Blood Bank Donation System

Database

**Created By:**

**Arhanjit Singh Sodhi**

**Project Summary**

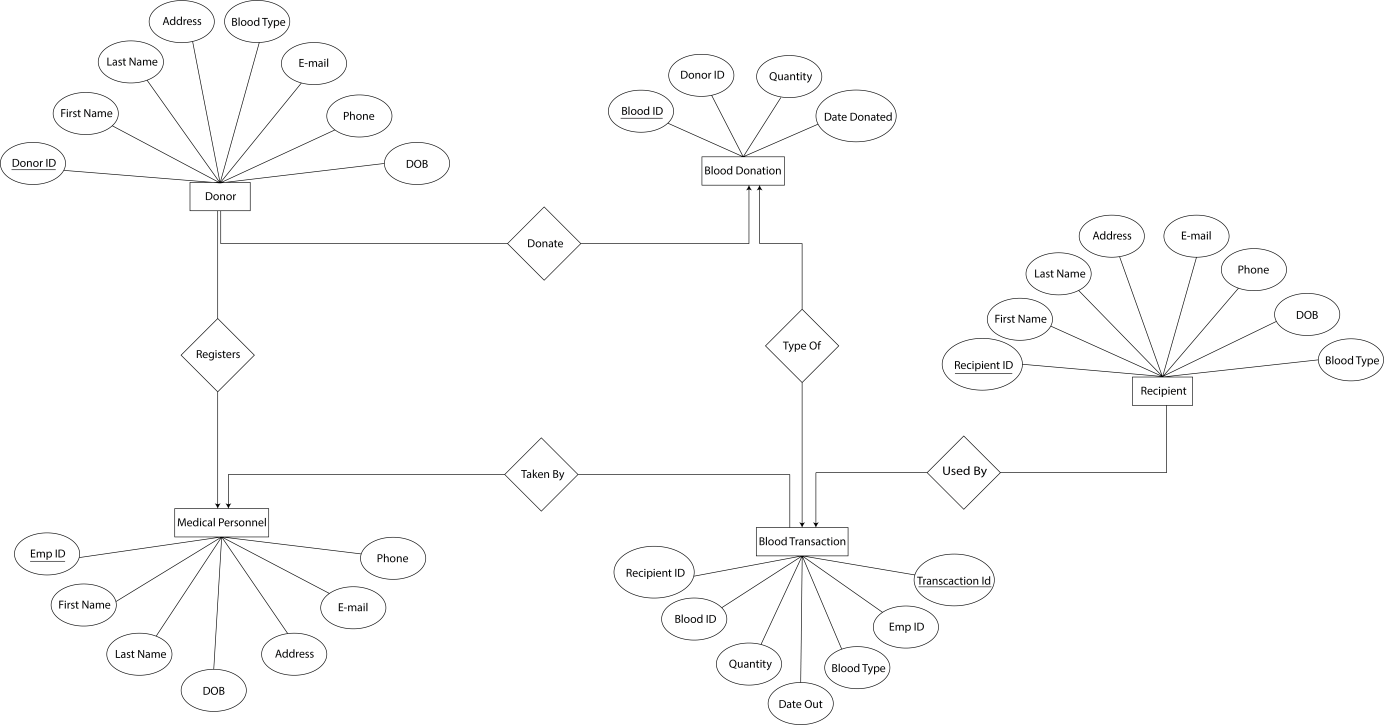
Our project is based on a Blood Bank Donation System Database. In our project we created multiple tables with multiple attributes each. Our purpose was to be able to have a database that stored information about the Donor, the Recipient, Medical personnel, how much and when the blood was donated, and whenever there was a transaction made between the donor, the medical personnel, and the recipient. All tables are linked together using foreign keys to enforce relationships. We created three different views: PatientSeen, BloodStock, and a PatientList. This adds security to the individual tables by not showing irrelevant data to unauthorized viewers. The views used joins to combine columns from different

tables. Our Donor and Recipient tables are extremely similar to each other. Both of these tables primary key is the ID attribute. They both include the person's first name, last name, address, email, phone number, date of birth, and blood type. The MedicalPersonnel table’s primary key is also the ID attribute. This table includes the same attributes as the Donor and Recipient table with the exception of not including their blood type. Our BloodDonation tables primary key is the bloodID attribute with the foreign key being donorID which is linked to the Donor table. This table also includes the date when the blood was donated and the quantity or amount donated. Lastly, our BloodTransaction table has one primary key and

three foreign keys. The primary key is the transacID and the three foreign keys are: empID which is linked to the MedicalPersonnel Table, the recipientID which is linked to the Recipient table, and the bloodID which is linked to the BloodDonation table. The first view that we created is the PatientSeen view. This view is for medical personnel who would like to see which patients they’ve seen. This view includes the medical personnel who either administered donated blood or took blood, the donor's or recipient's name, and the date it

happened. The second view we created was the BloodStock view. This view is seen by the medical personnel. This is just a simple view that shows all the blood types we have and the quantity of each we have in stock. The final and third view we created is the PatientList view. This view was made to show certain information about all the patients that came through the blood bank whether they were donating or receiving blood. The view will include the patient's full name, blood type, age, address, email, and phone number.

**ER Diagram**

****

**Tables**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table Name | Attributes | Constraint | Type |  |
| Donor | donorID | pk | int(11) | not null |
|  | first\_name |  | varchar(50) | not null |
|  | last\_name |  | varchar(50) | not null |
|  | address |  | varchar(60) | not null |
|  | email |  | varchar(100) | not null |
|  | phone |  | varchar(20) | not null |
|  | dob |  | date | not null |
|  | blood\_type |  | varchar(3) | not null |
|  |  |  |  |  |
| Recipient | Recipient\_id | pk | int(11) | not null |
|  | first\_name |  | varchar(50) | not null |
|  | last\_name |  | varchar(50) | not null |
|  | address |  | varchar(60) | not null |
|  | email |  | varchar(100) | not null |
|  | phone |  | varchar(20) | not null |
|  | dob |  | date | not null |
|  | blood\_type |  | varchar(3) | not null |
|  |  |  |  |  |
| BloodDonation | Blood\_id | pk | int(11) | not null |
|  | donorID | fk to Donor | int(11) | not null |
|  | Blood\_donated |  | datetime | not null |
|  | quantity |  | int | not null |
|  |  |  |  |  |
| MedicalPersonnel | empID | pk | int(11) | not null |
|  | firstName |  | varchar(50) | not null |
|  |  |  |  |  |
|  | lastName |  | varchar(50) | not null |
|  | address |  | varchar(60) | not null |
|  | email |  | varchar(100) | not null |
|  | phone |  | varchar(20) | not null |
|  | dob |  | date | not null |
|  |  |  |  |  |
| BloodTransaction | transactID | pk | int(11) | not null |
|  | empID | fk to MedPersonnel | int(11) | not null |
|  | dateOut |  | datetime | not null |
|  | quantity |  | int | not null |
|  | recipientID | fk to recipient | int(11) | not null |
|  | bloodType |  | varchar(3) | not null |
|  | bloodID | fk to BloodDonation | int(11) | not null |

**Statements**

create table Donor (

donorID INT not null,

first\_name VARCHAR(50) not null,

last\_name VARCHAR(50) not null,

address VARCHAR(60) not null,

email VARCHAR(100) not null,

phone VARCHAR(20) not null,

dob DATE not null,

blood\_type VARCHAR(3) not null,

Primary key(donorID)

);

create table Recipient (

recipient\_id INT not null,

first\_name VARCHAR(50) not null,

last\_name VARCHAR(50) not null,

address VARCHAR(60) not null,

email VARCHAR(100) not null,

phone VARCHAR(20) not null,

dob DATE not null,

blood\_type VARCHAR(3) not null,

Primary key(recipient\_id)

);

create table blood\_donation (

donorID INT(11) not null,

blood\_donated DATE not null,

blood\_id INT(11) not null,

quantity INT not null,

Primary key(blood\_id),

Foreign key (donorID) references

Donor(donorID)

);

create table MedicalPersonal (

empid INT not null,

first\_name VARCHAR(50) not null,

last\_name VARCHAR(50) not null,

email VARCHAR(100) not null,

gender VARCHAR(20) not null,

dob DATE not null,

address VARCHAR(60),

Primary key(empid)

);

create table BloodTransaction (

empid INT(11) not null,

transaction\_id INT(11) not null,

recipient\_id INT(11) not null,

blood\_type VARCHAR(50),

quantity INT not null,

DateOut DATE not null,

bloodID INT(11) not null

Primary key(transaction\_id),

Foreign key(empid) references

MedicalPersonal(empid),

Foreign key(recipient\_id) references

Recipient(recipient\_id),

Foreign key(blood\_id) references

Blood\_donation(blood\_id),

);

create view PatientSeen as select

concat\_ws(' ', m.first\_name, m.last\_name) as 'MedicalPersonal',

concat\_ws(' ', r.first\_name, r.last\_name) as 'Patient Name',

dateOut as 'Date Seen'

From MedicalPersonal m, BloodTransaction b,

Recipient r

where m.empid = b.empid AND r.recipient\_id = b.recipient\_id

order by m.last\_name ASC;

create view BloodStock as select

Donor.blood\_type as 'Blood Type',

sum(Blood\_donation.quantity) as 'In Stock' from Blood\_donation

join Donor on Blood\_donation.donorID = Donor.donorID

where Blood\_donation.blood\_id

not in (select blood\_id from BloodTransaction)

group by blood\_type

create view PatientList

as select

concat\_ws(', ', last\_name, first\_name) as Name,

bloodType as 'Blood Type',

date\_format(from\_days(to\_days(now())- to\_days(dob)),'%y')+0 as Age,

address as Address,

email as Email,

phone as Phone from Donor

union

select

concat\_ws(', ', last\_name, first\_name) as 'PatientName',

Blood\_type as 'Blood Type',

date\_format(from\_days(to\_days(now())- to\_days(dob)),'%y')+0 as Age,

address as Address,

email as Email,

phone as Phone

from Recipient

Order by Name

--Exception:

--If blood\_type not available

Declare

Bg varchar(15);

Check EXCEPTION;

Begin

Bg=&blood;

If NOT EXISTS(select \* from bloodTransaction where blood\_type=Bg) then

Raise check;

End if;

EXCEPTION

When check then

Dbms\_output.put\_line(‘Blood type not available’);

End;

Create or replace trigger quant before insert or update or delete of quantity on Blood Donation for each row;

Begin

Dbms\_output.put\_line(‘Blood quantity not available’);

End;

Create or replace trigger name\_not before insert or update or delete of bloodtype on BloodTransaction for each row;

Dbms\_output.put\_line(‘Blood Type not available’);

End;

Cursor c1is select first\_name,last\_name,address from donor where donorID=2;

Rec c1%rowtype;

Begin

Open c1;

Loop

Fetch c1 into rec;

Exit;

End loop;

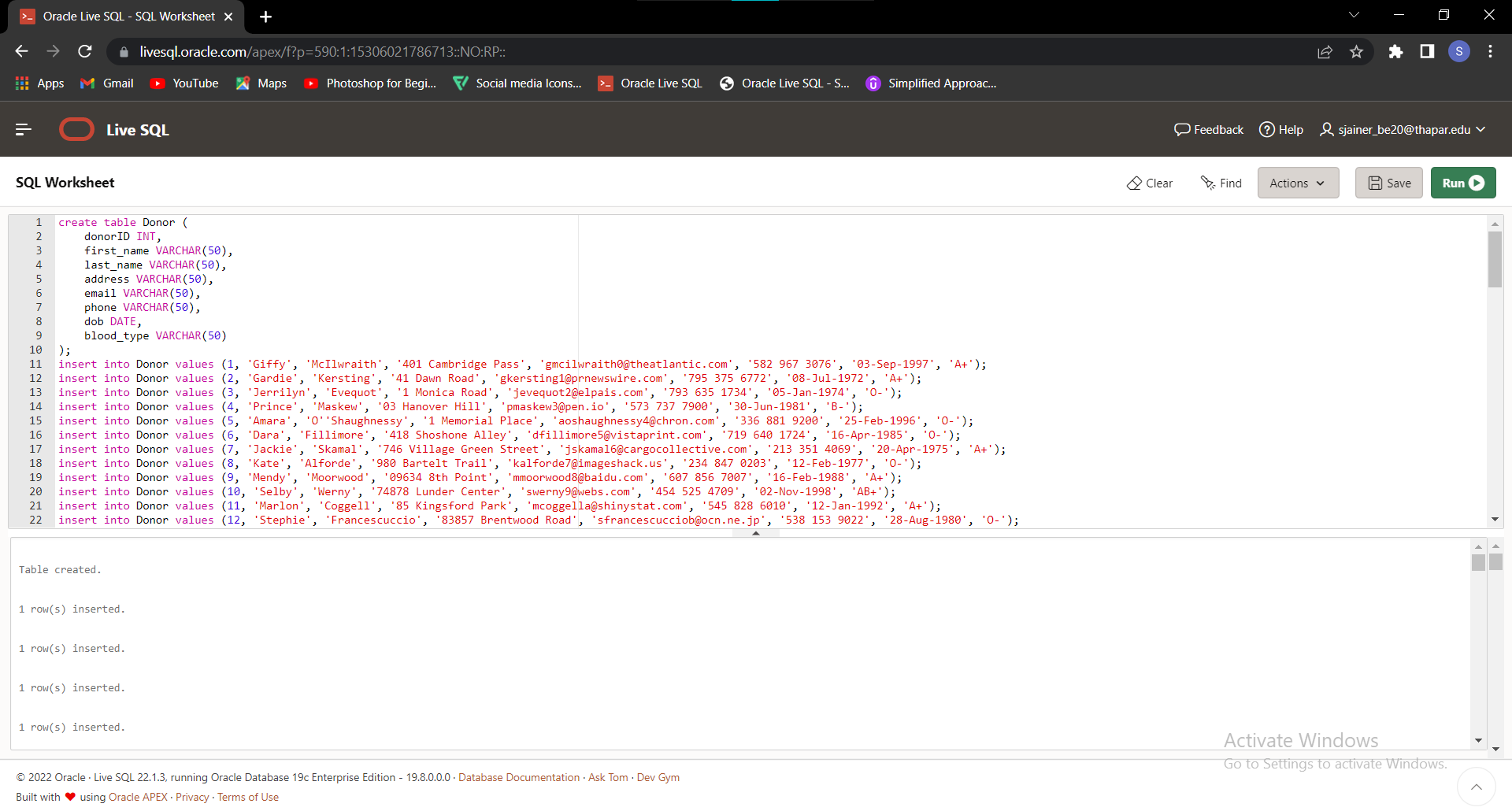
Close c1;

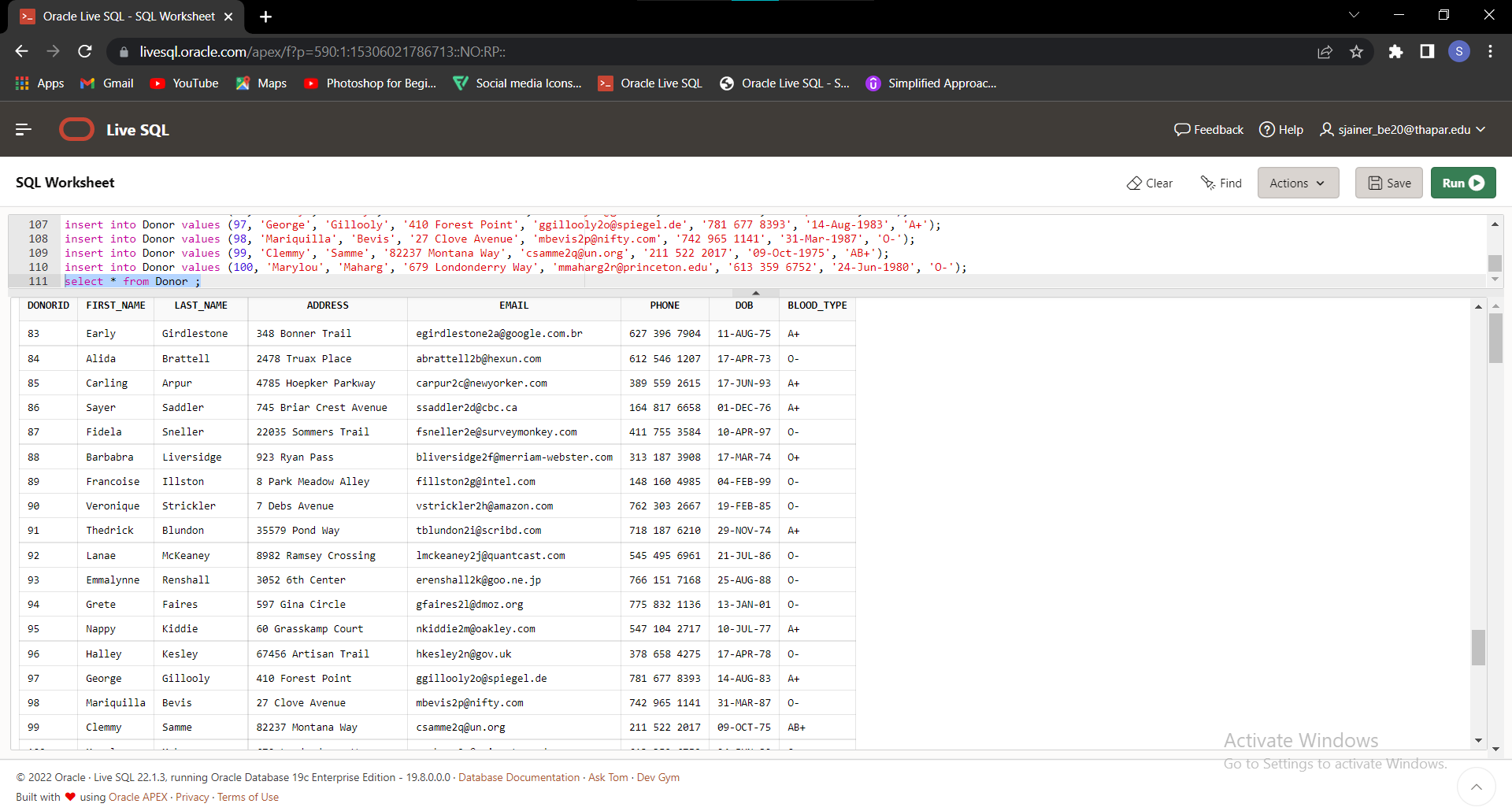
Create or replace procedure type\_blood(blood\_type) as

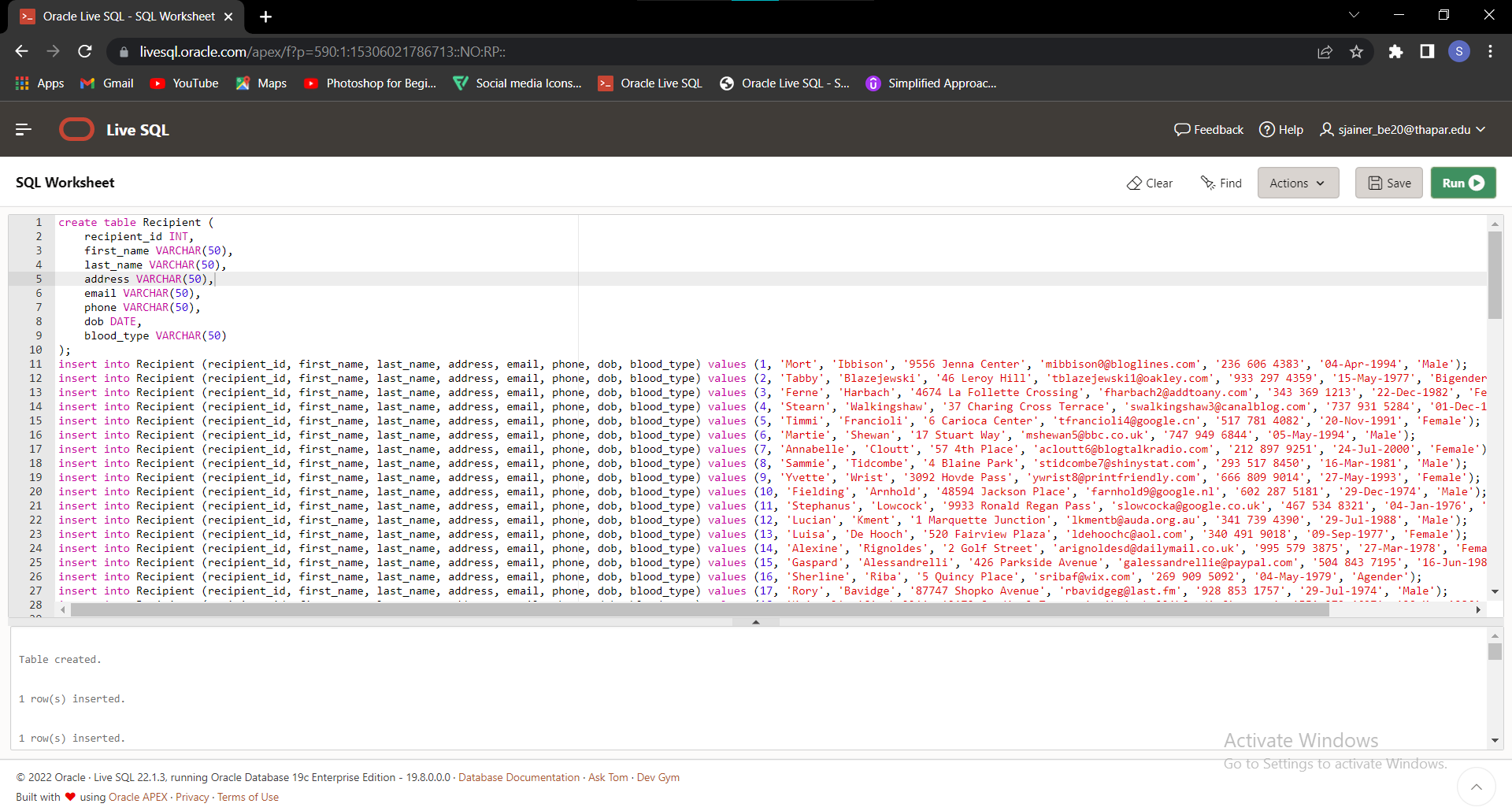
Begin

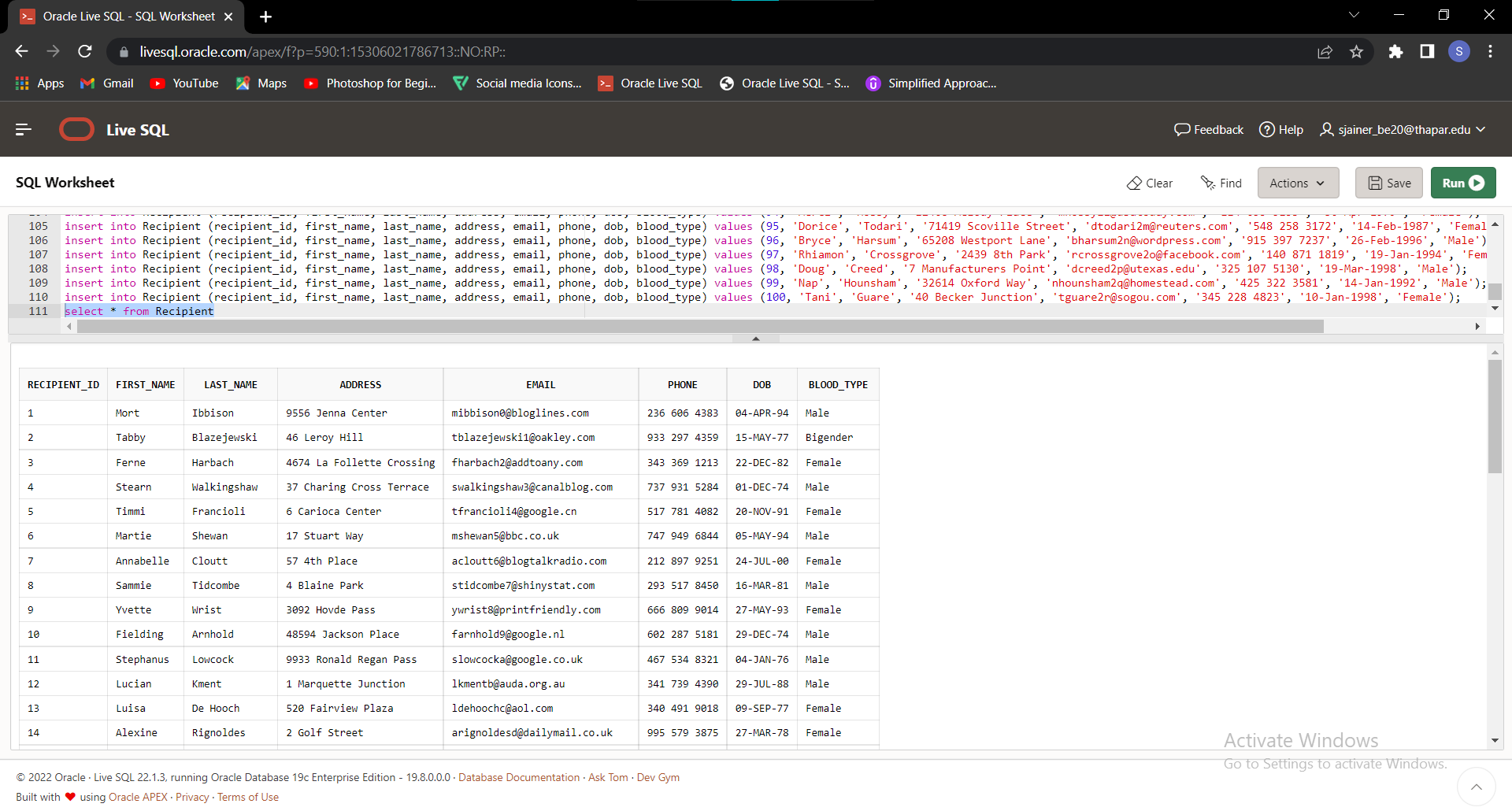
Delete from donor where blood\_type=’A+’

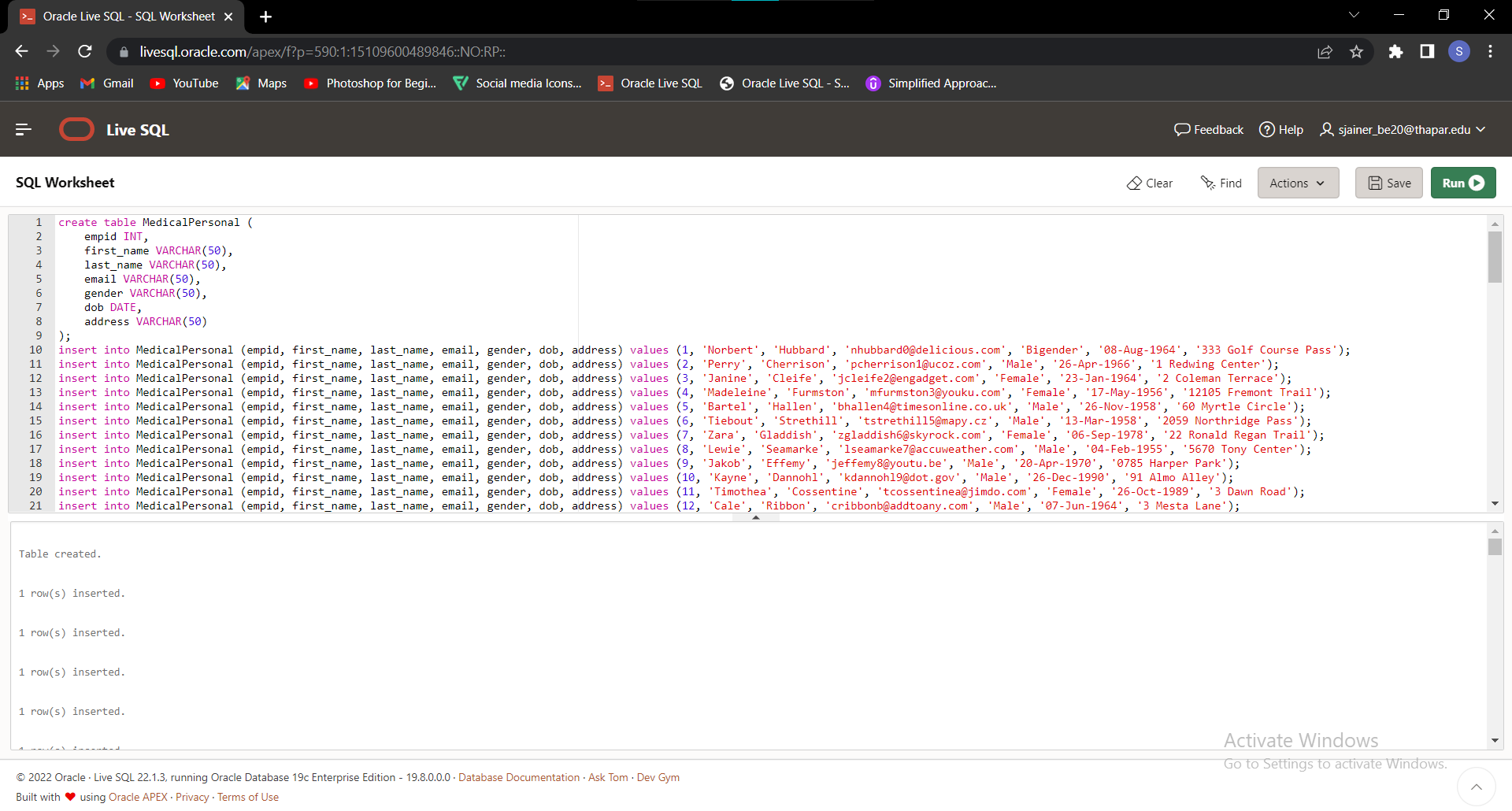
**Screenshots**

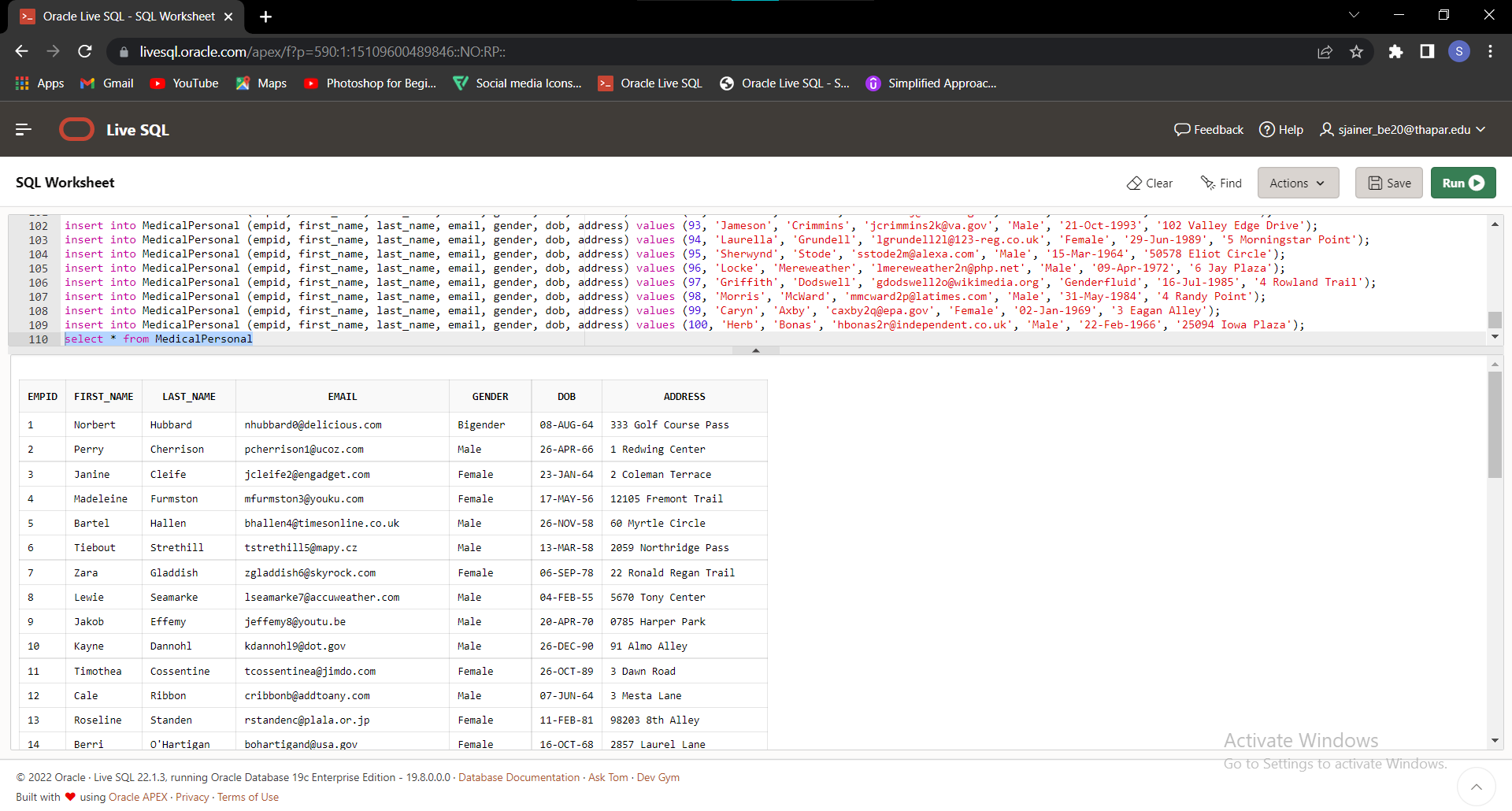
****

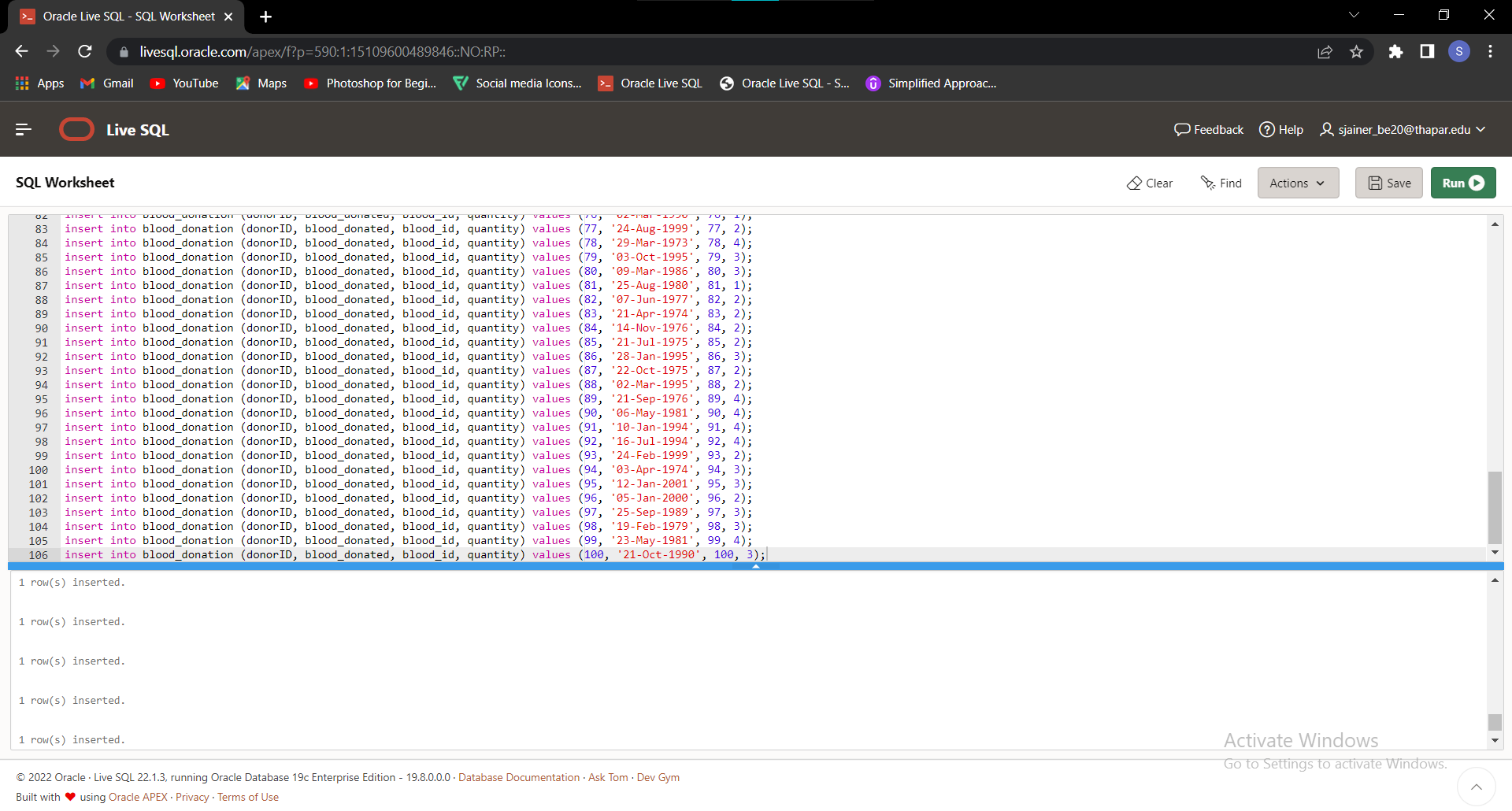
****

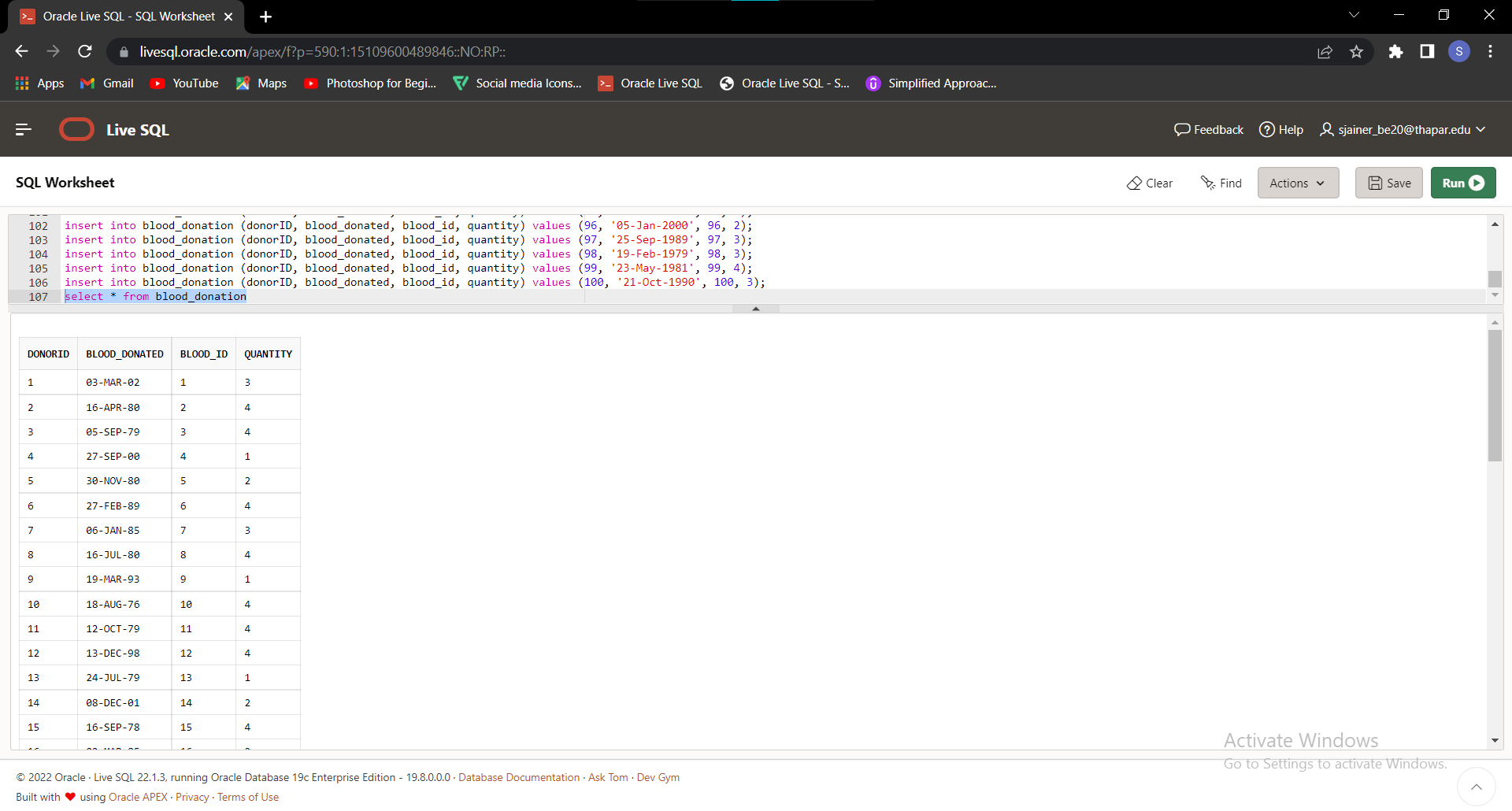
****

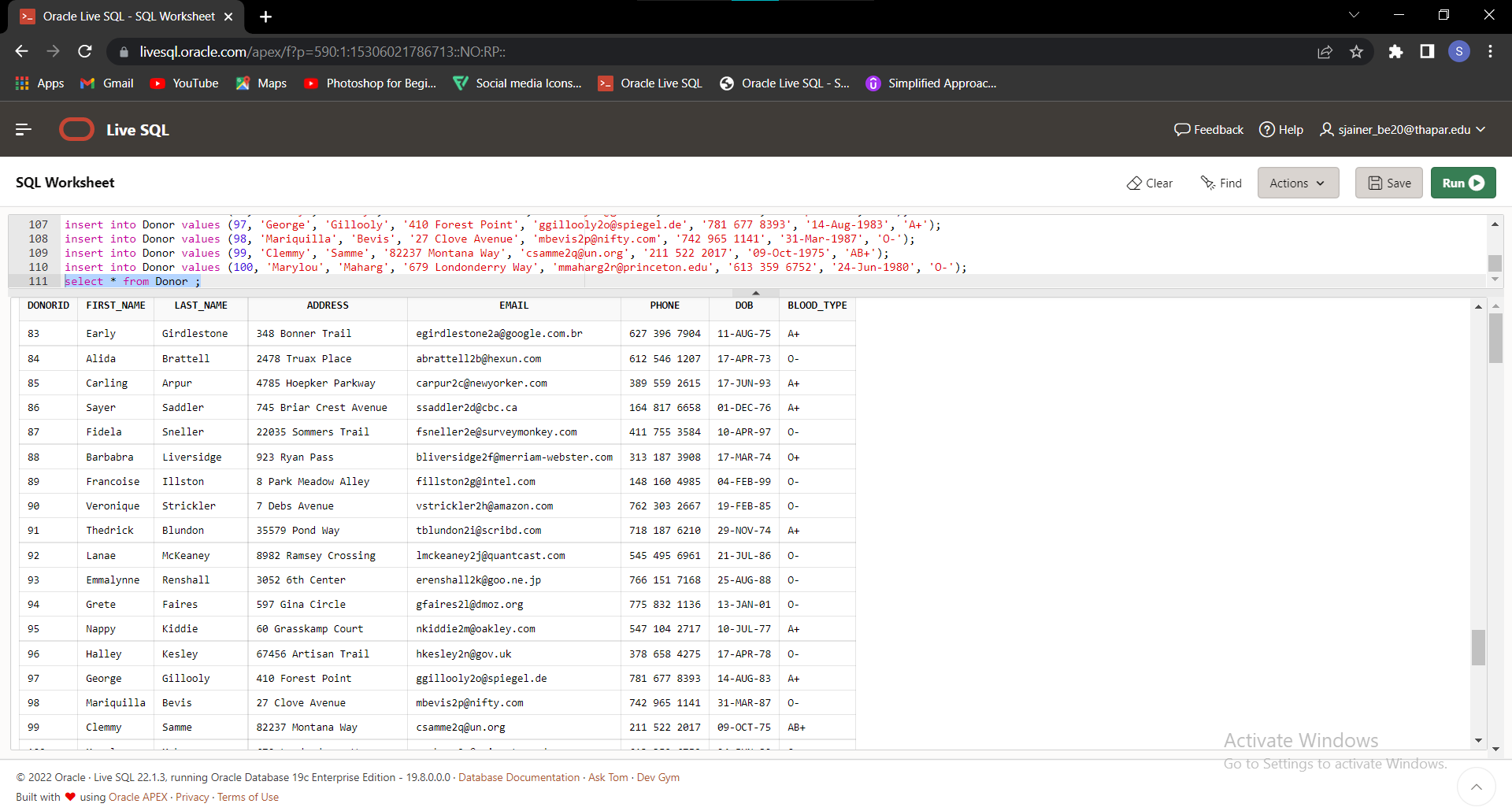
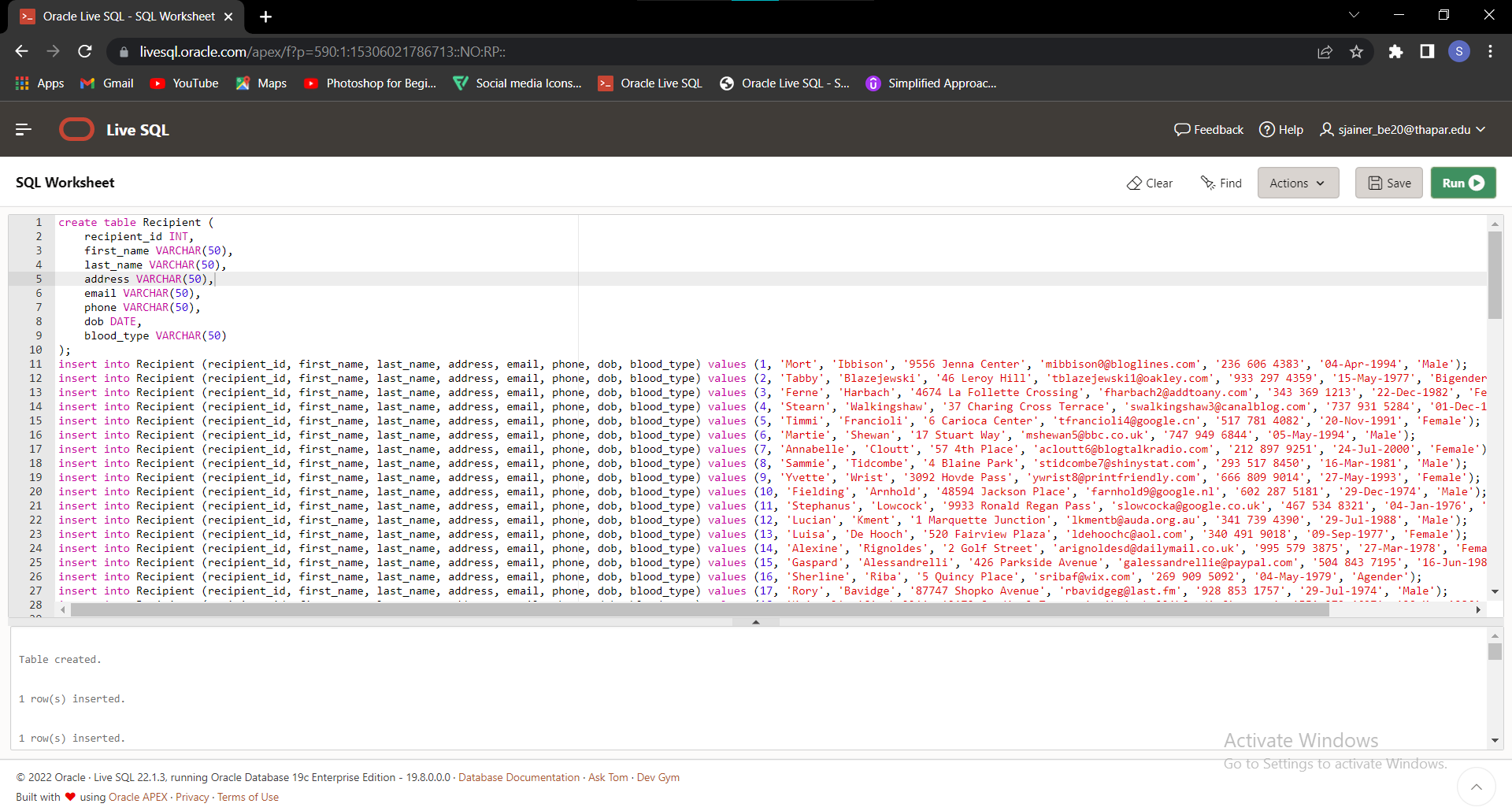
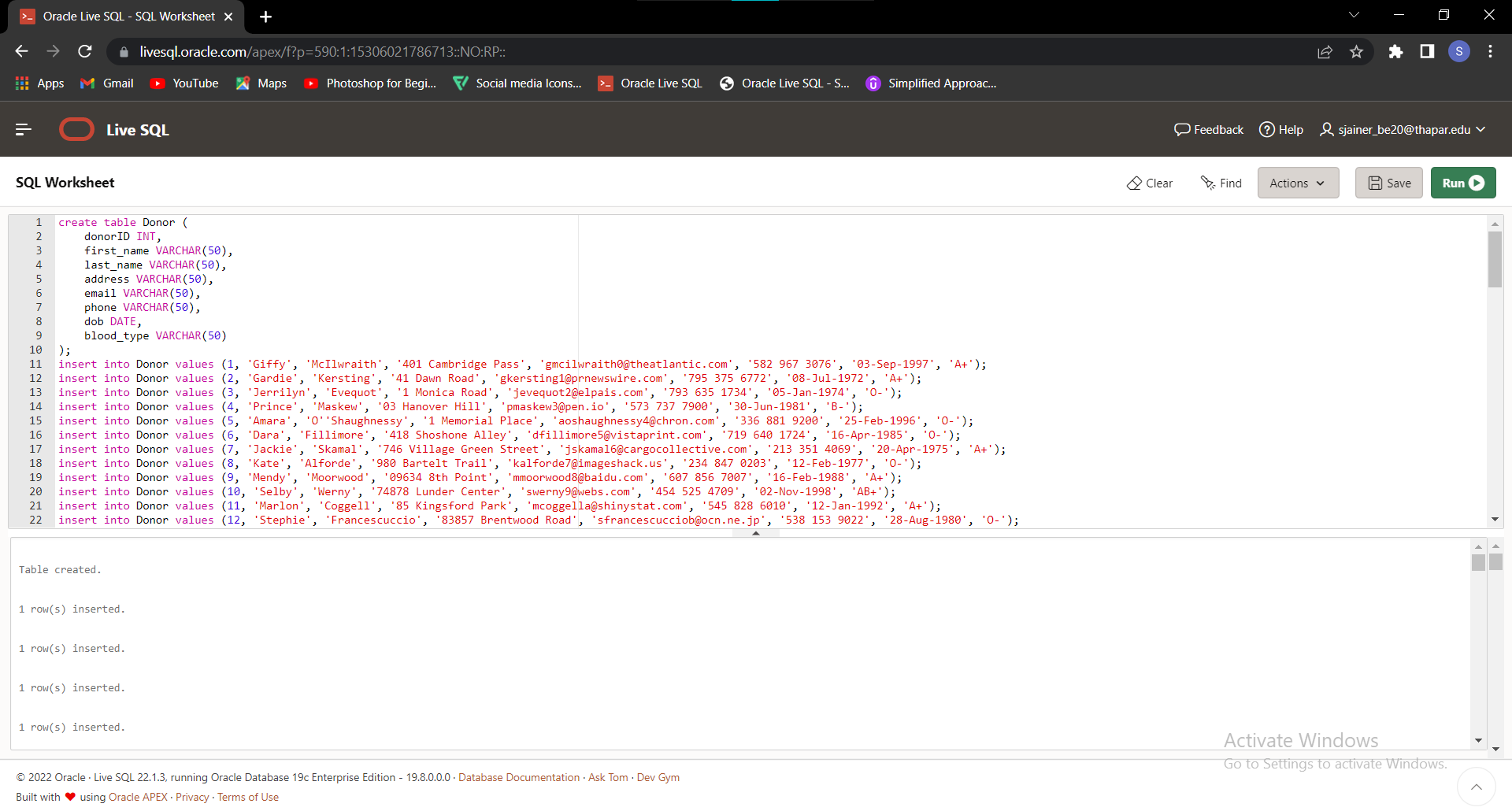
****

****

****

****

****

****

Created By:

Arhanjit Singh Sodhi

102003116