**DRIVING SCHOOL**

**MANAGEMENT**

**SYSTEM**

***DONE BY:-***

***LUKKA SHIVA CHARAN***

***IT B***

***1602-19-737-103***

**1**

**ABSTRACT:**

The Driving School Management System is a web- based system that provides a single-stop for registration and maintenance of driving school database of Cherry Driving School (Name of driving school). The database consists of all the details of customer, details of staff allotted to a particular customer, registration status, payment with license or without license, vehicle number and number of class the customer has opted for. We provide online as well as offline registrations whereas classes are not virtual, there will be people who will approach at your specified address and pick you up and train you on a kilometer basis for one particular day. Driving School Management System (DSMS) is important for ensuring the correct operation of driving school and in maintaining the services that run on them. This project has total of 6 tables. It describes how the network is being connected all across the customers and the particular staff. When you enter the data, it is stored in the data base and is displayed whenever you need it. The data can be updated or changed only by the people of the driving school.

**2**

**REQUIREMENT ANALYSIS:**

List of tables:

* Customer\_details
* Vehicle\_time
* Further\_details
* Staff\_details
* Id\_proofs & postal
* Payment\_status

List of attributes with their domain types:

1.Customer\_details:

Cust\_Id: cust\_id number (5)

C\_name: C\_name varchar2 (20)

Date of join: date of join varchar2 (20)

Licence\_type: licence\_type varchar2 (20)

Mobile\_num: Mobile\_num number(20)

Email\_id: email\_id varchar2 (20)

Address: Address varchar2 (20)

**3**

2.Vehicle\_time:

Cust\_name: cust\_name varchar2(20)

Period: period varchar2(15)

Time of attending: time of attending varchar2 (20)

Vehcile\_type: vehicle\_type varchar2 (20)

Vehcile name: vehicle name varchar2 (20)

3.Further details:

Customer\_id: customer\_id number (5)

Course days: course days number (2)

Licence\_required: Licence\_required varchar2 (20)

Pickup Address: pickup address varchar2 (20)

4.Staff details:

Staff\_id: staff\_id number (5)

Person\_Name: Person\_name varchar2 (20)

Mobile\_number: mobile\_number number (15)

Email: email varchar2 (30)

Staff\_add: Staff\_add varchar2 (30)

5.Id\_proofs& postal:

Person\_id: Person\_id number (5)

Name: name varchar2 (20)

Aadhar No: aadhar number (16)

Tenth\_memo: tenth memo varchar (20)

Postal\_Address: postal address varchar2 (20) **4**

6. Payment Status:

id: id number (5)

Customer\_Name: customer\_name varchar2 (20)

Total Money: total money number (5);

Money Paid: money paid number (5);

**5**

**MAPPING CARDINALITY AND PARTICIATION**

* **CONSTRAINTS**

The internet is connected to server. So, every servers connected to the internet, which is one to many participations. In this you can only opt for a one particular vehicle type at a time and the time of attending to driving classes cannot be changed once you are done with filling the form. You will be allotted one staff member who will pick you and drop you depending upon the pickup location. Your Id proofs are must in order to opt for a driving licence. You will receive licence only when your payment is completely done. For the driving registration you have two means i.e online and offline. Each computer will have only one operating system which is one to one participation.

**6**

ARCHITECTURE AND TECHNOLOGY

*Software used:*

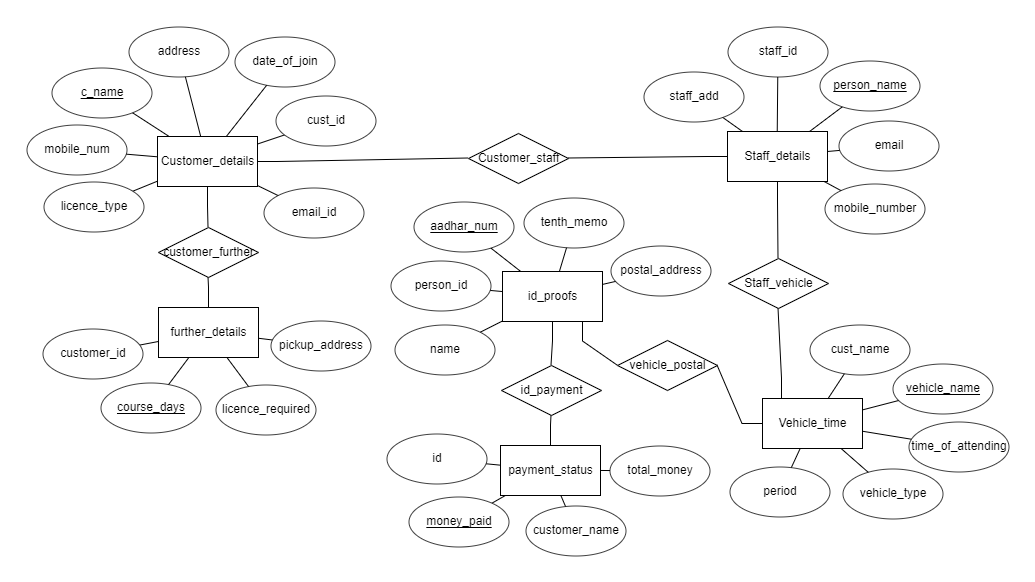
Java Eclipse, Oracle 11g Database,Java SE version 13, SQL\*Plus.Java

**AWT:**

Java AWT (Abstract Window Toolkit) is an API to develop GUI or window-based applications in java.Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavyweight i.e. its components are using the resources of OS.The java.awt package provides classes for AWT API such as TextField, Label, TextArea,RadioButton, CheckBox, Choice, List etc.

**SQL:**Structure Query Language(SQL) is a database query language used for storing and managing data in Relational DBMS. SQL was the first commercial language introduced for E.F Codd's Relational model of database. Today almost all RDBMS (MySql, Oracle, Infomix, Sybase, MS Access) use SQL as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

**7**



**8**

**DDL COMMANDS:**

1. create table **customer\_details** (cust\_id number (5), c\_name varchar2(20) primary key, date\_of\_join varchar2(20), mobile\_num number (20), email\_id varchar2(30), address varchar2(30));
2. create table **vehicle\_time** (cust\_name varchar2(20), period varchar (15), time\_of\_attending varchar2(30), vehicle\_type varchar2(20), vehicle\_name varchar2(20) primary key);
3. create table **further\_details** (customer\_id number (5), course\_days number (2) primary key, licence\_required varchar2(20), pickup\_address varchar2(30));
4. create table **staff\_details** (staff\_id number (5), person\_name varchar2(20) primary key, mobile\_number number (15), email varchar2(30), staff\_add varchar2(30));
5. create table **id proofs** (person\_id number (5), name varchar2(20), aadhar\_num number (16) primary key, tenth\_memo varchar2(20), postal\_address varchar2(30));
6. create table **payment\_status** (id number (5), customer\_name varchar2(20), total\_money number (5), money\_paid number (5) primary key);
7. create table **customer\_staff**(c\_name varchar2(20), person\_name varchar2(20), foreign key(c\_name) references customer\_details(c\_name), foreign key(person\_name) references staff\_details(person\_name));
8. create table **staff\_vehicle** (person\_name varchar2(20), vehicle\_name varchar2(20), foreign key(person\_name) references staff\_details(person\_name), foreign key(vehicle\_name) references vehicle\_time(vehicle\_name));
9. create table **vehicle\_postal** (vehicle\_name varchar2(20), aadhar\_num number (16), foreign key(vehicle\_name) references vehicle\_time(vehicle\_name), foreign key(aadhar\_num)

**9**

references id\_proofs(aadhar\_num));

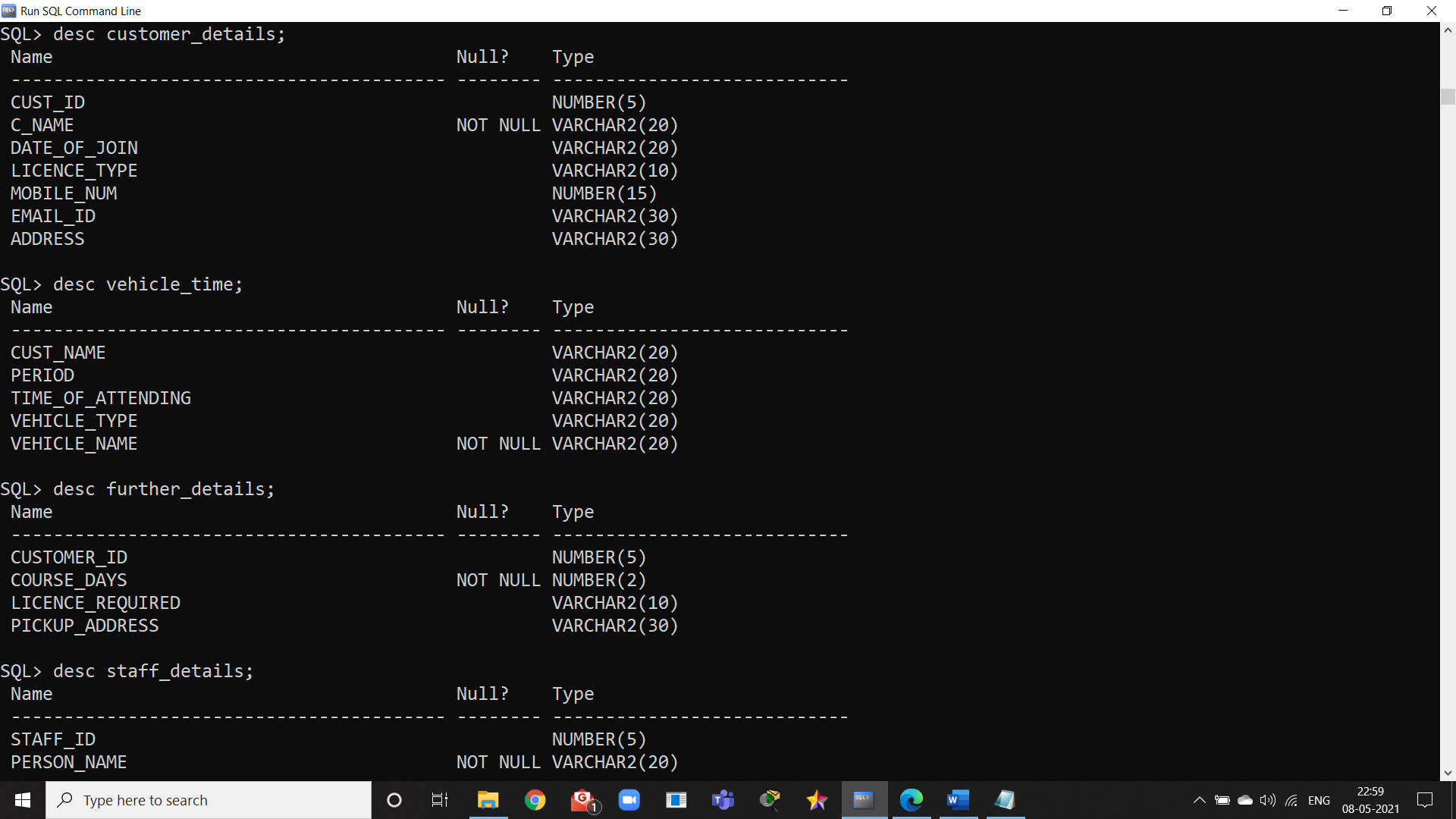
1. create table **id\_payment** (aadhar\_num number (16), money\_paid number (5), foreign key(aadhar\_num) references id\_proofs(aadhar\_num), foreign key(money\_paid) references payment\_status(money\_paid));
2. create table **customer\_further** (c\_name varchar2(20), course\_days number (2), foreign key(c\_name) references customer\_details(c\_name), foreign key(course\_days) references further\_details(course\_days));

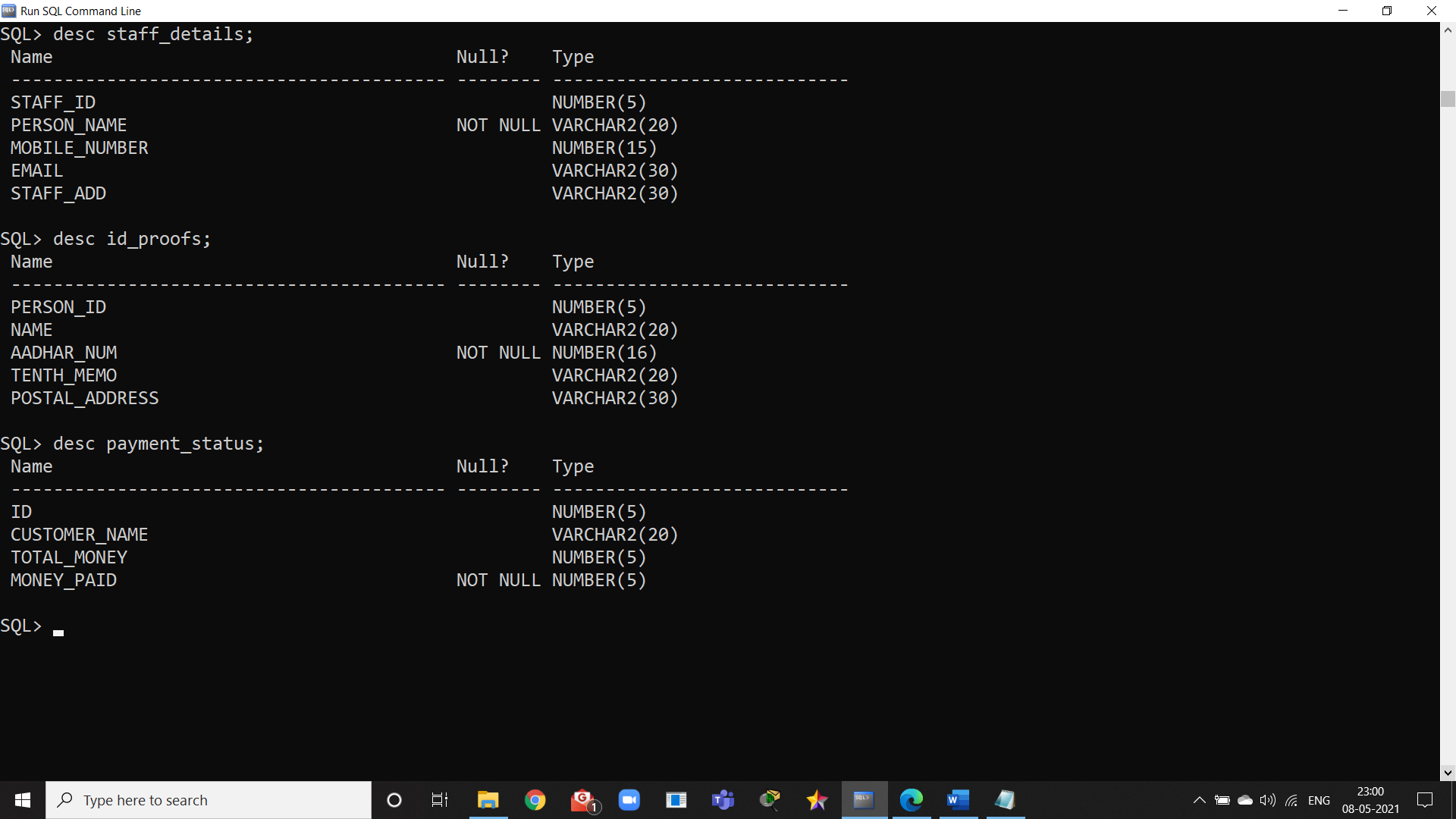
DML COMMANDS

1. insert into customer\_details values(&cust\_id,'&c\_name','&date\_of\_join','licence\_type',&mobile\_num,'&email\_id','&address');
2. insert into staff\_details values(&staff\_id,'&person\_name',&mobile\_number,'&email','&staff\_add');
3. insert into vehicle\_time values('&cust\_name','&period','&time\_of\_attending','&vehicle\_type','&vehicle\_name');
4. insert into id\_proofs values(&person\_id, ‘person\_name', &aadhar\_num, '&tenth\_memo', '&postal\_addess');
5. insert into payment\_status values (&id, '&customer\_name', &total\_money, &money\_paid);
6. insert into further\_details values (&customer id, &course\_days, '&licence\_required', '&pickup\_address');

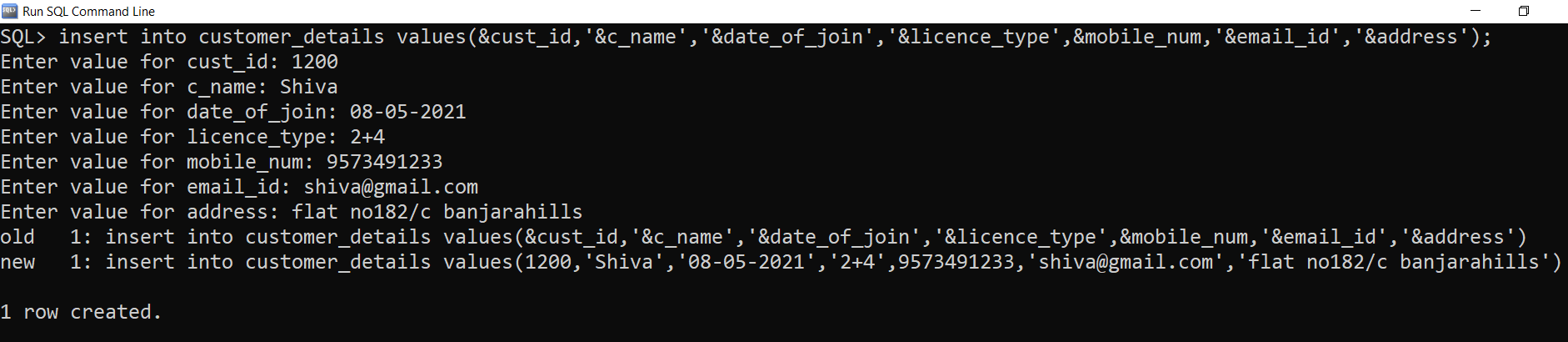
**10**

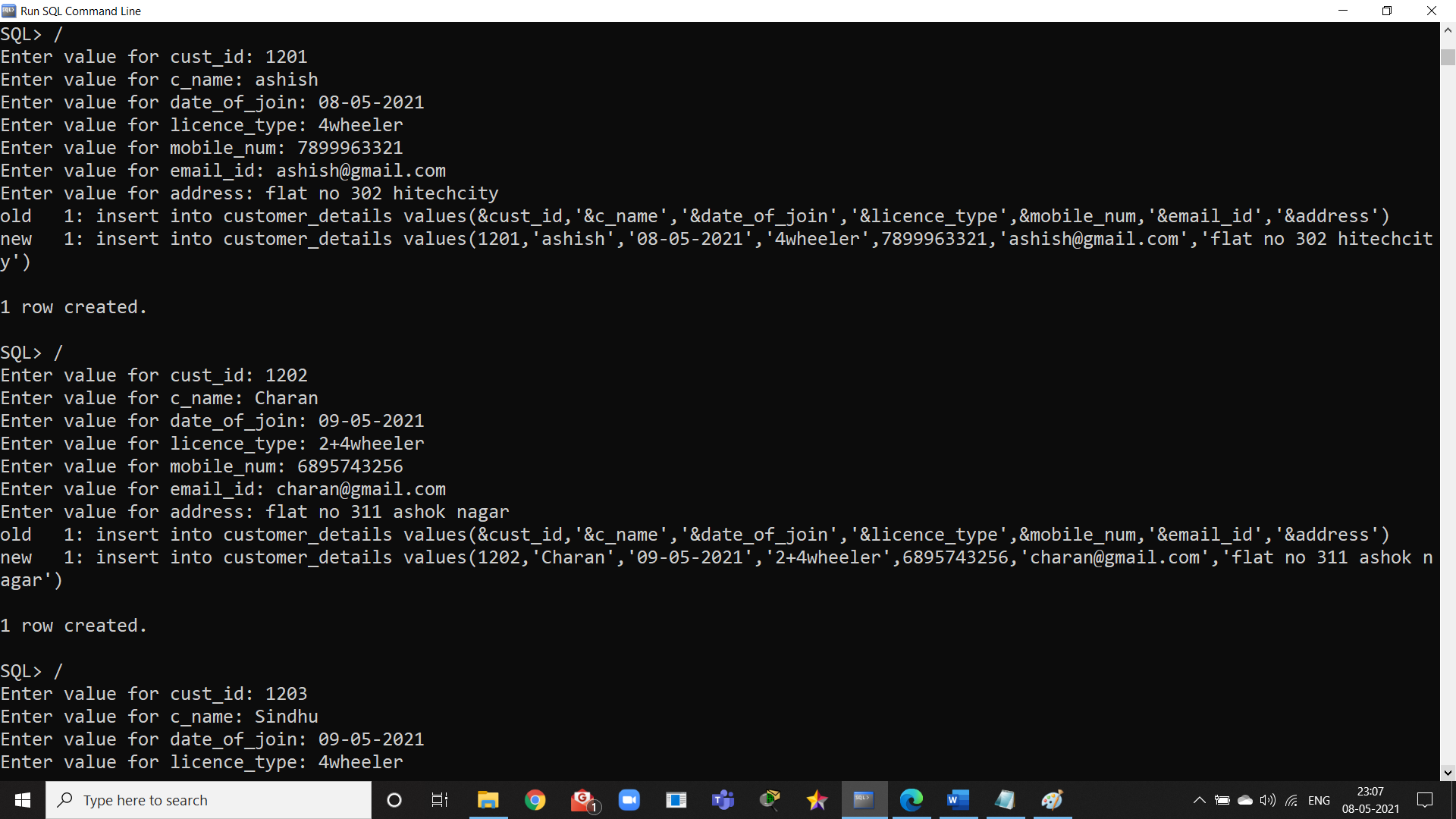
**OUTPUTS:**



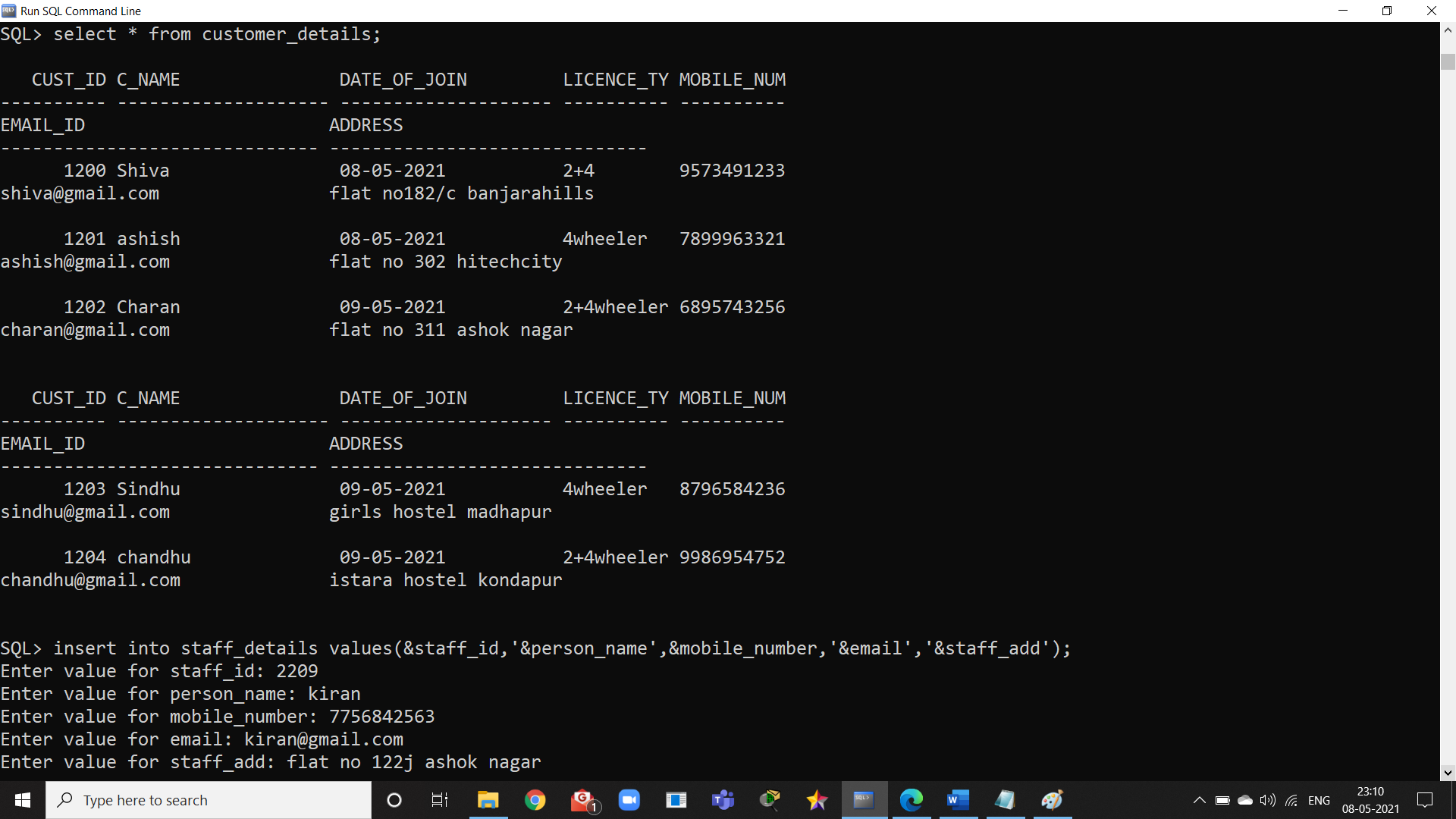


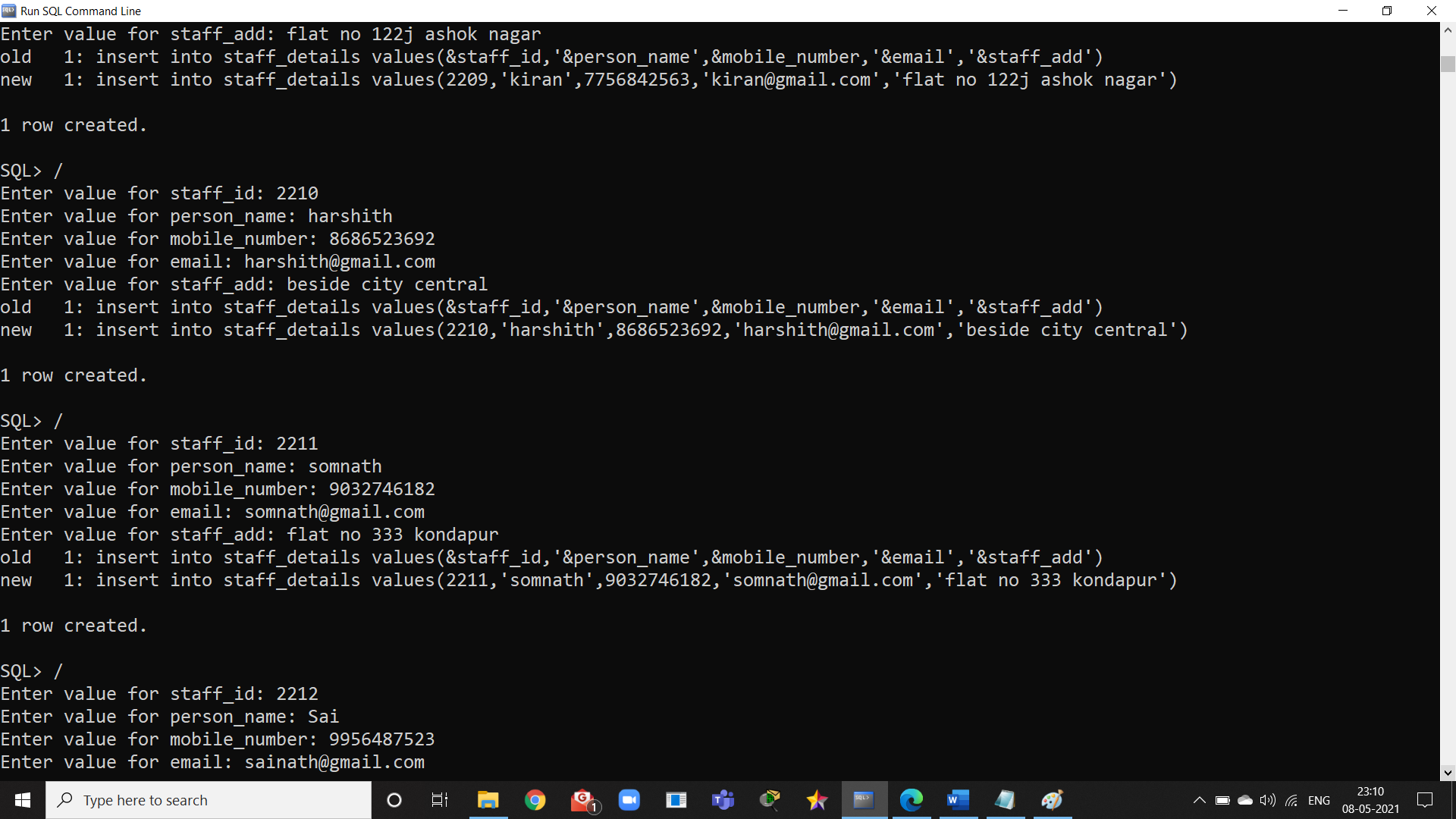
**11**

****

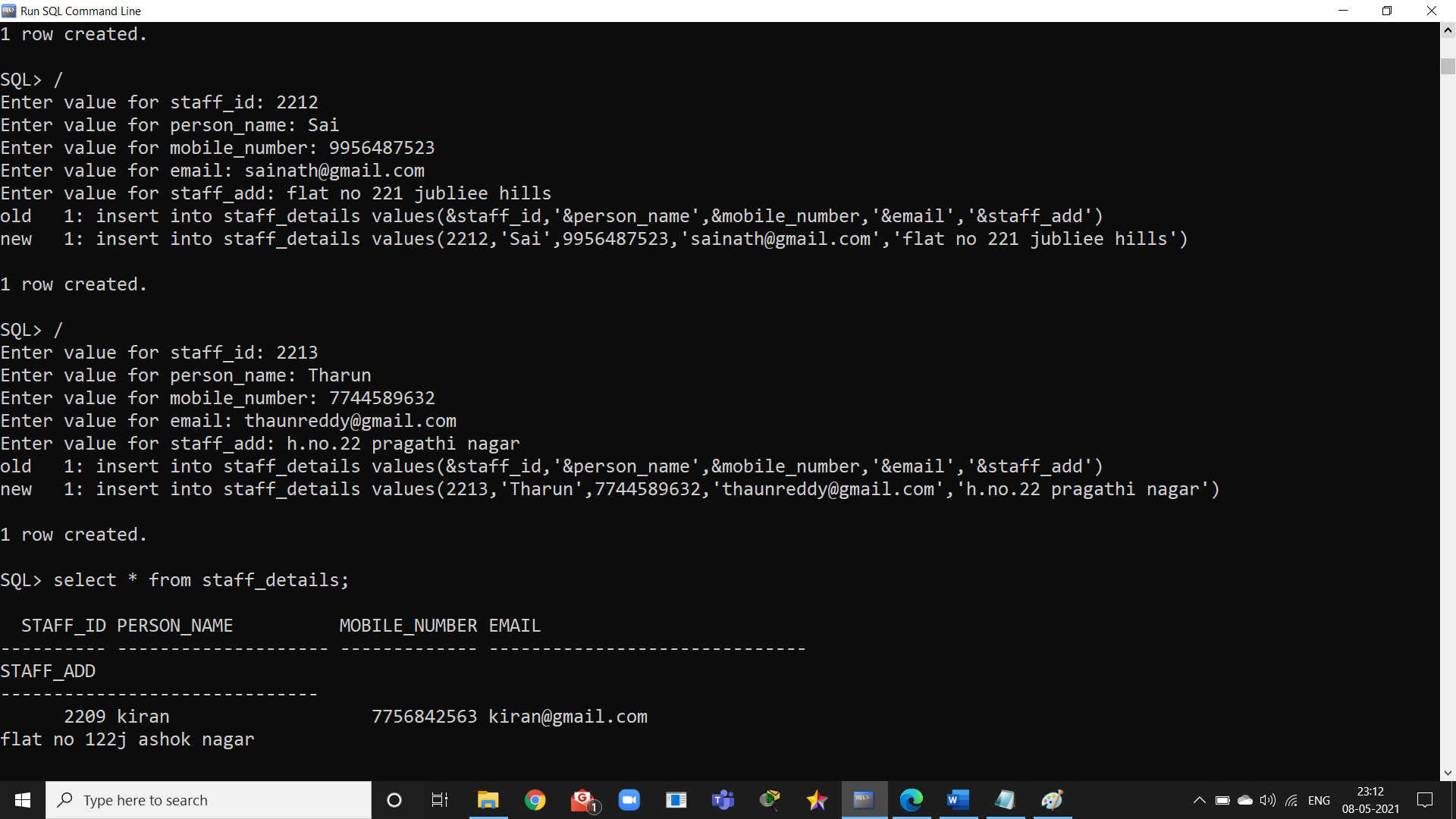


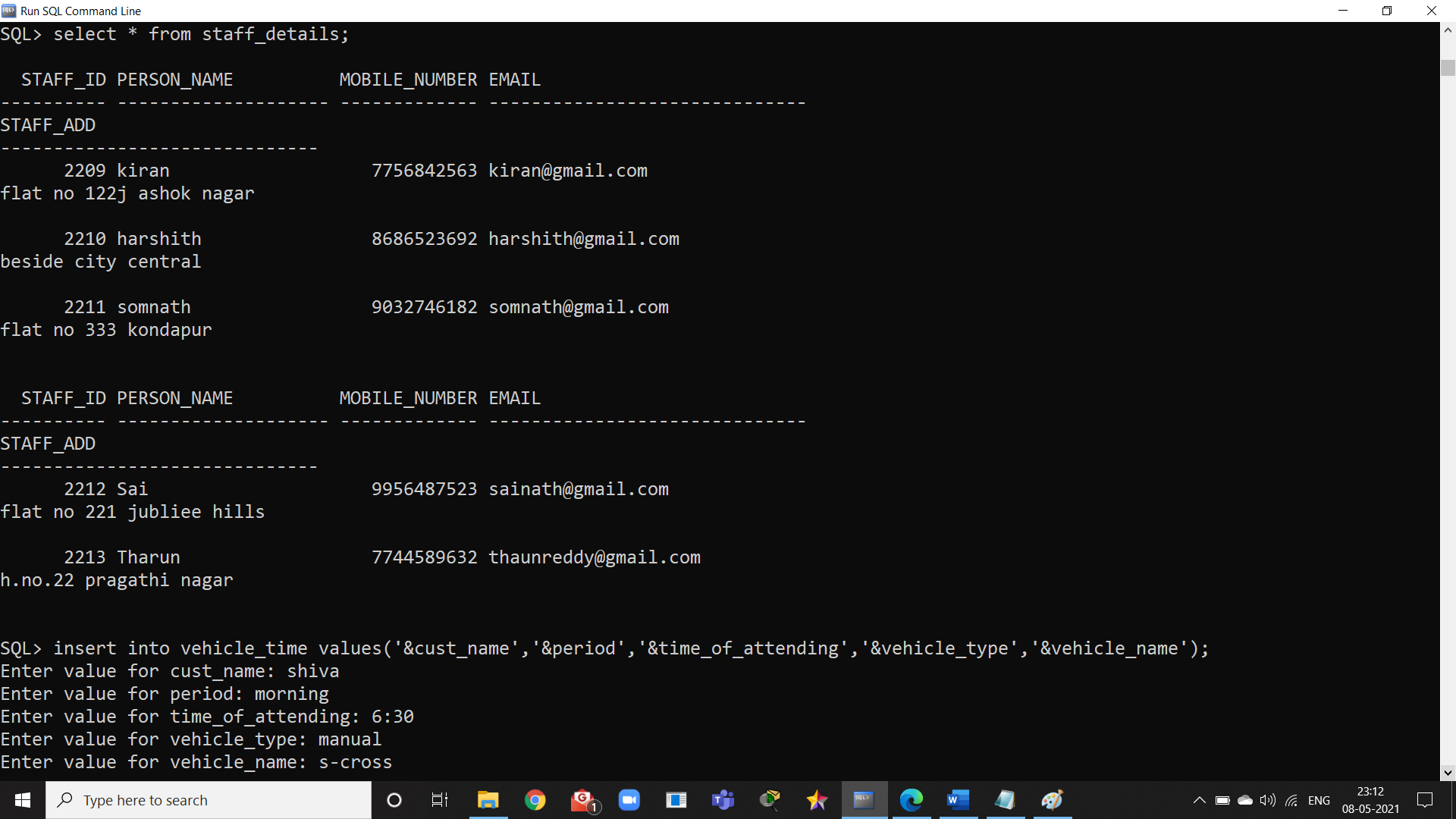
**12**



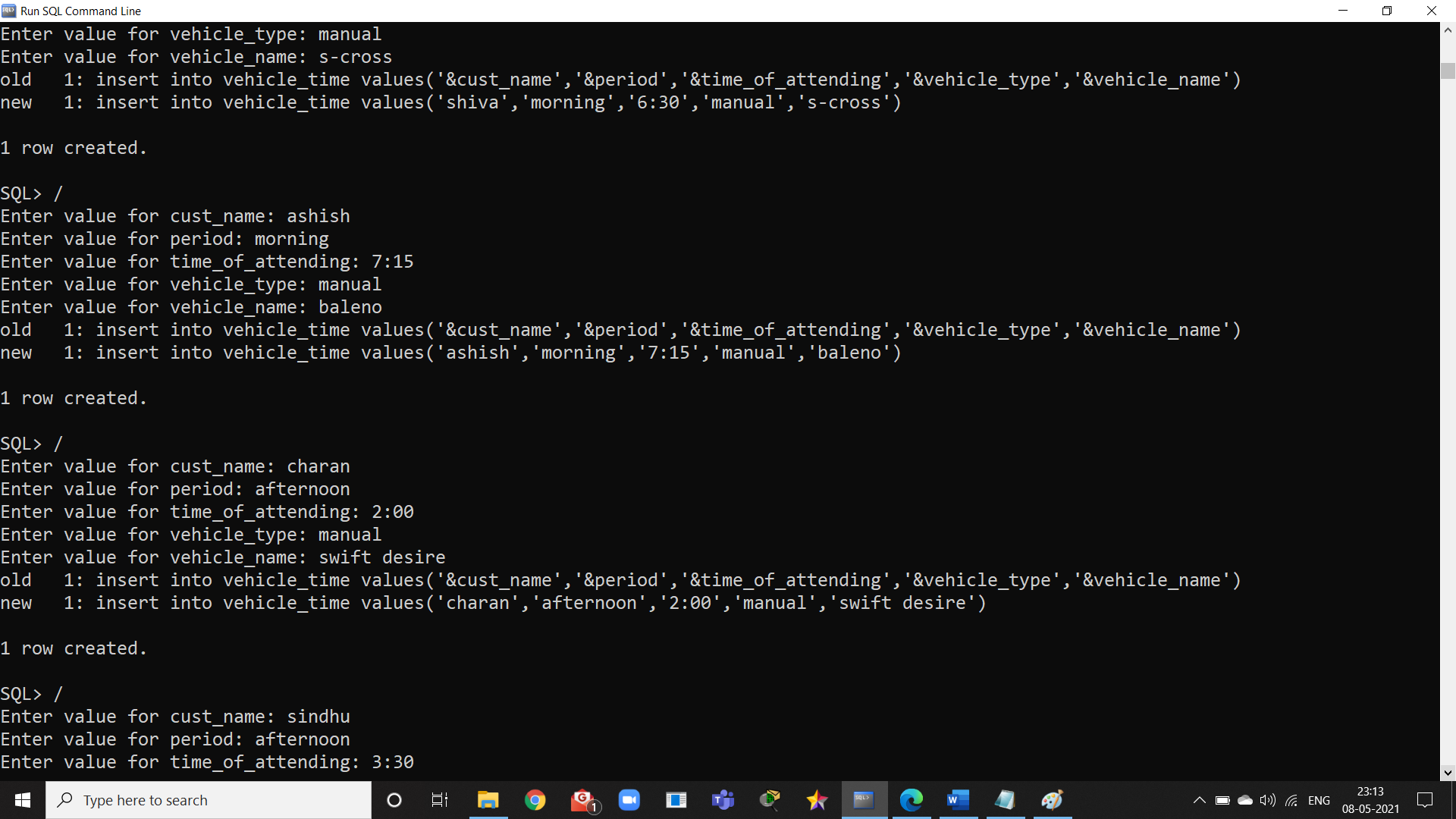


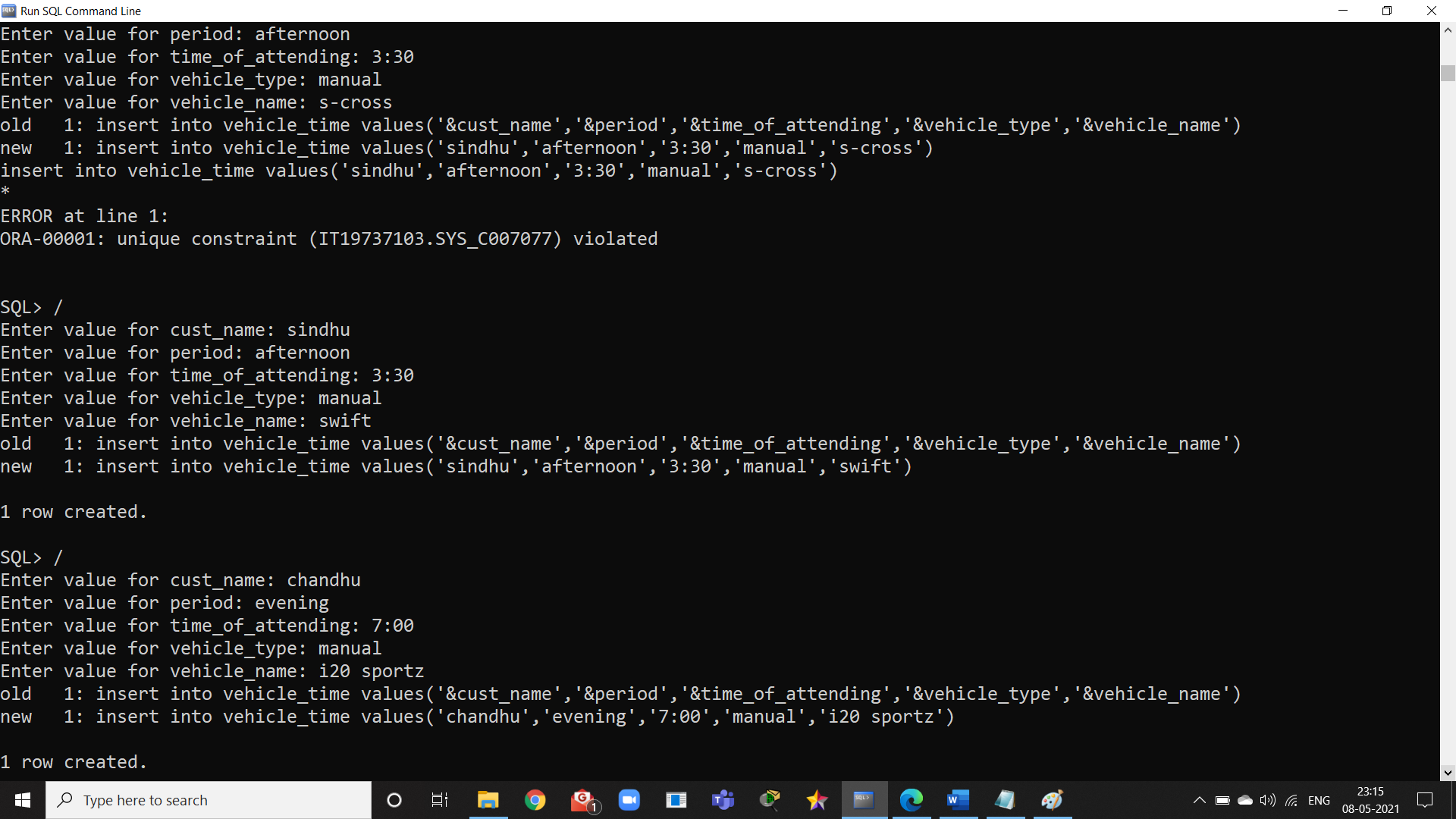
**13**



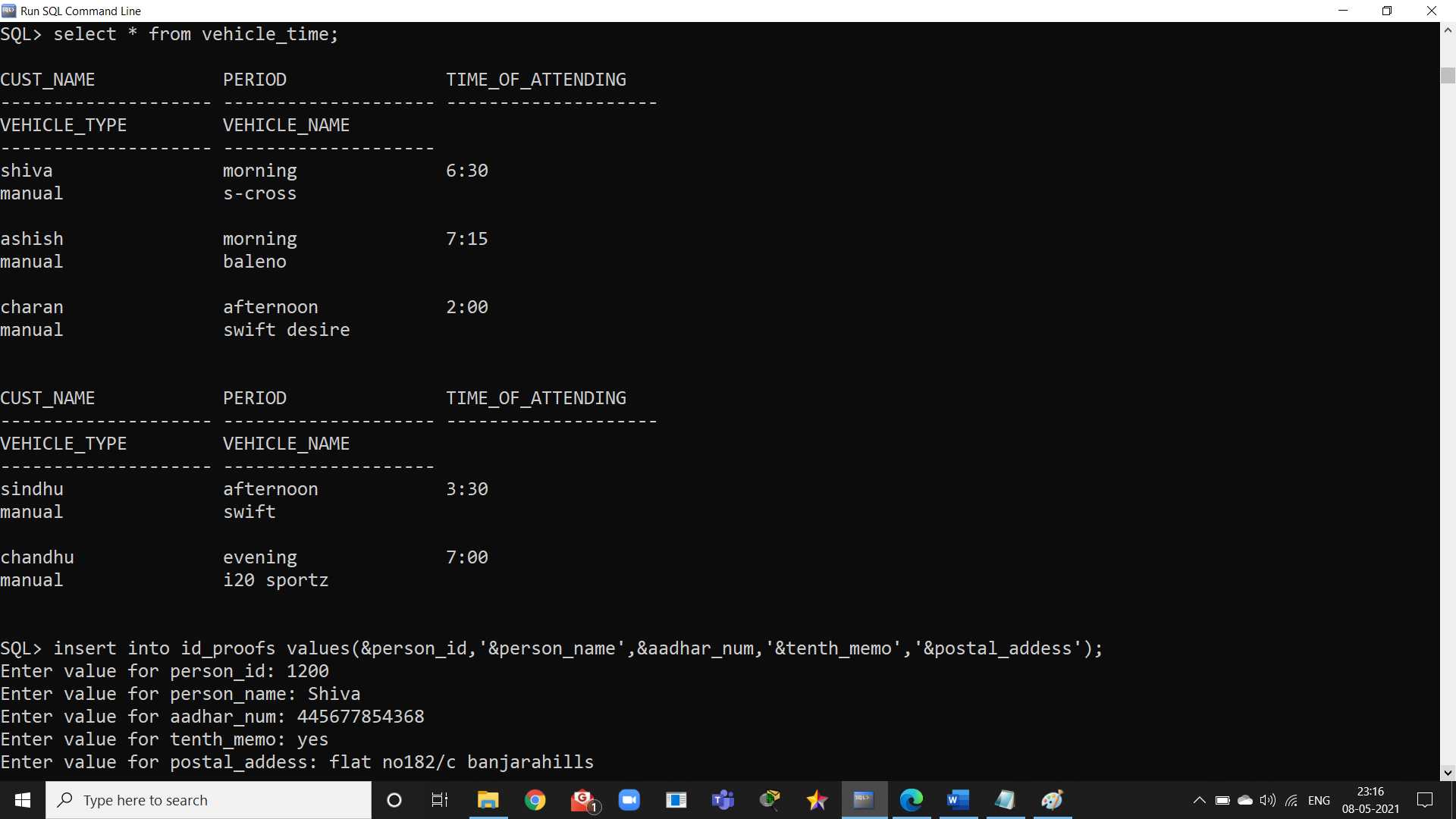


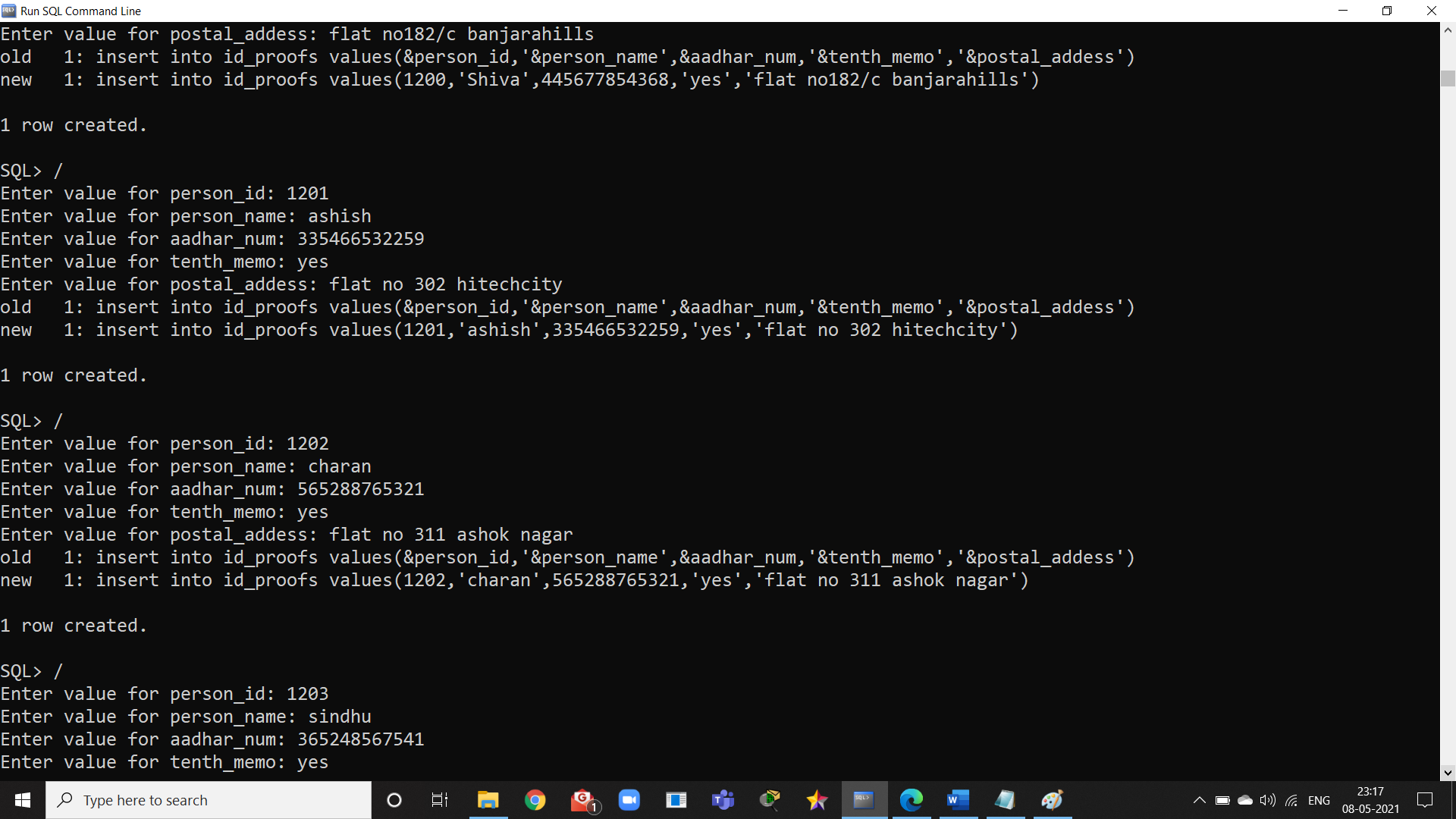
**14**



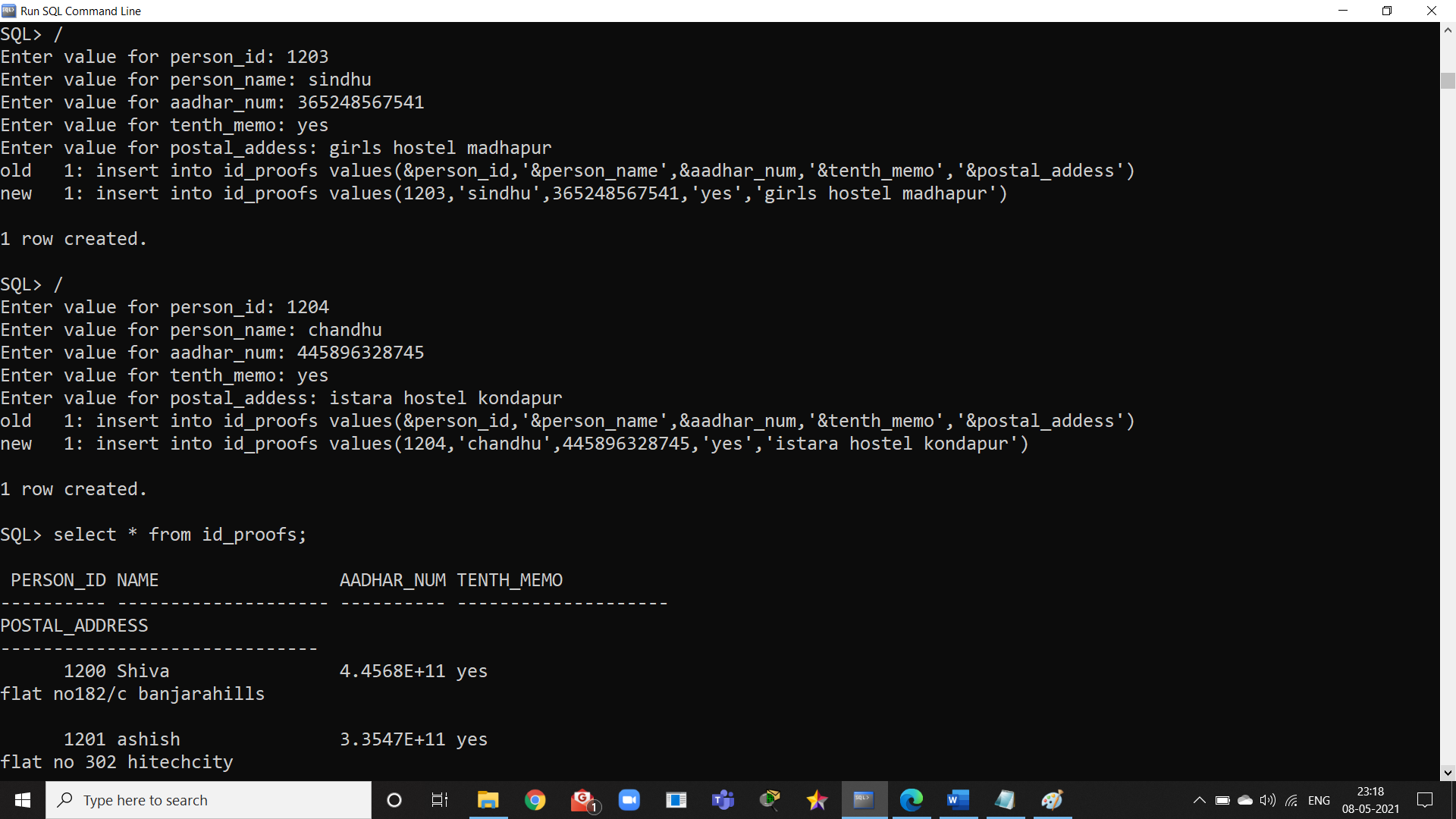


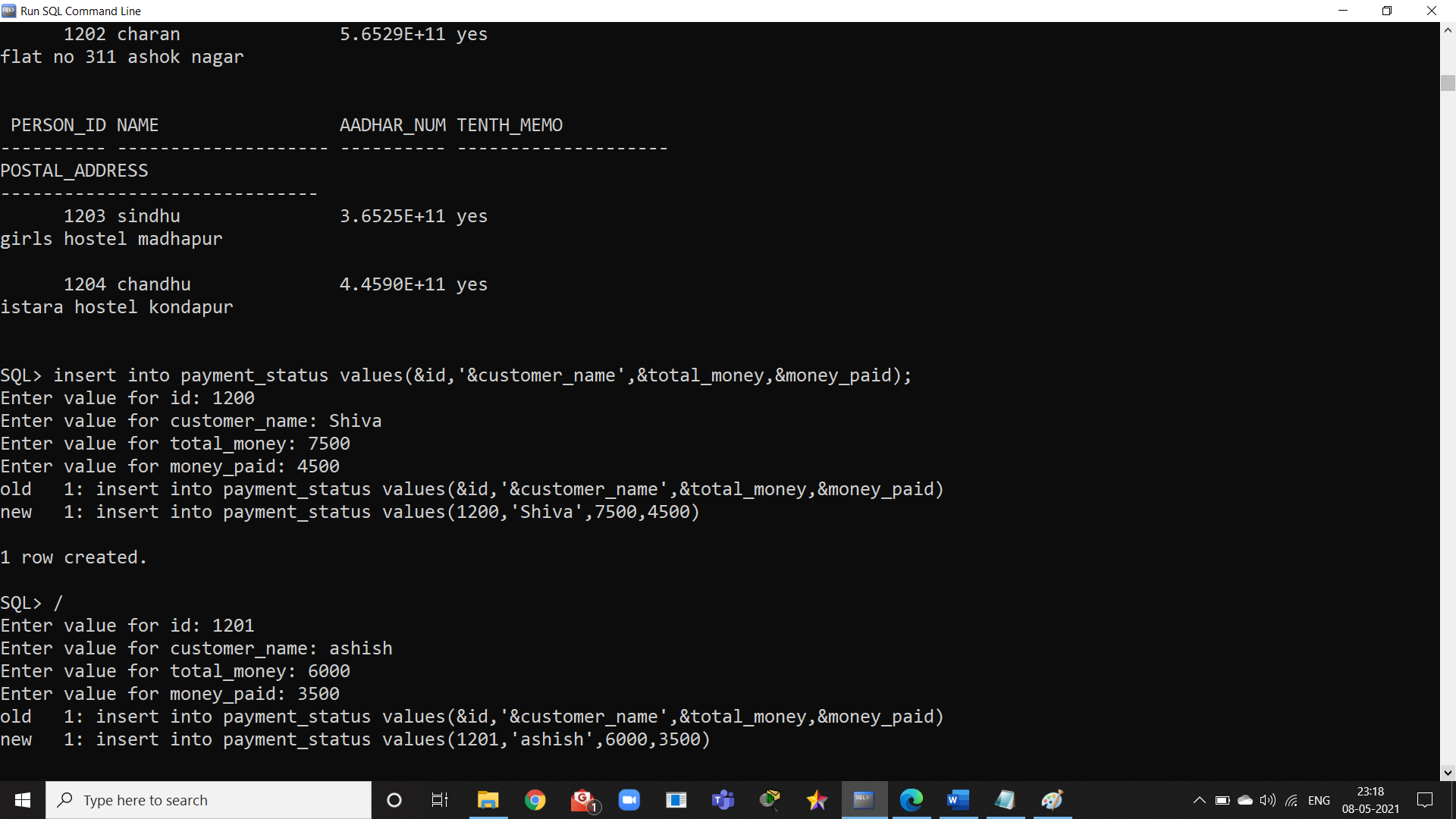
**15**



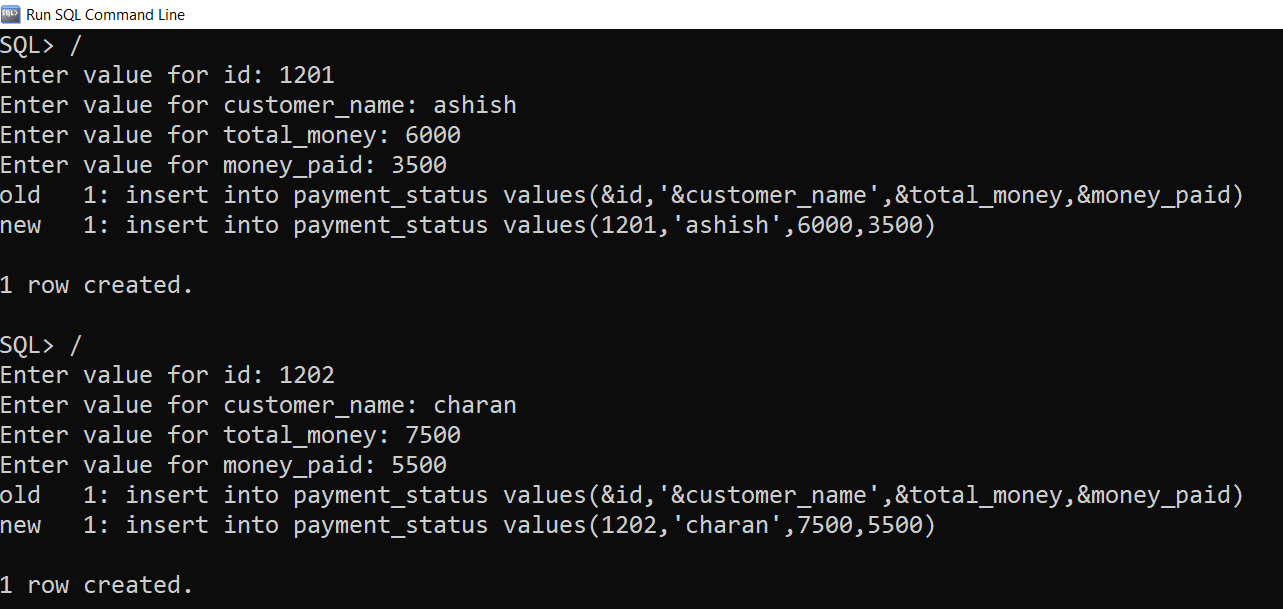


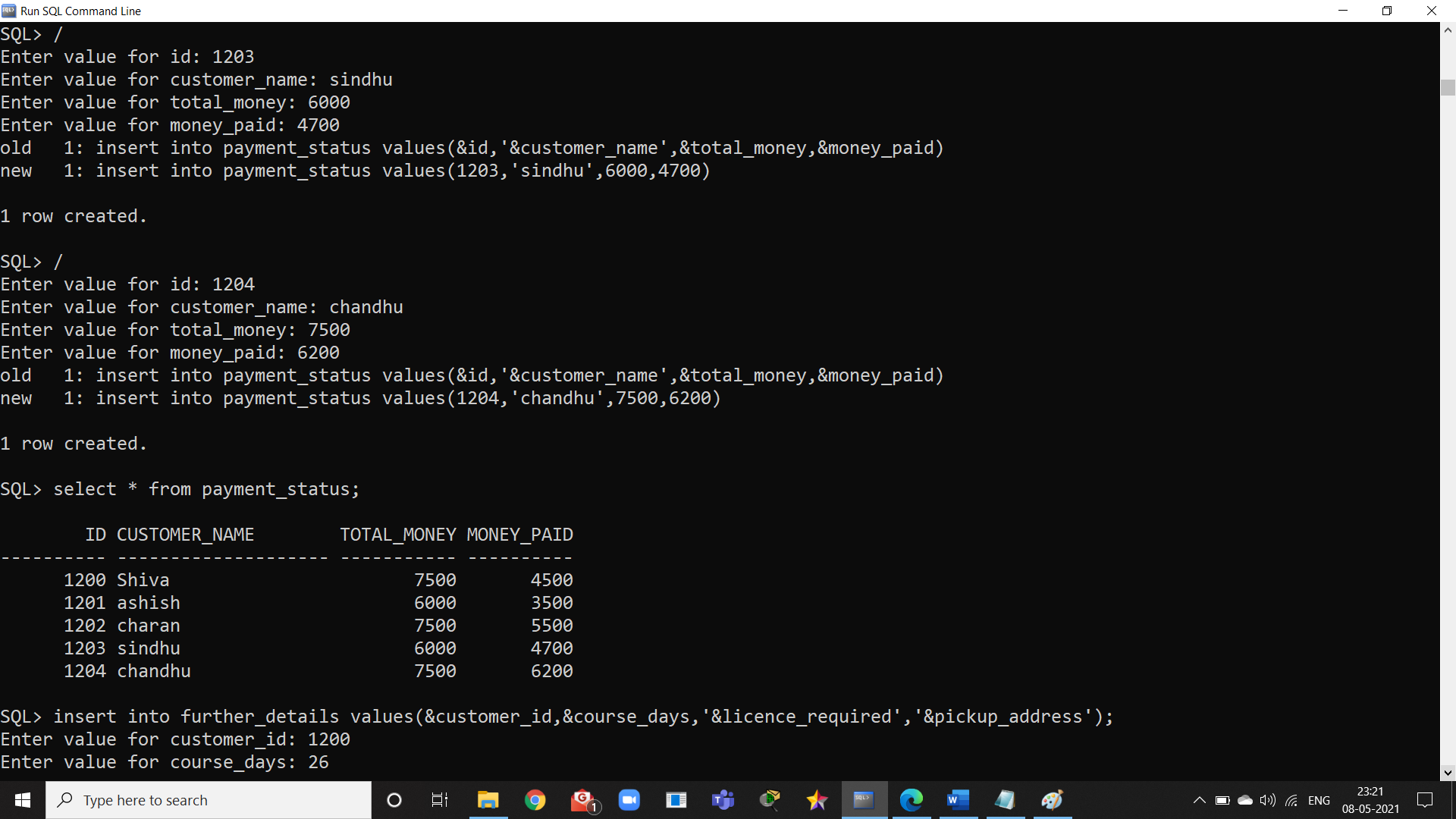
**16**



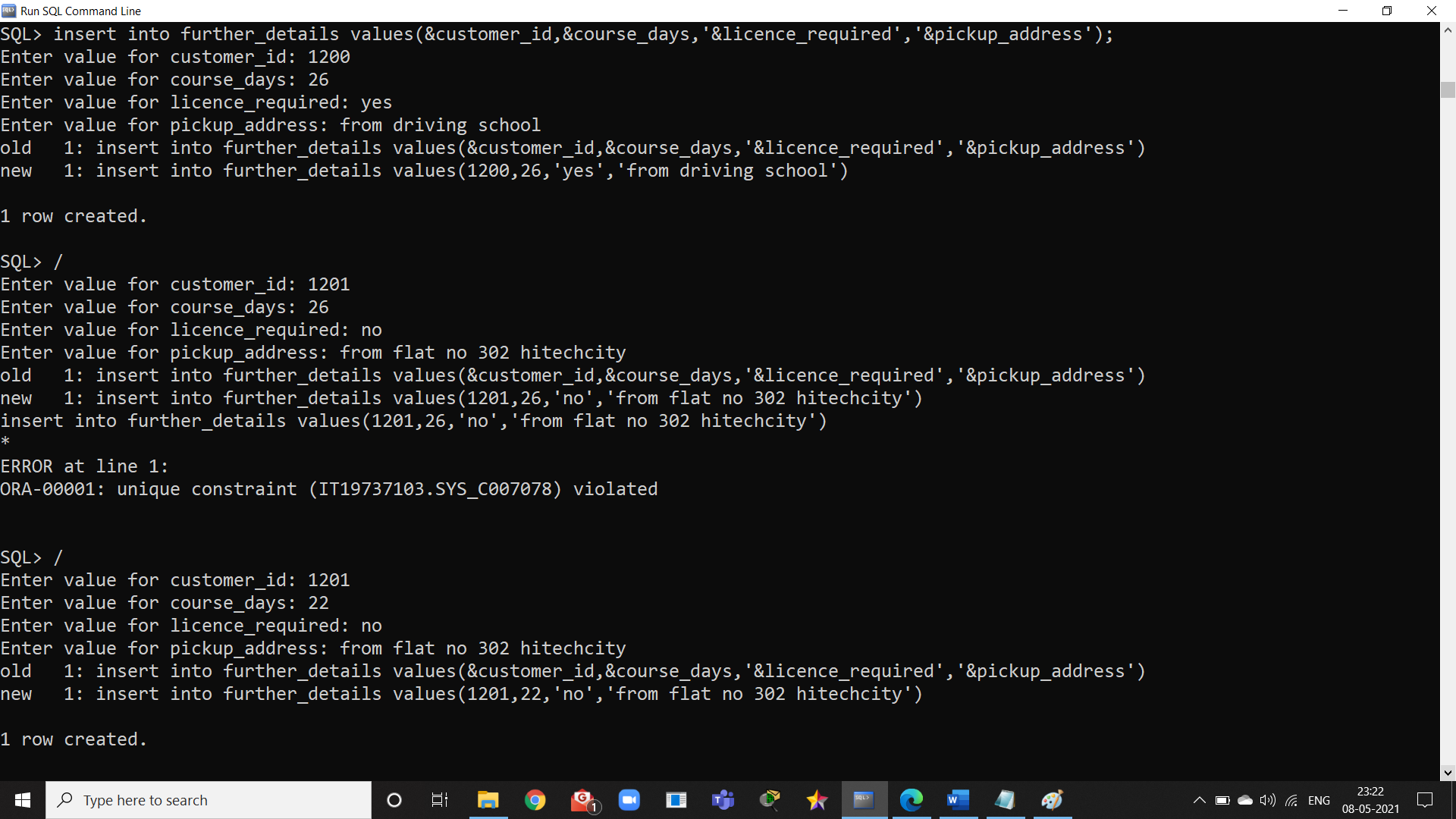


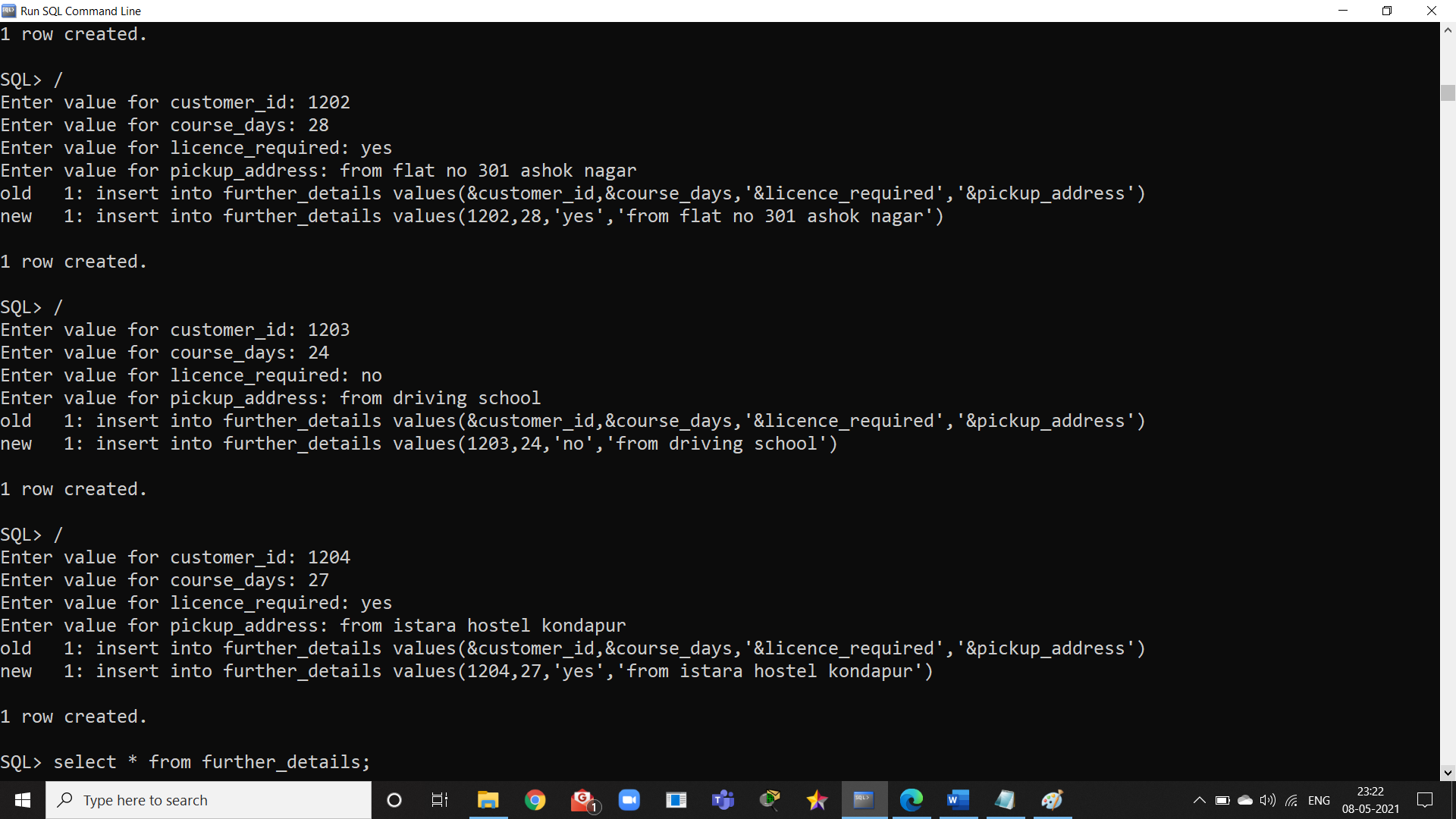
**17**

****

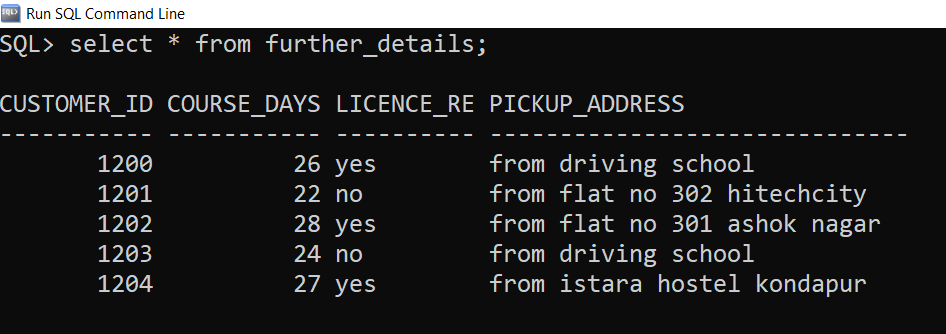


**18**





**19**

****

**--------------------------------------------------xxxxxxxxxxxxxxxxxxx----------------------------------------------------**

**20**