A

*Mobile banking*

*Management System*

a sql project

FALL 2022

Introduction To Database [SECTION- N]

DATE OF SUBMISSION: 26 Dec, 2022

**ProJect Title :** mobile banking management system

**Submitted by**

**GROUP: 1**

|  |  |
| --- | --- |
| **NAME** | **ID** |
| SHAFIN, TAFSIRUL ISLAM | **22-47325-1** |
| ISLAM, MD. MANSUR | **22-46550-1** |
| BUSHRA, ASIF | **22-46388-1** |
| ALAM, SHADMAN SHAKIB | **22-46262-1** |

**instructed by**

***RIFAT IBN ALAM***

###### Faculty of, Computer Science

AMERICAN INTERNATIONAL UNIVERSITY – BANGLADESH

*TABLE OF CONTENTS*

|  |  |
| --- | --- |
| * SCENARIO |  |
| * ER DIAGRAM BASED ON SCENARIO |  |
| * NORMALIZATION OF ER DIAGRAM |  |
| * TABLE CREATION WITH SCENARIO |  |
| * DATA INSERTION      * JOINING      * SUBQUERY      * VIEW |  |
| * CONSTRAINTS ADDING |  |
| * QUERY WRITING |  |

*SCENARIO*

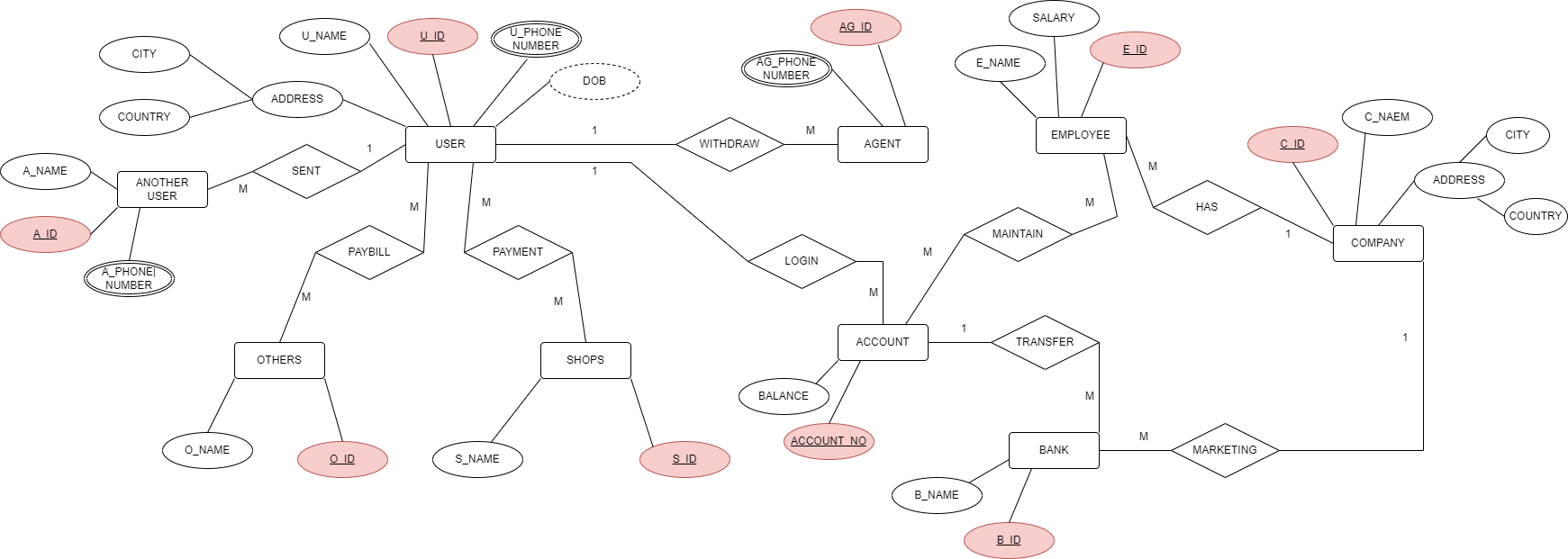
# Introduction:

Mobile banking is one of the latest tools for easy and convenient banking in the current world. Day by day mobile banking is becoming popular in Bangladesh. With proper guidelines, mobile banking can take Bangladesh to a different level of prosperity. Mobile banking is a digital financial service in the banking sector that facilitates banking via mobile devices. And this mobile banking management system’s primary benefit is that it makes the mobile banking system a lot easier. Reduced a lot of time and make life very comfortable that they can take out money or give money any time they want.

# Scenario Description:

**In a mobile banking system, user can sent money to many another users. And user have attributes like u\_id, u\_name, u\_phonenumber, dob and his or her living place like his or her city and country and another user also have attributes like a\_id, a\_name and a\_phonenumber. A user can pay multiple types of payment in various places like shops, and pay bill in various others like internet bill and electric bill so for that this others and shops has their own attributes like shops has s\_id and s\_name and o the other side other have attributes like o\_id, o\_name. A user can withdraw money from various agents and those agent have his own attributes like ag\_id and ag\_phonenumber. A user may-have multiple accounts and they can log-in to multiple accounts. And accounts has attributes like account\_no and balance. User can log-in multiple account and also this account is connected with their bank account but using this bank account they can withdraw their money from any time and any where so bank also have it’s attributes like b\_name, b\_id. User account is connected with employees this employees maintain their accounts and this employees are connected with a company and a company can have many employees. Company have attibutes like c\_id, c\_name, and address like city and country. At last company have many bank which connects with them.**

*ER DIAGRAM*



***NORMALIZATION***

**WITHDRAW:**

**This is a one to many relationship**

UNF : address, city ,country, u\_name ,u-id ,u\_phonenumber ,dob, ag\_ phone, ag \_id

1NF: city, country, u\_name, u-id ,dob, ag\_ phone, ag \_id

2NF: 1. city, country, u\_name, u-id , u\_phonenumber ,dob.

2. ag\_ phone, ag \_id , u-id (FK)

3NF: 1. city, u\_name, u-id , u\_phonenumber ,dob.

2. ag\_ phone, ag \_id , u-id (FK)

3. city , country

**MAINTAIN:**

**This is a many to many relationship**

UNF : account\_no , balance , e\_name, salary, e\_id

1NF: account\_no , balance , e\_name, salary, e\_id

2NF:

1.account\_no , balance

2. e\_name, salary, e\_id

3. account-no , e\_id(FK)

**3NF: ( same as 2nf)**

1.account\_no, balance

2. e\_id, e\_name, salary

3. account\_no, e\_id(FK)

**SENT:**

**This a one to many relationship**

UNF: u\_name, u\_phonenumber, dob, u\_id, address, city, country, a\_phonenumber, a\_id, a\_name

1NF: u\_name, dob, u\_id, city, country, a\_id, a\_name

2NF:

1. city, country, u\_name, u-id , u\_phonenumber ,dob.
2. a\_phonenumber, a\_id, a\_name, u\_id (FK)

3NF:

1. city, u\_name, u-id , u\_phonenumber ,dob.
2. A\_phonenumber, a\_id, a\_name, u\_id (FK)
3. City, country

**PAYBILL:**

**This is a many to many relationship**

UNF: u\_name,u\_phonenumber,dob,u\_id,address,city,country, O\_id, O\_name

1NF: u\_name,dob, u\_Id, city, country, O\_Id, O\_name

2NF:

1. u\_name, u\_phonenumber, dob, u\_id, city, country

2. O\_id, O\_name

3. u\_id, O\_id(FK)

3NF:(same as 2NF)

1. u\_name, u\_phonenumber, dob, u\_id, city

2. O\_id, O\_name

3. u\_id, O\_id(FK)

**PAYMENT:**

**This is a many to many relationship**

UNF: u\_name,u\_phonenumber,dob,u\_id,address,city,country, s\_id, s\_name

1NF: u\_name, dob, u\_Id, city, country, s\_Id, s\_name

2NF:

1. u\_name, u\_phonenumber, dob, u\_id, city, country

2. s\_id, s\_name

3. u\_id, \_id(FK)

3NF:(same as 2NF)

1. u\_name, u\_phonenumber, dob, u\_id, city

2. s\_id, s\_name

3. u\_id, s\_id(FK)

**LOGIN:**

**This is a one to many relationship**

UNF: u\_name,u\_phonenumber,dob,u\_id,address,city,country, account\_no, balance

1NF: u\_name,dob,u\_id,city,country, account\_no, balance

2NF:

1. U\_name,u\_phone number,dob,u\_id,city,country
2. account\_no, balance, u\_id (fk)

3NF:

1. U\_name,u\_phonenumber,dob,u\_id,city
2. account\_no, balance, u\_id (fk)
3. city, country

**TRANSFER:**

**This is a one to many relationship**

UNF: account\_no, balance , bank\_id, b\_name

1NF: account\_no, balance, bank\_id, b\_name

2NF:

1.account\_no,balance

2. bank\_id, b\_name, account\_no (FK)

3NF:

1.account\_no,balance

2. bank\_id, b\_name, account\_no (FK)

**MARKATING:**

**This is a one to many relationship**

UNF: bank\_id, b\_name, c\_name, c\_id, address, city, country

1NF: bank\_id, b\_name, c\_name, c\_id, city,country

2NF:

1.bank\_id, b\_name, c\_id(FK)

2. c\_name, c\_id, city, country

3NF:

1.bank\_id, b\_name, c\_id(FK)

2. c\_name, c\_id,city

3. city, country

**HAS:**

**This is a one to many relationship**

UNF:e\_id, e\_name, salary, c\_name, c\_id, address, city, country

1NF: e\_id, e\_name, salary, c\_name, c\_id, city, country

2NF:

1. c\_name,c\_id,city,country
2. e\_name, e\_id, salary, c\_id (FK)

3NF:

1. c\_name, c\_id, city
2. e\_name, e\_id, salary, c\_id(FK)
3. city , country.

***FINALIZATION***

**FINAL TABLE:**

1. u-id, u\_name, u\_phonenumber, city, dob.
2. ag \_id, ag\_ phonenumber, u-id (FK)
3. city, country
4. account\_no, balance
5. e\_id ,e\_name, salary,
6. account-no, e\_id(FK)
7. a\_id , a\_name ,a\_phonenumber, u\_id (FK)
8. O\_id, O\_name
9. u\_id, O\_id(FK)
10. s\_id, s\_name
11. u\_id, s\_id(FK)
12. account\_no, balance, u\_id (fk)
13. b\_id, b\_name, account\_no (FK)
14. c\_name, c\_id,city
15. b\_id, b\_name, c\_id(FK)
16. e\_name, e\_id, salary, c\_id(FK)

***LIST OF TABLES***

1. USER1.
2. AGENT.
3. COUNTRY.
4. ACCOUNT.
5. EMPLOYEE.
6. ACC\_MAINTAIN.
7. ANOTHER\_USER
8. OTHERS.
9. USER\_PAYBILL.
10. SHOPS.
11. USER\_PAYMENT.
12. USER\_LOGIN.
13. BANK.
14. COMPANY.
15. C\_BANK.
16. C\_EMP.

***TABLE CREATION***

**USER1:**

CREATE TABLE User1 (

u\_id INT PRIMARY KEY NOT NULL,

u\_name VARCHAR(20) NOT NULL,

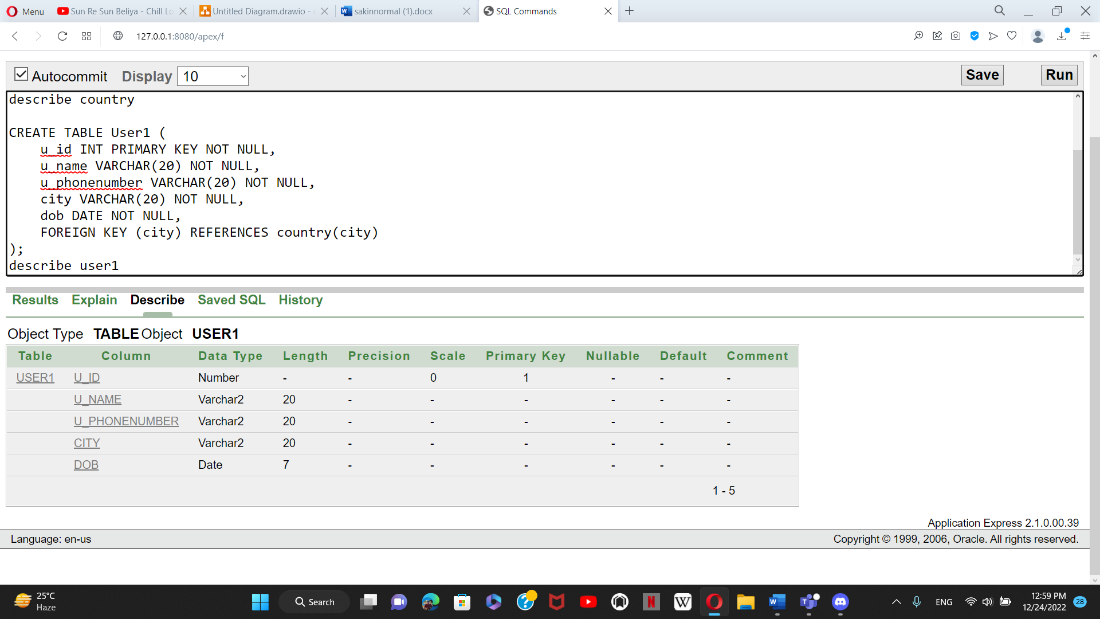
u\_phonenumber VARCHAR(20) NOT NULL,

city VARCHAR(20) NOT NULL,

dob DATE NOT NULL,

FOREIGN KEY (city) REFERENCES country(city)

);



**AGENT:**

CREATE TABLE agent (

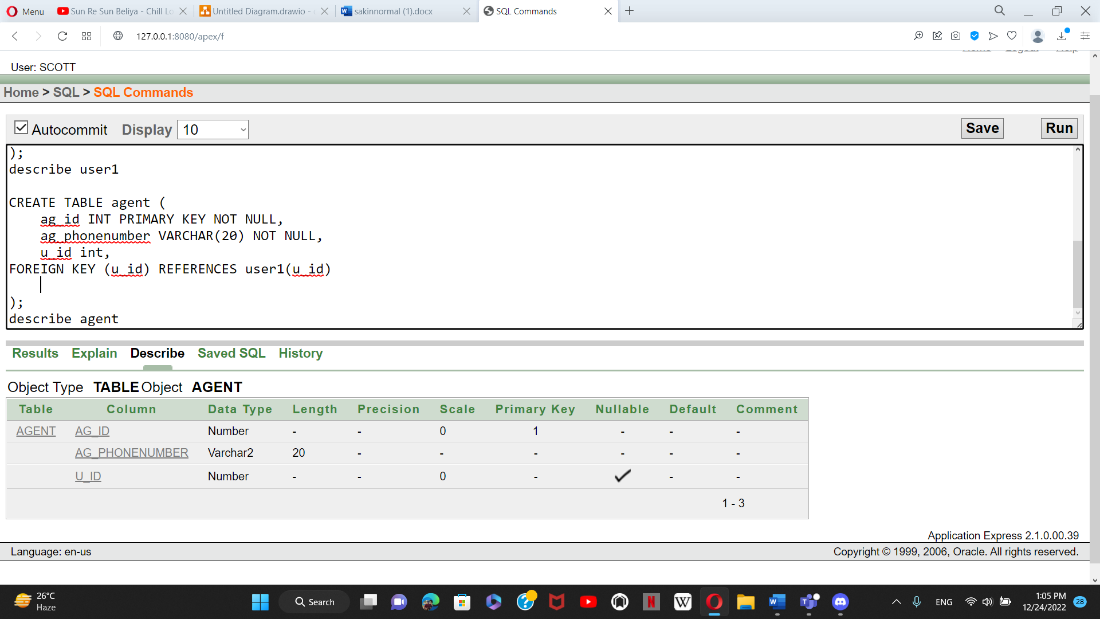
ag\_id INT PRIMARY KEY NOT NULL,

ag\_phonenumber VARCHAR(20) NOT NULL,

u\_id int,

FOREIGN KEY (u\_id) REFERENCES User1(u\_id)

);



**COUNTRY:**

CREATE TABLE country

city VARCHAR(20) PRIMARY KEY NOT NULL,

country VARCHAR(20) NOT NULL

);



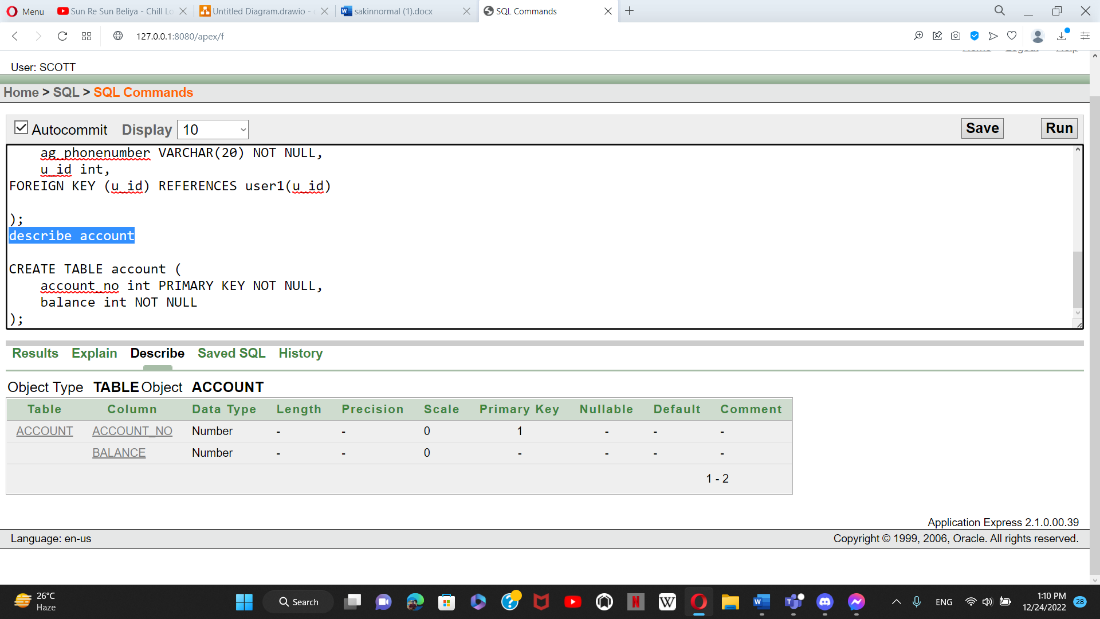
**ACCOUNT:**

CREATE TABLE account (

account\_no int PRIMARY KEY NOT NULL,

balance int NOT NULL

);



**EMPLOYEE:**

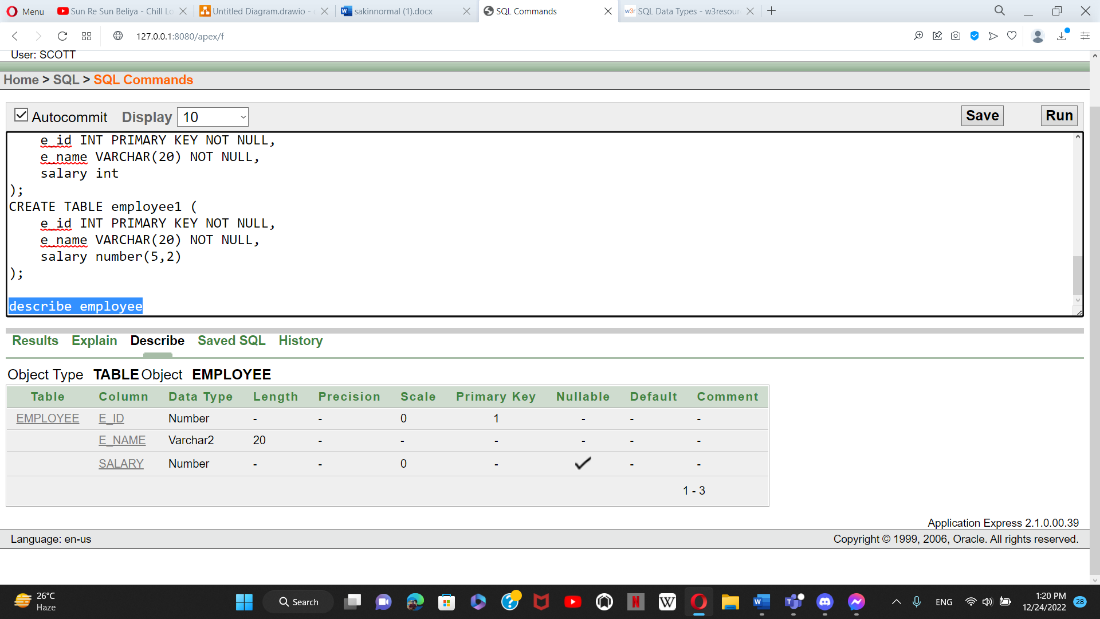
CREATE TABLE employee (

e\_id INT PRIMARY KEY NOT NULL,

e\_name VARCHAR(20) NOT NULL,

salary int

);



**ACC\_MIANTAIN:**

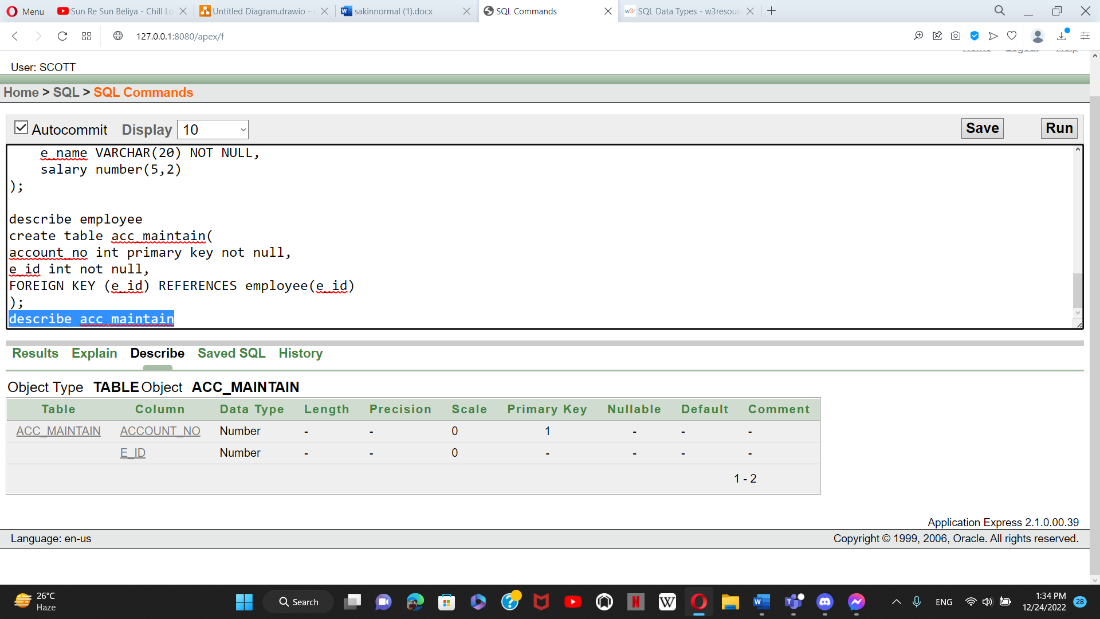
create table acc\_maintain(

account\_no int primary key not null,

e\_id int not null,

FOREIGN KEY (e\_id) REFERENCES employee(e\_id)

);



**ANOTHER\_USER:**

create table another\_user(

a\_id int primary key not null,

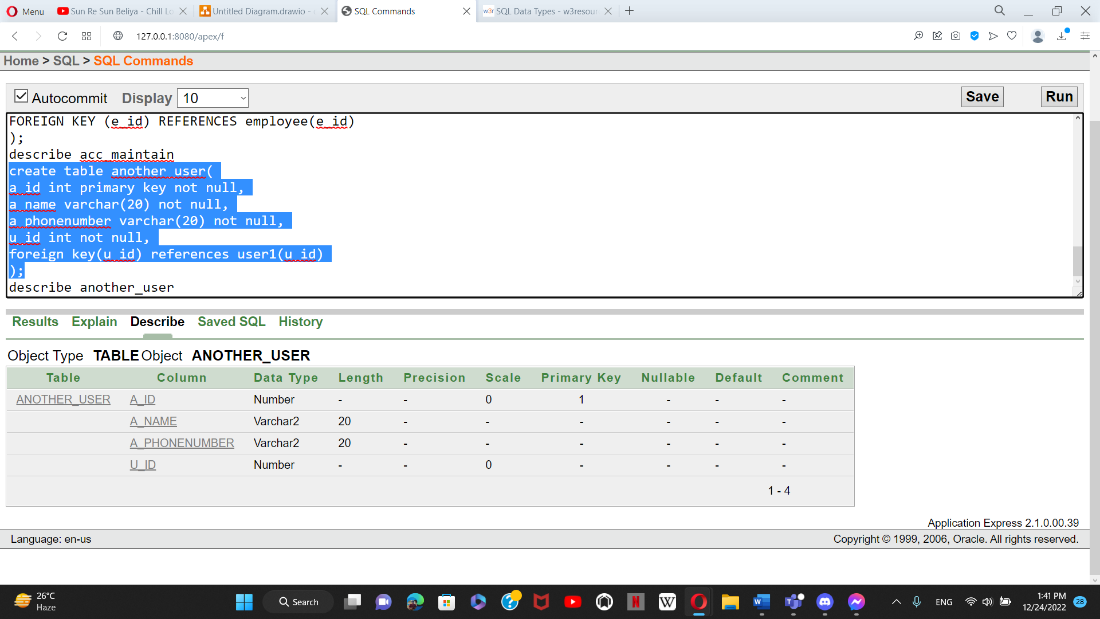
a\_name varchar(20) not null,

a\_phonenumber varchar(20) not null,

u\_id int not null,

foreign key(u\_id) references user1(u\_id)

);



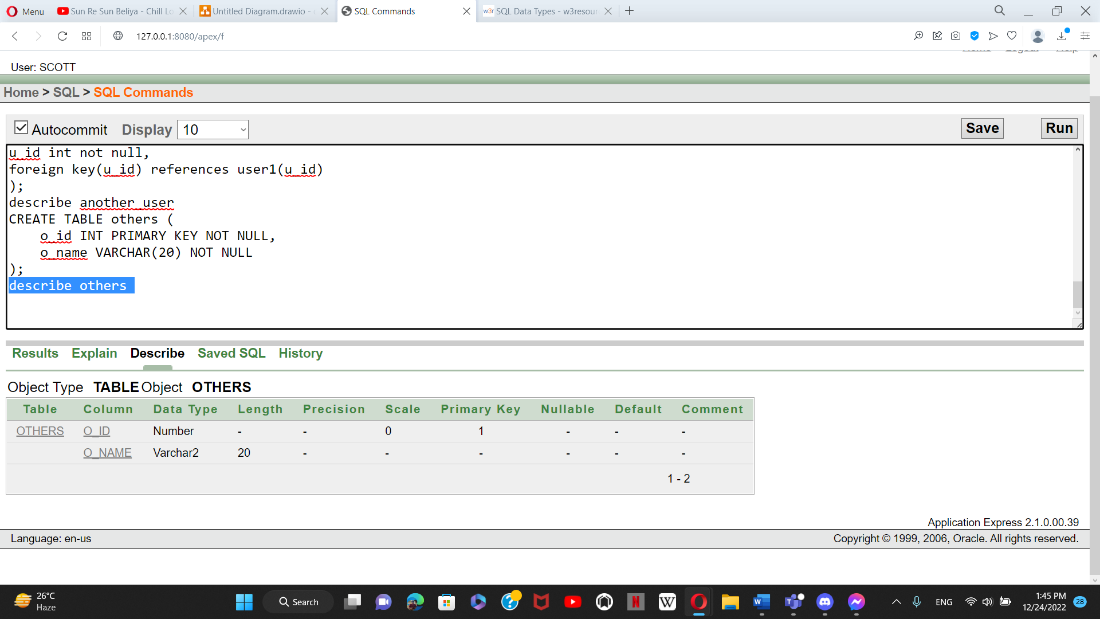
**OTHERS:**

CREATE TABLE others (

o\_id INT PRIMARY KEY NOT NULL,

o\_name VARCHAR(20) NOT NULL

);



**USER\_PAYBILL:**

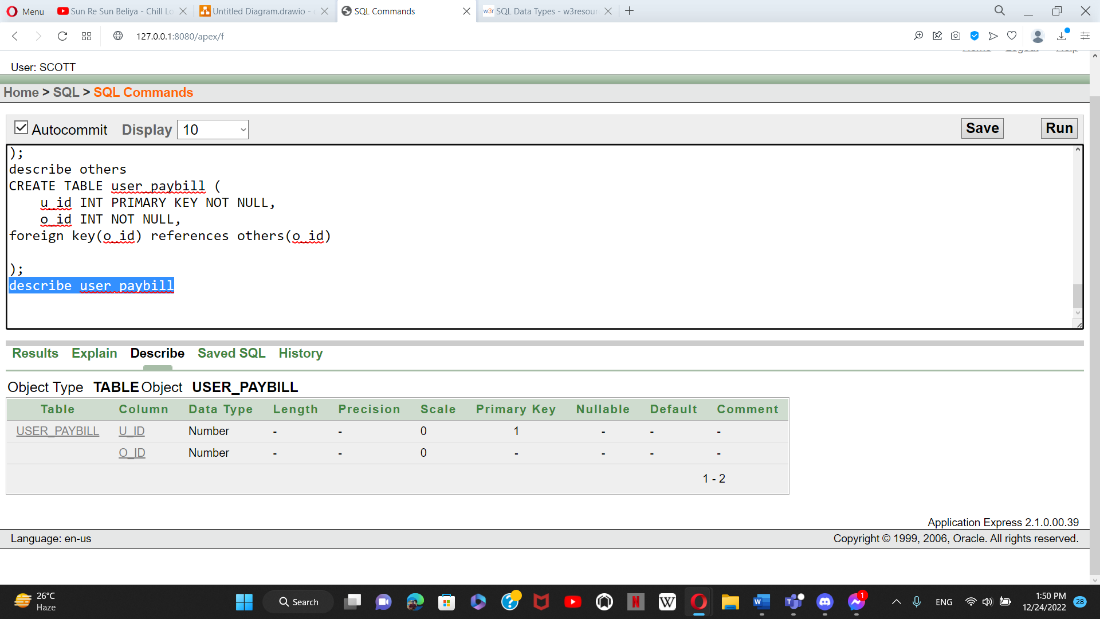
CREATE TABLE user\_paybill (

u\_id INT PRIMARY KEY NOT NULL,

o\_id INT NOT NULL,

foreign key(o\_id) references others(o\_id)

);



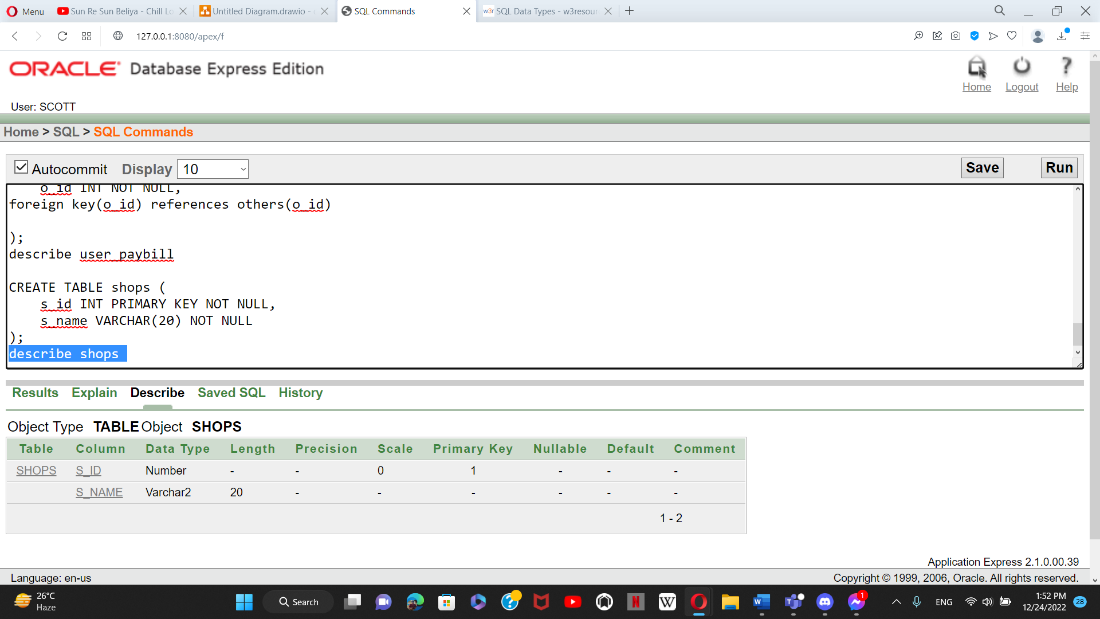
**SHOPS:**

CREATE TABLE shops (

s\_id INT PRIMARY KEY NOT NULL,

s\_name VARCHAR(20) NOT NULL

);



**USER\_PAYMENT:**

CREATE TABLE user\_payment (

u\_id INT PRIMARY KEY NOT NULL,

s\_id INT NOT NULL,

foreign key(s\_id) references shops(s\_id)

);



**USER\_LOGIN:**

create table user\_login(

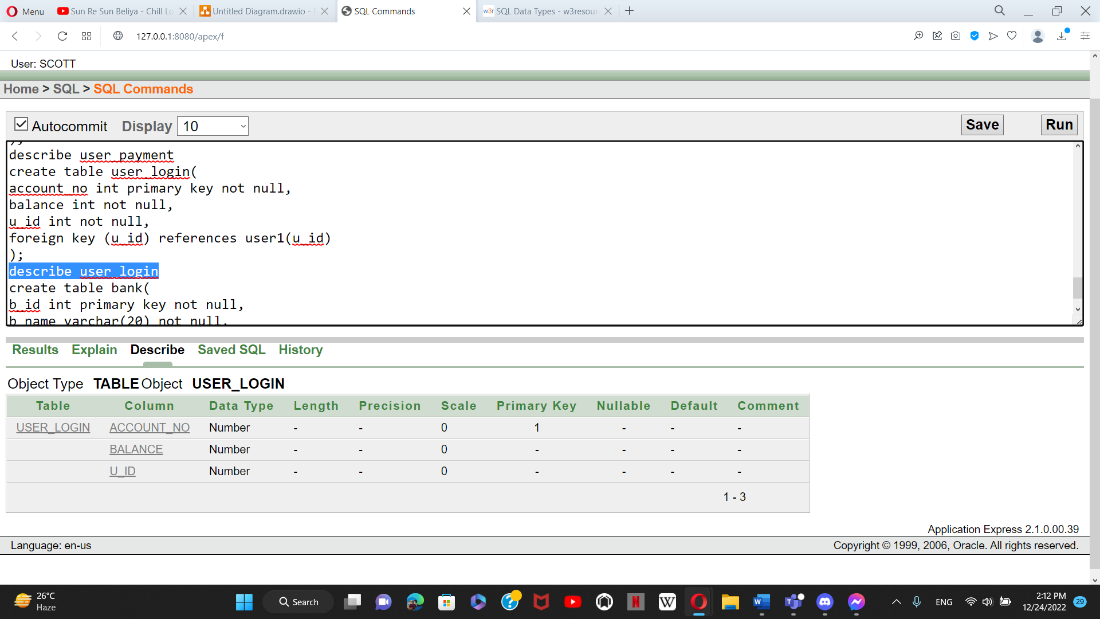
account\_no int primary key not null,

balance int not null,

u\_id int not null,

foreign key (u\_id) references user1(u\_id)

);



**BANK:**

create table bank(

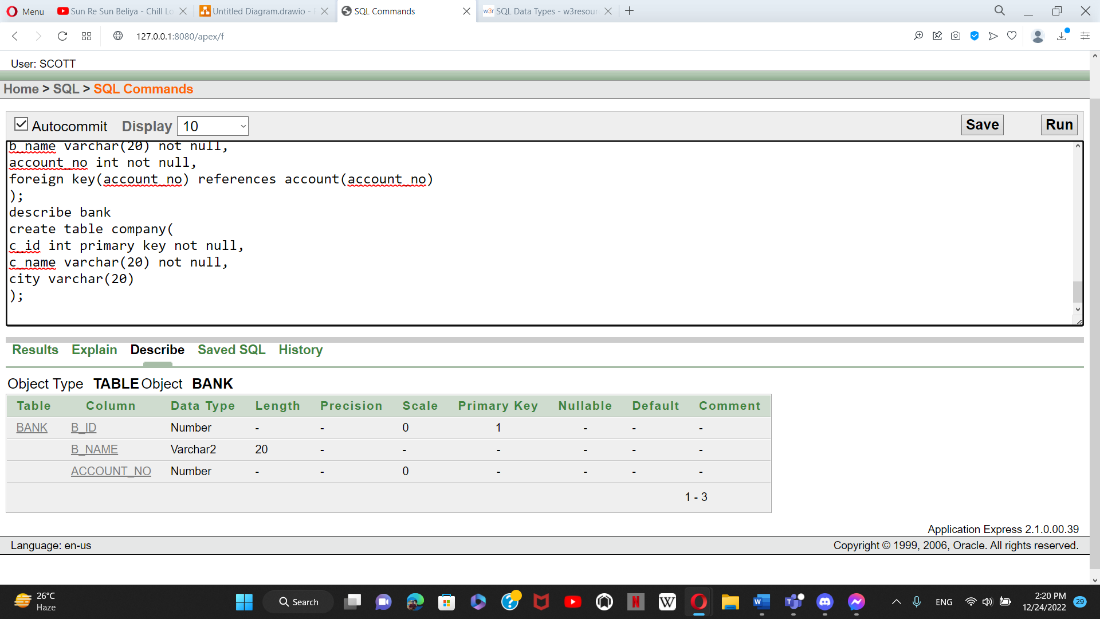
b\_id int primary key not null,

b\_name varchar(20) not null,

account\_no int not null,

foreign key(account\_no) references account(account\_no)

);



**COMPANY:**

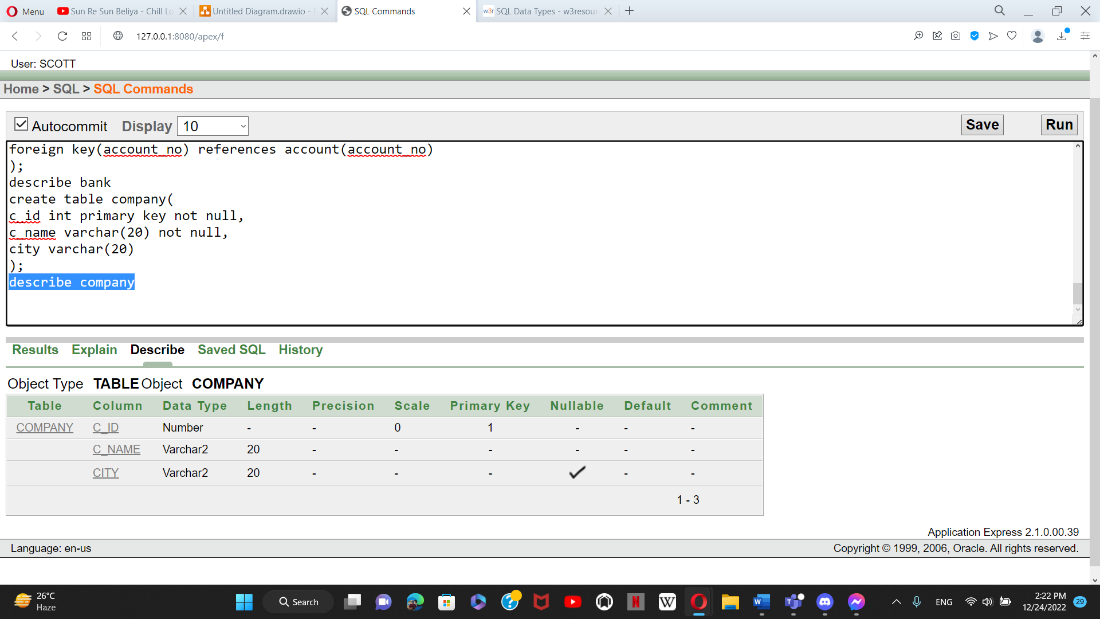
create table company(

c\_id int primary key not null,

c\_name varchar(20) not null,

city varchar(20)

);



**C\_BANK:**

create table c\_bank(

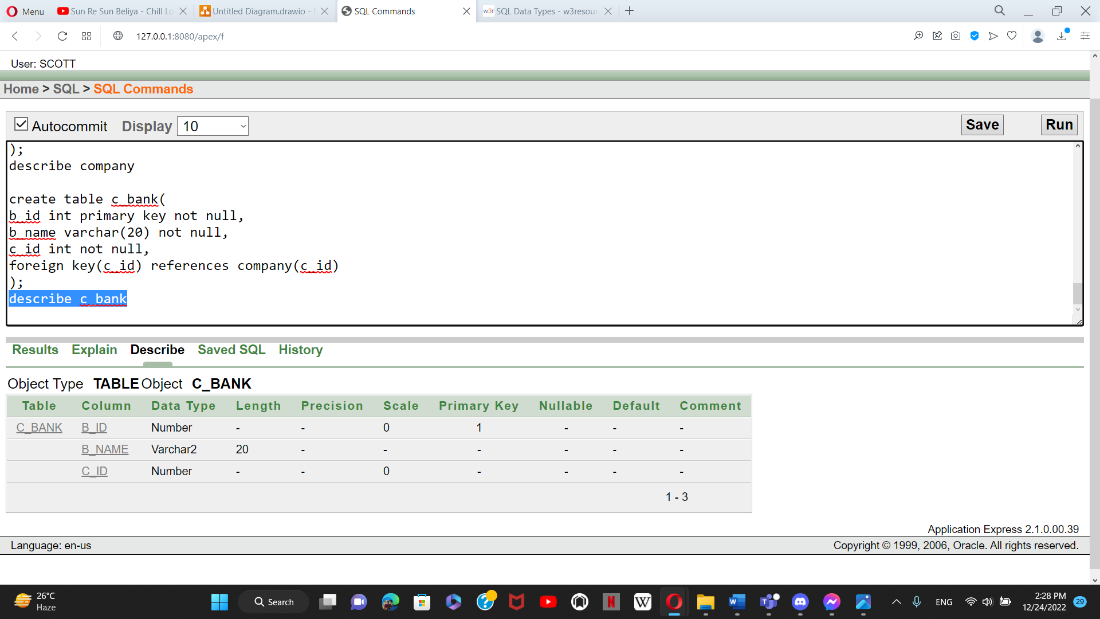
b\_id int primary key not null,

b\_name varchar(20) not null,

c\_id int not null,

foreign key(c\_id) references company(c\_id)

);



**C\_EMP:**

CREATE TABLE c\_emp (

e\_id INT PRIMARY KEY NOT NULL,

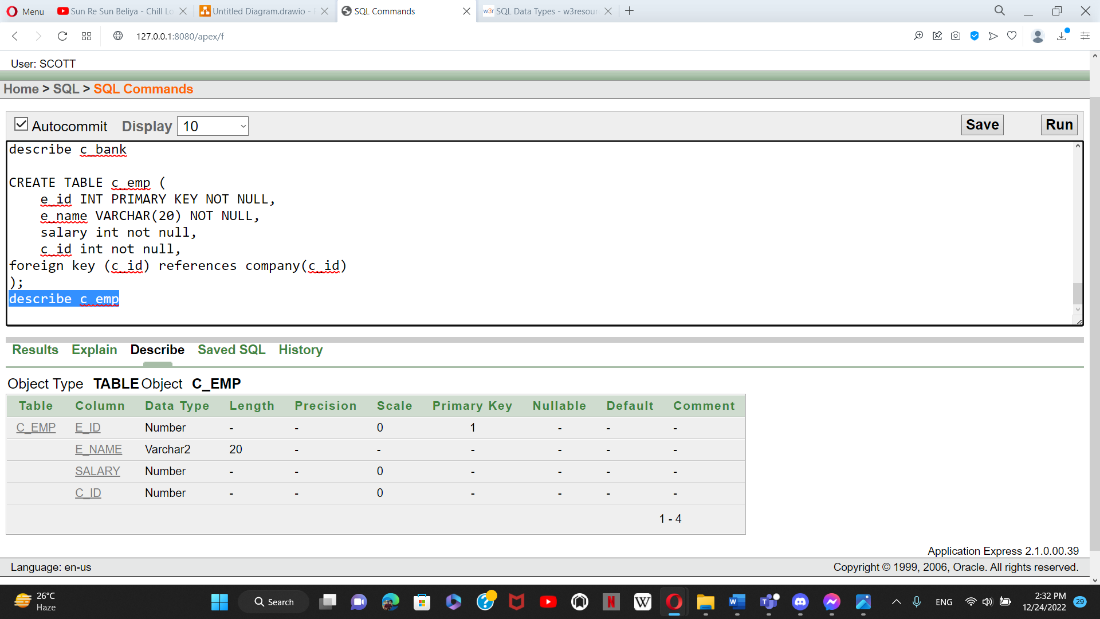
e\_name VARCHAR(20) NOT NULL,

salary int not null,

c\_id int not null,

foreign key (c\_id) references company(c\_id)

);



**VALUE INSERTION**

**COUNTRY TABLE INSERTION:**

INSERT INTO country VALUES ('Dhaka', 'Bangladesh');

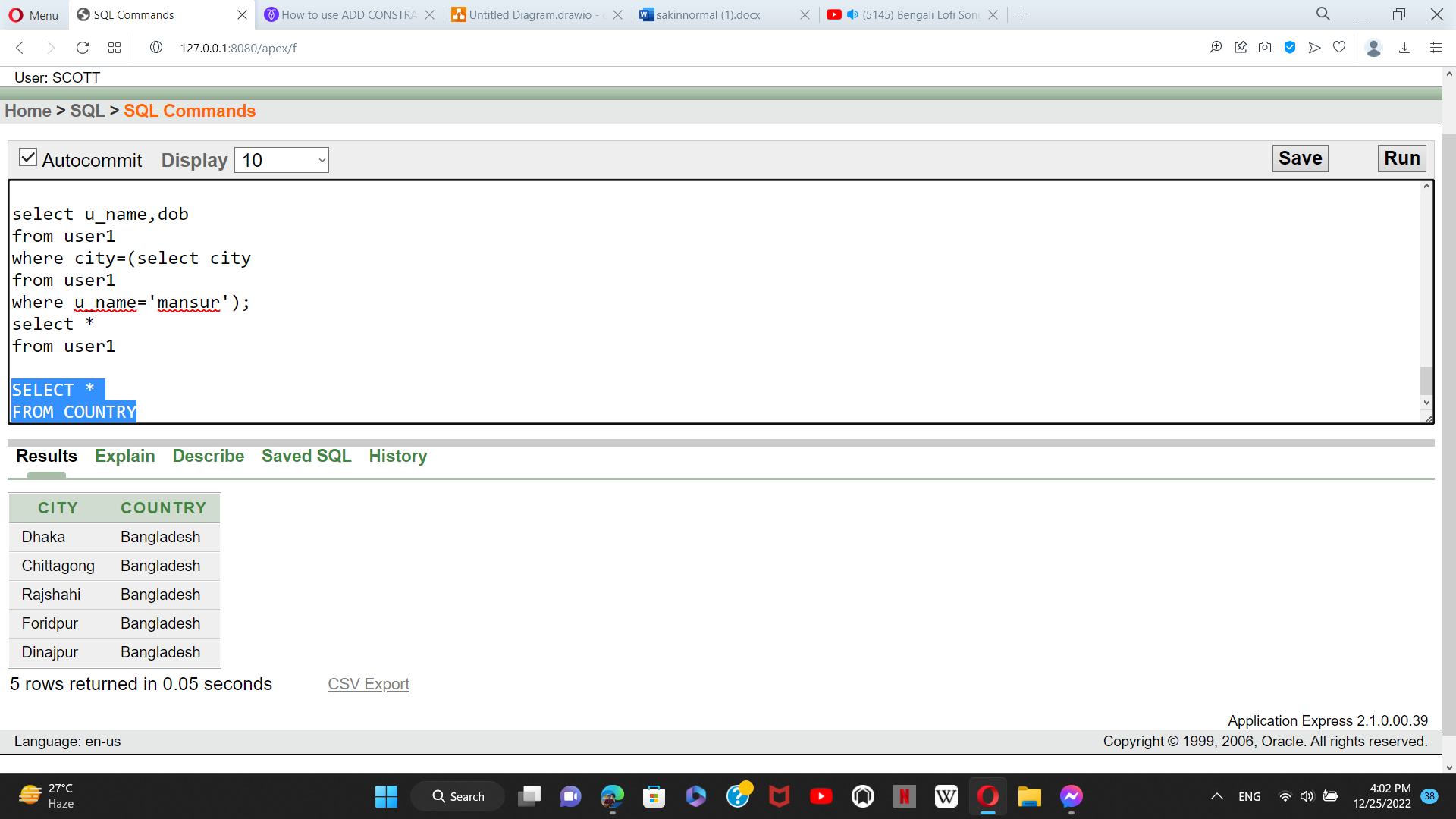
INSERT INTO country VALUES ('Chittagong', 'Bangladesh');

INSERT INTO country VALUES ('Rajshahi', 'Bangladesh');

INSERT INTO country VALUES ('Foridpur','Bangladesh');

INSERT INTO country VALUES ('Dinajpur','Bangladesh');

**TABLE:**



**USER1 TABLE INSERTION:**

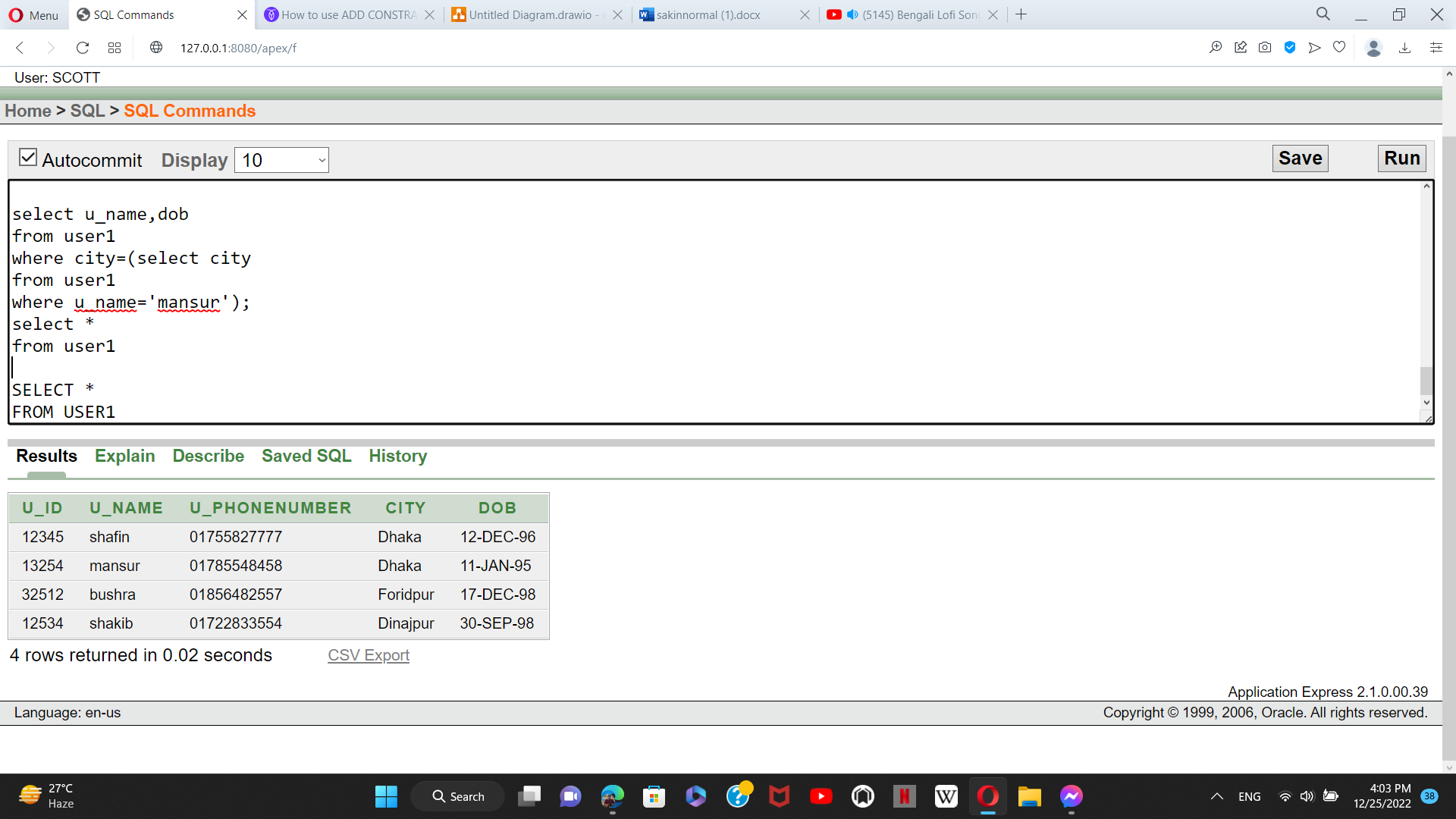
INSERT INTO User1 VALUES (12345,'shafin','01755827777','Dhaka','12-DEC-1996');

INSERT INTO User1 VALUES (13254, 'mansur','01785548458','Dhaka','11-JAN-1995');

INSERT INTO User1 VALUES (32512,'bushra','01856482557','Foridpur', '17-DEC-1998');

INSERT INTO User1 VALUES (12534,'shakib','01722833554','Dinajpur', '30-SEP-1998');

**TABLE:**



**AGENT TABLE INSERTION:**

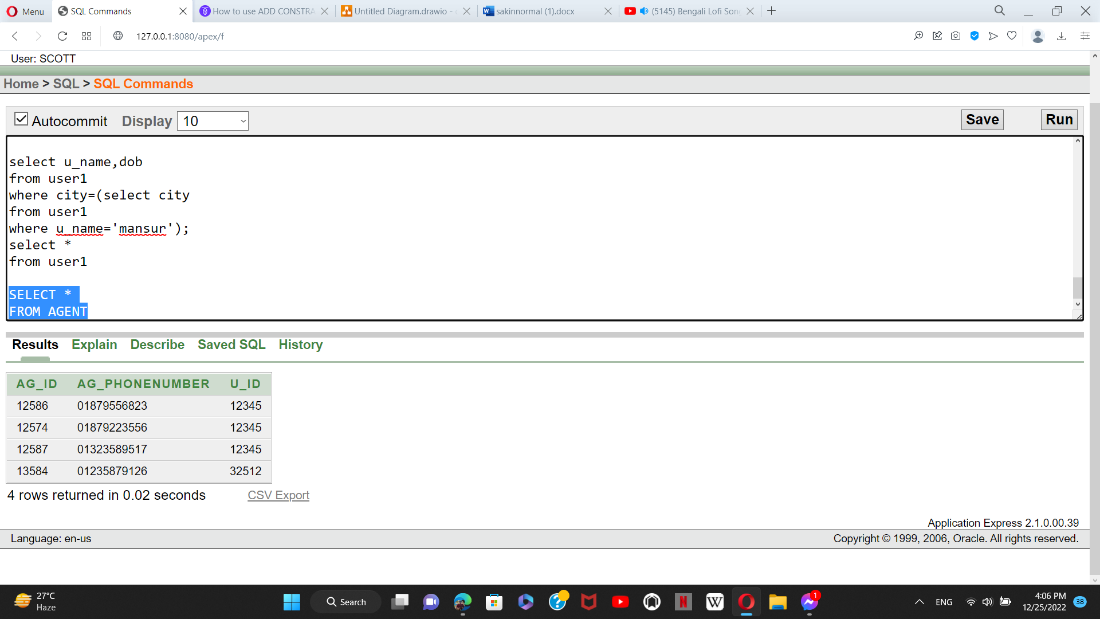
INSERT INTO agent VALUES (12586,'01879556823',12345);

INSERT INTO agent VALUES (12574,'01879223556',12345);

INSERT INTO agent VALUES (12587,'01323589517',12345);

INSERT INTO agent VALUES (13584,'01235879126',32512);

**TABLE:**



**ACCOUNT TABLE INSERTION:**

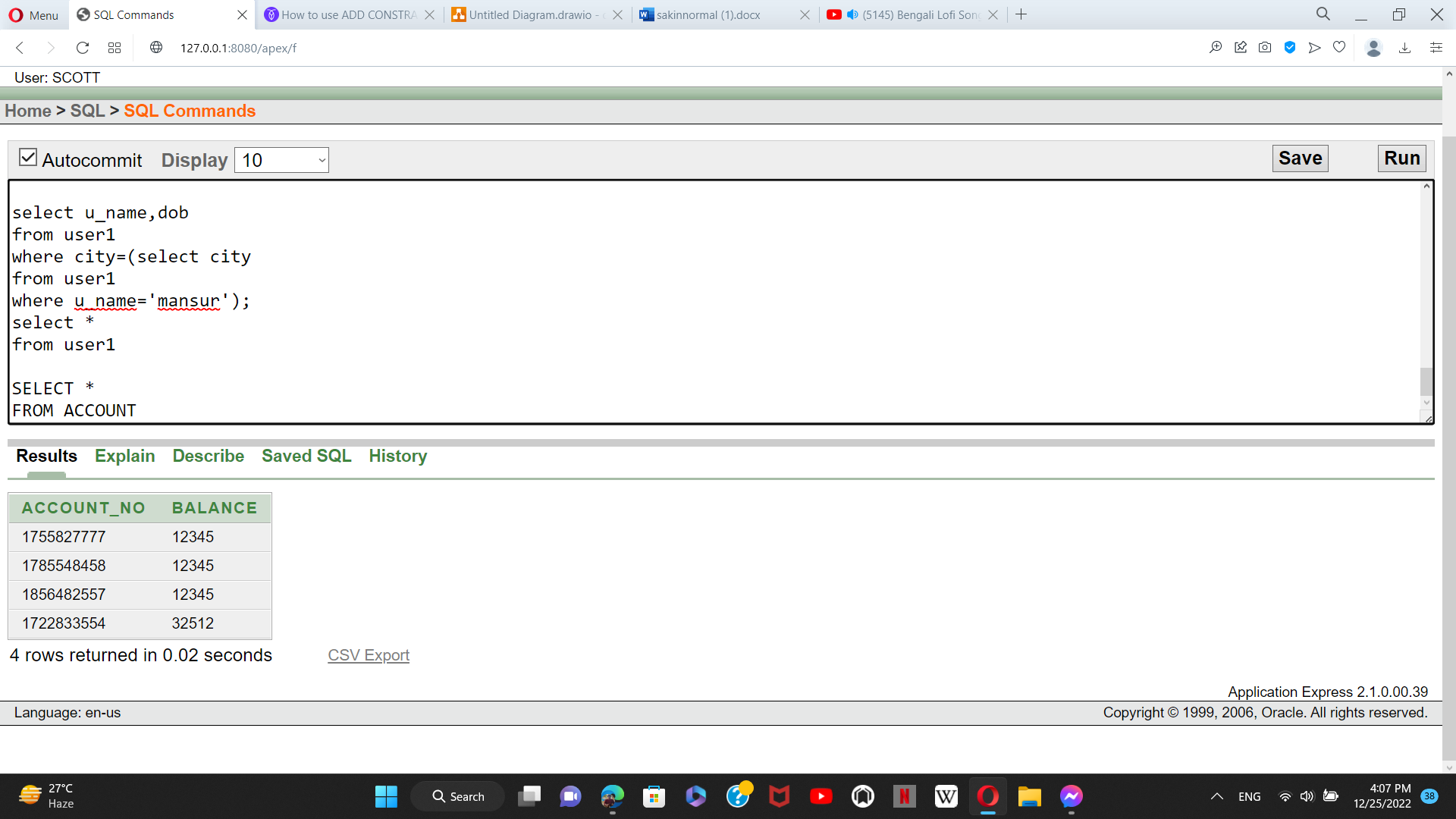
INSERT INTO account VALUES (01755827777,12345);

INSERT INTO account VALUES (01785548458,12345);

INSERT INTO account VALUES (01856482557,12345);

INSERT INTO account VALUES (01722833554,32512);

**TABLE:**



**EMPLOYEE TABLE INSERTION:**

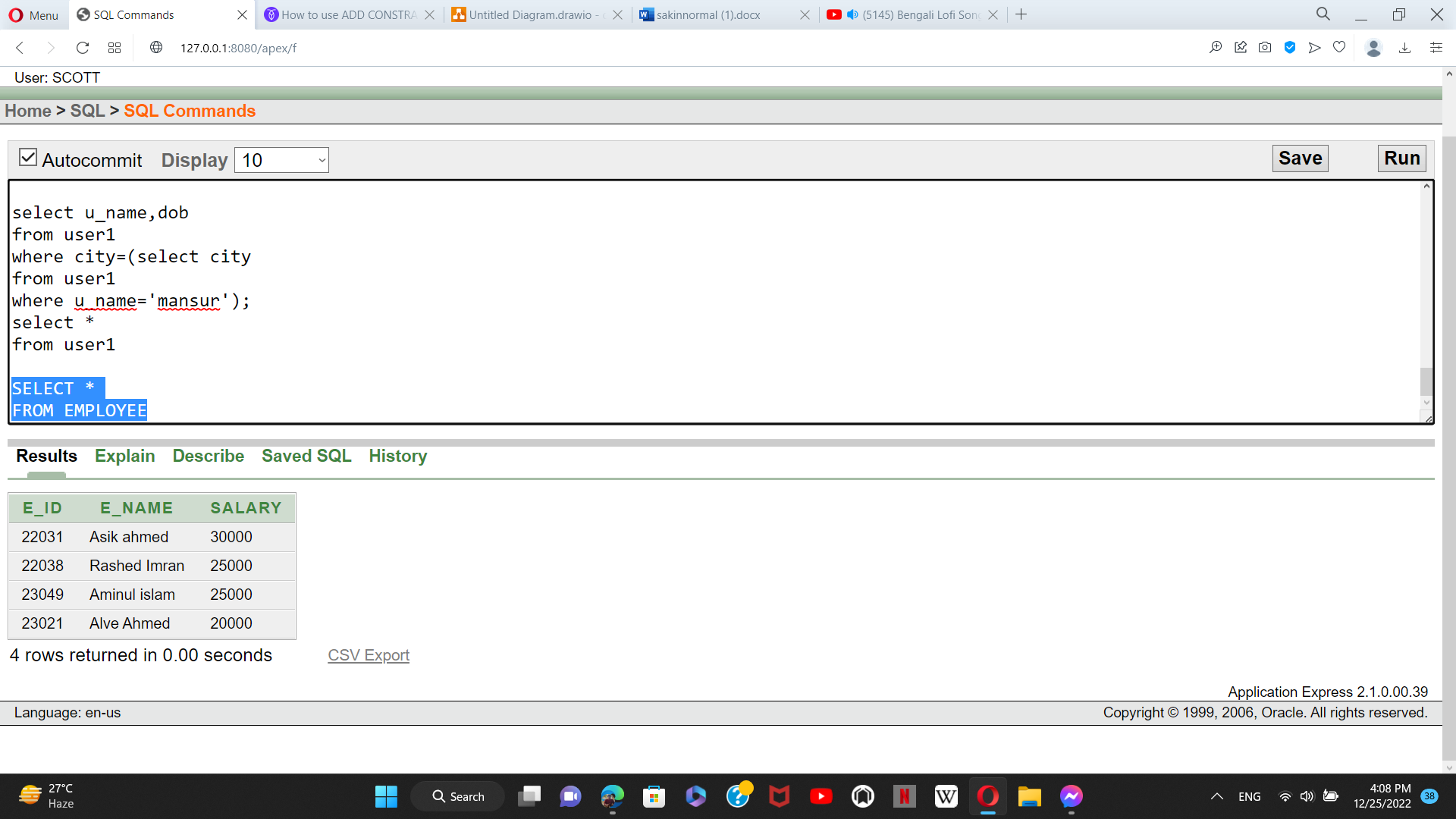
insert into employee values(22031,'Asik ahmed',30000);

insert into employee values(22038,'Rashed Imran',25000);

insert into employee values(23049,'Aminul islam',25000);

insert into employee values(23021,'Alve Ahmed',20000);

**TABLE:**



**ACC\_MAINTAIN TABLE INSERTION:**

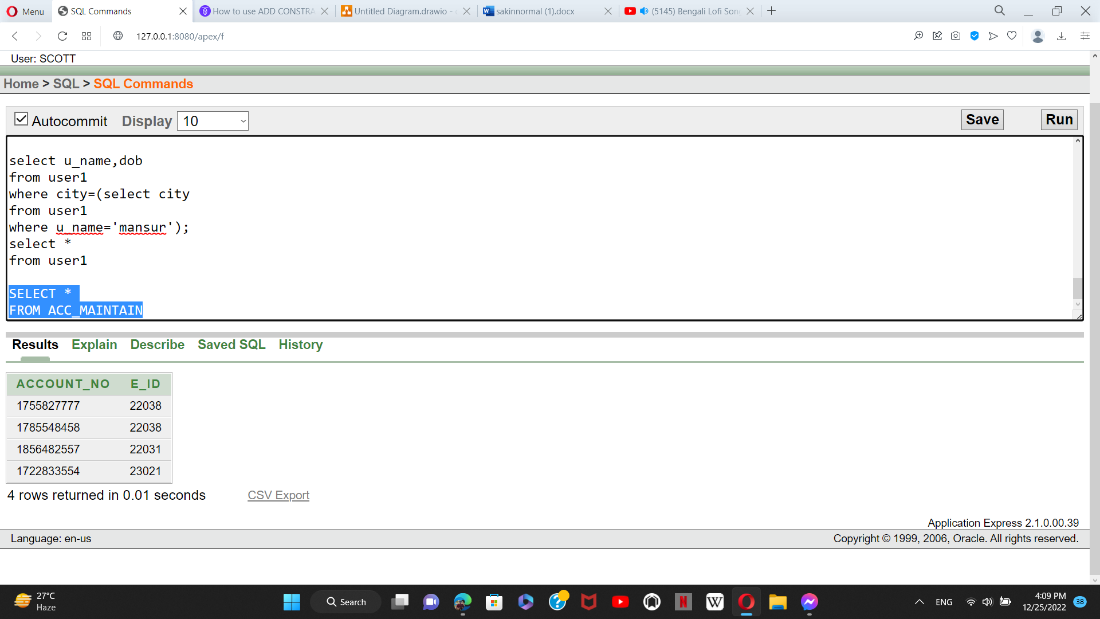
INSERT INTO acc\_maintain VALUES (01755827777,22038);

INSERT INTO acc\_maintain VALUES (01785548458,22038);

INSERT INTO acc\_maintain VALUES (01856482557,22031);

INSERT INTO acc\_maintain VALUES (01722833554,23021);

**TABLE:**



**ANOTHER\_USER TABLE INSERTION:**

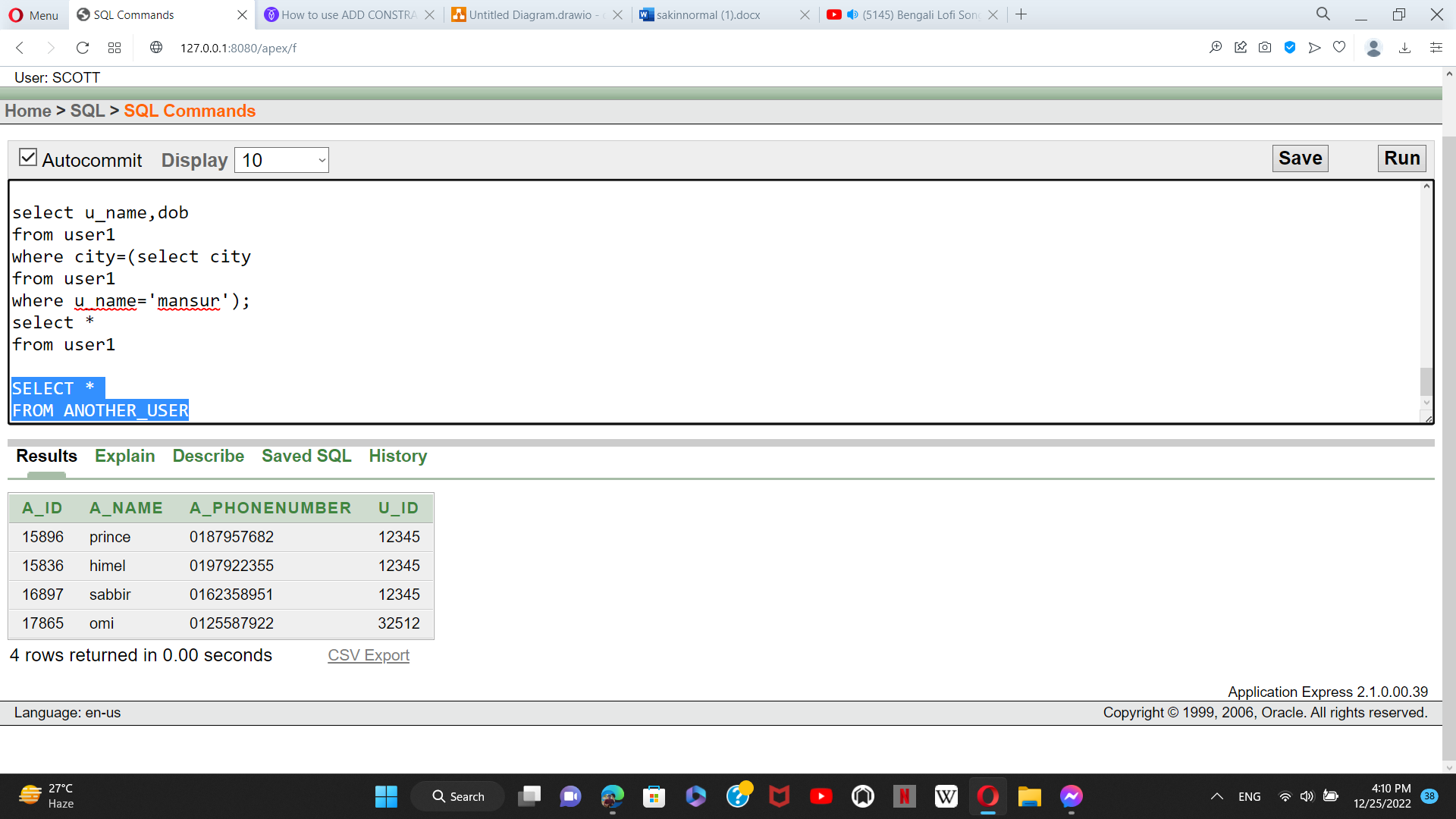
INSERT INTO another\_user VALUES (15896,'prince','0187957682',12345);

INSERT INTO another\_user VALUES (15836,'himel','0197922355',12345);

INSERT INTO another\_user VALUES (16897,'sabbir','0162358951',12345);

INSERT INTO another\_user VALUES (17865,'omi','0125587922',32512);

**TABLE:**



**OTHERS TABLE INSERTION:**

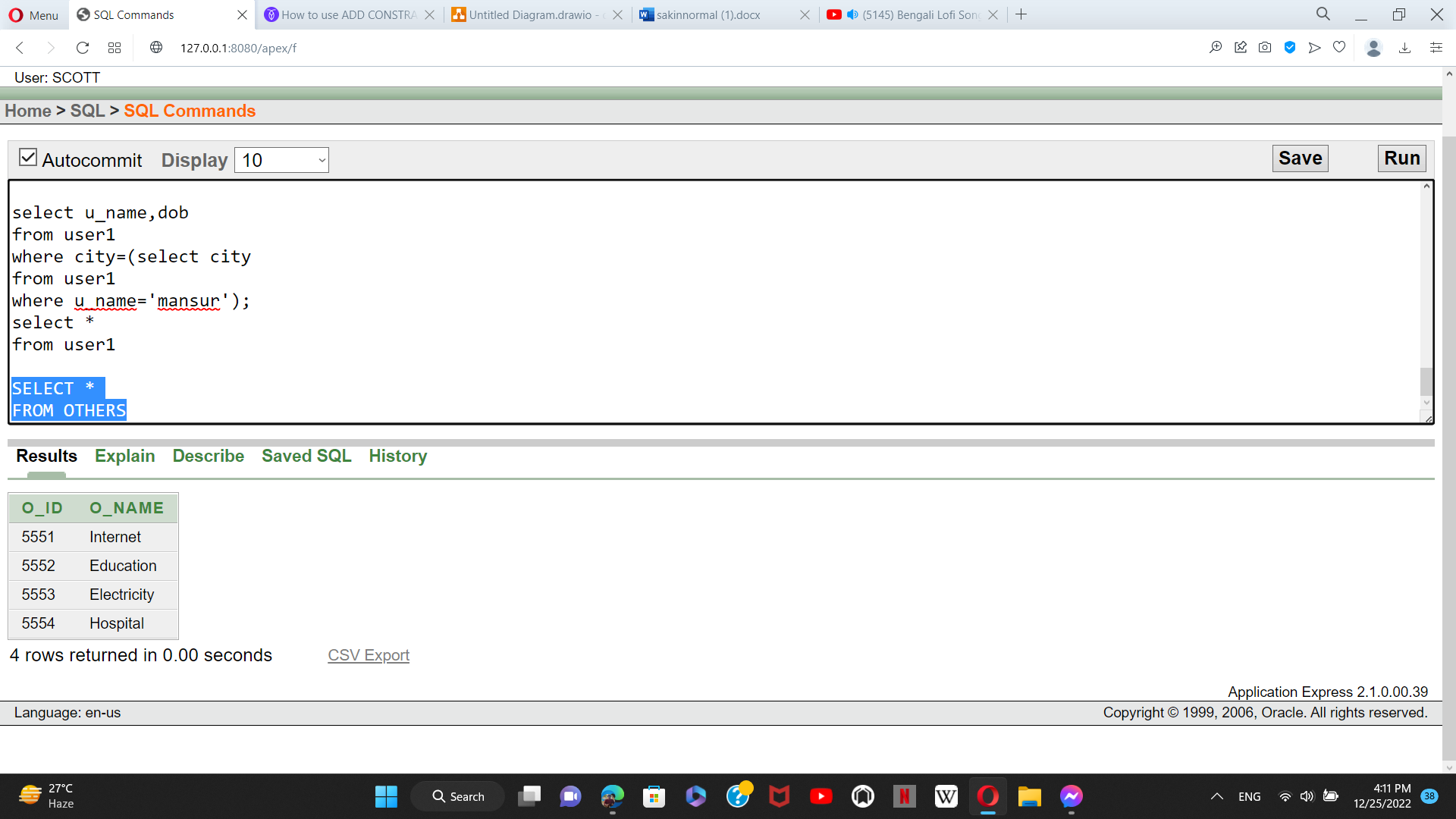
insert into OTHERS values(5551,'Internet');

insert into OTHERS values(5552,'Education');

insert into OTHERS values(5553,'Electricity');

insert into OTHERS values(5554,'Hospital');

**TABLE:**



**USER\_PAYBILL TABLE INSERTION:**

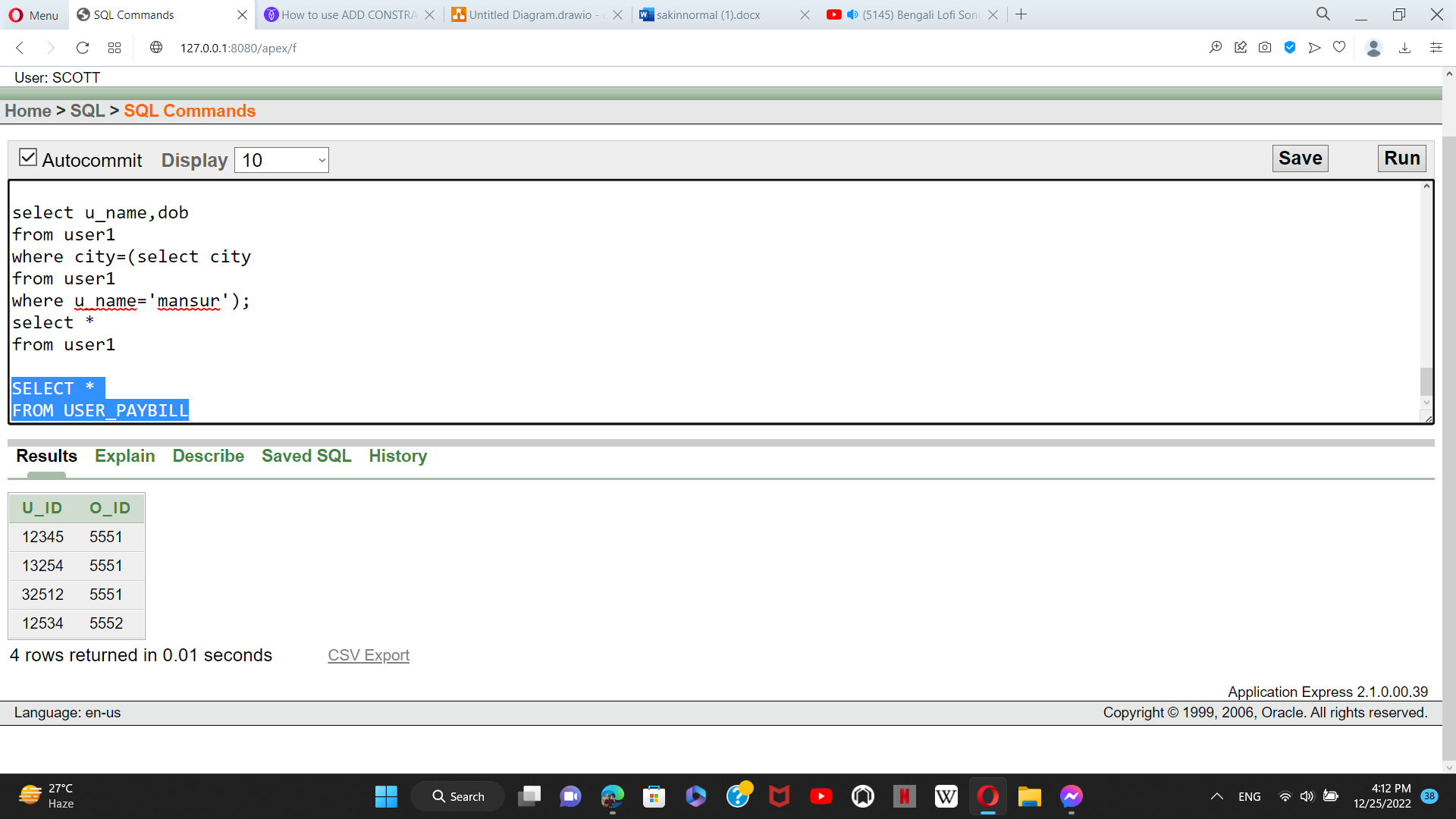
insert into USER\_PAYBILL values(12345,5551);

insert into USER\_PAYBILL values(13254,5551);

insert into USER\_PAYBILL values(32512,5551);

insert into USER\_PAYBILL values(12534,5552);

**TABLE:**



**SHOPS TABLE INSERTION:**

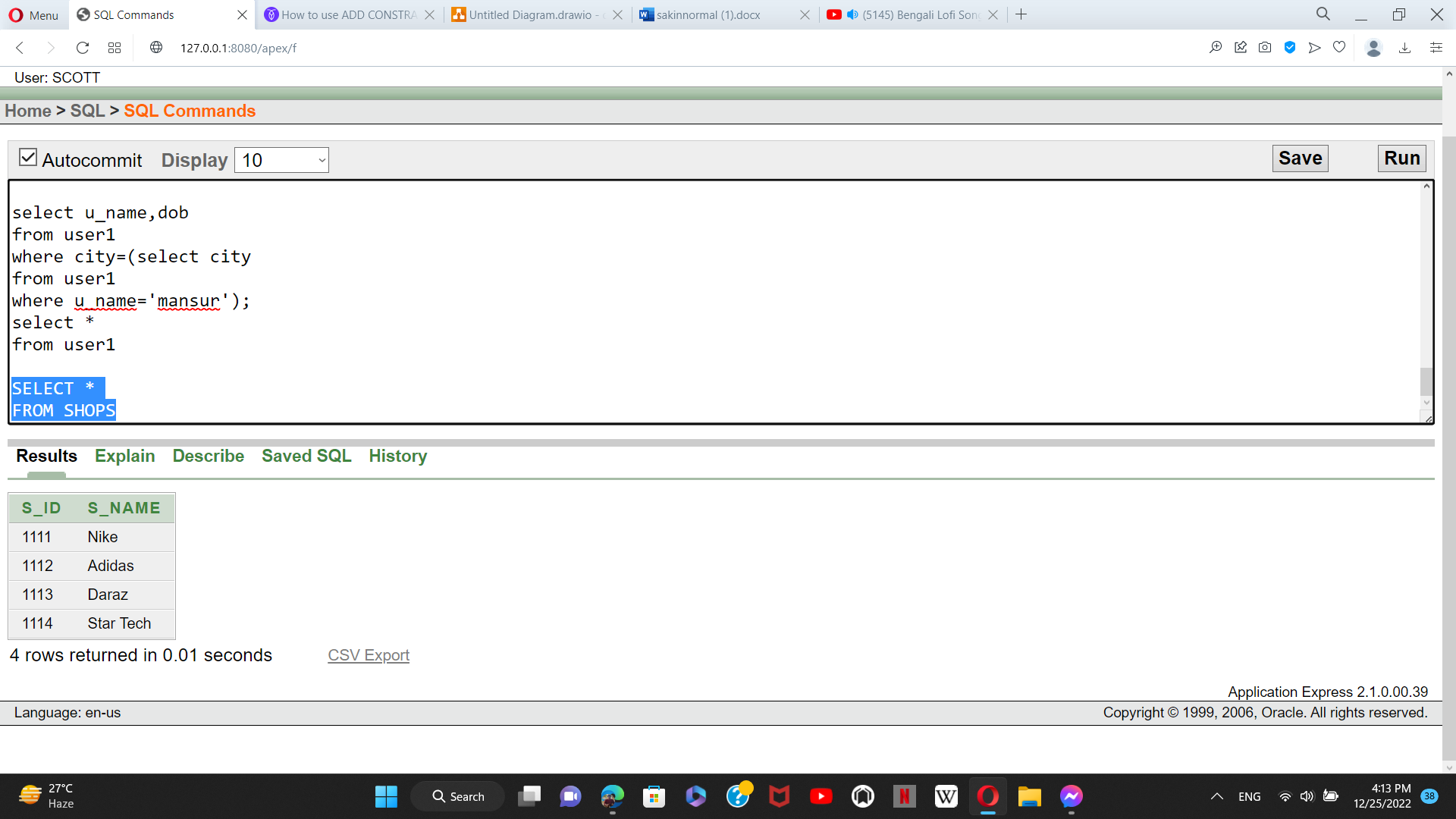
insert into SHOPS values(1111,'Nike');

insert into SHOPS values(1112,'Adidas');

insert into SHOPS values(1113,'Daraz');

insert into SHOPS values(1114,'Star Tech');

**TABLE:**



**USER\_PAYMENT TABLE INSERTION:**

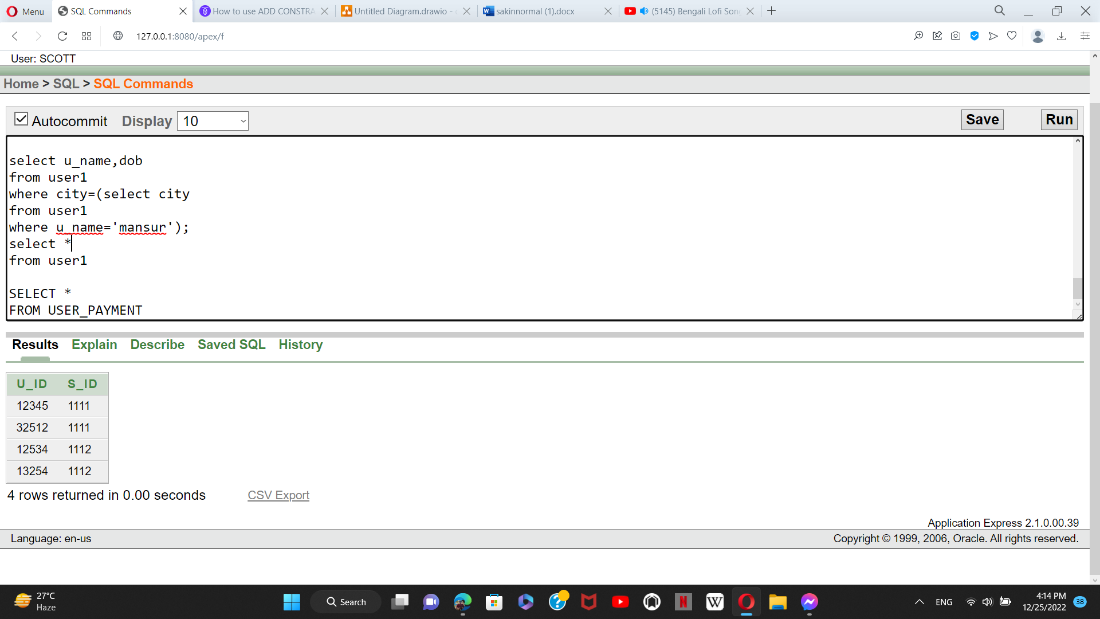
insert into USER\_PAYMENT values(12345,1111);

insert into USER\_PAYMENT values(32512,1111);

insert into USER\_PAYMENT values(12534,1112);

insert into USER\_PAYMENT values(13254,1112);

**TABLE:**



**USER\_LOGIN TABLE INSERTION:**

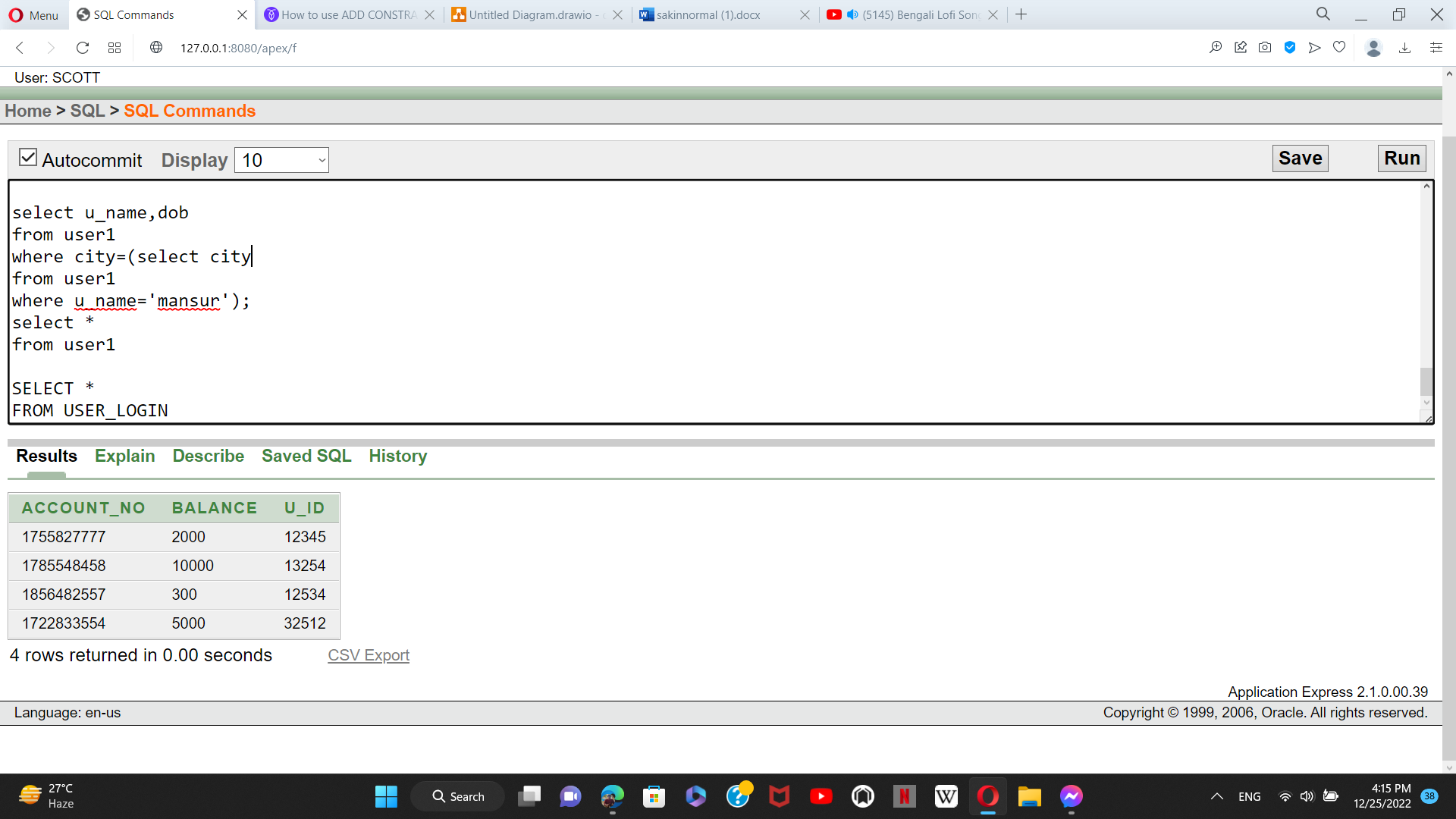
insert into USER\_LOGIN values(01755827777,2000,12345);

insert into USER\_LOGIN values(01785548458,10000,13254);

insert into USER\_LOGIN values(01856482557,300,12534);

insert into USER\_LOGIN values(01722833554,5000,32512);

**TABLE:**



**BANK TABLE INSERTION:**

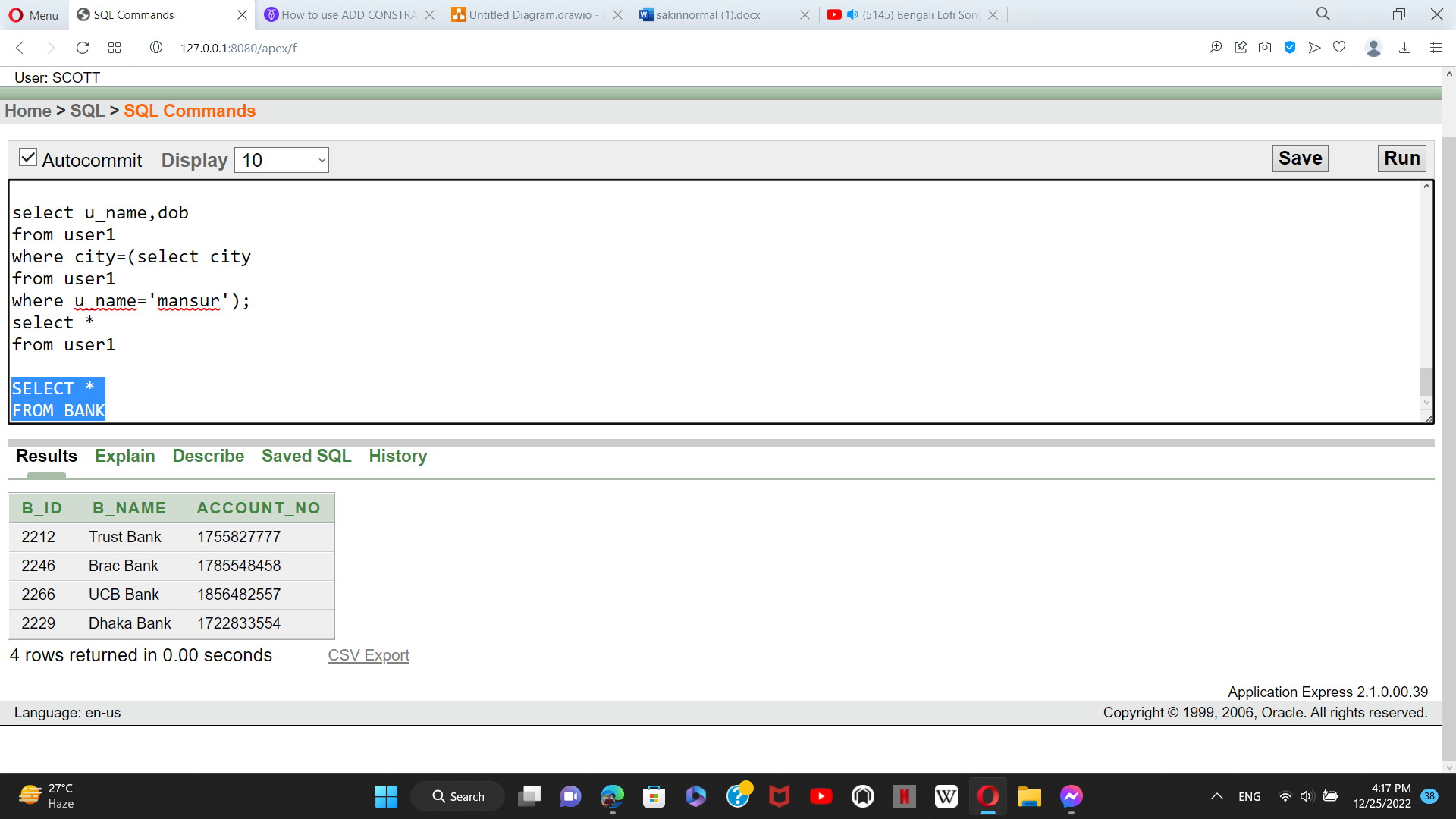
insert into BANK values(2212,'Trust Bank',01755827777);

insert into BANK values(2246,'Brac Bank',01785548458);

insert into BANK values(2266,'UCB Bank',01856482557);

insert into BANK values(2229,'Dhaka Bank',01722833554);

**TABLE:**



**COMPANY TABLE INSERTION:**

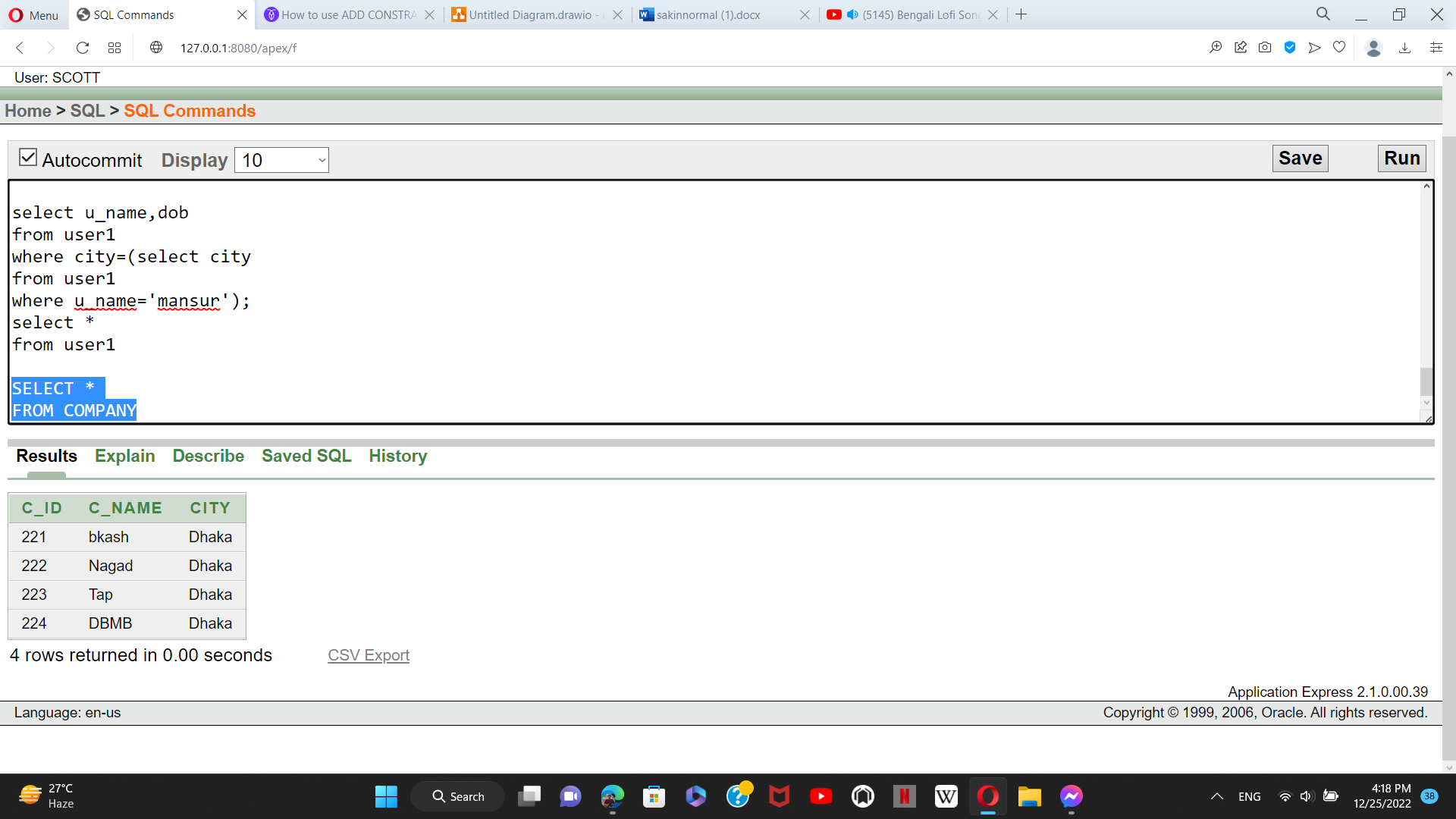
insert into COMPANY values(221,'bkash','Dhaka');

insert into COMPANY values(222,'Nagad','Dhaka');

insert into COMPANY values(223,'Tap','Dhaka');

insert into COMPANY values(224,'DBMB','Dhaka');

**TABLE:**



**C\_BANK TABLE INSERTION:**

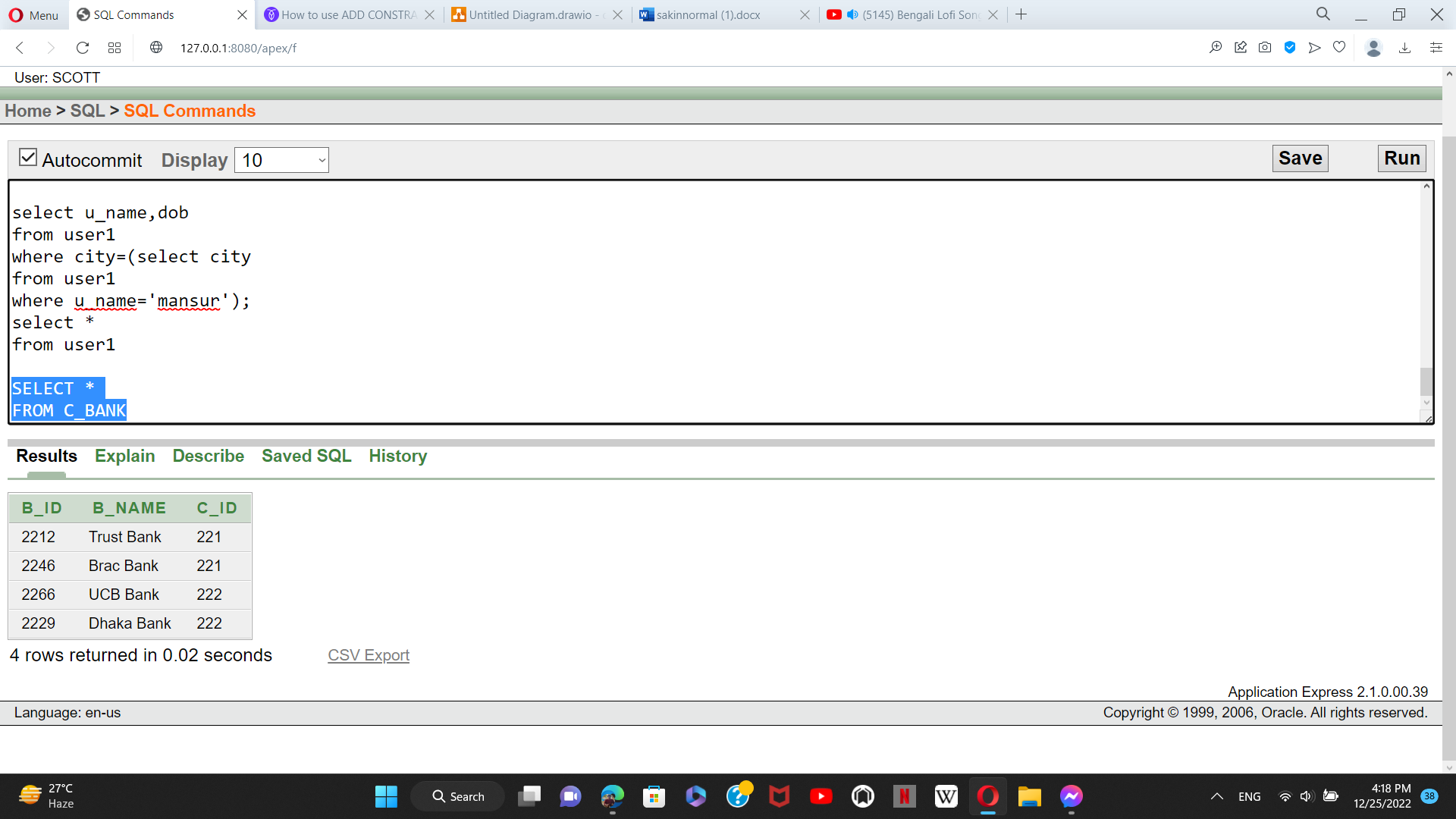
insert into C\_BANK values(2212,'Trust Bank',221);

insert into C\_BANK values(2246,'Brac Bank',221);

insert into C\_BANK values(2266,'UCB Bank',222);

insert into C\_BANK values(2229,'Dhaka Bank',222);

**TABLE:**



**C\_EMP TABLE INSERTION:**

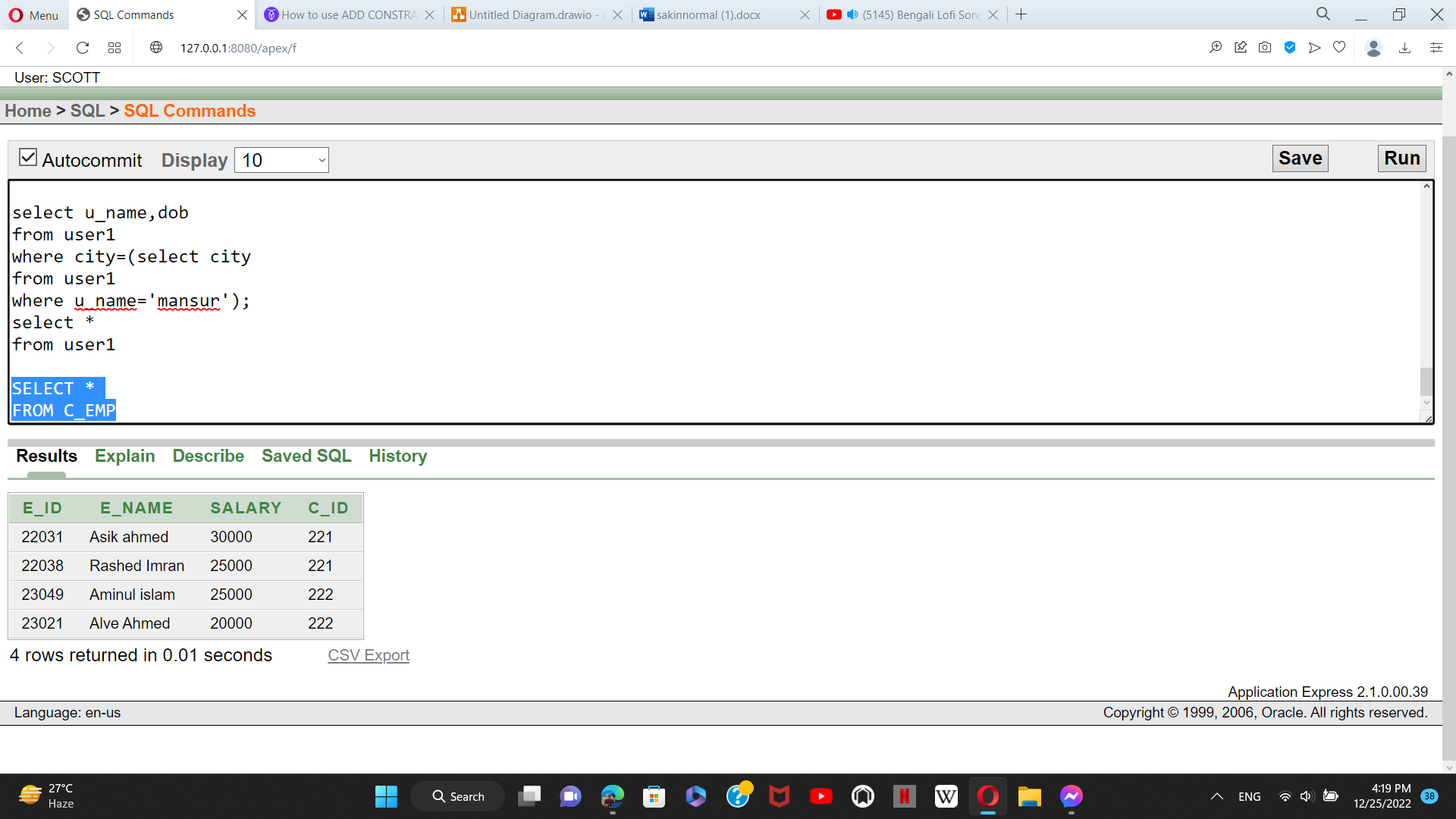
insert into C\_EMP values(22031,'Asik ahmed',30000,221);

insert into C\_EMP values(22038,'Rashed Imran',25000,221);

insert into C\_EMP values(23049,'Aminul islam',25000,222);

insert into C\_EMP values(23021,'Alve Ahmed',20000,222);

**TABLE:**



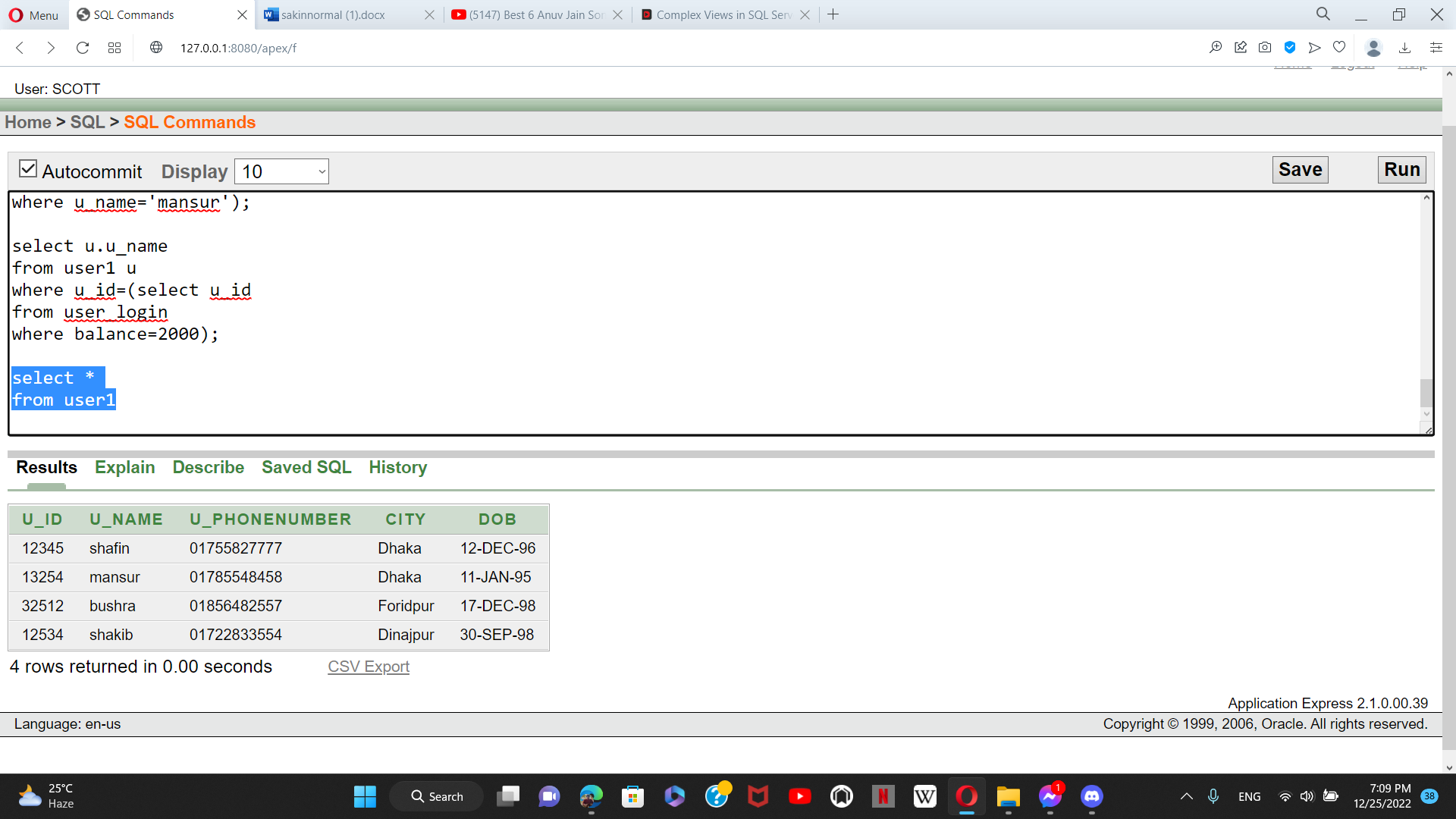
**QUERIES**

**SIMPLE QUERIES:**

1. Display all the user information by using sql queries

**Select \***

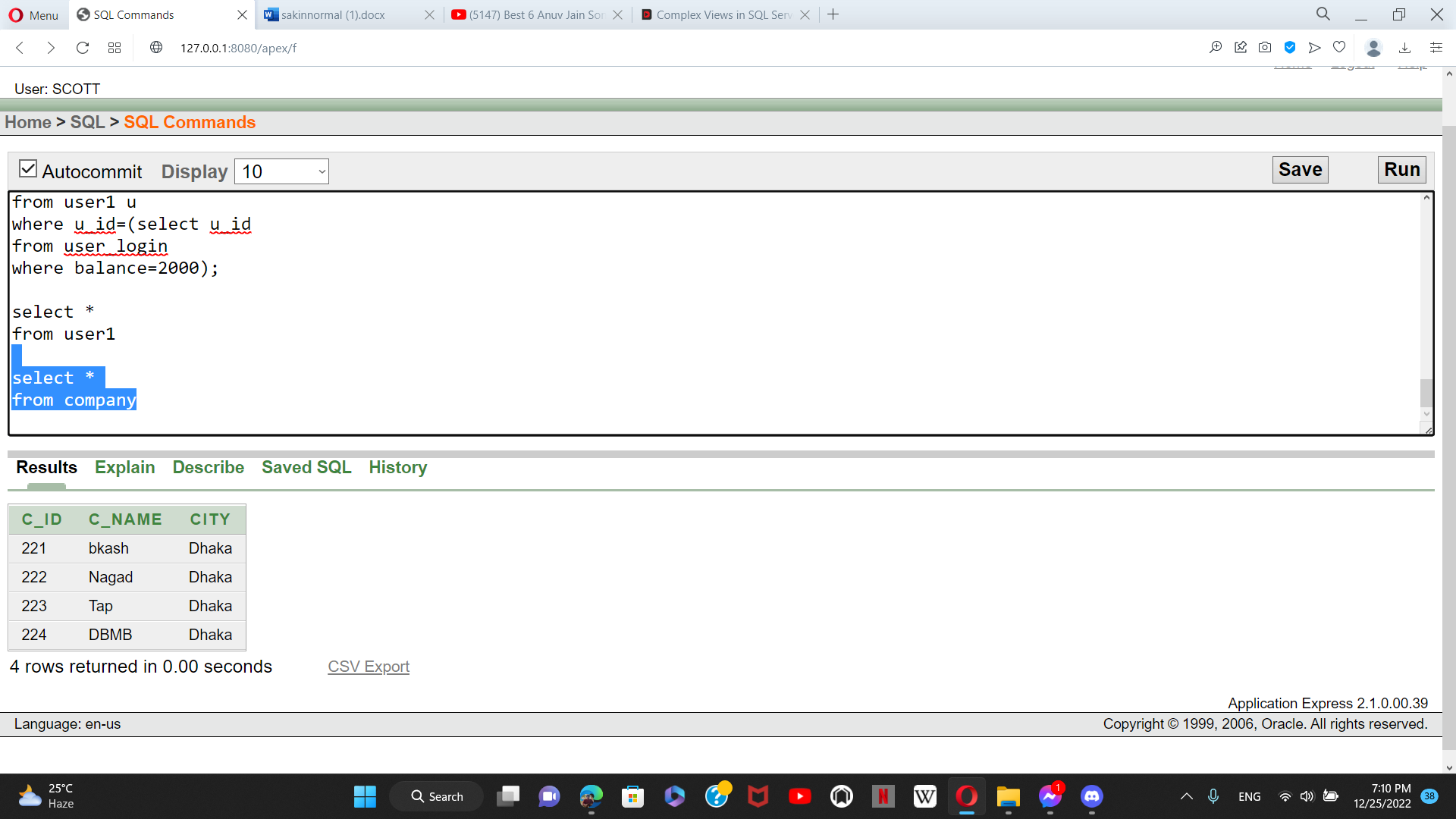
**From user1;**



1. Display all the company information by using sql queries

**Select \***

**From company**



**JOINING QUERIES:**

1. **Self join:**

**Create a query which appears like ‘shakib live in dinajpur’ by using self joining**

**select a.u\_name ||' live in '|| a.city as user\_location**

**from user1 a, user1 b**

**where a.u\_id=b.u\_id**

**order by a.u\_name;**



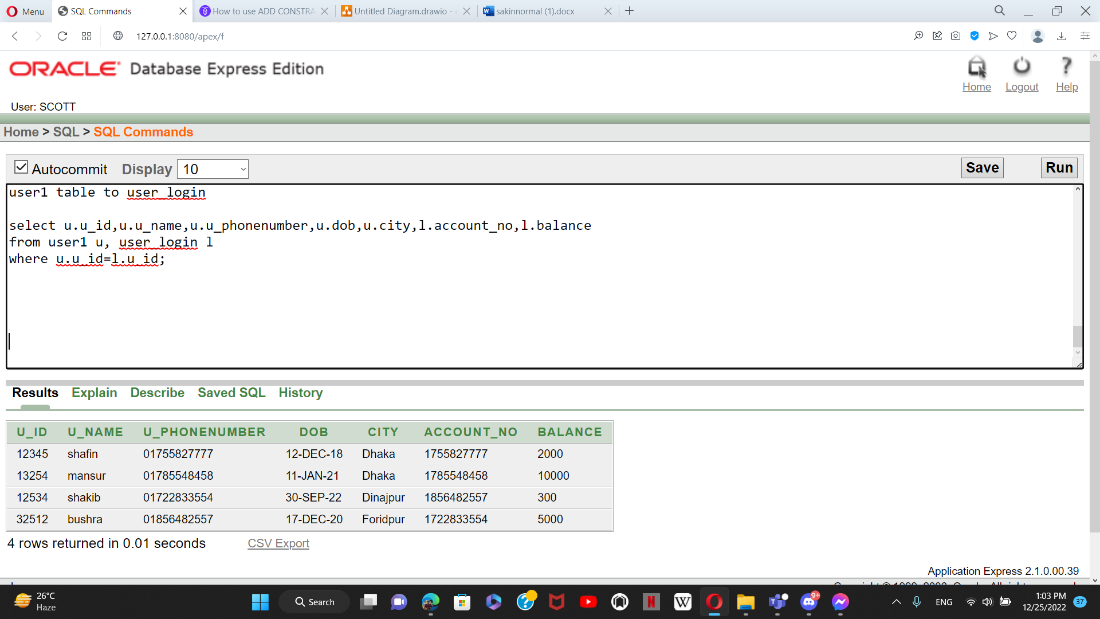
**2. Equijoin:**

**Create a join to display user id, user name, user phonenumber, user date of birth, user city, user account number, and user balance from user1 table and user\_login table.**

**select u.u\_id,u.u\_name,u.u\_phonenumber,u.dob,u.city,l.account\_no,l.balance**

**from user1 u, user\_login l**

**where u.u\_id=l.u\_id;**



3.**OUTHER JOIN**

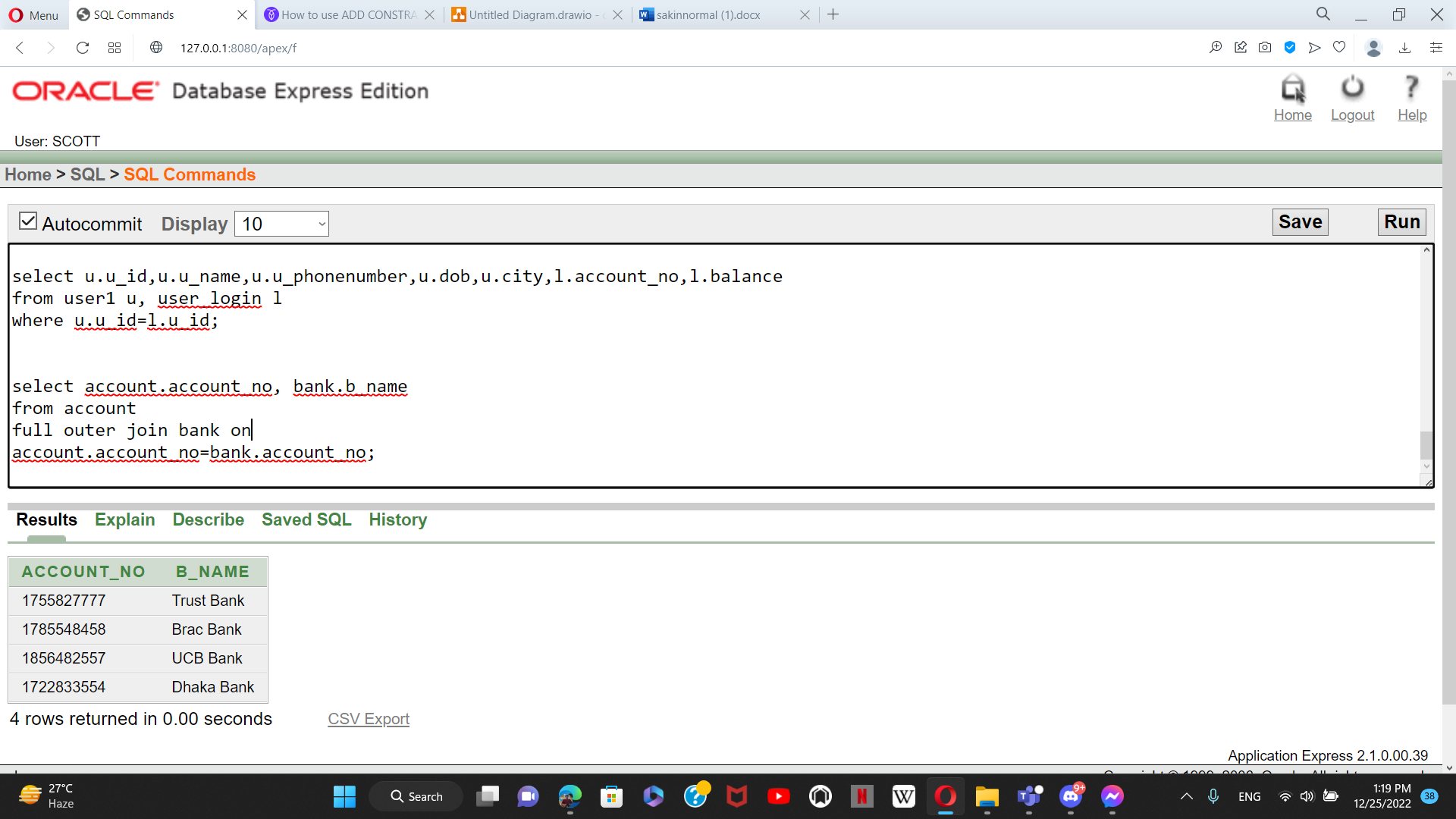
**Create a query to join user account number, and bank name from account table**

**select account.account\_no, bank.b\_name**

**from account**

**full outer join bank on**

**account.account\_no=bank.account\_no;**



**VIEW**

1. **Simple view:**

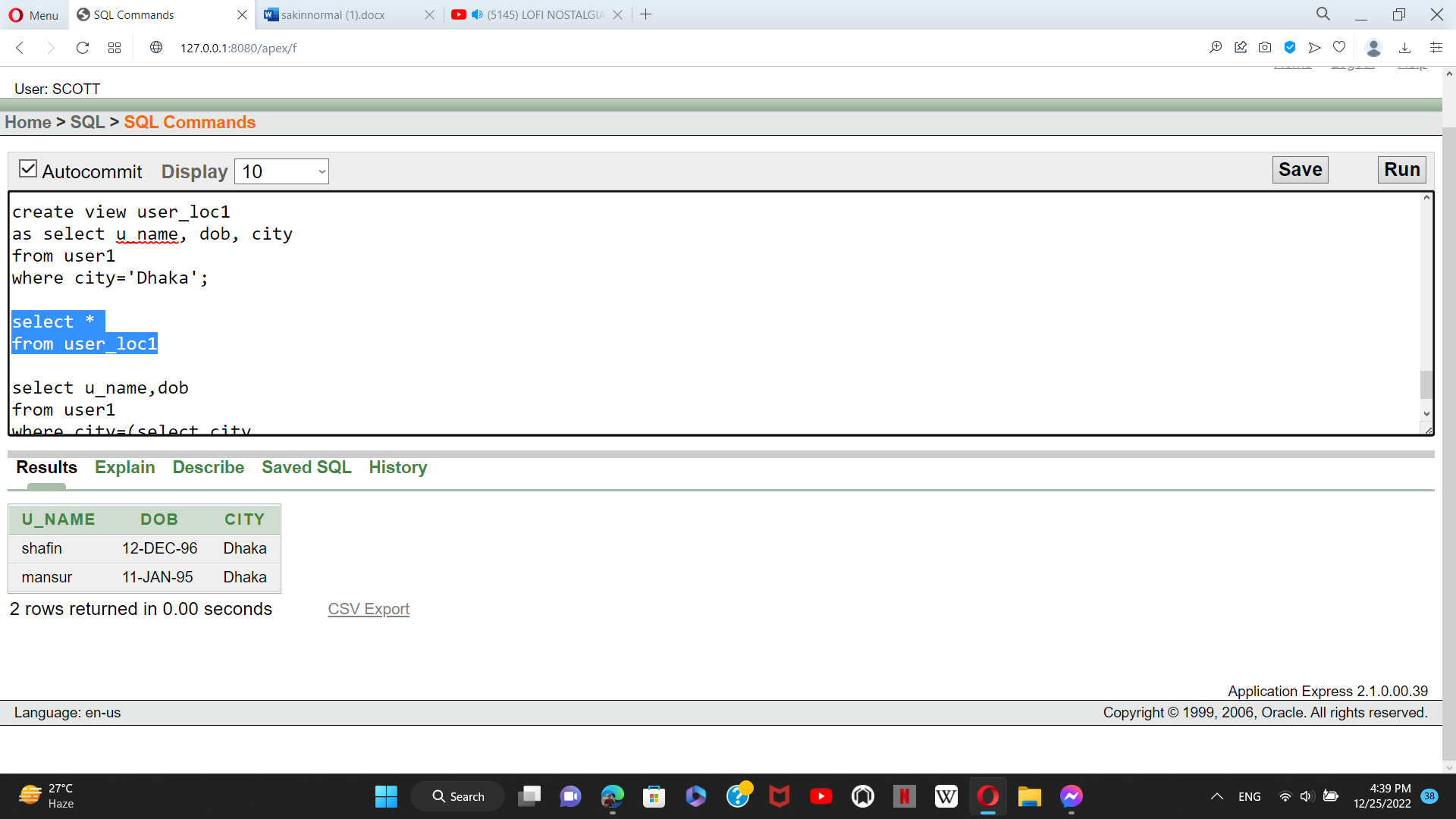
create a view called **user\_loc1** based on the user name, user date of birth, and user city from the user1 table. and their live is in dhaka city.

**create view user\_loc1**

**as select u\_name, dob, city**

**from user1**

**where city='Dhaka';**



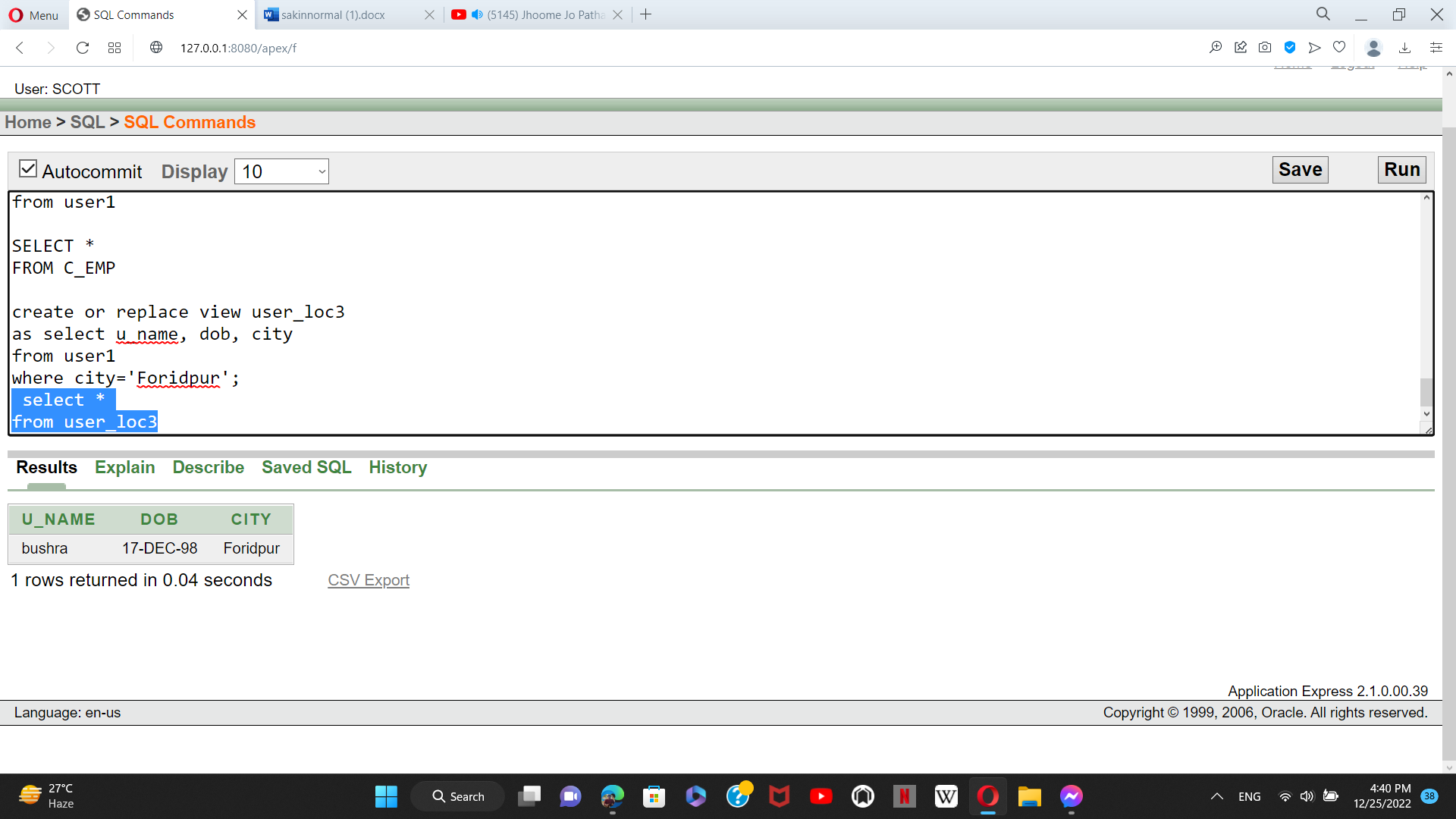
**2. c**reate a view called **user\_loc3** based on the user name, user date of birth, and user living city from the user1 table. This time their living in foridpur city

**create or replace view user\_loc3**

**as select u\_name, dob, city**

**from user1**

**where city='Foridpur';**



3.

create a view called **transfer** based on the account number, and bank name from account table.

**Complex view:**

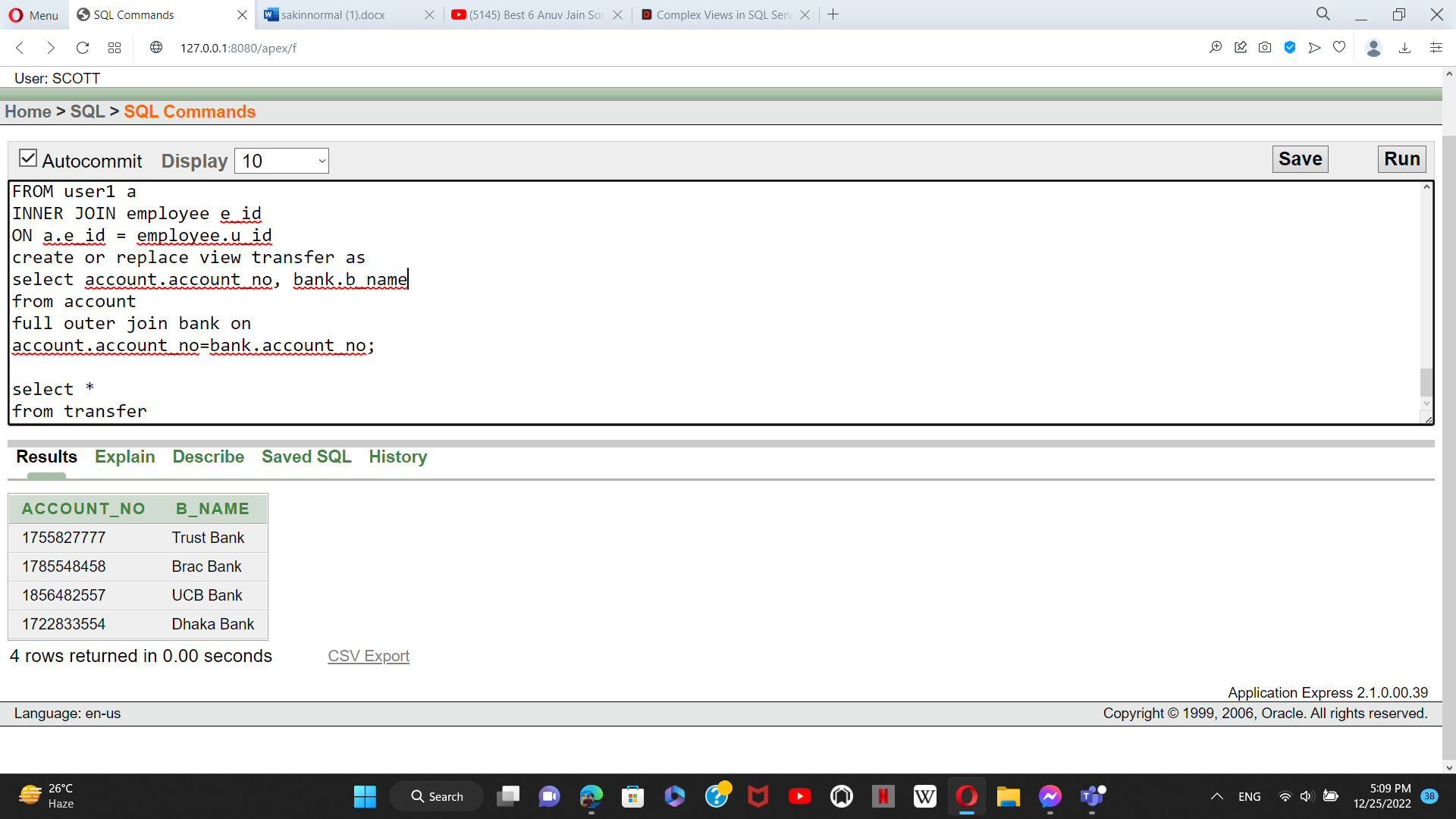
**create or replace view transfer as**

**select account.account\_no, bank.b\_name**

**from account**

**full outer join bank on**

**account.account\_no=bank.account\_no;**



**SUBQUERY**

1.write a query to display user name and data of birth those who live in same place as mansue

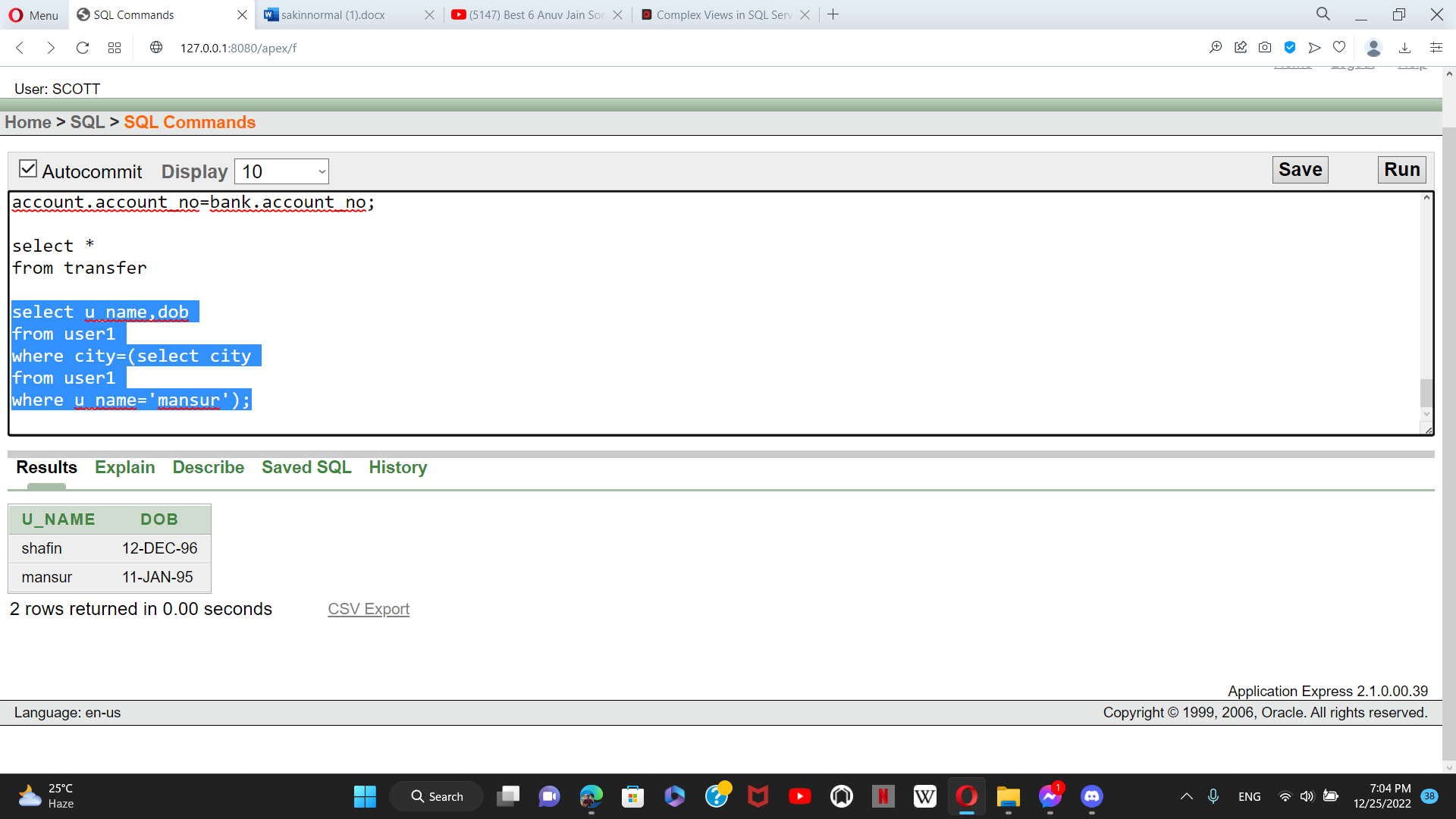
**select u\_name,dob**

**from user1**

**where city=(select city**

**from user1**

**where u\_name='mansur');**



1. Write a query to display whose balance is 2000 taka in his or her account

**select u.u\_name**

**from user1 u**

**where u\_id=(select u\_id**

**from user\_login**

**where balance=2000);**



CONSTRAINTS ADDING

**CREATE TABLE User1 (**

**u\_id INT PRIMARY KEY NOT NULL,**

**u\_name VARCHAR(20) NOT NULL,**

**u\_phonenumber VARCHAR(20) NOT NULL,**

**city VARCHAR(20) NOT NULL,**

**dob DATE NOT NULL,**

**);**

**ALTER TABLE USER1**

**ADD FOREIGN KEY (CITY) REFERENCES COUNTRY(CITY);**



SEQUENCE

**1.**

**create sequence USER34**

**start with 1**

**increment by 1**

**maxvalue 500**

**NOCACHE**

**NOCYCLE**

