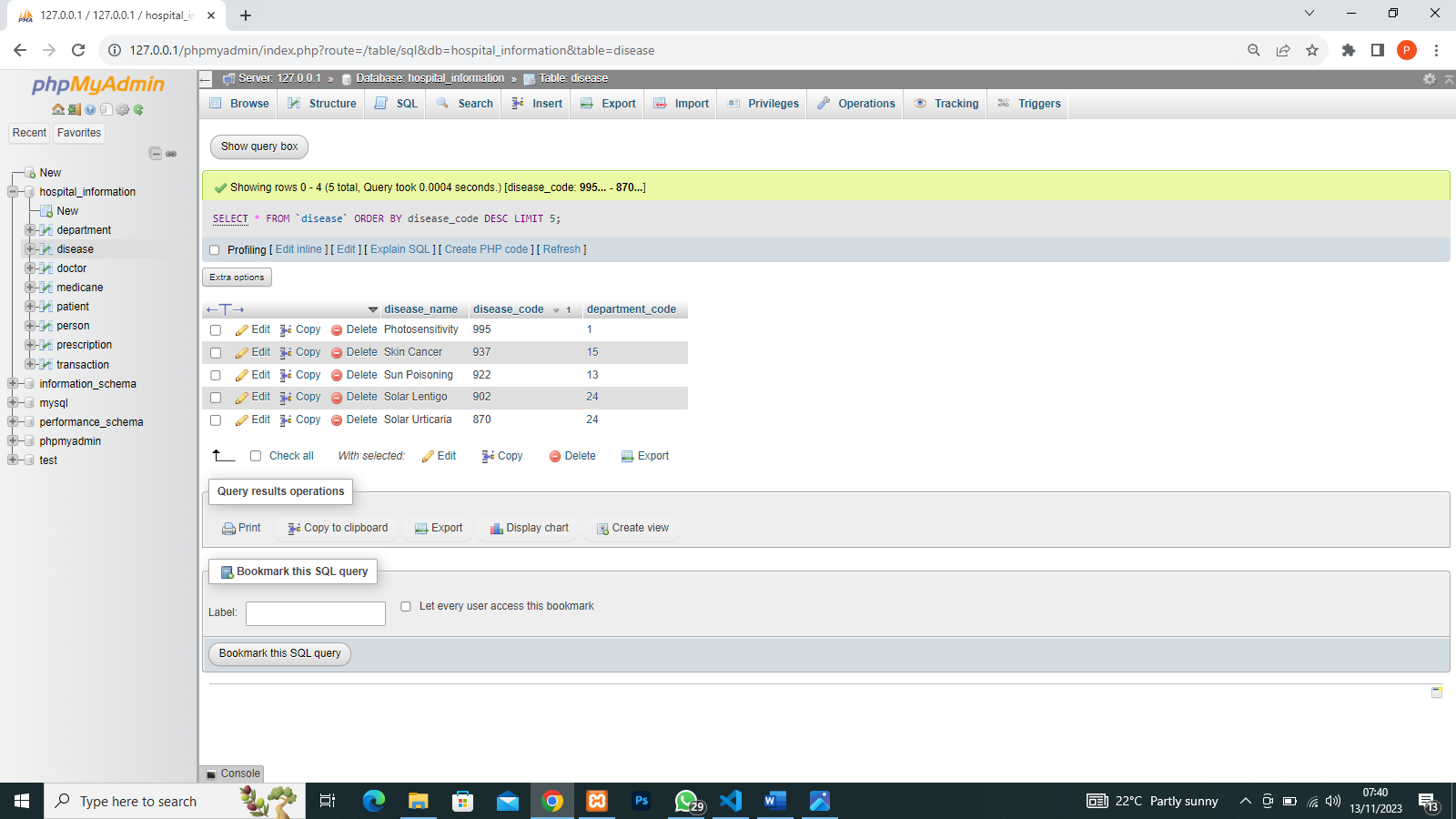
1. SQL TOP CLAUSE

SELECT \* FROM doctor LIMIT 5



1. SQL ORDER BY CLAUSE

SELECT \* FROM `disease` ORDER BY disease\_code DESC LIMIT 5



1. GROUP BY

SELECT patient\_id, SUM(amount) as amount\_payed FROM transaction GROUP BY patient\_id

