**AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH(AIUB)**

**INTRODUCTION TO DATABASE**

SUPERVISE BY

**SAYED NAFIUL SHEEFAT**

**PROJECT TOPIC: ORGAN DONOR MANAGEMENT SYSTEM**

**GROUP NO: 05**

**Section: G**

**Submitted by:**

|  |  |
| --- | --- |
| **NAME** | **ID** |
| A.F.M. Rafiul Hassan | 22-47048-1 |
| MD Shohanur Rahman Shohan | 22-46013-1 |
| Mahmuda Khatun | 22-47016-1 |

**TABLE OF CONTENTS**

Topics

**1. Abstract & Introduction**

**2. Scenario**

**3. Entity Relationship Diagram**

**4. Normalization**

**5. Finalization**

**6. Table Creation**

**7. Data Insertion**

**8. Query Test**

**9. Conclusion**

**ABSTRACT**

This project aims to develop a Organ Bank Management System. A Organ Bank Management System can be used in any clinic, hospital, labs or any emergency situation which requires Organ units for survival. Our system can be used to find required type of blood in emergency situations from either Organ bank or even Organ donors. Current system uses a grapevine communication for finding Organ in cases of emergency, may it be by a donor or Organ bank. The intentions of proposing such a system is to abolish the panic caused during an emergency due to unavailability of organ**.**

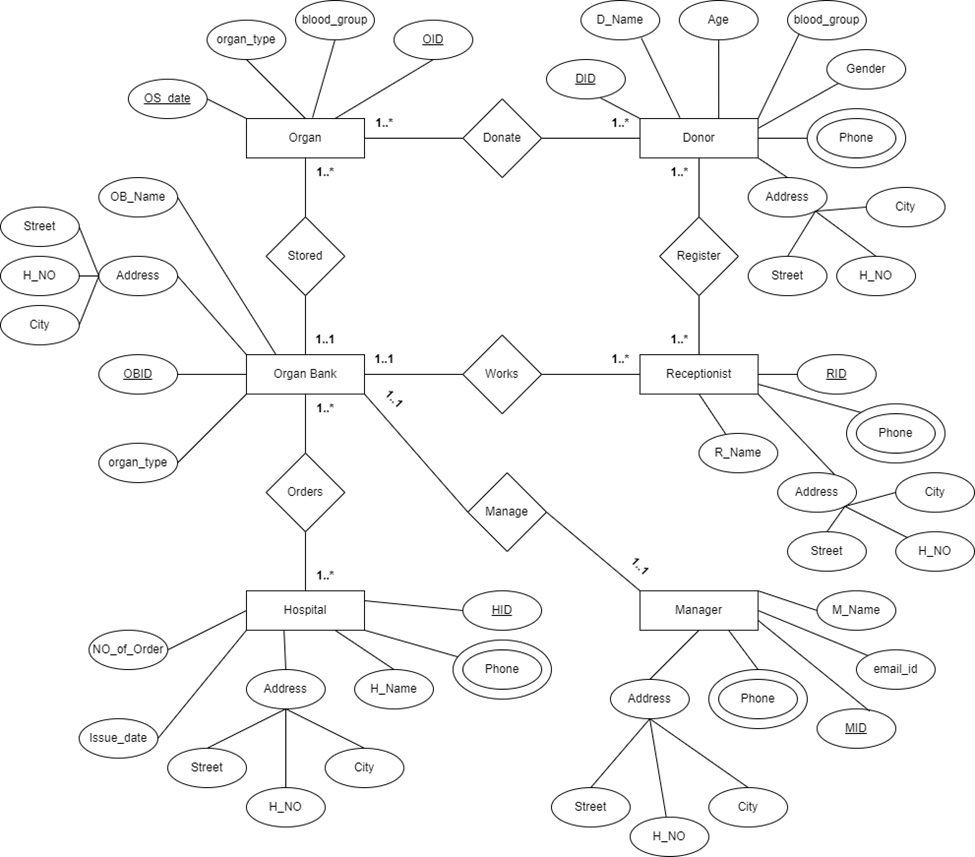
**INTRODUCTION**

Organ banks collect, store and provide collected organ to the patients who are in need of organ. The people who donate organ are called ‘donors’. The banks then group the organ which they receive according to the organ types. The main mission of the organ bank is to provide the organ to the hospitals and health care systems which saves the patient’s life. The major concern each organ bank has is to monitor the quality of the organ and monitor the people who donates the organ, that is ‘donors’. But this a tough job. The existing system will not satisfy the need of maintaining quality organ and keep track of donors. To overcome all these limitations we introduced a new system called ‘Organ Bank Management System’.The ‘Organ Bank Management System’ allows us to keep track of quality of organ and also keeps track of available organ types when requested by the acceptor. The existing systems are Manual systems which are time consuming and not so effective. ‘Organ Bank Management system’ automates the distribution of organ. This database consists of thousands of records of each organ bank.By using this system searching the available organ becomes easy and saves lot of time than the manual system. It will hoard, operate, recover and analyse information concerned with the administrative and inventory management within a organ bank. This system is developed in a manner that it is manageable, time effective, cost effective, flexible and much man power is not required.

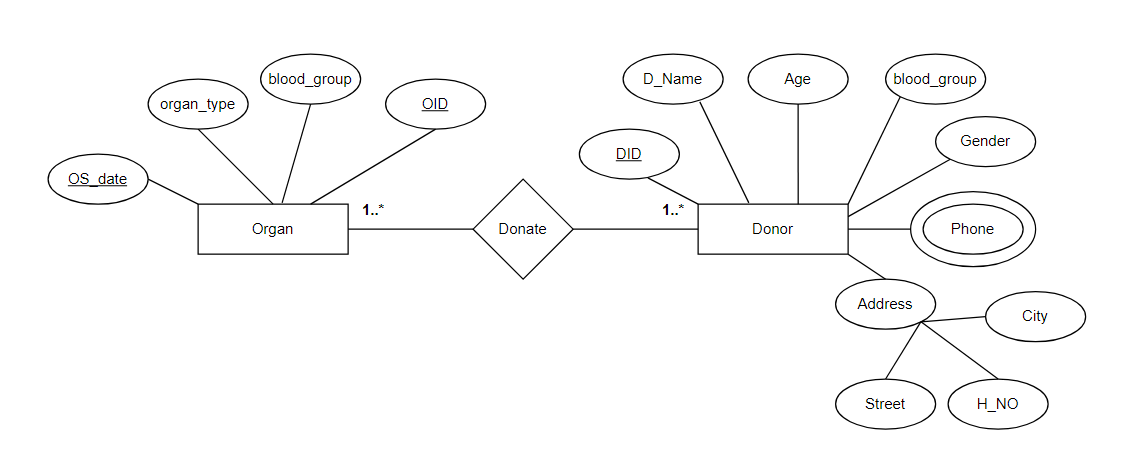
**SCENARIO**

**An organ bank stores various types of organs. Many donors can donate different type of organ. A donor may donate organ more than once and he/she is identified by a donor id, name, age, gender, address, and phone number. The organ donated by the donor is distinguish by organ type, organ id, blood group and organ stored date. Before each donor donates his/her organ, he/she is required to register himself as a donor with the receptionist who works at the organ bank. The receptionist is characterized by receptionist id, name, address, and phone number. The organ banks receive orders for organ from many hospitals for emergency purposes and other surgical requirements and organ bank issues the same of required type of organ. The hospitals are identified by hospital id, name, address, organ issue date, orders, and phone number. Organ bank has organ bank id, name, address, and organ types stored. The organ bank is managed by the manager who is identified by manager id, name, e-mail id, address, and phone number. He/she is responsible for the proper management of the organ bank.**

**Entity Relationship Diagram**

****

**NORMALIZATION**



**Donate:**

**UNF:**

DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, PHONE, H\_NO, STREET, CITY, OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE

**1NF:** PHONE is a multivalued attribute.

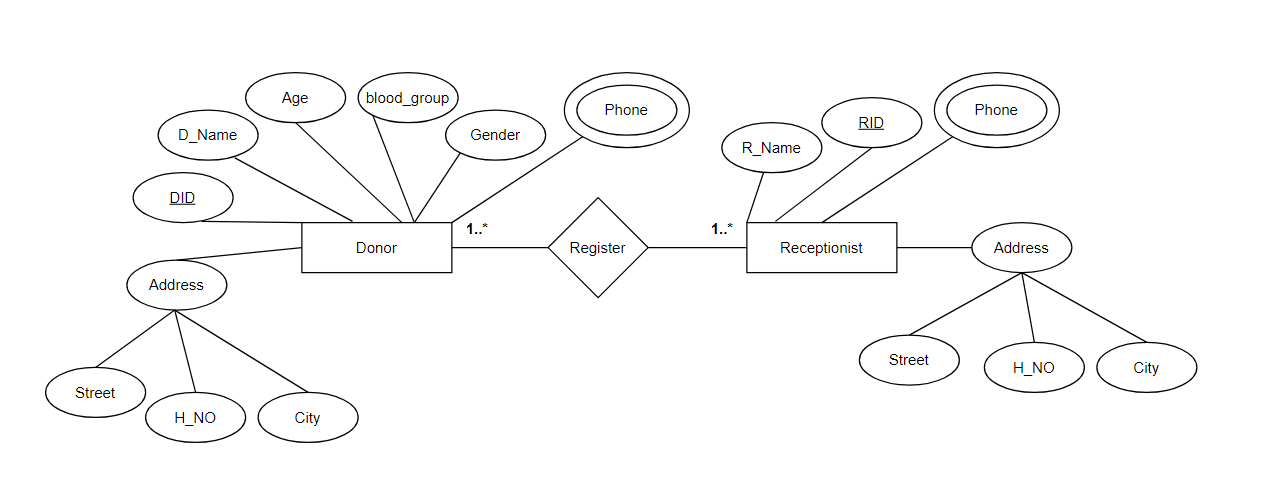
DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, H\_NO, STREET, CITY, OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE

**2NF:**

1. DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, H\_NO, STREET, CITY, OID, PID
2. PHONE, PID
3. OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE

**3NF:**

1. DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, OID, PID, AID
2. PID, PHONE
3. AID, H\_NO, STREET, CITY
4. OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE



**Register:**

**UNF:**

DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, PHONE, H\_NO, STREET, CITY, RID, R\_NAME, PHONE, H\_NO, STREET, CITY

**1NF:** PHONE is a multivalued attribute.

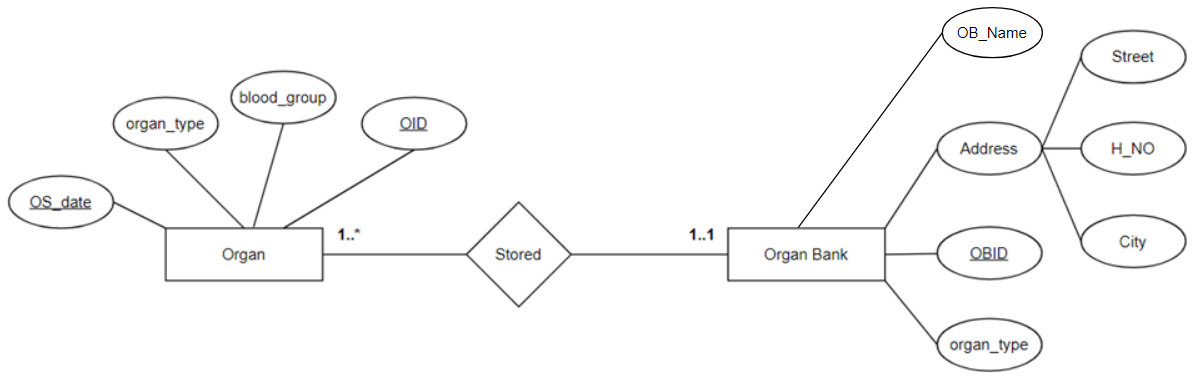
DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, H\_NO, STREET, CITY, RID, R\_NAME, PHONE, H\_NO, STREET, CITY

**2NF:**

1. DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, H\_NO, STREET, CITY, RID, PID
2. PID, PHONE
3. RID, R\_NAME, H\_NO, STREET, CITY, PID

**3NF:**

1. DID, D\_NAME, AGE, BLOOD\_GROUP, RID, PID, AID
2. PID, PHONE
3. AID, H\_NO, STREET, CITY
4. RID, R\_NAME, PID, AID



**Stored:**

**UNF:**

OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE, OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY

**1NF:** **NO** is a multivalued attribute. (SAME AS **UNF**)

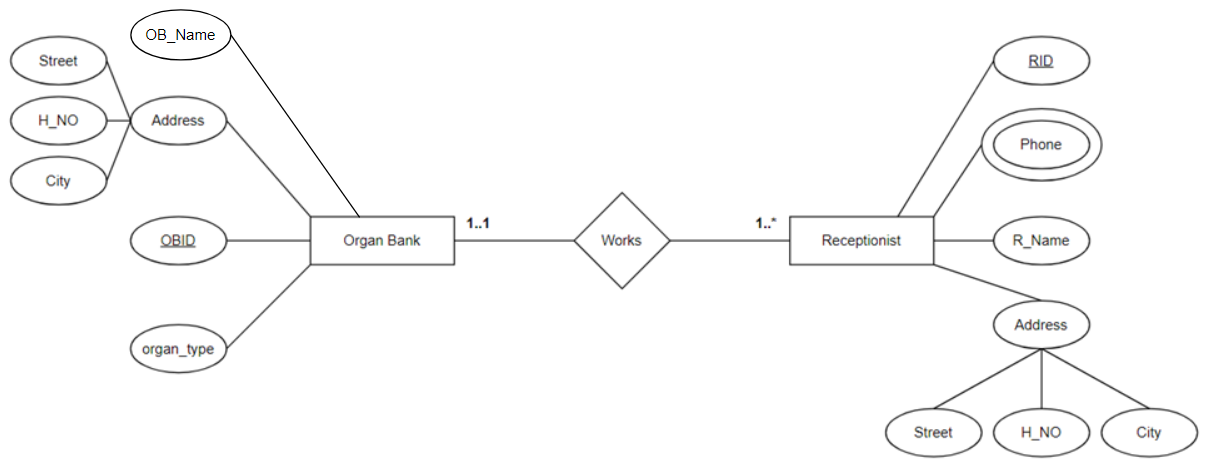
OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE, OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY

**2NF:**

1. OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE, OBID
2. OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY

**3NF:**

1. OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE, OBID
2. OBID, OB\_NAME, ORGAN\_TYPE, AID
3. AID, H\_NO, STREET, CITY



**Works:**

**UNF:**

OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY, RID, R\_NAME, PHONE, H\_NO, STREET, CITY

**1NF:** PHONE is a multivalued attribute.

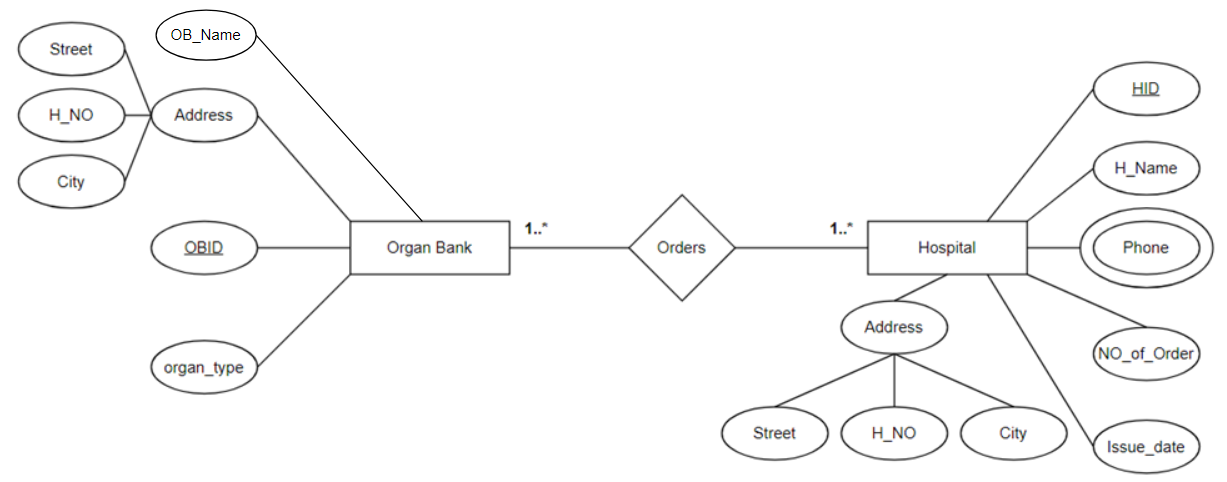
OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY, RID, R\_NAME, H\_NO, STREET, CITY

**2NF:**

1. OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY
2. PID, PHONE
3. RID, R\_NAME, PHONE, H\_NO, STREET, CITY, OBID, PID

**3NF:**

1. OBID, OB\_NAME, ORGAN\_TYPE, AID
2. PID, PHONE
3. AID, H\_NO, STREET, CITY
4. RID, R\_NAME, OBID, PID, AID



**Orders:**

**UNF:**

OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY, HID, H\_NAME, PHONE, NO\_OF\_ORDER, ISSUE\_DATE, H\_NO, STREET, CITY

**1NF:** PHONE is a multivalued attribute.

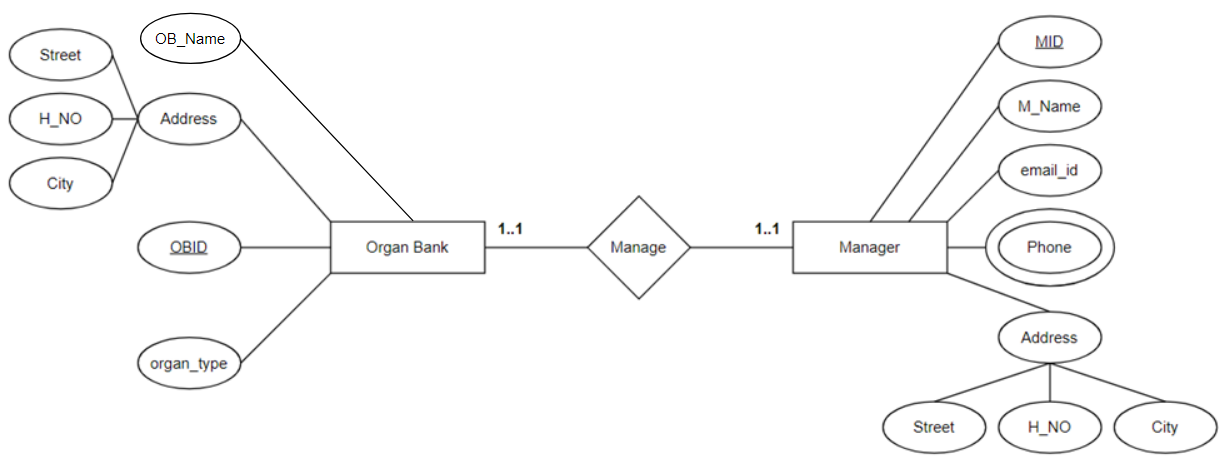
OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY, HID, H\_NAME, NO\_OF\_ORDER, ISSUE\_DATE, H\_NO, STREET, CITY

**2NF:**

1. OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY
2. PID, PHONE
3. HID, H\_NAME, NO\_OF\_ORDER, ISSUE\_DATE, H\_NO, STREET, CITY, OBID, PID

**3NF:**

1. OBID, OB\_NAME, ORGAN\_TYPE, AID
2. PID, PHONE
3. AID, H\_NO, STREET, CITY
4. HID, H\_NAME, NO\_OF\_ORDER, ISSUE\_DATE, OBID, PID, AID



**Manage:**

**UNF:**

OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY, MID, M\_NAME, EMAIL\_ID, PHONE, H\_NO, STREET, CITY

**1NF:** PHONE is a multivalued attribute.

OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY, MID, M\_NAME, EMAIL\_ID, H\_NO, STREET, CITY

**2NF:**

1. OBID, OB\_NAME, ORGAN\_TYPE, H\_NO, STREET, CITY
2. PID, PHONE
3. MID, M\_NAME, EMAIL\_ID, H\_NO, STREET, CITY, OBID, PID

**3NF:**

1. OBID, OB\_NAME, ORGAN\_TYPE, AID
2. PID, PHONE
3. AID, H\_NO, STREET, CITY
4. MID, M\_NAME, EMAIL\_ID, OBID, PID, AID

**FINALIZATION**

**1. DID, D\_NAME, AGE, BLOOD\_GROUP, GENDER, OID, PID, AID**

**2. PID, PHONE**

**3. AID, H\_NO, STREET, CITY**

**4. OID, BLOOD\_GROUP, ORGAN\_TYPE, OS\_DATE, OBID**

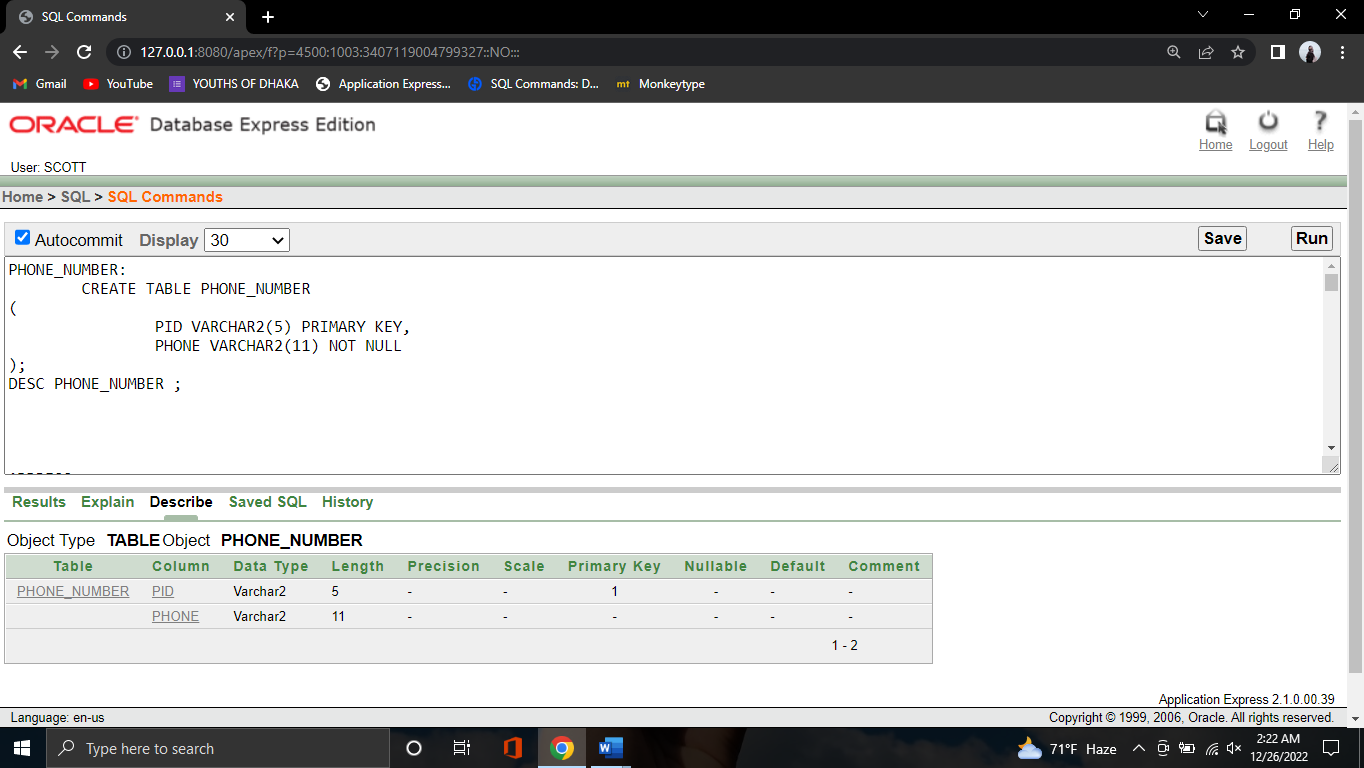
**5. OBID, OB\_NAME, AID**

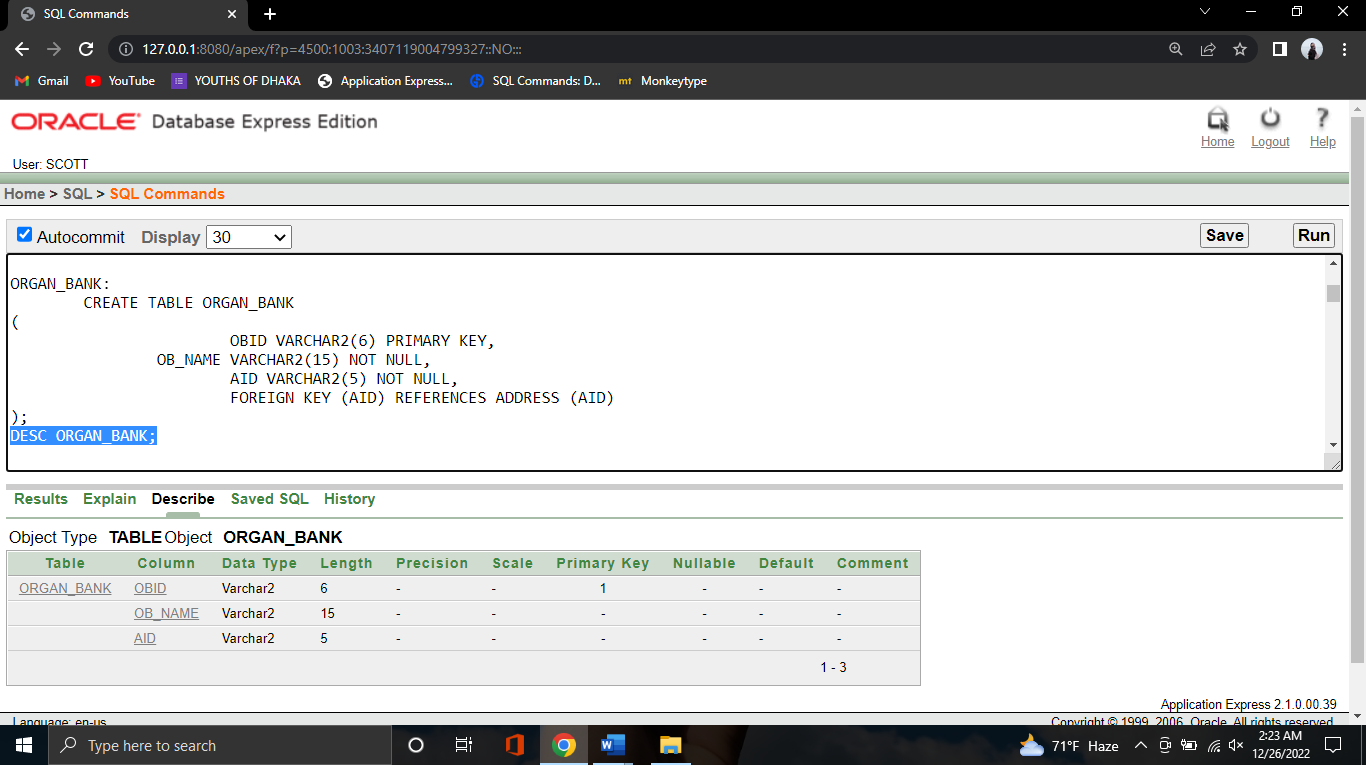
**6. RID, R\_NAME, OBID, PID, AID**

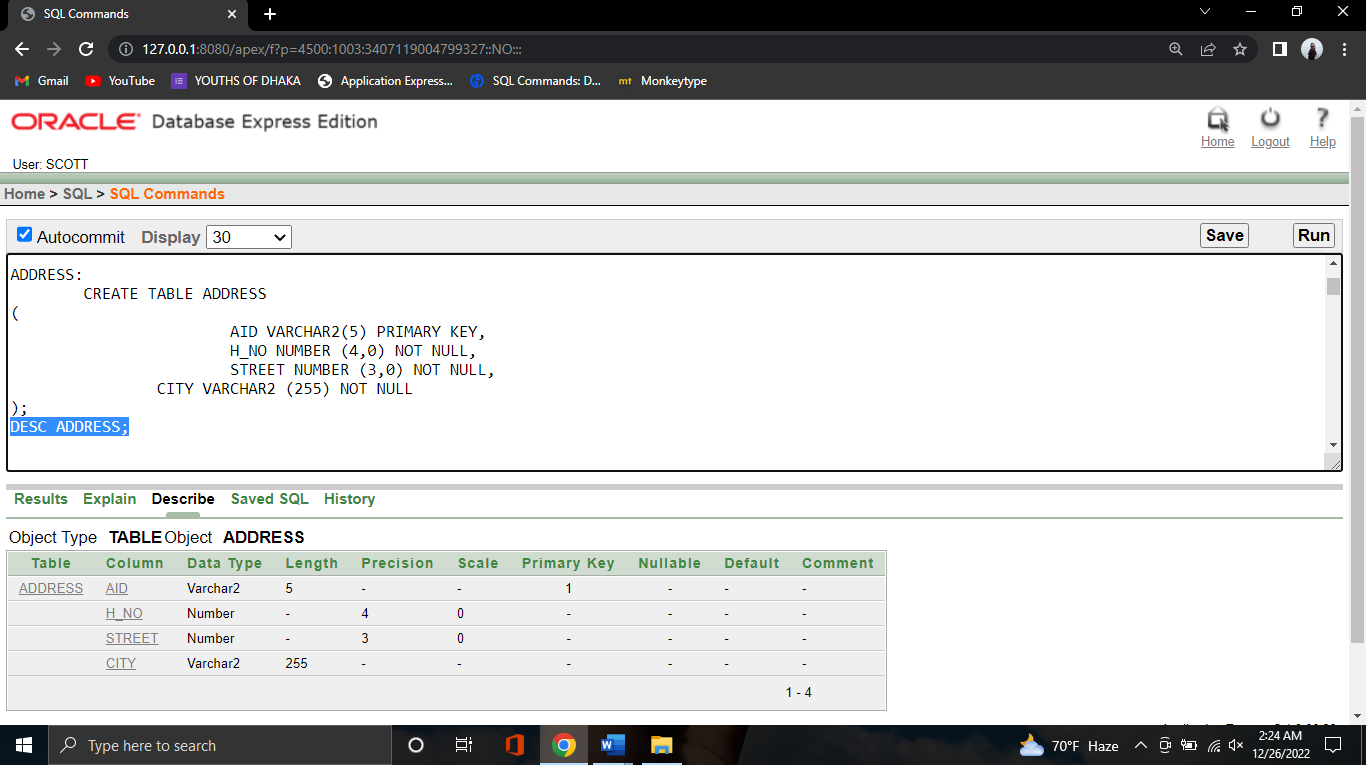
**7. HID, H\_NAME, NO\_OF\_ORDER, ISSUE\_DATE, OBID, PID, AID**

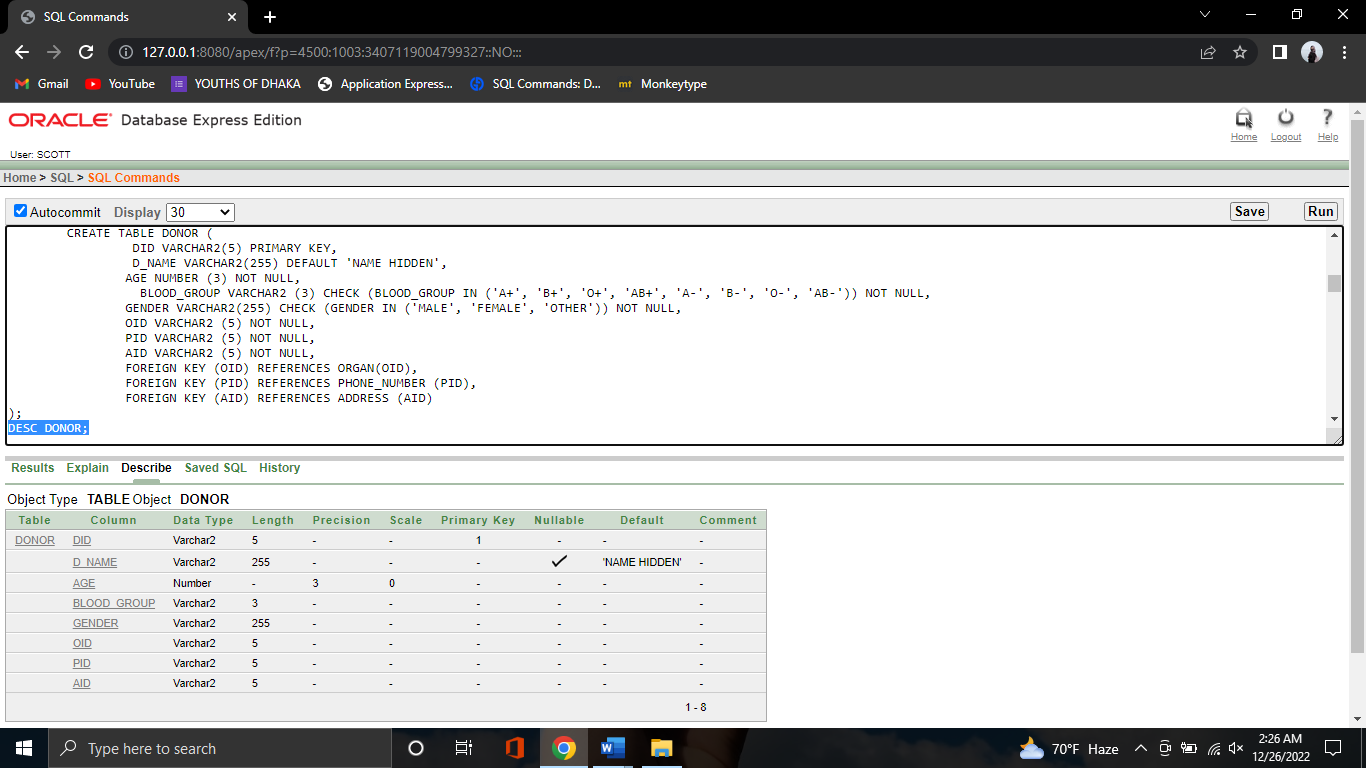
**8. MID, M\_NAME, EMAIL\_ID, OBID, PID, AID**

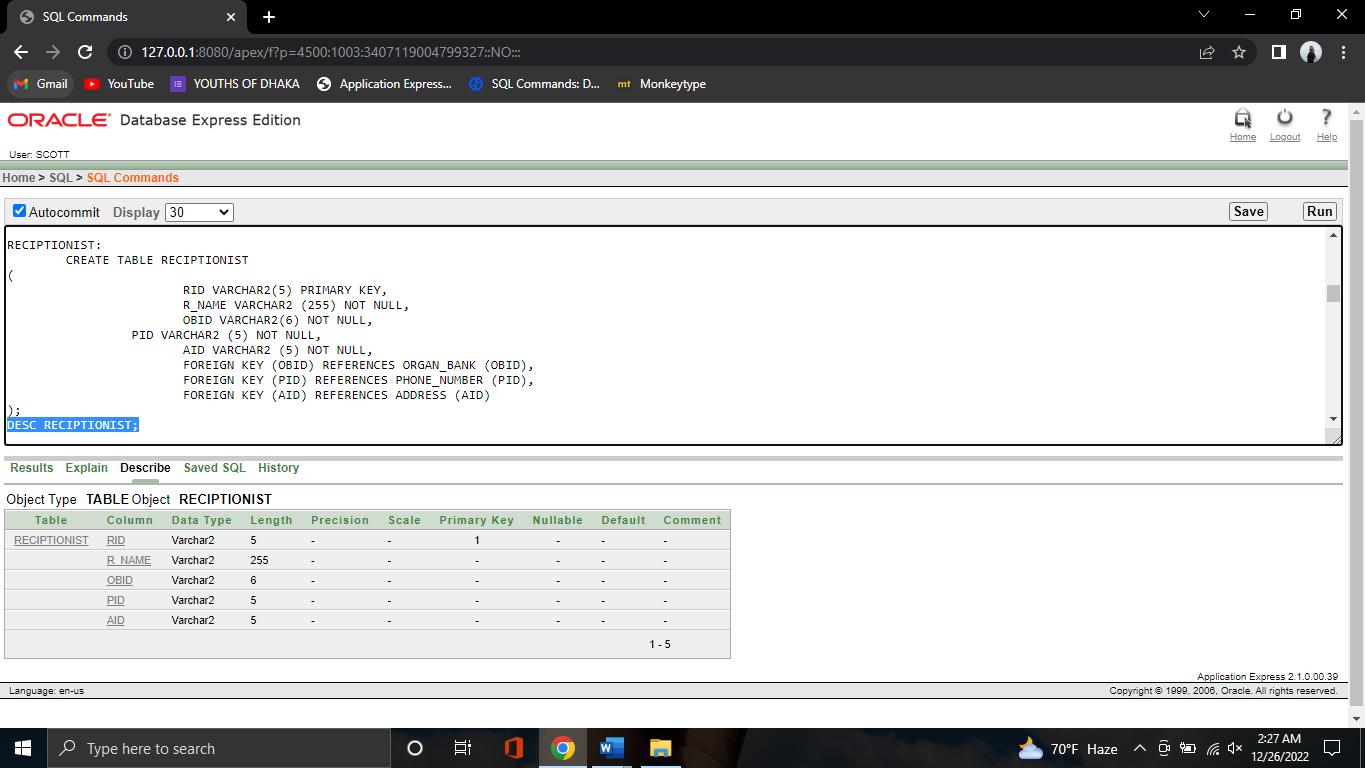
**TABLE CREATION**

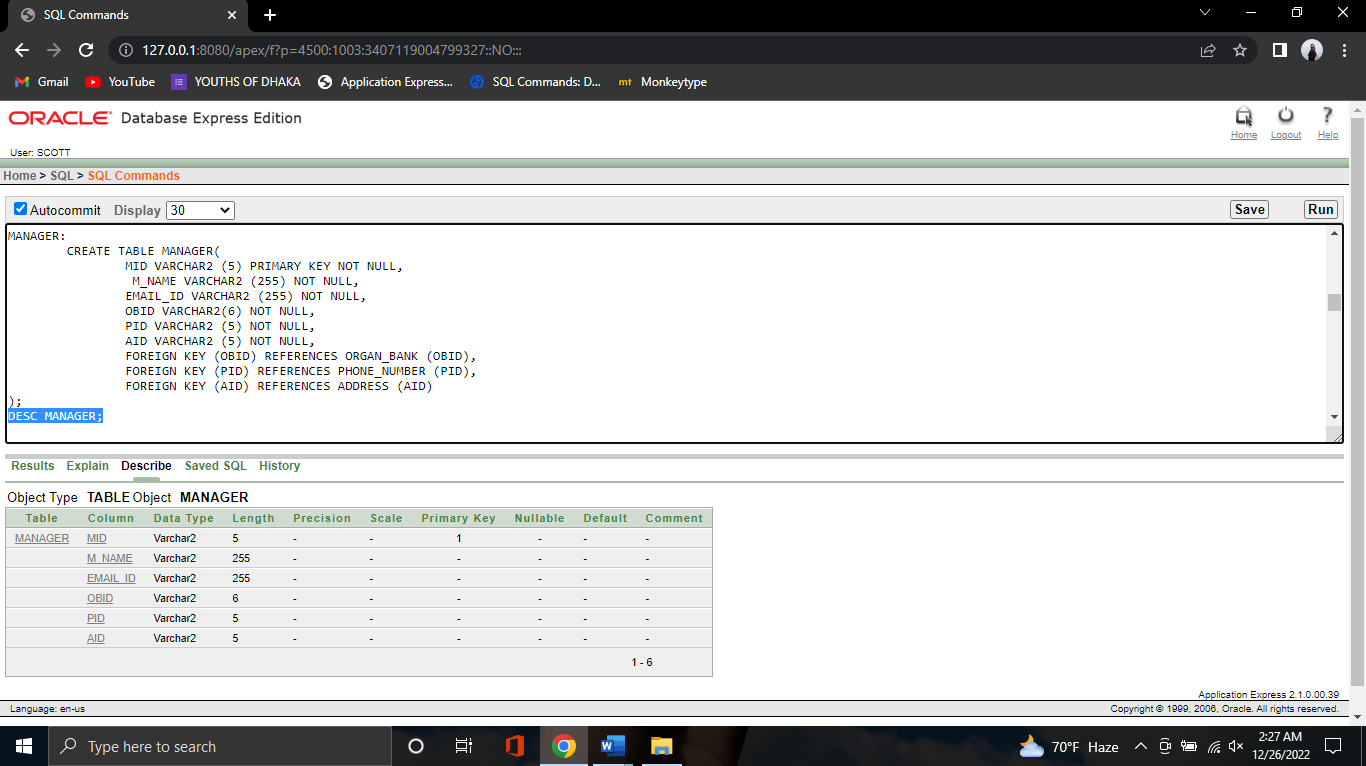


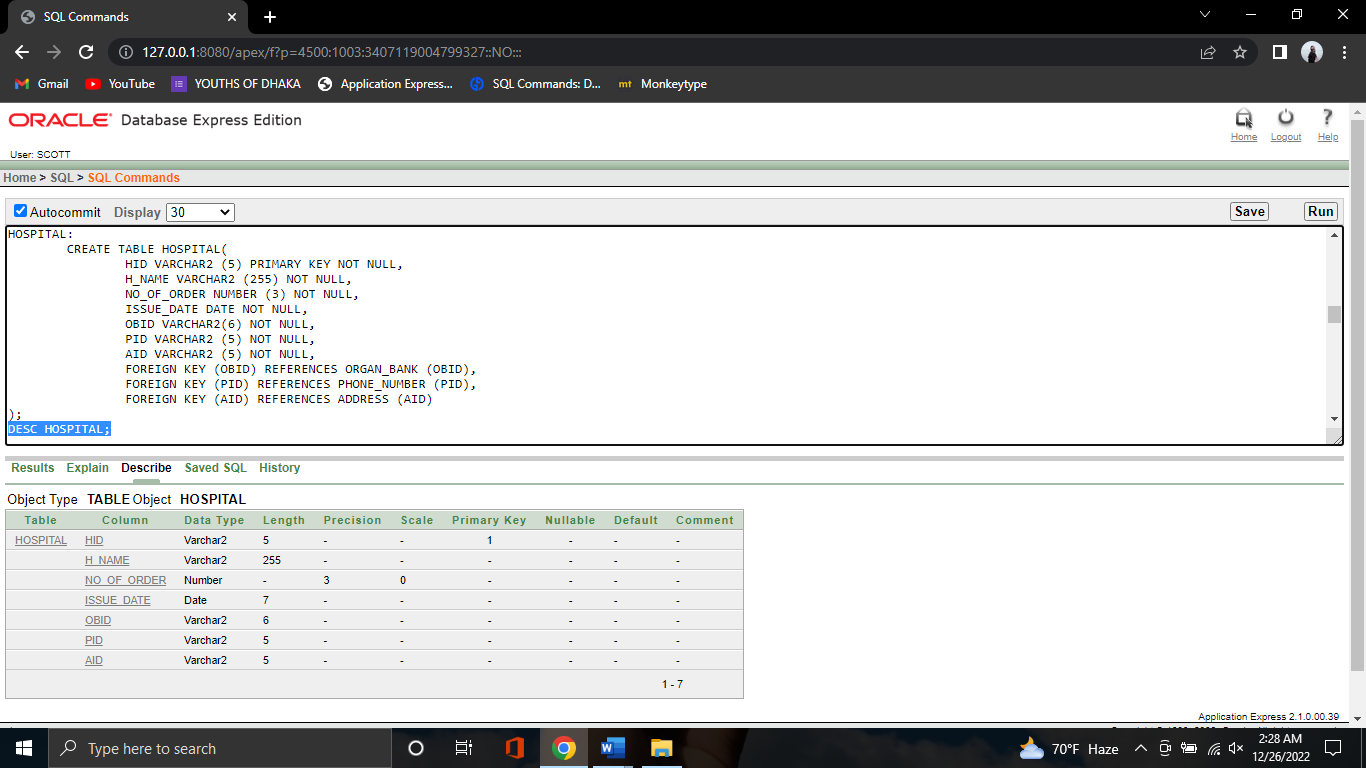




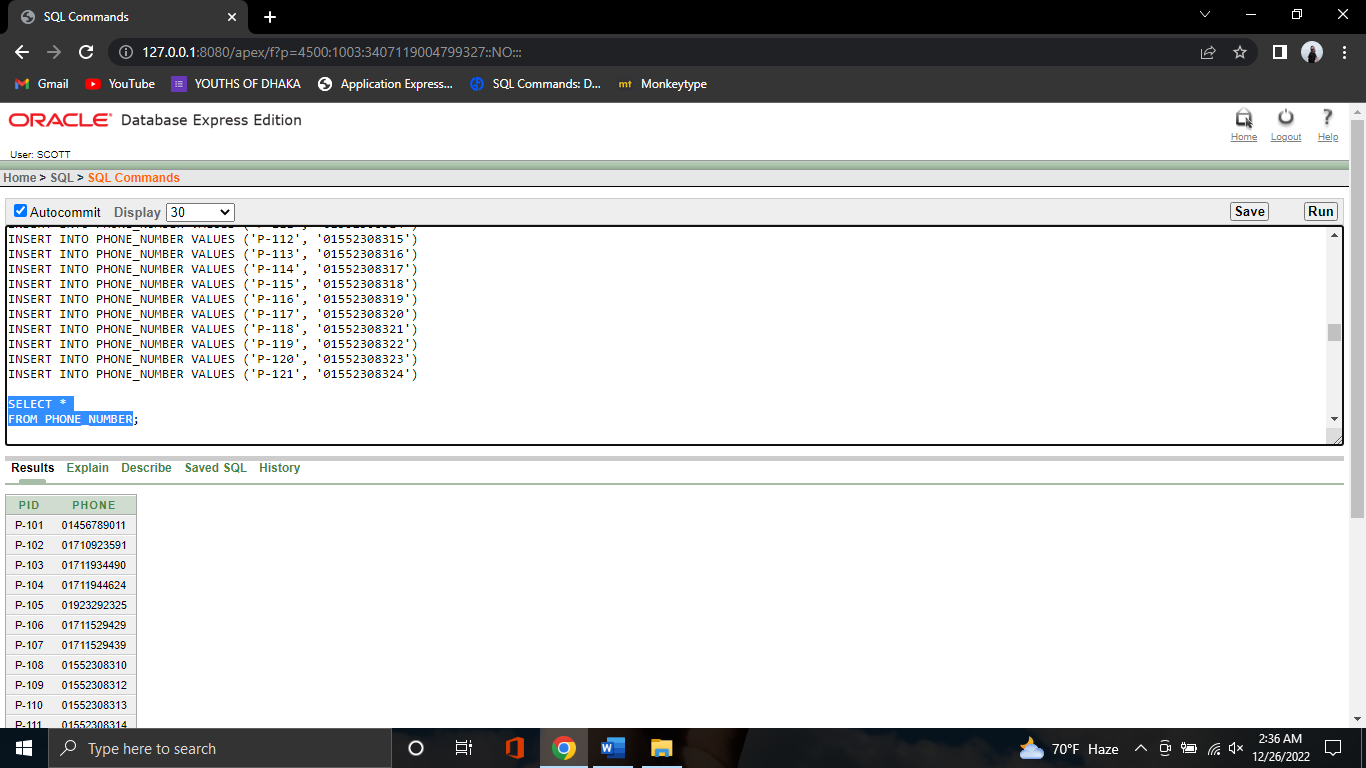


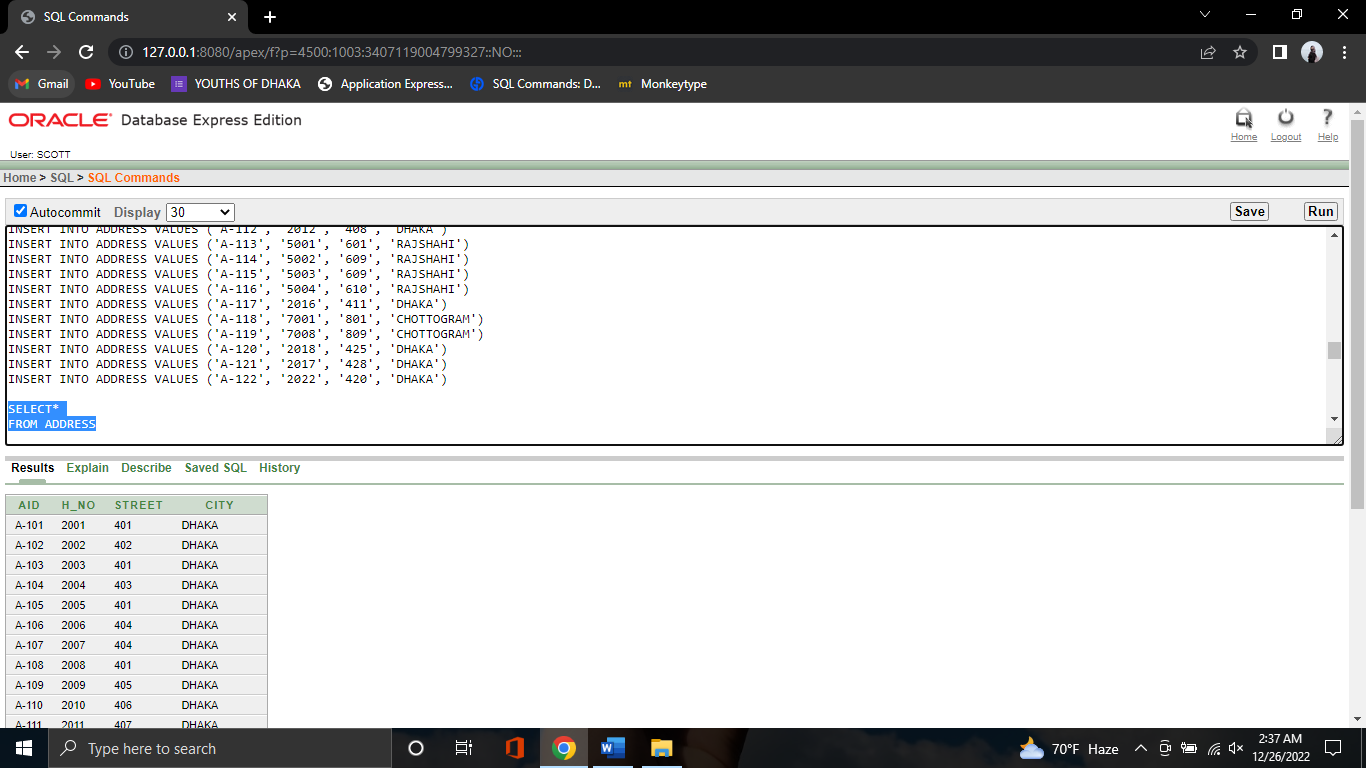


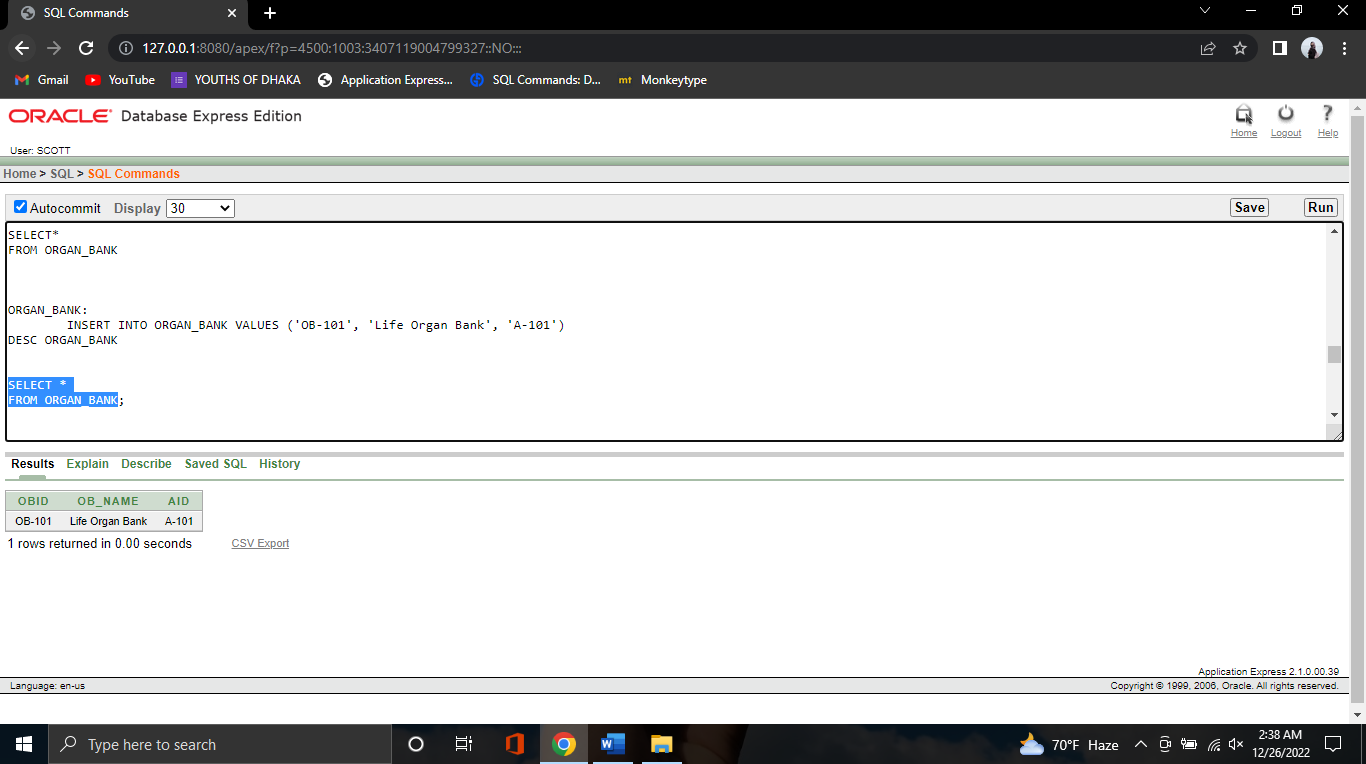


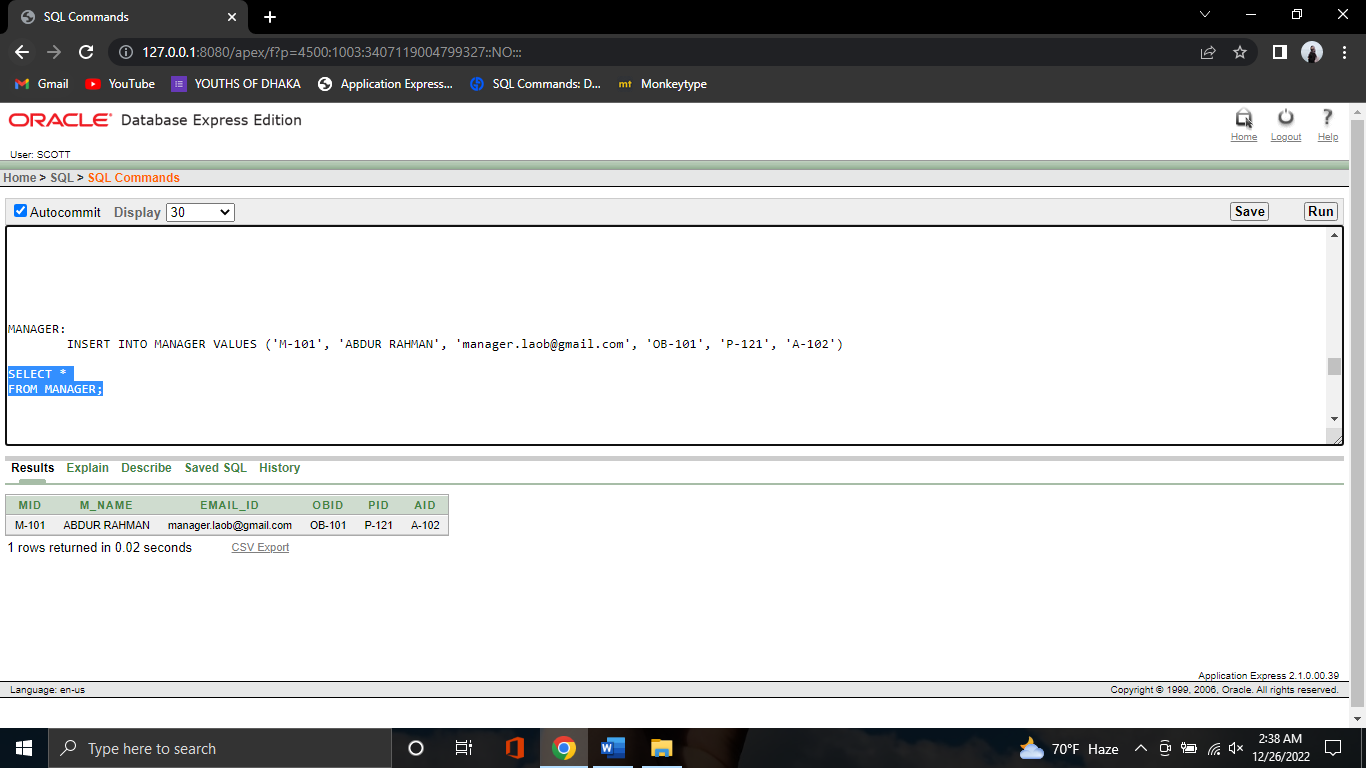


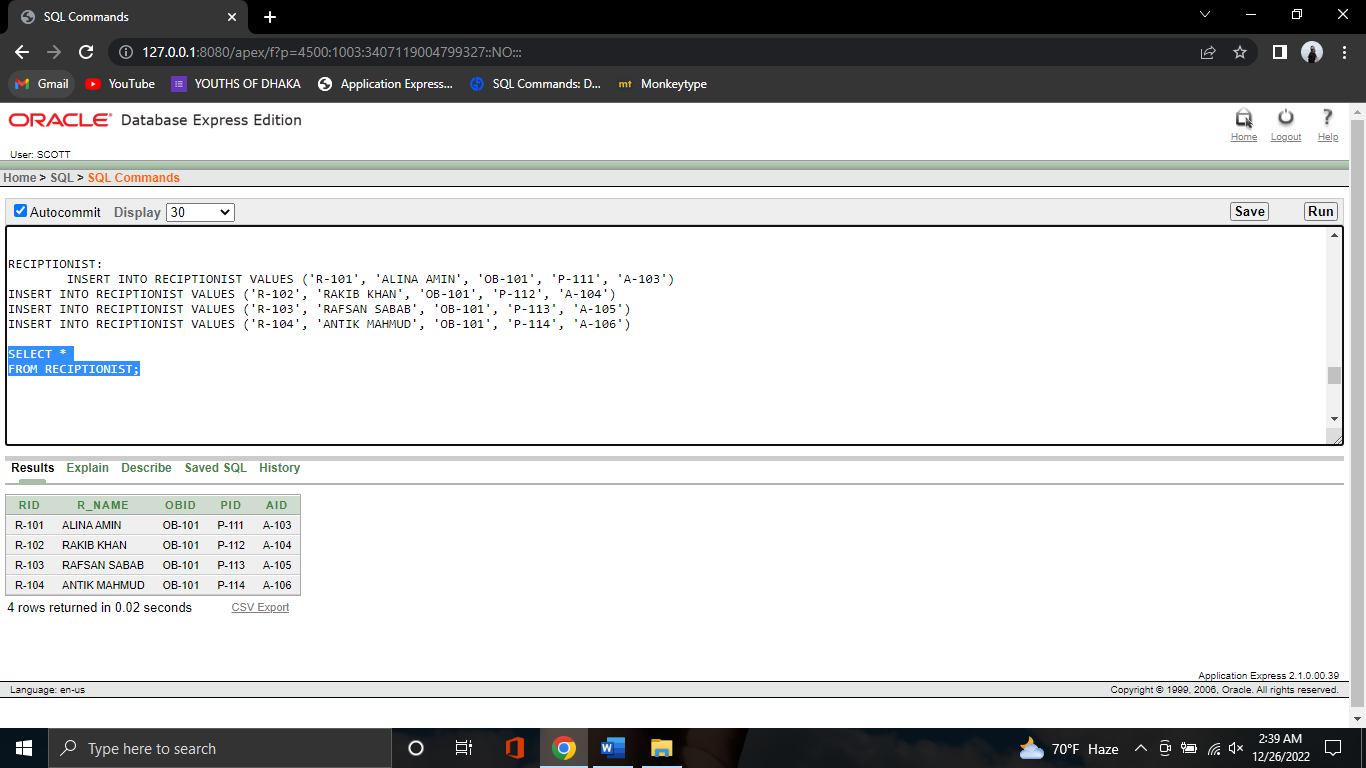
**Data Insertation**

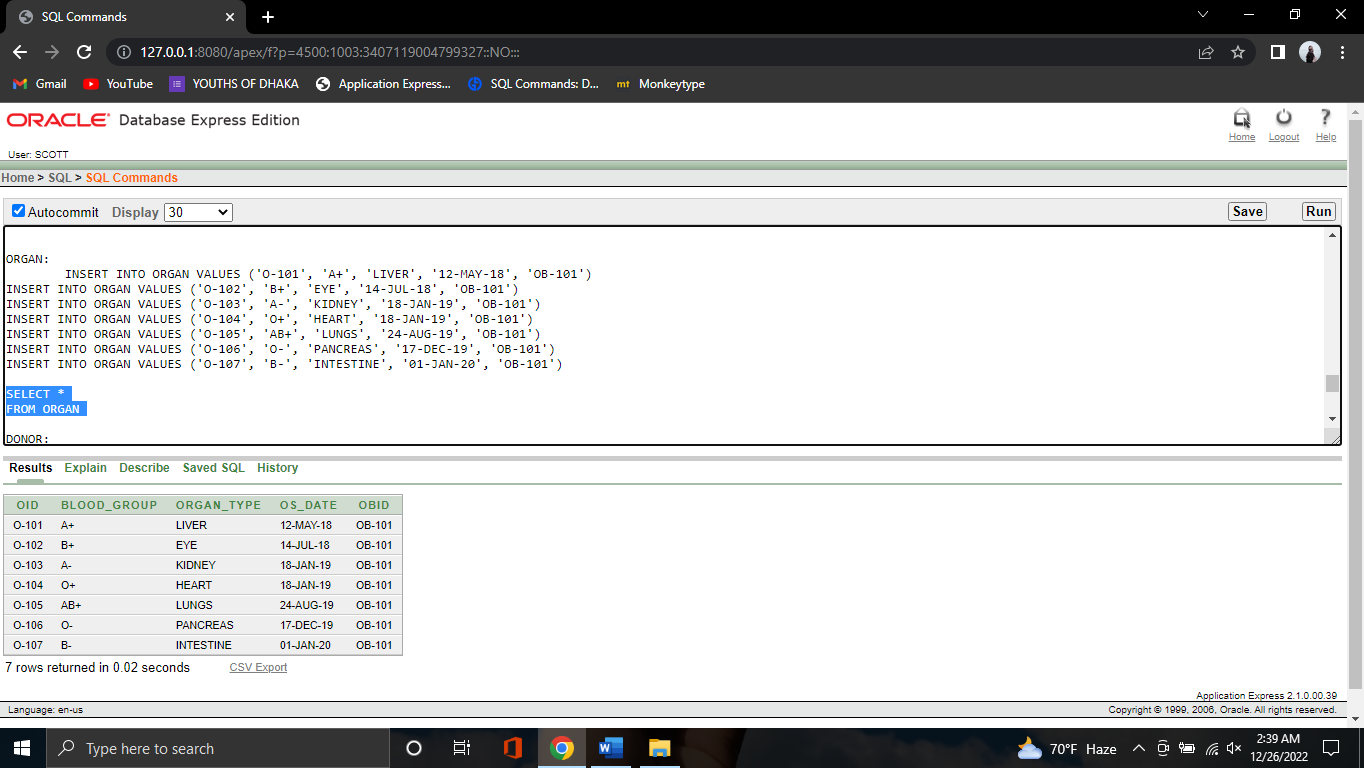


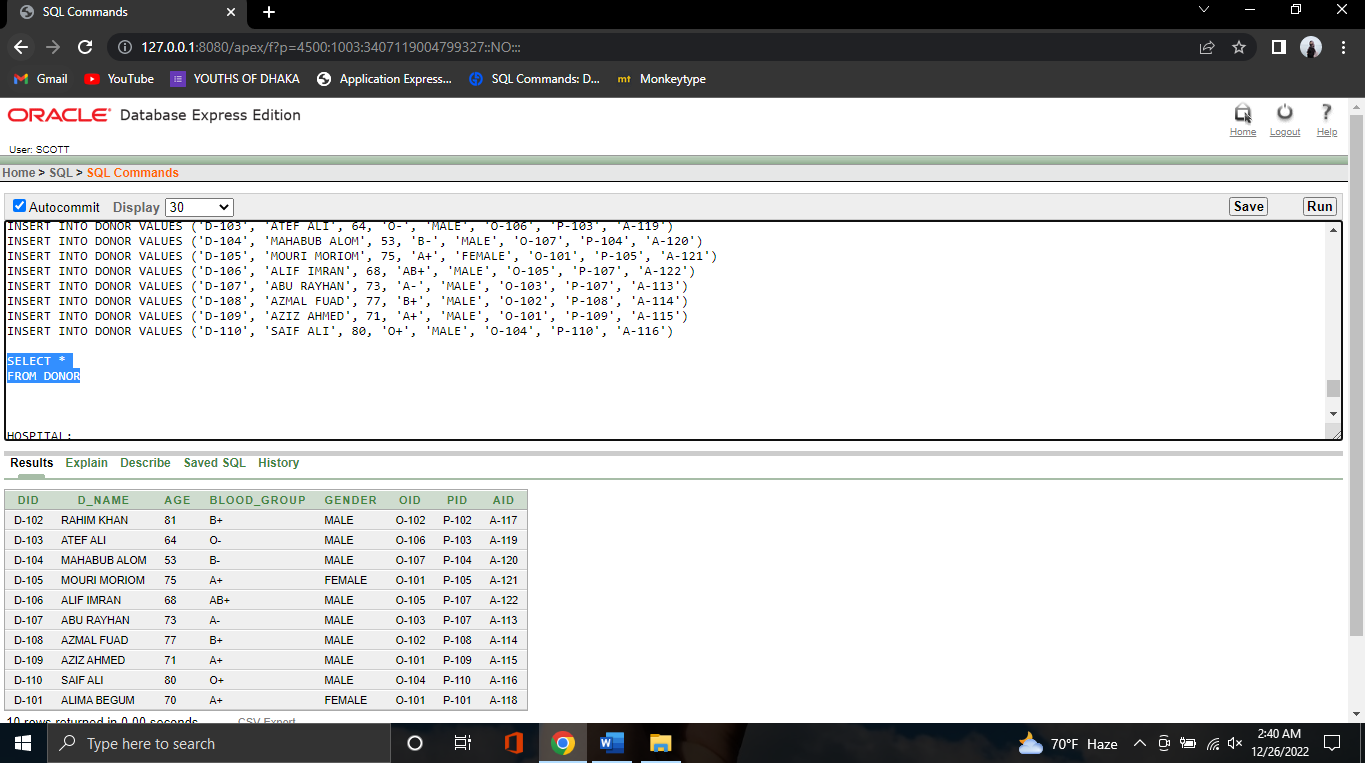


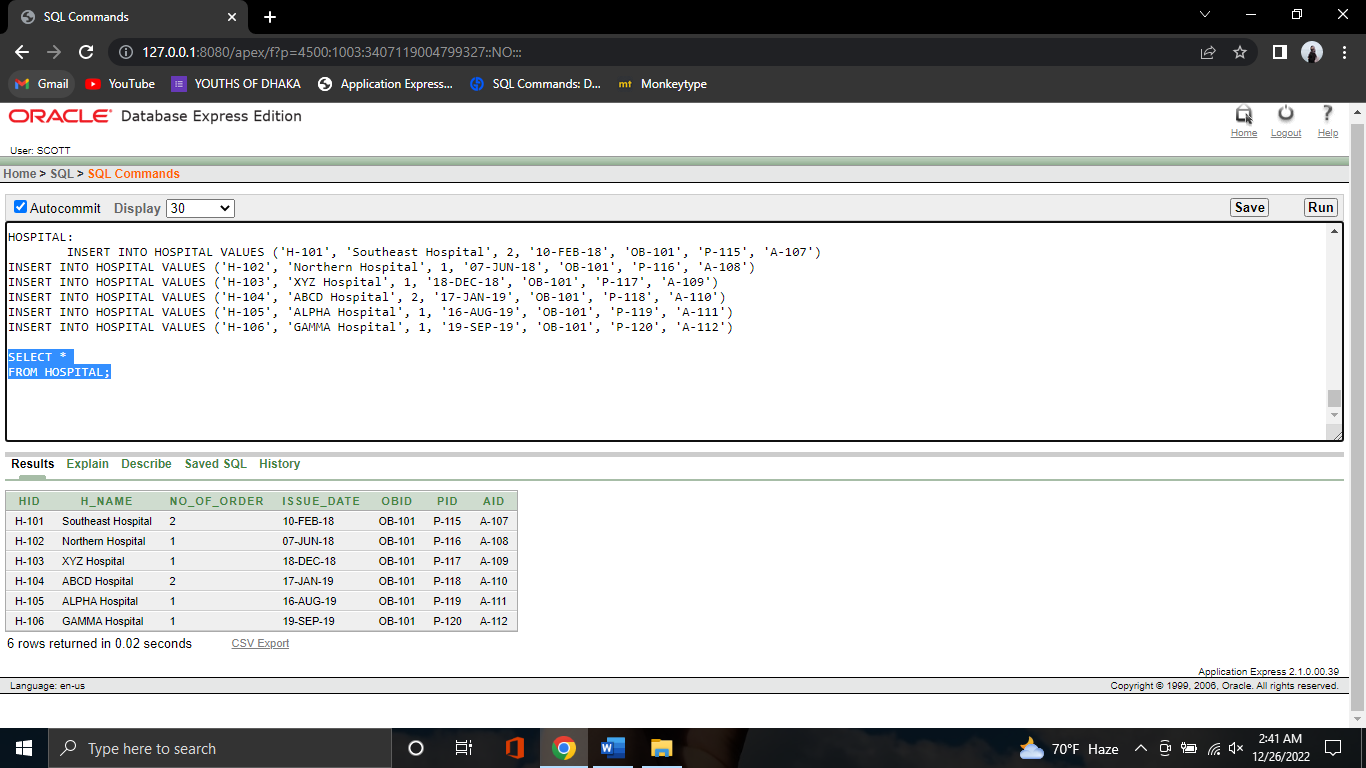












**Query Testing**

**Conclusion**

Our Database project is about “ORGAN DONOR MANAGEMENT SYSTEM” management system. While doing this project we had learnmany things. When we were doing ER Diagram we faced some difficulties in arranging everything. Then comes normalization. It was a big challenge for us because there are too many rows. After completing normalization, we created our all tables. Then while inserting values we faced some difficulties with foreign keys. Then Query part and it was really fun. Overall, we had many new experiences while doing this project. Solving every problem felt like a great achievement for us. After finishing this project, we have a clearer concept in Databases.