Hotel Database Management System:**”MilNurAn”**

Created by: Anel Askarkyzy,Duisenov Nurdaulet,Kulimzhayeva Milana

**a)INTRODUCTION: Overview of project & analyze the requirements**

The main objective of this project is to create a database management system for a hotel. We will be managing the below areas of the hotel database management system.

-The hotel chains, their details.

The hotel and their details and other information like the rooms and their description and discounts, etc.

Information about employees and departments they work in.

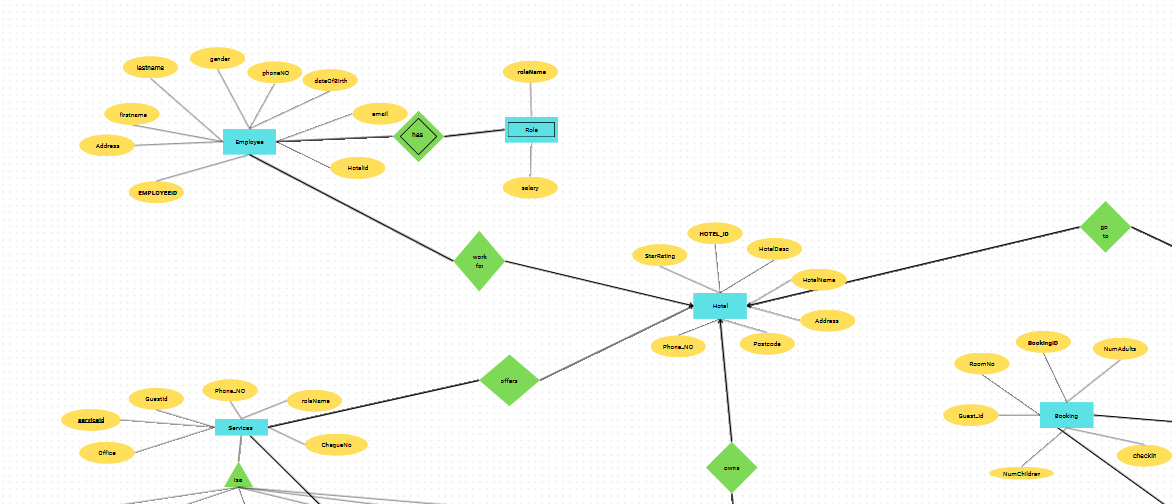
- Information about guests.

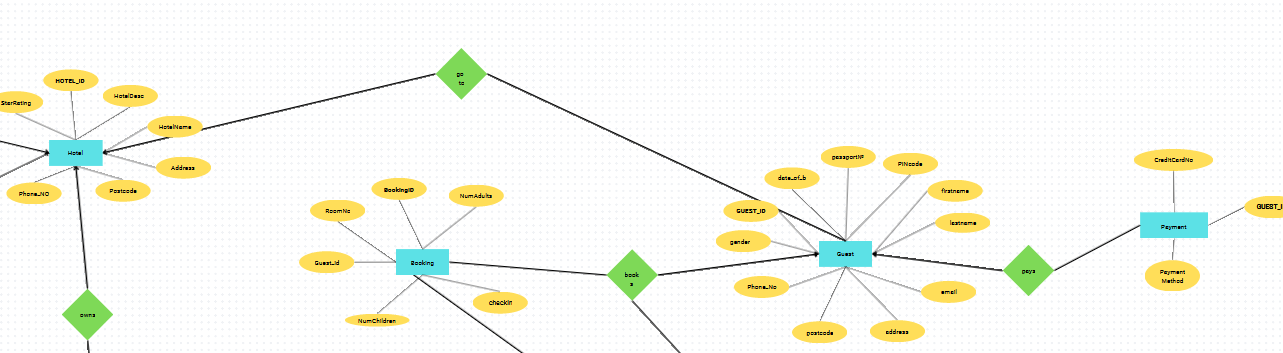
Managing bookings and other services used by the guests. The end-users are the guests who stay in the hotel.

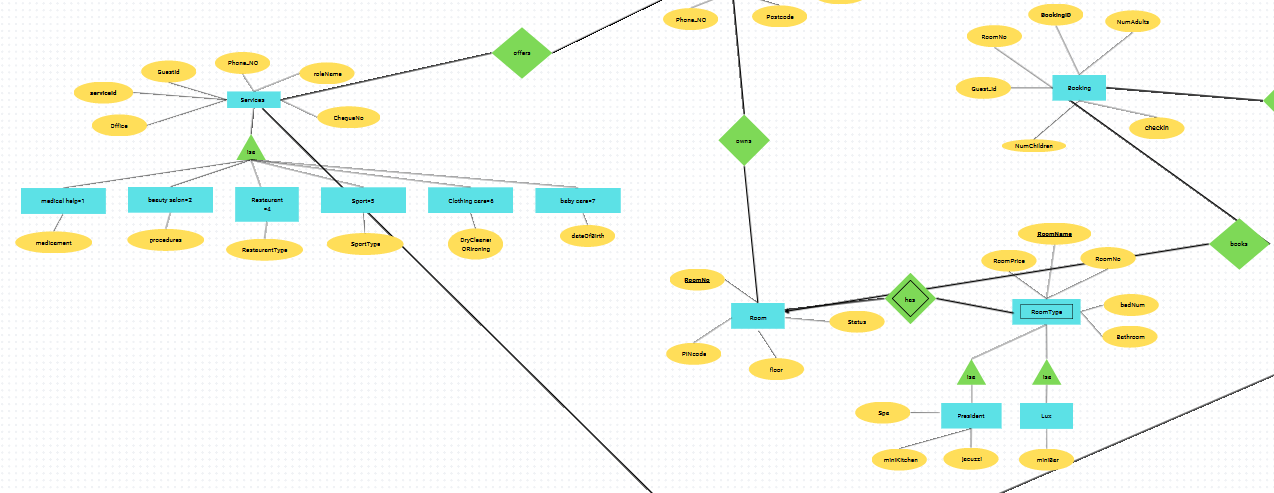
**b)DATABASE DESIGNING: Starting with drawing the ERR diagram**

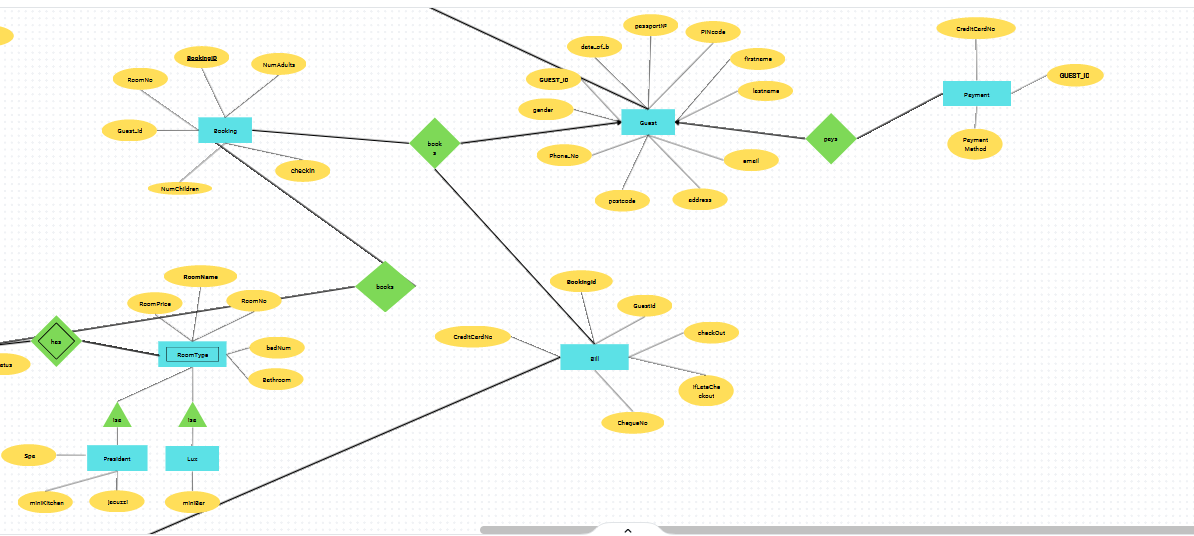
We started with designing the structure of the database. We drew the ERR diagram on a paper, noting down all the tables required. We designed each table with columns and attributes respectively and trying to make an idea about the relationships between tables. We tried to keep the tables in a form where we can reduce the data redundancy and tried to make it accessible in an easy and simple way. The ERR diagram created is as below:

**https://www.canva.com/design/DAFT4XI2ODQ/44Ua543BiBpoDeYJrEQSvg/edit?utm\_content=DAFT4XI2ODQ&utm\_campaign=designshare&utm\_medium=link2&utm\_source=sharebutton**









**c) Normalization:**

HOTEL

| hotelID | hotelName | hotelDesc | phoneNo | starRating | postCode |
| --- | --- | --- | --- | --- | --- |

hotelId -> hotelName,HotelDesc,starRating,postCode

| postCode | address |
| --- | --- |

postCode -> address

GUEST

| guestID | passportNo | email | phoneNumber | postcode |
| --- | --- | --- | --- | --- |

guestId -> passportNo,email,phoneNumber,postCode

| passportNo | firstName | lastName | gender | dateOfBirth |
| --- | --- | --- | --- | --- |

passportNo -> firstName,latName,gender,postCode

| postcode | address |
| --- | --- |

postCode-> address

PAYMENT

| guestID | paymentMethod | **creditCardNo** |
| --- | --- | --- |

creditCardNo -> guestId,paymentMethod

BILL

| bookingID | guestID | chequeNo | checkOut | ifLateCheckOut | creditCardNo |
| --- | --- | --- | --- | --- | --- |

BOOKING

| bookingID | numAdults | roomNo | numChildren | guestID |
| --- | --- | --- | --- | --- |

bookingId -> numAdults,roomNo,numChildren,guestId

SERVICES

| serviceID | guestID | roleName | chequeNo | office |
| --- | --- | --- | --- | --- |

serviceId -> guestId,roleName,chequeNo

| office | phoneNumb |
| --- | --- |

office -> phoneNumb

BEAUTY SALON

| serviceID | beautyProcedures |
| --- | --- |

serviceId -> beautyProcedures

MEDICAL HELP

| serviceID | medicament |
| --- | --- |

serviceId -> medicament

BABY CARE

| serviceID | dateOfBirth |
| --- | --- |

serviceId -> dateOfBirth

SPORT

| serviceID | sportType |
| --- | --- |

serviceId -> sportType

RESTAURANT

| serviceID | restaurantType |
| --- | --- |

serviceId -> restaurantType

CLOTHING CARE

| serviceID | dry cleaner | ironing |
| --- | --- | --- |

serviceId -> dry cleaner, ironing

ROOM

| roomNo | PINCode | floor | status | hotelID |
| --- | --- | --- | --- | --- |

roomNo ->PINcode, floor,status

ROOMTYPE

| roomName | roomPrice | bedNum | bathroom | roomNo |
| --- | --- | --- | --- | --- |

roomName -> roomPrice, bedNum,bathroom,roomNo

LUX

| roomName | miniBar |
| --- | --- |

roomName -> miniBar

PRESIDENT

| roomName | miniKitchen | SPA | jacuzzi |
| --- | --- | --- | --- |

roomName -> miniKitchen, SPA,jacuzzi

EMPLOYEE

| employeeID | lastName | firstName | email | phoneNo | dateOfBirth | address | gender | HotelId |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |

employeeId -> lastName,firstName,phoneNo,email,dateOfBirth,address,gender

ROLE

| employeeID | roleName | salary |
| --- | --- | --- |

**d) Physical Design (physical is like table in database)**

**in github**

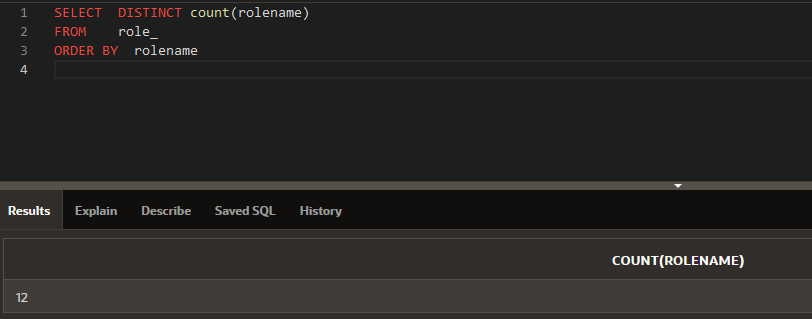
**e)Query**

**1.How many different professions work in the hotel ? (order by)**

SELECT DISTINCT count(rolename)

FROM role\_

ORDER BY rolename



δ

τ rolename

π COUNT (rolename)

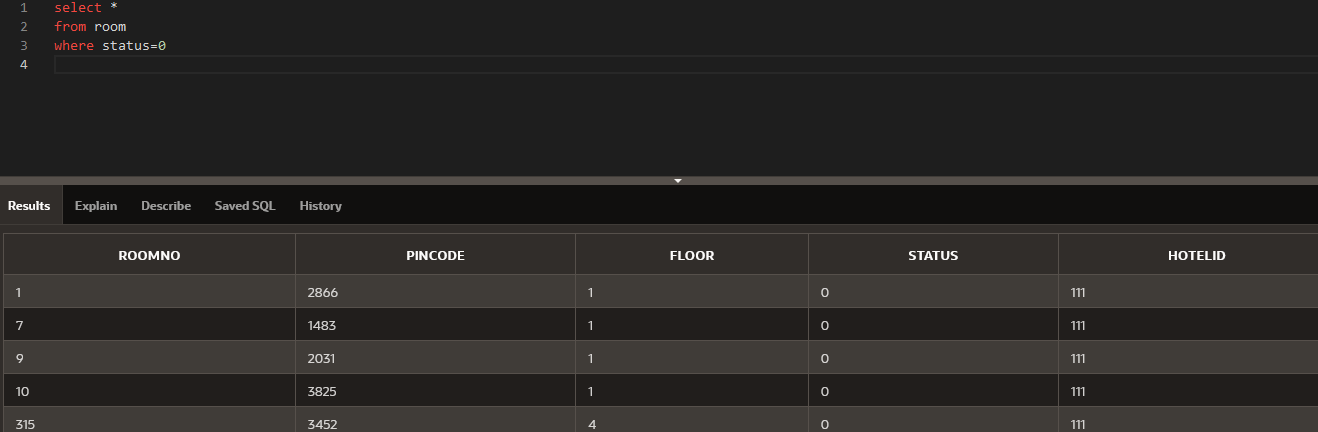
γ COUNT (rolename) role\_

**2.Display the rooms which are available.**

select \*

from room

where status=0



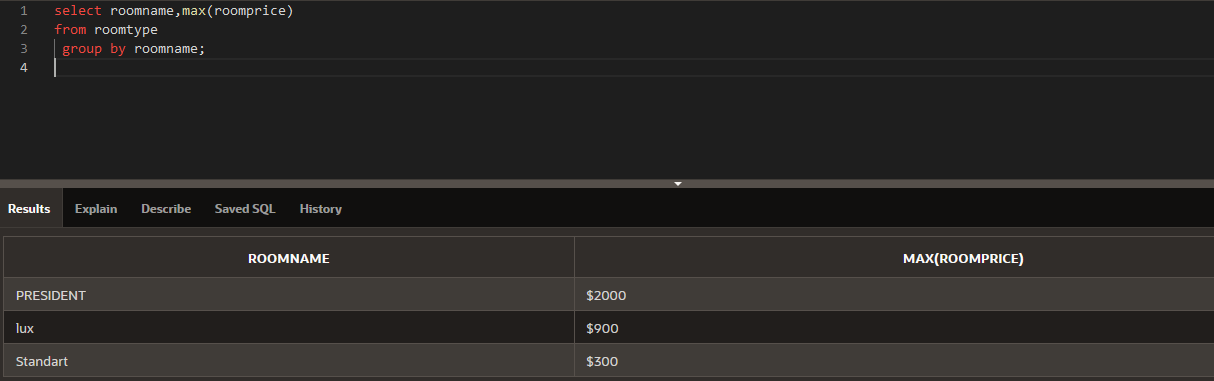
σ status = 0 room

**3.Display the highest roomprice for each room type. (group by)**

select roomname,max(roomprice)

from roomtype

group by roomname;



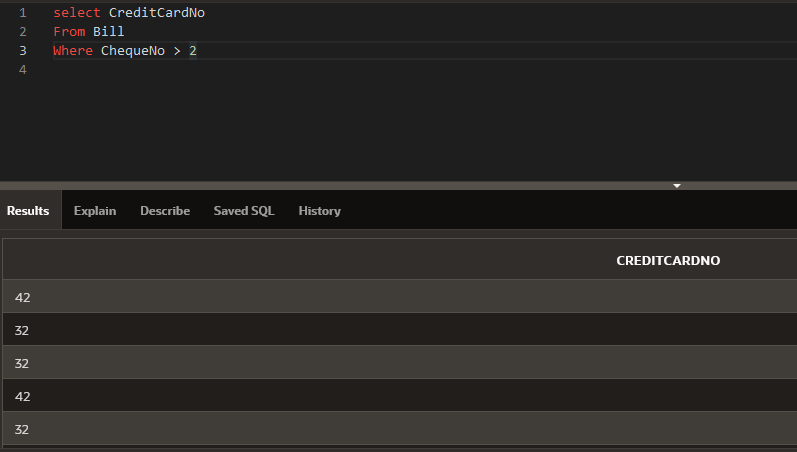
γ roomname, MAX (roomprice) roomtype

**4.**

select CreditCardNo

From Bill

Where ChequeNo > 2



π creditcardno  σ chequeno > 2 bill

**5.Trigger1**

create or replace TRIGGER test1

BEFORE UPDATE ON hotel1

FOR EACH ROW

BEGIN

IF( :new.hotelname!= 'MilNurAn' )

THEN

:new.hotelname := :old.hotelname;

END IF;

END;

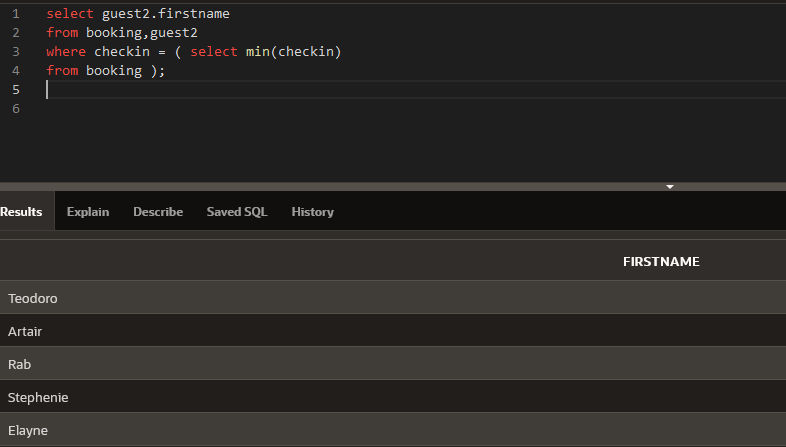
**6.**

select guest2.firstname

from booking,guest2

where checkin = ( select min(checkin)

from booking );



**7.**

SELECT

EXTRACT(YEAR FROM checkin) AS year,

EXTRACT(MONTH FROM checkin) AS month,

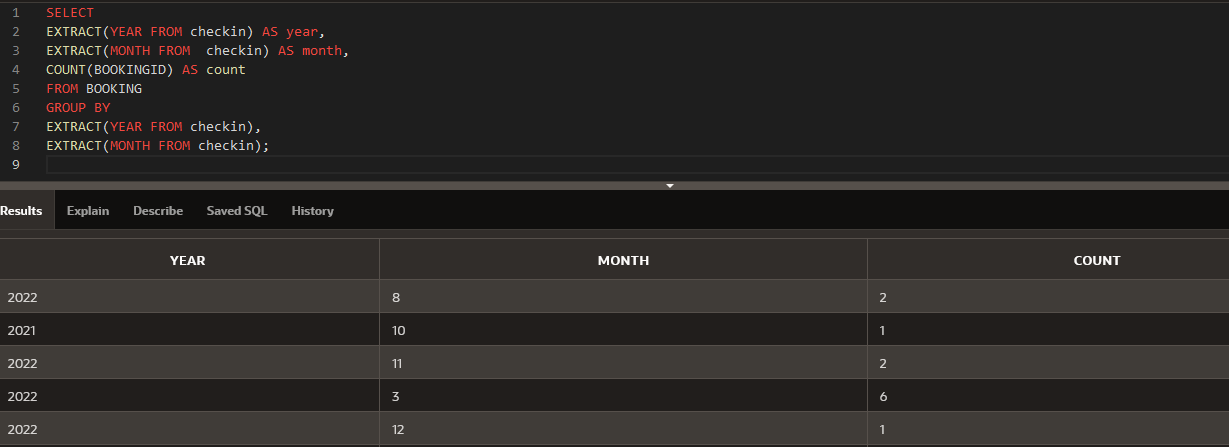
COUNT(BOOKINGID) AS count

FROM BOOKING

GROUP BY

EXTRACT(YEAR FROM checkin),

EXTRACT(MONTH FROM checkin);



**8**.**Trigger2**

CREATE OR REPLACE TRIGGER "increaseLuxPrice"

AFTER

DELETE on ROOMTYPE

for each row

declare

pragma autonomous\_transaction;

begin

if :old.roomname like 'PRESIDENT' then

(update roomtype set roomprice = :old.roomprice

where roomname like 'lux');

commit;

end if;

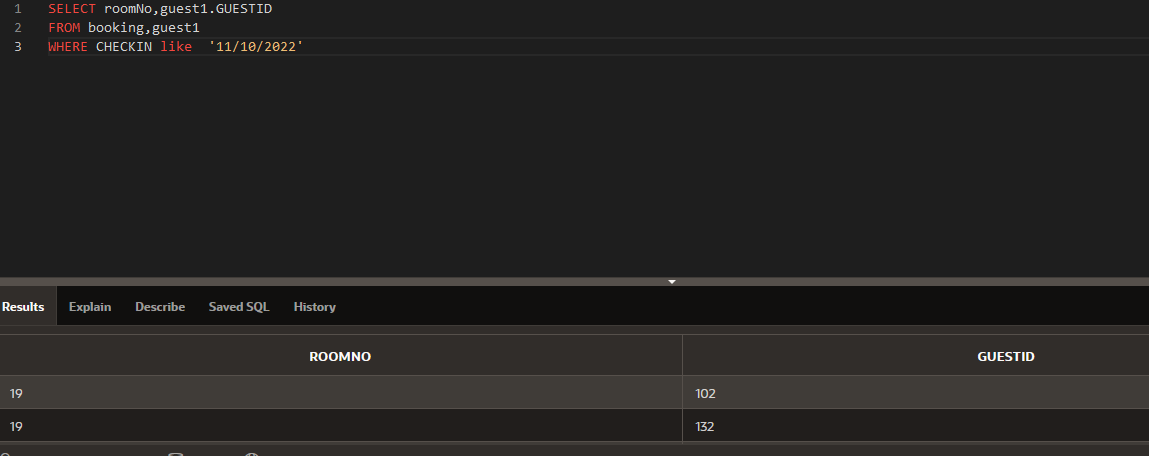
end; TRUE

**9.**

SELECT roomNo,guest1.GUESTID

FROM booking,guest1

WHERE CHECKIN like '11/10/2022'



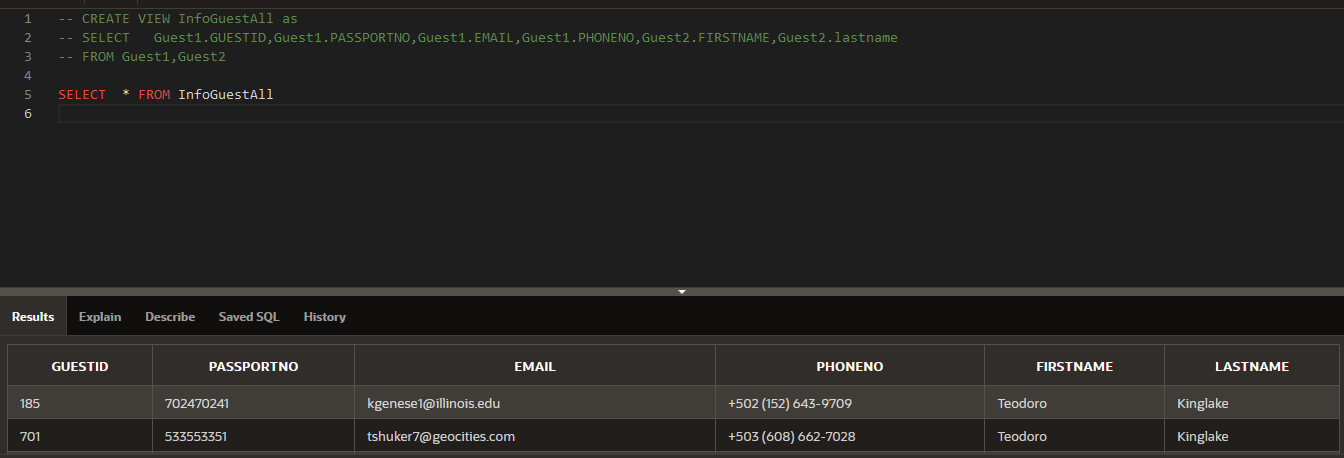
**10.VIEW**

CREATE VIEW InfoGuestAll as

SELECT Guest1.GUESTID,Guest1.PASSPORTNO,Guest1.EMAIL,Guest1.PHONENO,Guest2.FIRSTNAME,

Guest2.lastname

FROM Guest1,Guest2



**11.Trigger3**

CREATE OR REPLACE TRIGGER "roleAdding"

after insert on role\_

for each row

declare

pragma AUTONOMOUS\_TRANSACTION;

begin

if:new.rolename like 'Master Chef' then

update role\_ set salary = :new.salary - 5 where rolename like 'Chef';

commit;

end if;

end;

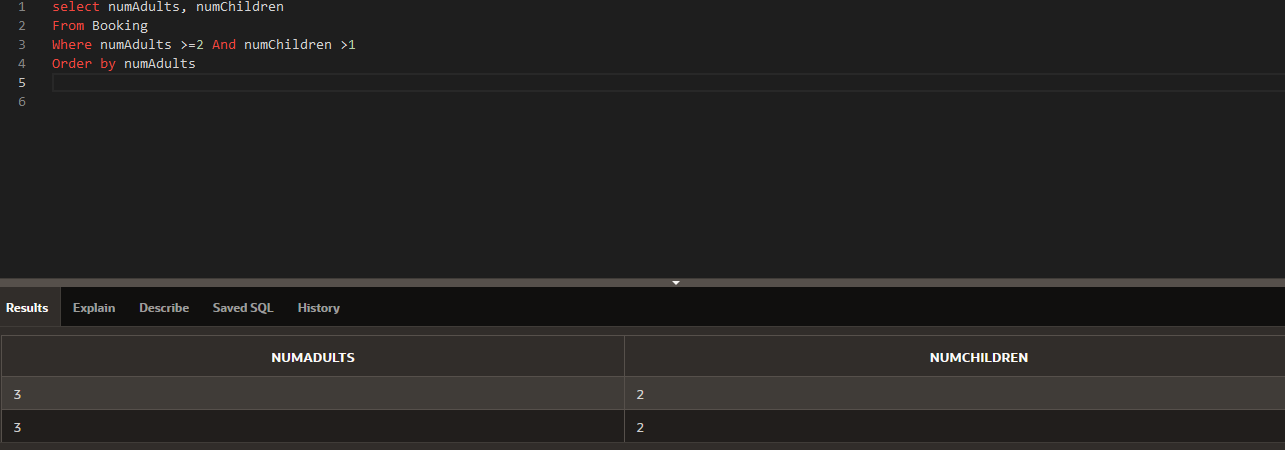
**12.**

select numAdults, numChildren

From Booking

Where numAdults >=2 And numChildren >1

Order by numAdults



13.

select guest1.passportNo

from guest1

join payment on payment.guestID = guest1.guestID

join bill on bill.creditCardNo = payment.creditCardNo

where bill.chequeNo = 60



**14.View**

CREATE VIEW hotel\_employees AS

SELECT FIRSTNAME, LASTNAME, EMAIL, PHONENO

FROM employee

JOIN role\_

ON role\_.EMPLOYEEID = employee.EMPLOYEEID;

**15**.select distinct count(\*)

from guest1

δ

π COUNT (\*)

γ COUNT (\*) guest1

**16**,ALTER TABLE Guest1

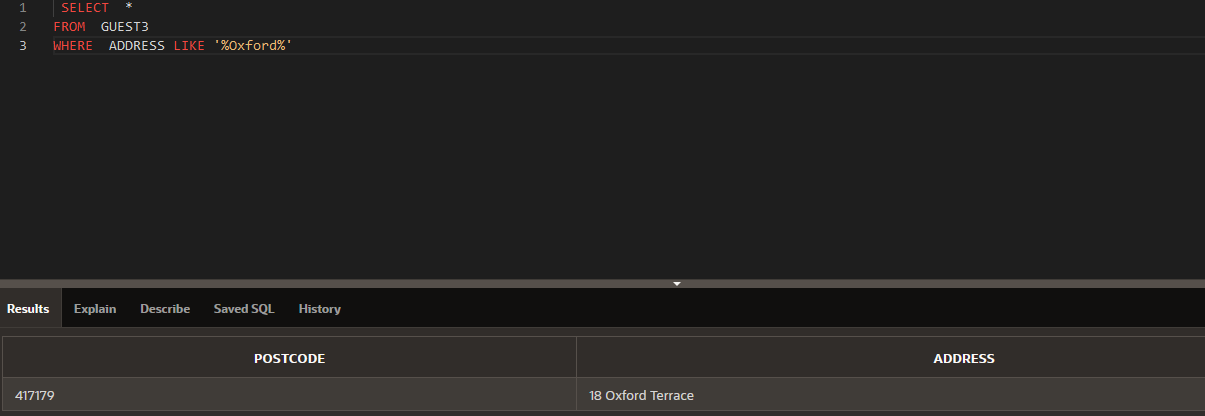
DROP column Email

17.ALTER TABLE Employee

RENAME TO Workers;

**18**. SELECT \*

FROM GUEST3

WHERE ADDRESS LIKE '%Oxford%'

σ address LIKE "%Oxford%" guest3

**Transactions**

**1.Booking a room:**

BEGIN TRANSACTION;

INSERT INTO BOOKING (BOOKINGID, NUMADULTS, ROOMNO, NUMCHILDREN, GUESTID, CHECKIN)

VALUES (1001, 2, 101, 0, 5001, '12/25/2022');

COMMIT;

**2.Checking out a guest:**

BEGIN TRANSACTION;

UPDATE BOOKING

SET CHECKOUT = '12/27/2022'

WHERE BOOKINGID = 1001;

INSERT INTO BILL (BOOKINGID, GUESTID, CHEQUENO, CHECKOUT, IFLATECHECKOUT, CREDITCARDNO)

VALUES (1001, 5001, 10001, '12/27/2022', NULL, NULL);

COMMIT;

**3.Paying for a booking with a credit card:**

BEGIN TRANSACTION;

INSERT INTO PAYMENT (CREDITCARDNO, GUESTID, PAYMENTMETHOD)

VALUES (1234567812345678, 621, cash);

UPDATE CHEQUE

SET CREDITCARDNO = 1234567812345678, CHEQUENO = NULL

WHERE BOOKINGID = 1001;

COMMIT;

**Indexes**

1.CREATE INDEX employee\_lastname\_idx ON EMPLOYEE (LASTNAME);

2.CREATE INDEX booking\_numadults\_idx ON BOOKING (NUMADULTS);

3.CREATE INDEX bill\_checkout\_idx ON BILL (CHECKOUT);