**Title of the Project: Organ Donation Management System**

**Table of Contents**

1. Abstract
2. Motivation of the Project
3. Problems Solved by the Project
4. Problem Solving Explanation
5. State of Art
6. How this is better than the solutions in market?
7. Features of the project
8. ER Diagram
9. Software Used
10. New features
11. Better options than used ones
12. Testing/Implementation
13. Queries(Creation and Insertion)
14. Connection with Heidi SQL
15. Conclusion
16. REFERENCES

**Abstract**

So, we will start from a question that what is data? Data are characteristics or information, usually numerical, that are collected through observation. In a more technical sense, data is a set of values of qualitative or quantitative variables about one or more persons or objects, while a datum is a single value of a single variable.

So the place where we store the data is database. We go to the definition of database, A database is an organized collection of data, generally stored and accessed electronically from a computer system. Where databases are more complex they are often developed using formal design and modeling techniques. And if we go further then the system which helps the user to manage the database is called Database Management System.

The Organ Donation Management System is a database an administration framework that utilizes database innovation to develop, keep up and control different sorts of information about an individual's gift or obtainment of a specific organ. It keeps up a far-reaching clinical history and other basic data like blood gathering, age, and so on of each individual in the database structure. To put it plainly, it keeps up a database containing measurable data with respect to a system of organ gift and acquisition of various nations.

**Motivation of the Project**

Organ Transplantation is a procedure in which organs from one body are removed and they are placed into the other body which is in need of that organ. Thus the organs may need to be transported from one place to other. So the organization which are working on this donation process, they are only responsible for the procurement of organs. Thus it is very important for them to have a proper database which can be used by hospitals immediately in case of emergency as these organizations represent the front-line of organ procurement, having direct contact with the hospital and the family of a recently deceased donor. The work of such organizations includes to identify the best candidates for the available organs and to coordinate with the medical institutions to decide on each organ recipient.

**Problem solved by the project**

This database solves a major issue/problem which is faced by almost all the organ donation organization which is organ wastage, this issue can only be solved by keeping proper record of the database of patients and donors, which makes the whole process easy. This database can also help in getting statistics which can help government to actually make rules for organ donation this is also a great problem solved.

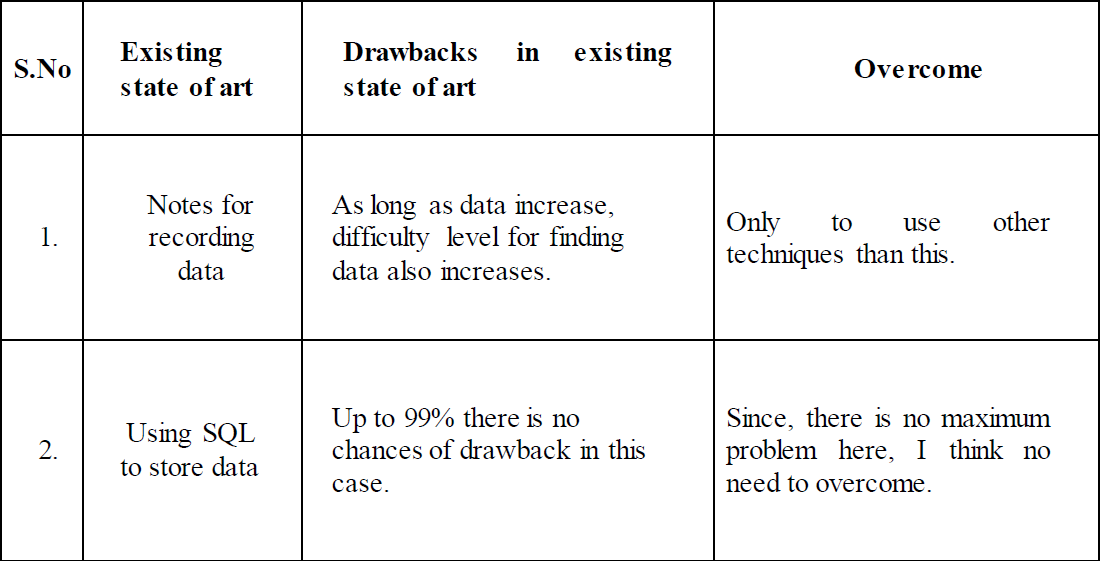
**Problem solved explained**

This database serves a variety of purposes and is critical to the proper functioning of Organ Donation, especially in today’s health care environment. These records provide statistical data showing the number of organs needed and available at a particular point of time, which makes it very important to take proper care of procurement, evaluation, planning etc. to take further steps in donation which is done by the help of the database.

Even in the THO act of 1994 it is clearly mentioned that only the hospitals which are government authorised can proceed with the organ transplantation work so it is very important by the organ donation organization to maintain a database like this to have proper information about the hospitals.

**State of Art**

A **management system** is a set of policies/processes and procedures used by an organization to ensure that it can fulfil the tasks required to achieve its objectives. These objectives cover many aspects of the organization's operations. For instance, an environmental management system enables organizations to improve their environmental performance and an occupational health and safety management system (OHSMS) enables an organization to control its occupational health and safety risks, etc.

\_

**How this solution is different than any other in the market?**

The other solutions in the market are mostly based on MongoDB, but this project is based on MariaDB based which is very similar to MySQL. There are a lot of more differences which make the novelty of the project stand out like this database contains much more information which generally other miss out.

These information include the name of the hospitals which are only government authorised, and due to this reason there are a lot of wrong organ transplantation done which were also seen in the past that how organs were taken out instead and were sold because all these organ transplantation procedure was done non authorised hospitals.

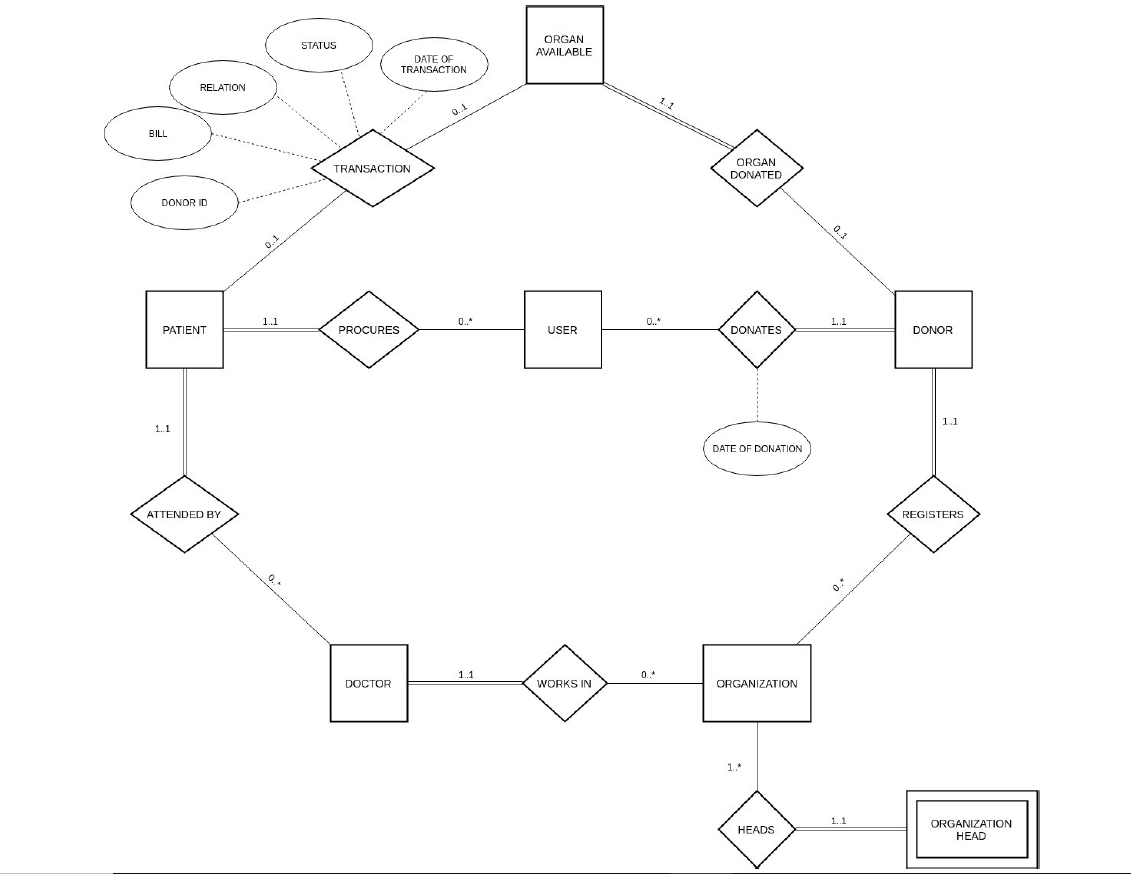
This is also much faster because of the base of MySQL, as we can retrieve data much faster than the web systems already available, in which huge chains of statements are need to be put to retrieve the data which is very hectic.

**Features of Project**

There are a lot of features that a DBMS Project provides from which some of some are, It provides security and removes redundancy, self-describing nature of a database system, insulation between programs and data abstraction, support of multiple views of the data, sharing of data and multiuser transaction processing, DBMS allows entities and relations among them to form tables, it follows the ACID concept ( Atomicity, Consistency, Isolation, and Durability), DBMS supports multi-user environment that allows users to access and manipulate data in parallel.

If we talk in the context of the project then I believe it is very important that a database needs to be protected as it a containing such important information about the donor and receptor. Multi user environment is very important in this case as all government approved hospitals will use this database.

**ER Diagram**

****

**Software Used**

**MariaDB**: Which was used to store all the data and make a database.

**HeidiSQL**: Which was used to present the database which was made in a CLI mode in a GUI mode.

**SQL:** means Structured Query Language which is the standard language for dealing with Relational Databases.

**New Features**

Well in this project the new features involved in the use of HeidiSQL to provide a graphical interface for the database for the user as well as the creator. Because some time if we want to share the whole database with the user than the user may not be comfortable in using the CLI based database to make changes, thus the GUI based database could help in this situation. Well this is the technical change/ new feature in the project.

Now the non-technical change is that the database also contains the details of hospitals which are authorised from the government, so that the organ wastage and wrong transplantation problem could be solved.

**Better options than used ones**

Better option could only be possible if MongoDB would have been used instead of MariaDB, because it has the special feature of dealing which unstructured data in a large quantity. Other than this technical part I don’t think there is anything to offer, also a web interface could have been a better option to deal with the database.

**Testing/Implementation**

Testing/Implementation both were done in the MariaDB interface as well as in HeidiSQL. This testing was done by extracting data with different conditions. Using **Views** also was the part of testing to check the security element of the project. Then primary keys and foreign keys were also a major part of implementation as they only helped in connecting the tables with each other so that the data could be retrieved in a proper manner and appropriate data could be extracted with correct queries.

Different types of other queries like **UPDATE, ALTER, DROP, DELETE** etc were used to test the database.

**Queries(Creation/Insertion)**

Database Creation:-

CREATE DATABASE DBMS\_PROJECT;

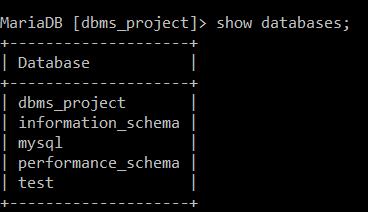
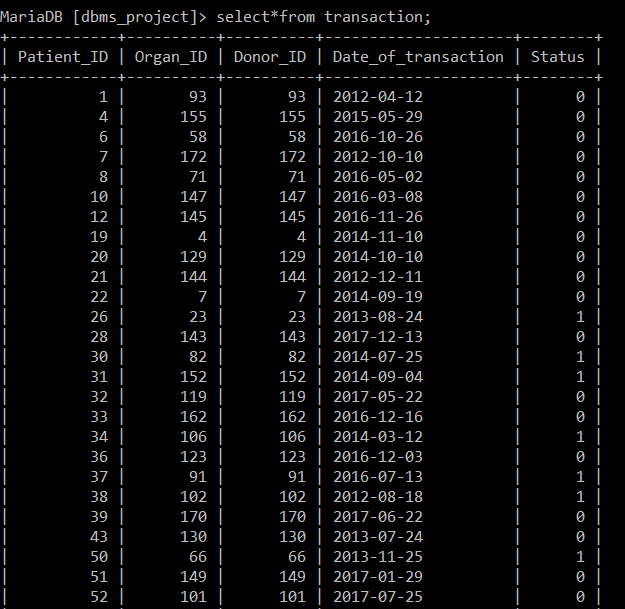


Table Creation:-

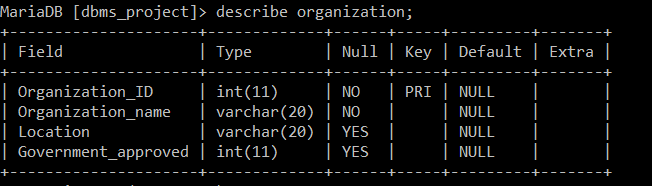
|  |
| --- |
| CREATE TABLE User( |
| User\_ID int NOT NULL, |
| Name varchar(20) NOT NULL, |
| Date\_of\_Birth date NOT NULL, |
| Medical\_insurance int, |
| Medical\_history varchar(20), |
| Street varchar(20), |
| City varchar(20), |
| State varchar(20), |
| PRIMARY KEY(User\_ID) |
| ); |

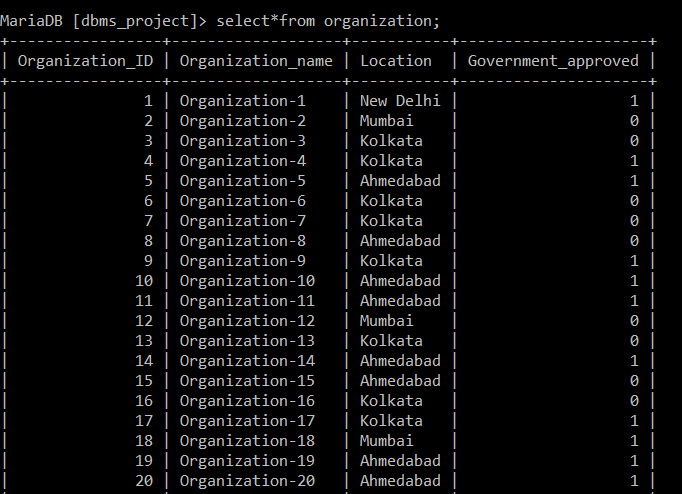




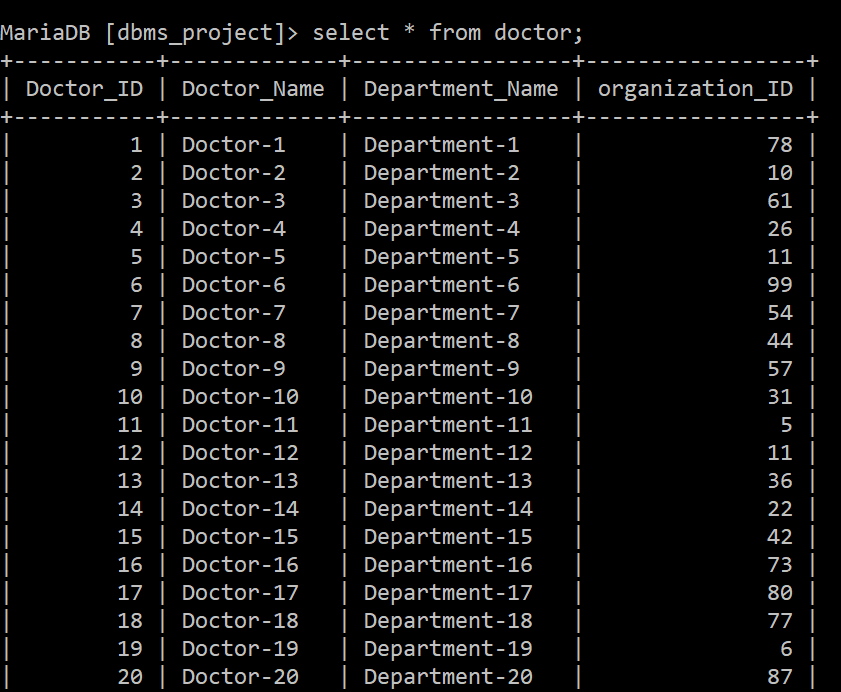
|  |
| --- |
| CREATE TABLE Organization( |
| Organization\_ID int NOT NULL, |
| Organization\_name varchar(20) NOT NULL, |
| Location varchar(20), |
| Government\_approved int, # 0 or 1 |
| PRIMARY KEY(Organization\_ID) |
| ); |





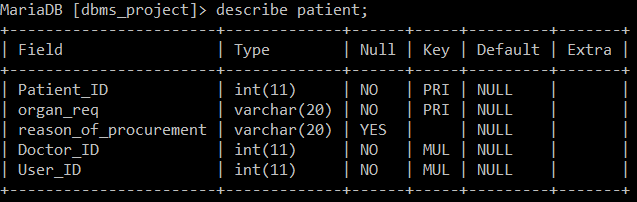


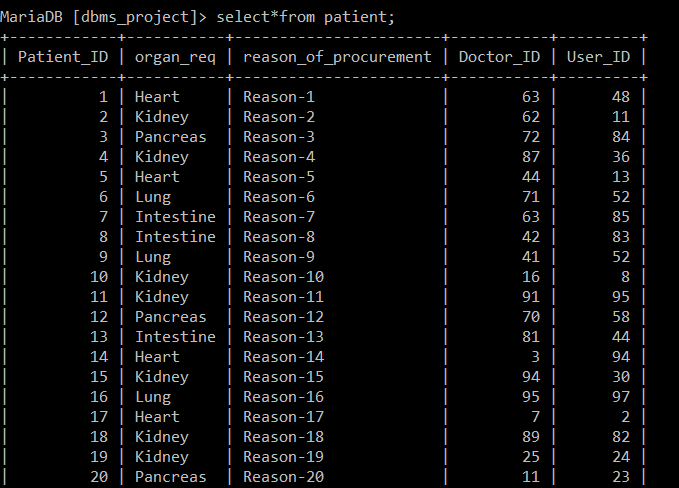
|  |
| --- |
| CREATE TABLE Doctor( |
| Doctor\_ID int NOT NULL, |
| Doctor\_Name varchar(20) NOT NULL, |
| Department\_Name varchar(20) NOT NULL, |
| organization\_ID int NOT NULL, |
| FOREIGN KEY(organization\_ID) REFERENCES Organization(organization\_ID) ON DELETE CASCADE, |
| PRIMARY KEY(Doctor\_ID) |
| ); |





|  |
| --- |
| CREATE TABLE Patient( |
| Patient\_ID int NOT NULL, |
| organ\_req varchar(20) NOT NULL, |
| reason\_of\_procurement varchar(20), |
| Doctor\_ID int NOT NULL, |
| User\_ID int NOT NULL, |
| FOREIGN KEY(User\_ID) REFERENCES User(User\_ID) ON DELETE CASCADE, |
| FOREIGN KEY(Doctor\_ID) REFERENCES Doctor(Doctor\_ID) ON DELETE CASCADE, |
| PRIMARY KEY(Patient\_Id, organ\_req) |
| ); |



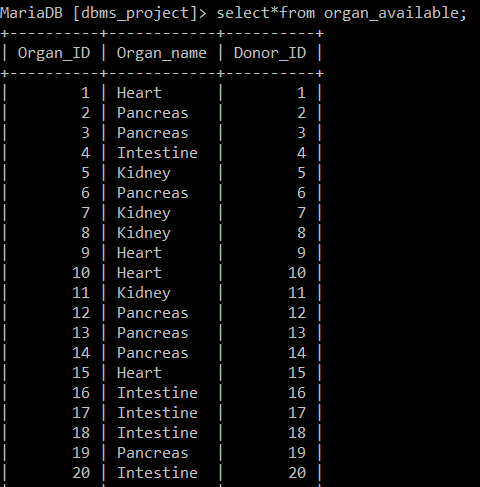




|  |
| --- |
| CREATE TABLE Donor( |
| Donor\_ID int NOT NULL, |
| organ\_donated varchar(20) NOT NULL, |
| reason\_of\_donation varchar(20), |
| Organization\_ID int NOT NULL, |
| User\_ID int NOT NULL, |
| FOREIGN KEY(User\_ID) REFERENCES User(User\_ID) ON DELETE CASCADE, |
| FOREIGN KEY(Organization\_ID) REFERENCES Organization(Organization\_ID) ON DELETE CASCADE, |
| PRIMARY KEY(Donor\_ID, organ\_donated) |
| ); |
|  |

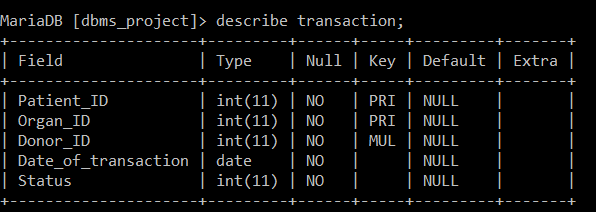


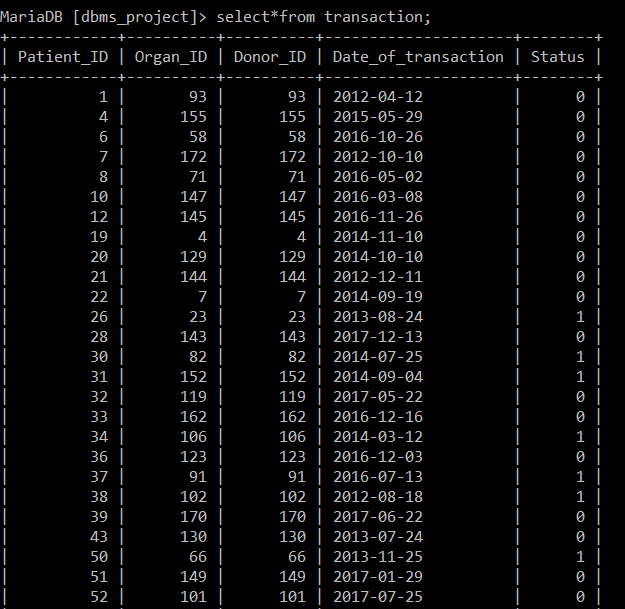
|  |
| --- |
| CREATE TABLE Organ\_available( |
| Organ\_ID int NOT NULL AUTO\_INCREMENT, |
| Organ\_name varchar(20) NOT NULL, |
| Donor\_ID int NOT NULL, |
| FOREIGN KEY(Donor\_ID) REFERENCES Donor(Donor\_ID) ON DELETE CASCADE, |
| PRIMARY KEY(Organ\_ID) |
| ); |
|  |





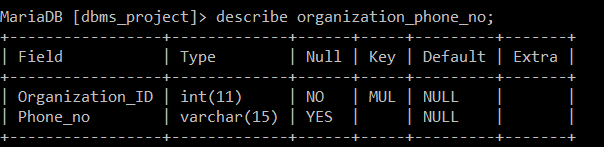
|  |
| --- |
| CREATE TABLE Transaction( |
| Patient\_ID int NOT NULL, |
| Organ\_ID int NOT NULL, |
| Donor\_ID int NOT NULL, |
| Date\_of\_transaction date NOT NULL, |
| Status int NOT NULL, #0 or 1 |
| FOREIGN KEY(Patient\_ID) REFERENCES Patient(Patient\_ID) ON DELETE CASCADE, |
| FOREIGN KEY(Donor\_ID) REFERENCES Donor(Donor\_ID) ON DELETE CASCADE, |
| PRIMARY KEY(Patient\_ID,Organ\_ID) |
| ); |
|  |

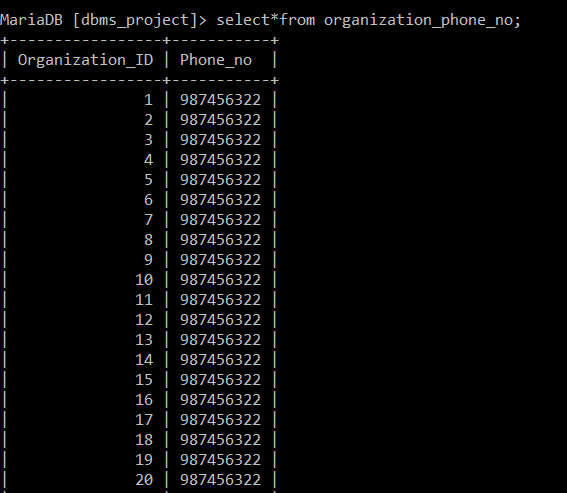






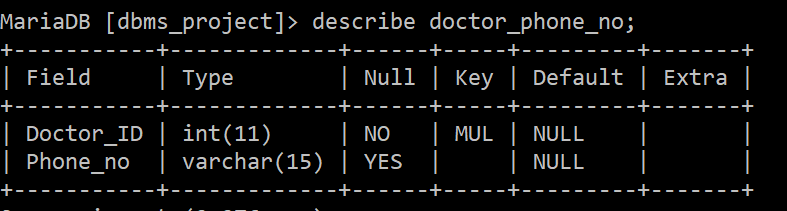
|  |
| --- |
| CREATE TABLE Organization\_phone\_no( |
| Organization\_ID int NOT NULL, |
| Phone\_no varchar(15), |
| FOREIGN KEY(Organization\_ID) REFERENCES Organization(Organization\_ID) ON DELETE CASCADE |
| ); |

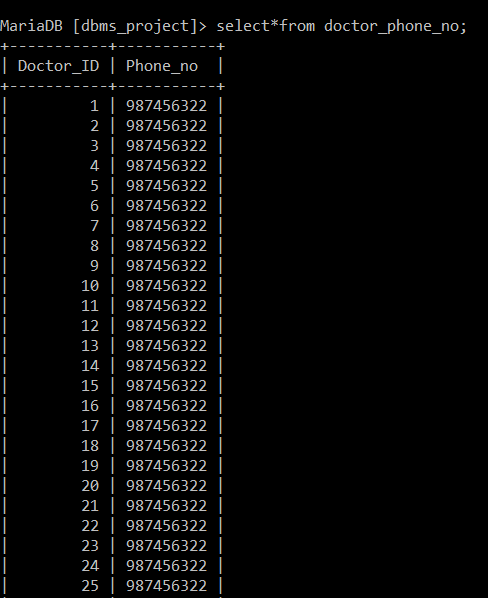






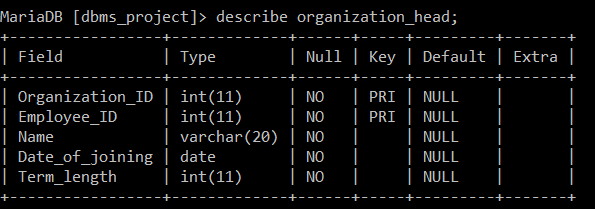
|  |
| --- |
| CREATE TABLE Doctor\_phone\_no( |
| Doctor\_ID int NOT NULL, |
| Phone\_no varchar(15), |
| FOREIGN KEY(Doctor\_ID) REFERENCES Doctor(Doctor\_ID) ON DELETE CASCADE |
| ); |







|  |  |
| --- | --- |
|  | CREATE TABLE Organization\_head( |
|  | Organization\_ID int NOT NULL, |
|  | Employee\_ID int NOT NULL, |
|  | Name varchar(20) NOT NULL, |
|  | Date\_of\_joining date NOT NULL, |
|  | Term\_length int NOT NULL, |
|  | FOREIGN KEY(Organization\_ID) REFERENCES Organization(Organization\_ID) ON DELETE CASCADE, |
|  | PRIMARY KEY(Organization\_ID,Employee\_ID) |
|  | ); |



1) Trigger for adding Donor information to Log table.

delimiter //

create trigger ADD\_DONOR\_LOG

after insert

on Donor

for each row

begin

insert into log values

(now(), concat("Inserted new Donor",

cast(new.Donor\_Id as char)));

end //

delimiter ;

2) Trigger for adding “Update” action information in Log table.

create trigger UPD\_DONOR\_LOG

after update

on Donor

for each row

begin

insert into log values

(now(), concat("Updated Donor Details",

cast(new.Donor\_Id as char)));

end //

delimiter ;

3) Trigger for adding “Delete” action information in Log table.

create trigger DEL\_DONOR\_LOG

after delete

on Donor

for each row

begin

insert into log values

(now(), concat("Deleted Donor ",

cast(old.Donor\_Id as char)));

end //

delimiter ;

4) Trigger for adding “Add patient” action information in Log table

create trigger ADD\_PATIENT\_LOG

after insert

on Patient

for each row

begin

insert into log values

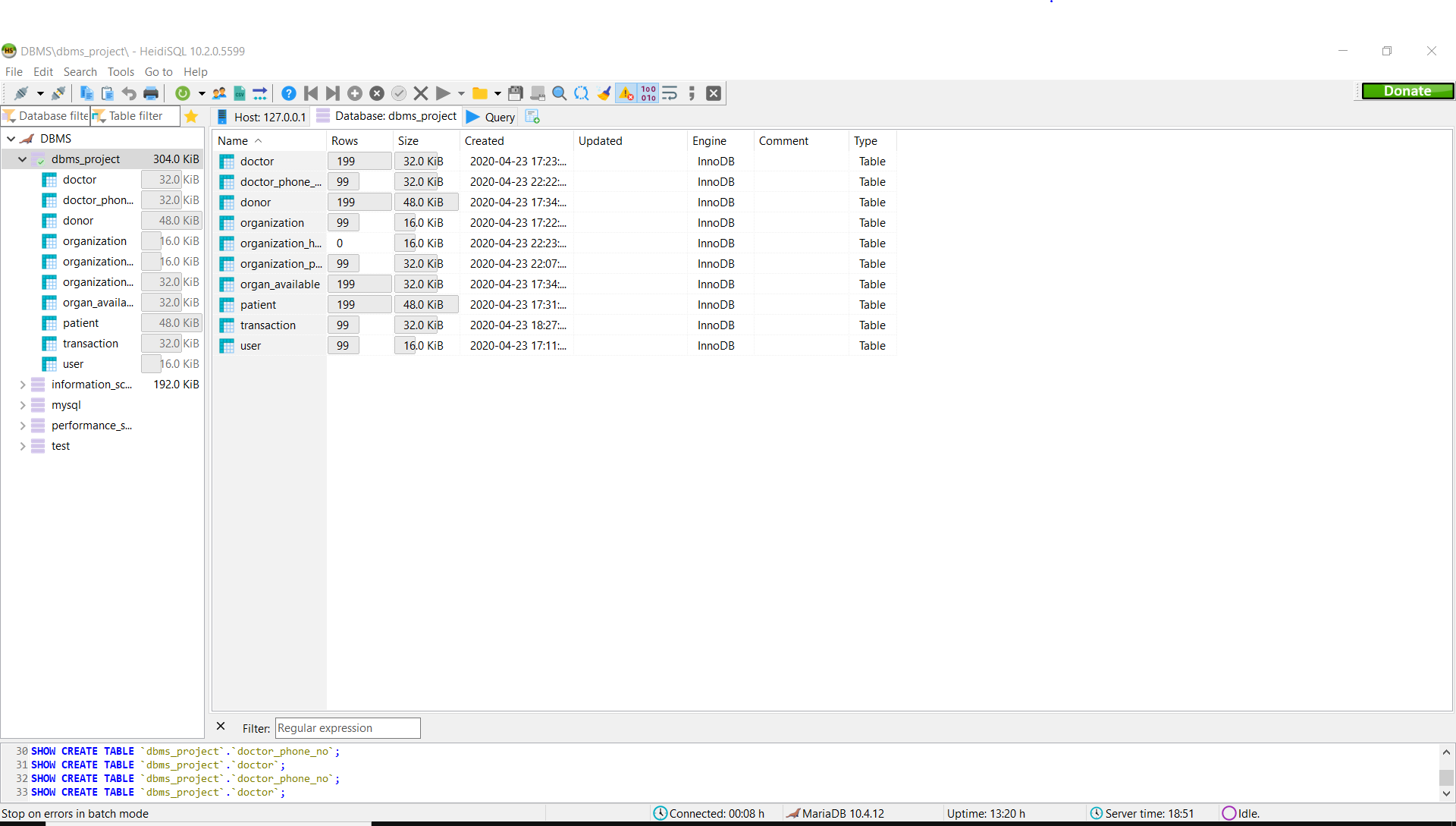
(now(), concat("Inserted new Patient

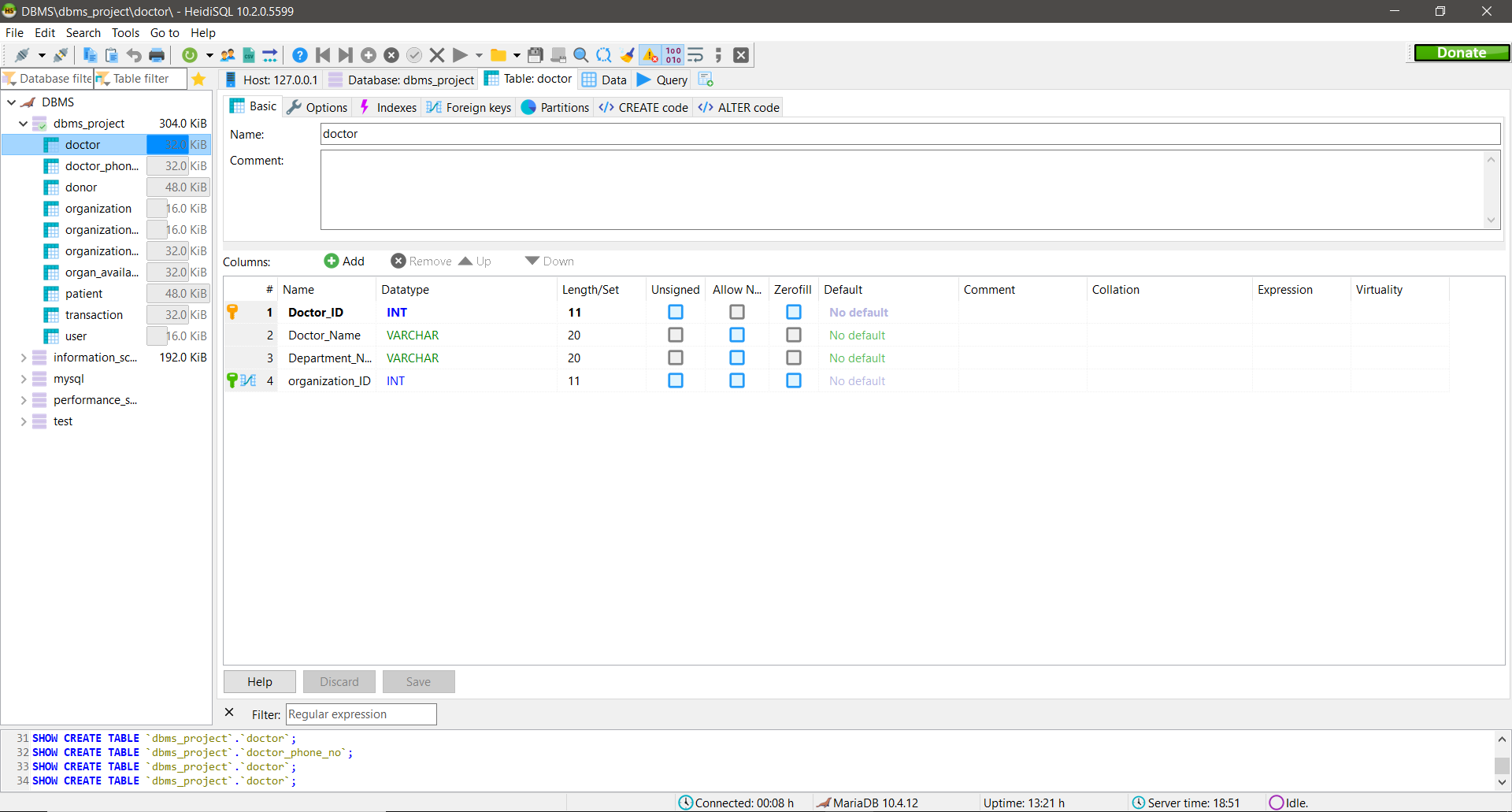
", cast(new.Patient\_Id as char)));

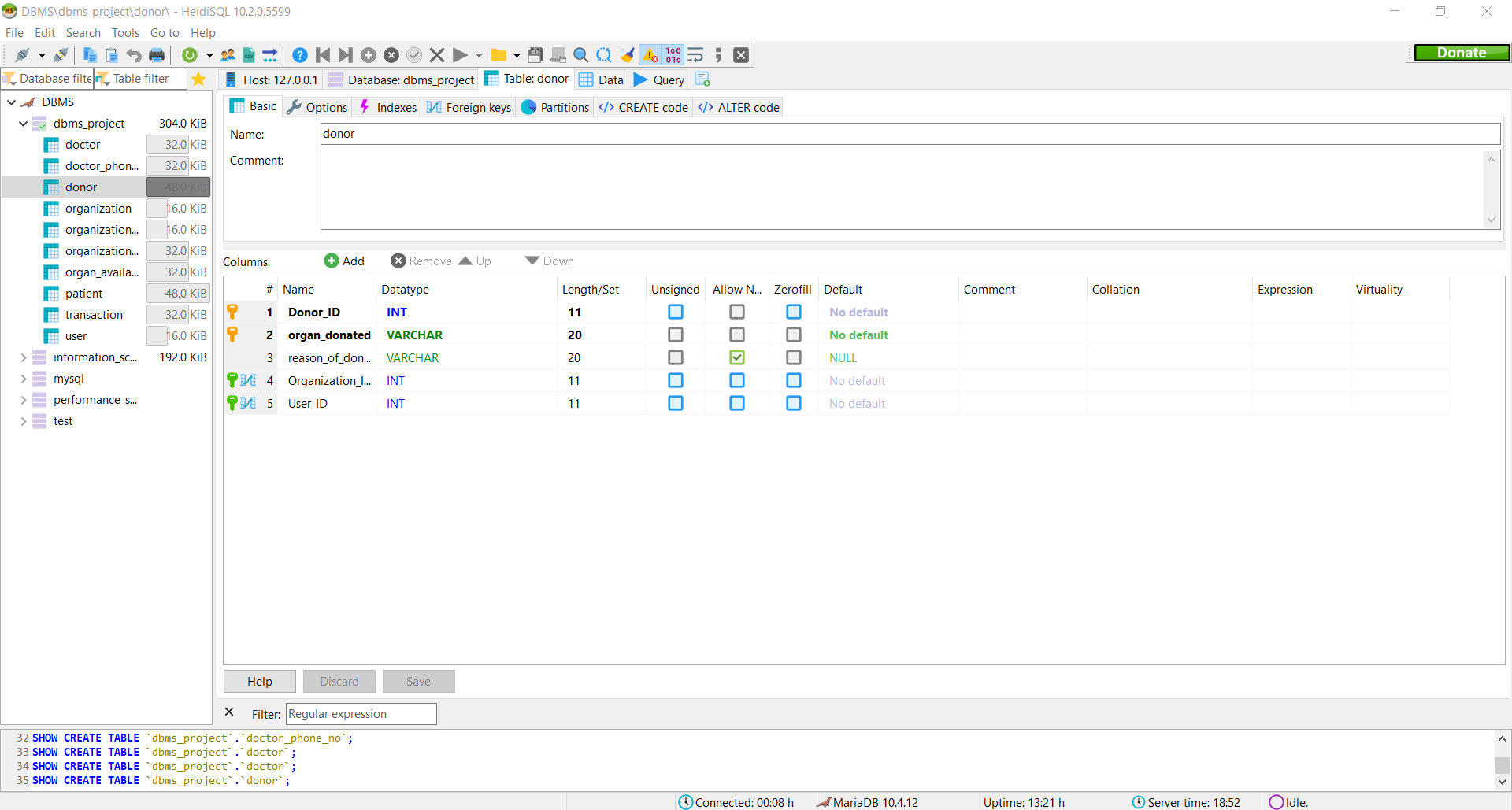
end //

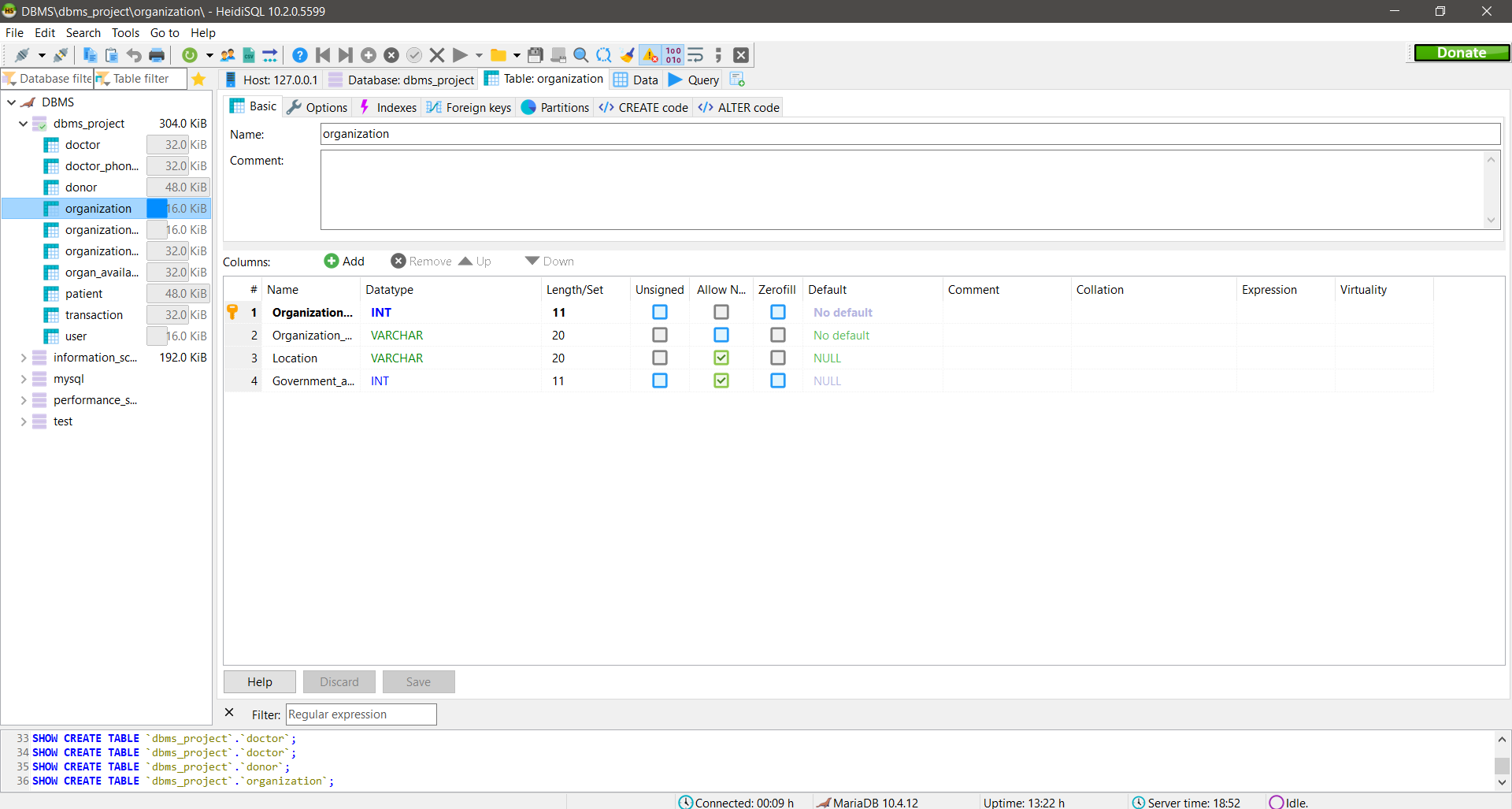
delimiter ;

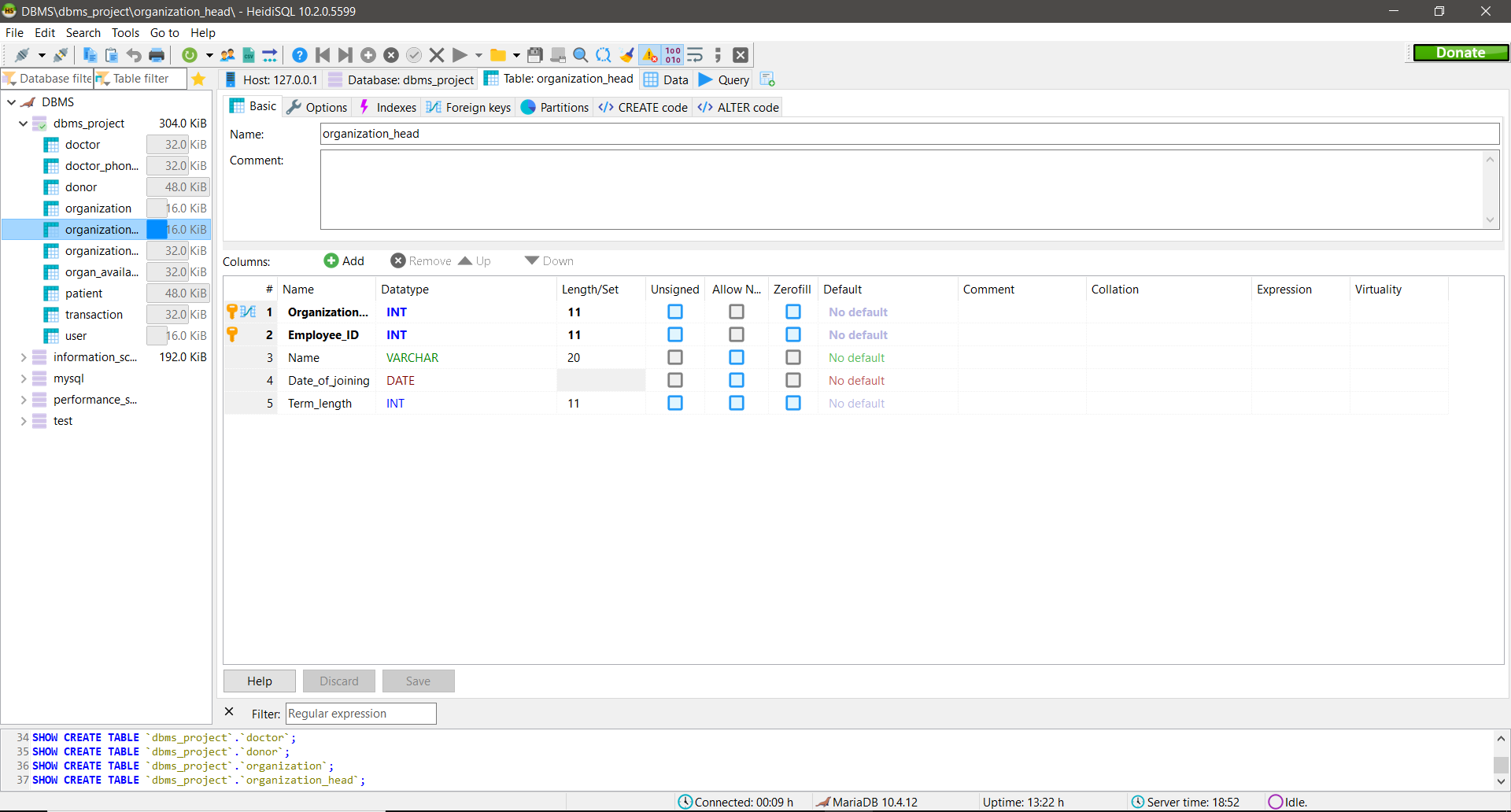
**Connection with HeidiSQL**

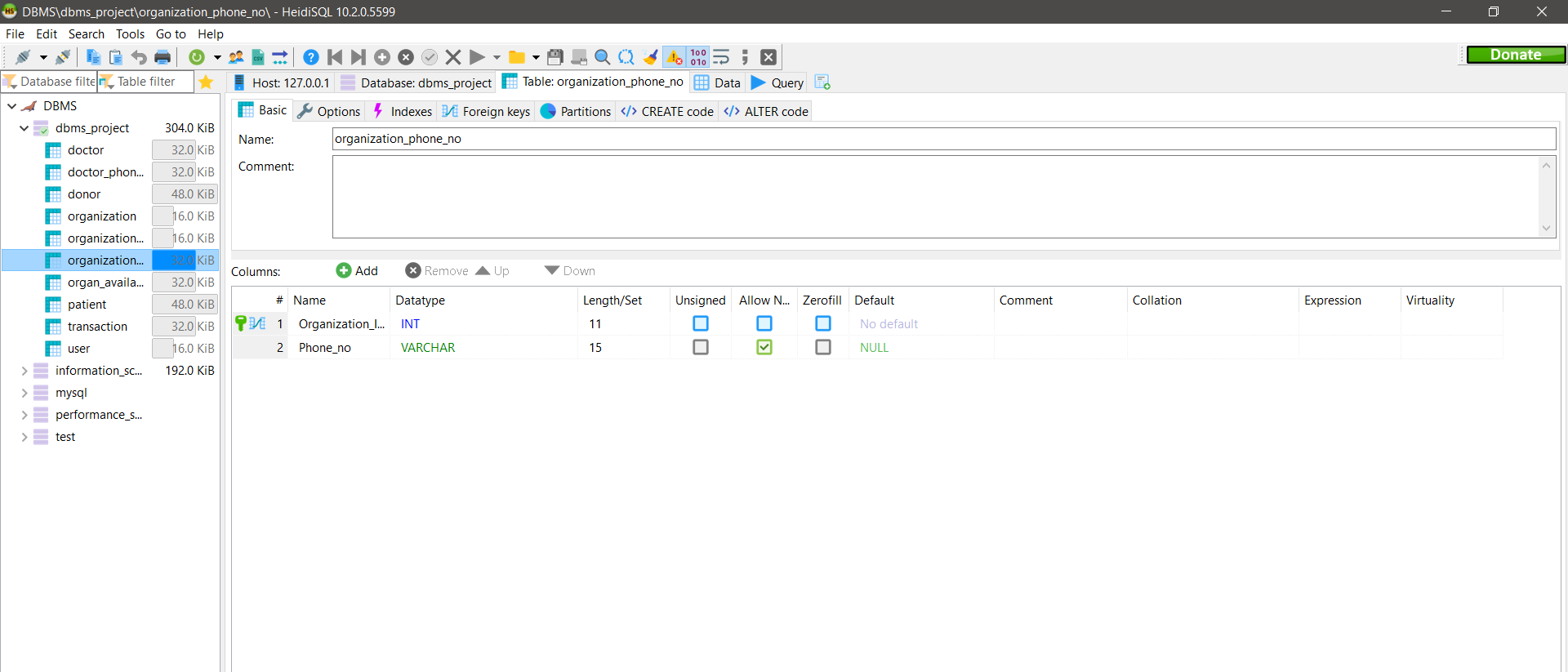
****

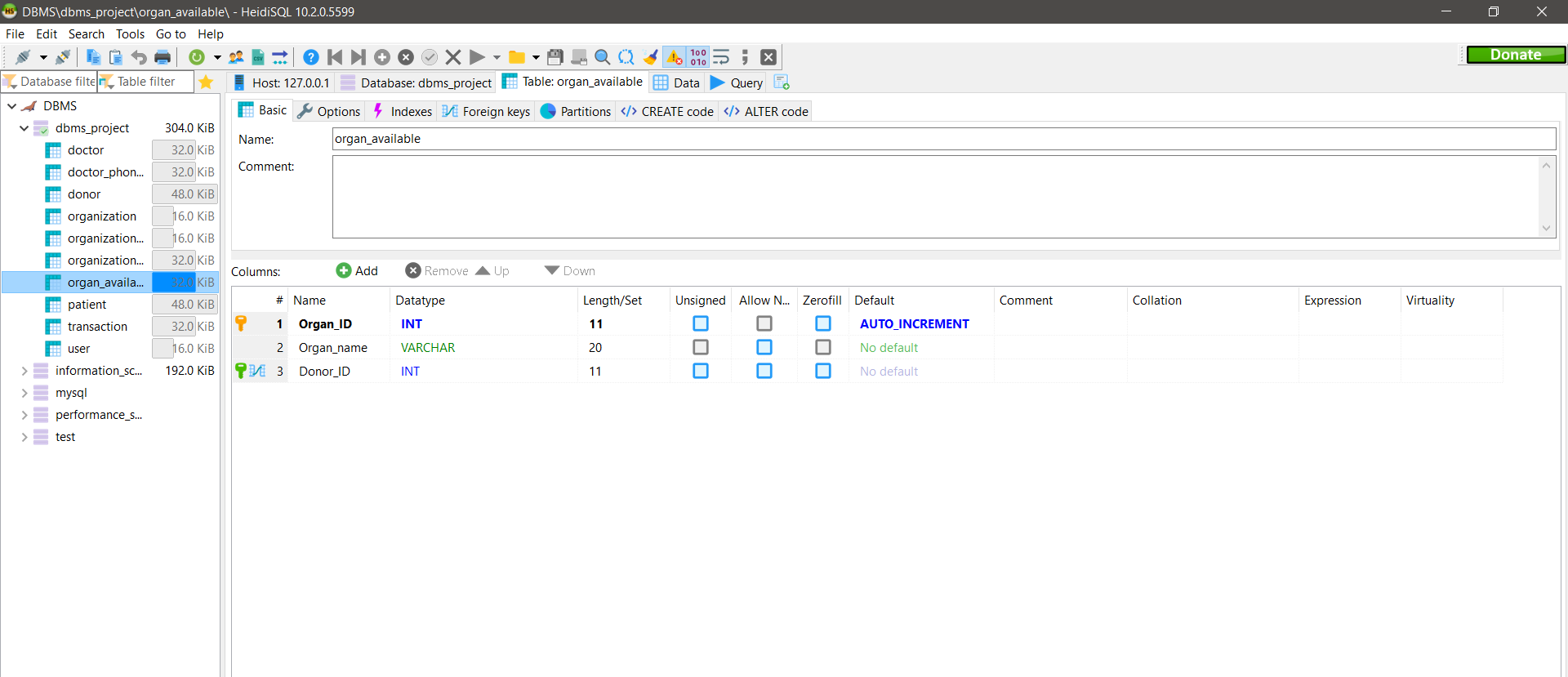
****

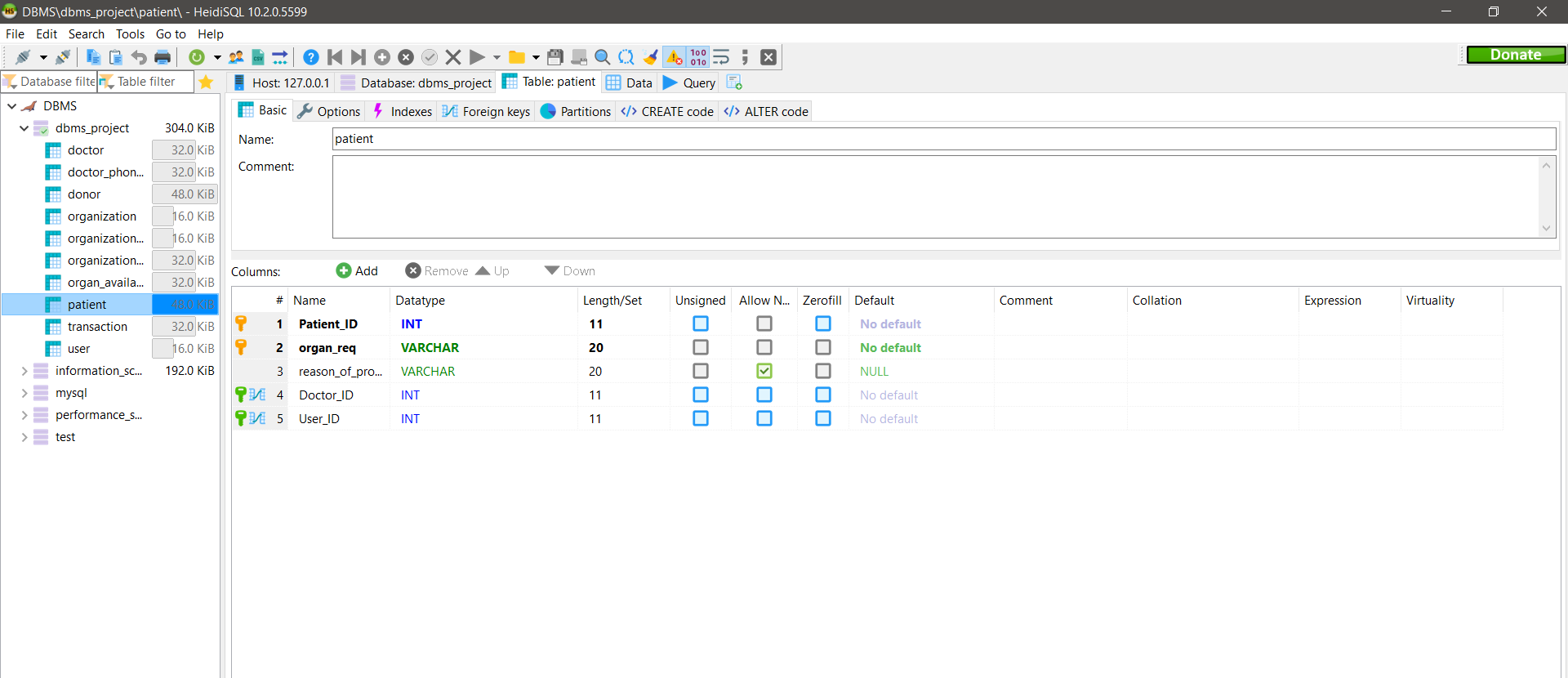
****

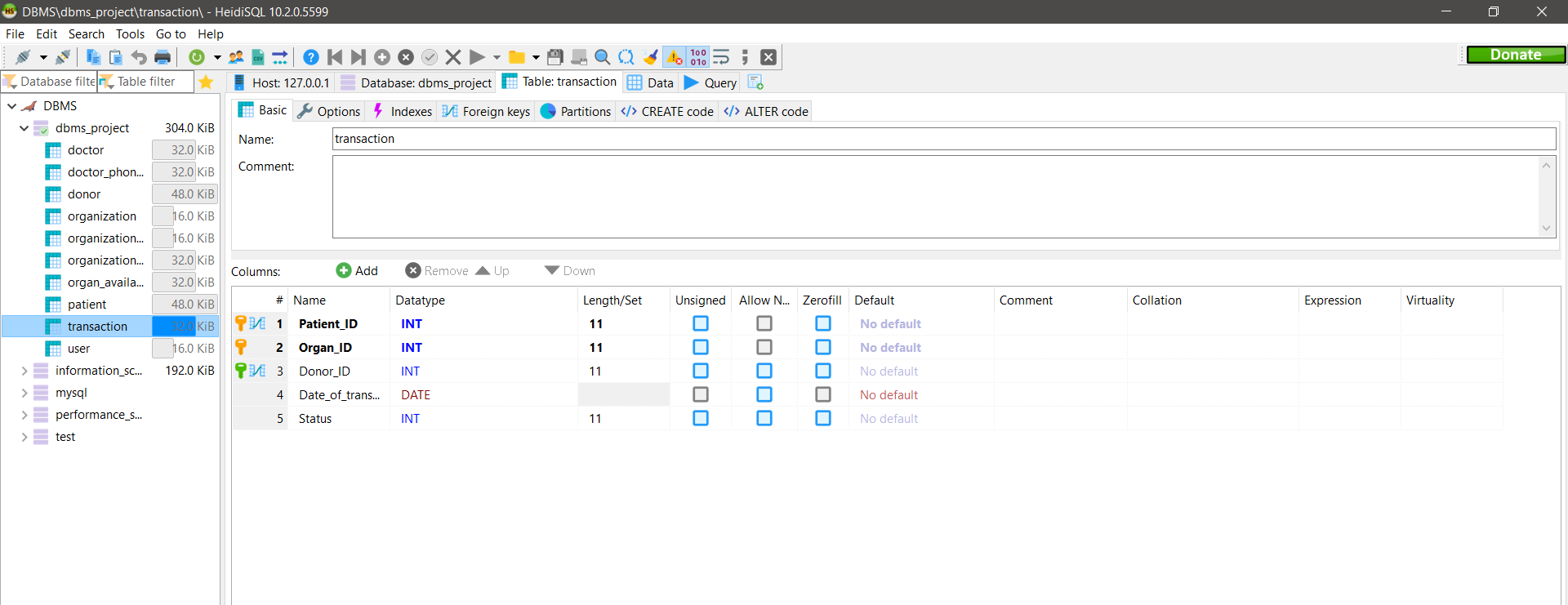
****

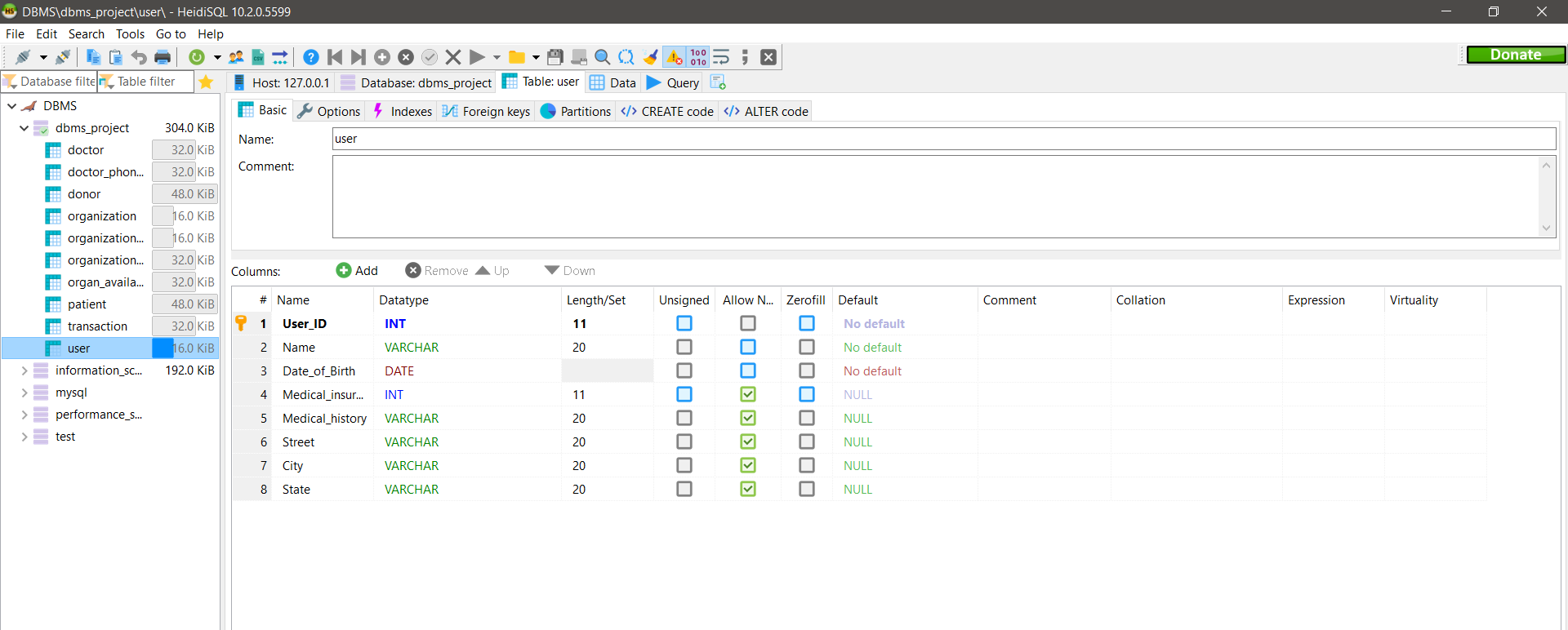
****

****

****

****

****

****

**Conclusion**

This record serves a variety of purposes and is critical to the proper functioning of Organ Donation and Procurement Network, especially in today’s complicated health care environment. These records provide statistical information regarding the number of organs needed and available at a particular point of time. It is essential for planning, evaluating and coordinating organ donation and procurement. In India, the Transplantation of organs is done according to the Transplantation of Human Organs (THO) Act,1994. Many new rules had been added to the act, later on, to cater to current needs. According to this Act, every transplantation operation should be approved by the Government Organization. So the records of transplantation are there with the organization. Also, these operations can only be done in Government-authorized Hospitals. Our aim to create a solution that effectively deals with the problems of finding donors and

also providing Statistical data of the transplants that can help the government to form better rules and regulations.

**REFERENCES:**

https://www.w3schools.com/sql/

https://dev.mysql.com/doc/workbench/en/

https://en.wikipedia.org/wiki/Database

https://www.geeksforgeeks.org/introduction-of-dbms-database-management-system-set-1/

https://www.javatpoint.com/dbms-tutorial