Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India
(Autonomous College Affiliated to University of Mumbai)

MINI PROJECT PHASE 1 REPORT

HOUSE RENTING, BUYING AND SELLING DATABASE MANAGEMENT SYSTEM

NAME: Janhavi Ram Deshmukh
UID: 2021700017
BRANCH: CSE(DS)
BATCH: D1
GROUP: 3 (UID: 5,9,17)

INDEX:

AIM: To create a database management system for a house buying, selling, and renting platform

SCOPE:

- **ABOUT THE PLATFORM:**

The online platform allows users to search for houses on rent or for sale based on their preferences, or to sell or rent out their house(s) on the platform. Primary purpose of the platform is to connect buyers and tenants with homeowners. The platform also provides company-hired agents for home seekers who visit a house on seeker's behalf and resolve any queries they may have; in case the seeker cannot visit the house themself.

Houses meeting an eligible home seeker's criteria will be recommended to him/her, and the seeker can take help of an agent experienced with the houses of that area. An owner can also register his/her house(s) on the platform for it to be recommended to home seekers. A seeker can also pay the security deposit if they wish to rent a house, or token of confirmation if they wish to buy a house, on the platform.

DATABASE REQUIREMENTS:
(CASE STUDY)

- Database records the information about every user (USER) who has an account on the platform. Each user has a unique EMail ID (EMAIL_ID) and a password (PASSWORD) essential to log in to the platform. The system assigns a unique numeric ID (USER_ID) to every user at the time of account creation. Phone number (PHONE_NO), name (FIRST_NAME, MIDDLE_NAME and LAST_NAME), date of birth (DOB) and address (STATE, CITY, AREA, PINCODE) of the user are also recorded whenever the user chooses to provide this information. Account creation date (ACCOUNT_CREATION_DATE) and date of last activity (LAST_ACTIVE_DATE) is also stored to determine the account's age and if the account is still active respectively.

- A user could be searching for a house (HOMESEEKER) or could be an owner wanting to rent or sell their house (HOMEOWNER). For a home seeker, his/her house preferences are stored on the site if provided. These preferences include information about: city (PREFERRED_CITY), area (PREFERRED_AREA), whether the user is looking for an apartment or a bungalow (PREFERRED_HOUSE_TYPE), floor (PREFERRED_FLOOR), whether the user strictly wants pets to be allowed in the house (PREFERS_PET_ALLOWANCE), number of rooms (NO_OF_ROOMS), minimum required size in square feet (SIZE), minimum required parking spaces

(NO_OF_PARKING_SPACES), and whether the user prefers a furnished house or an unfurnished one (PREFERRED_FURNISHING_STATUS).

- A home seeker could further be looking for a house to rent (TENANT) or to buy (BUYER). For a buyer, his/her preferred price range (PRICE_RANGE) is recorded in the database. For a tenant, following information is stored: preferred rent range (RENT_RANGE), marital status (MARITAL_STATUS), whether the tenant is vegetarian or non-vegetarian (FOOD_HABIT) and the maximum number of tenants the user can share a house with (MAXIMUM_NO_OF_ROOMMATES).

- For a homeowner, a government ID is compulsory. Type (GOVT_ID_TYPE) and number (GOVT_ID_NO) of the ID provided is recorded.

- A homeowner can register multiple houses on the website for renting or selling. Each registered house (HOUSE) has an auto generated numeric unique house ID (HOUSE_ID). Information recorded for each house is as follows: address (STATE, CITY, AREA, PINCODE), whether it is an apartment or a bungalow (TYPE), the year it was bought in (BOUGHT_IN), floor of the house (if it is an apartment) or the number of floors (if it is a bungalow) (FLOOR), number of rooms (NO_OF_ROOMS), whether pets are allowed (PET_ALLOWANCE), furnishing status (FURNISHING_STATUS), number of bathrooms (NO_OF_BATHROOMS), and number of parking spaces (NO_OF_PARKING_SPACES), hours per day for which water is available (WATER_AVAILABILITY), whether a deal for buying or renting the house has been finalised, in which case the house will no longer appear in recommendation pages or search results of other users (FINALISED); and ID of the user who booked the house (BOOKED_BY) if the house has been booked on the site.

- A house can be up for renting (RENT) or for sale (SALE). For a house on rent, following information is to be recorded: monthly rent (MONTHLY_RENT), maximum number of tenants the owner will allow if a group is renting the house (NO_OF_TENANTS), preferred marital status of the tenant (TENANT_MARITAL_STATUS), whether owner would allow a non-vegetarian to rent the house or not (TENANT_FOOD_HABIT), mandatory contract period DURATION_OF_CONTRACT), security deposit amount (SECURITY_DEPOSIT_AMOUNT). For a house on sale, total price (TOTAL_PRICE) and amount to be paid as a token of confirmation (TOKEN_AMOUNT) is recorded.

- The company hires agents to help home seekers by visiting a house or neighbourhood on their behalf. The company has to keep track of following information about the agent (AGENT): E-Mail id (EMAIL), phone number (PHONE_NO), name (FIRST_NAME, MIDDLE_NAME, LAST_NAME), salary (SALARY), joining date (JOINING_DATE), and working address (STATE, CITY, AREA, PINCODE)

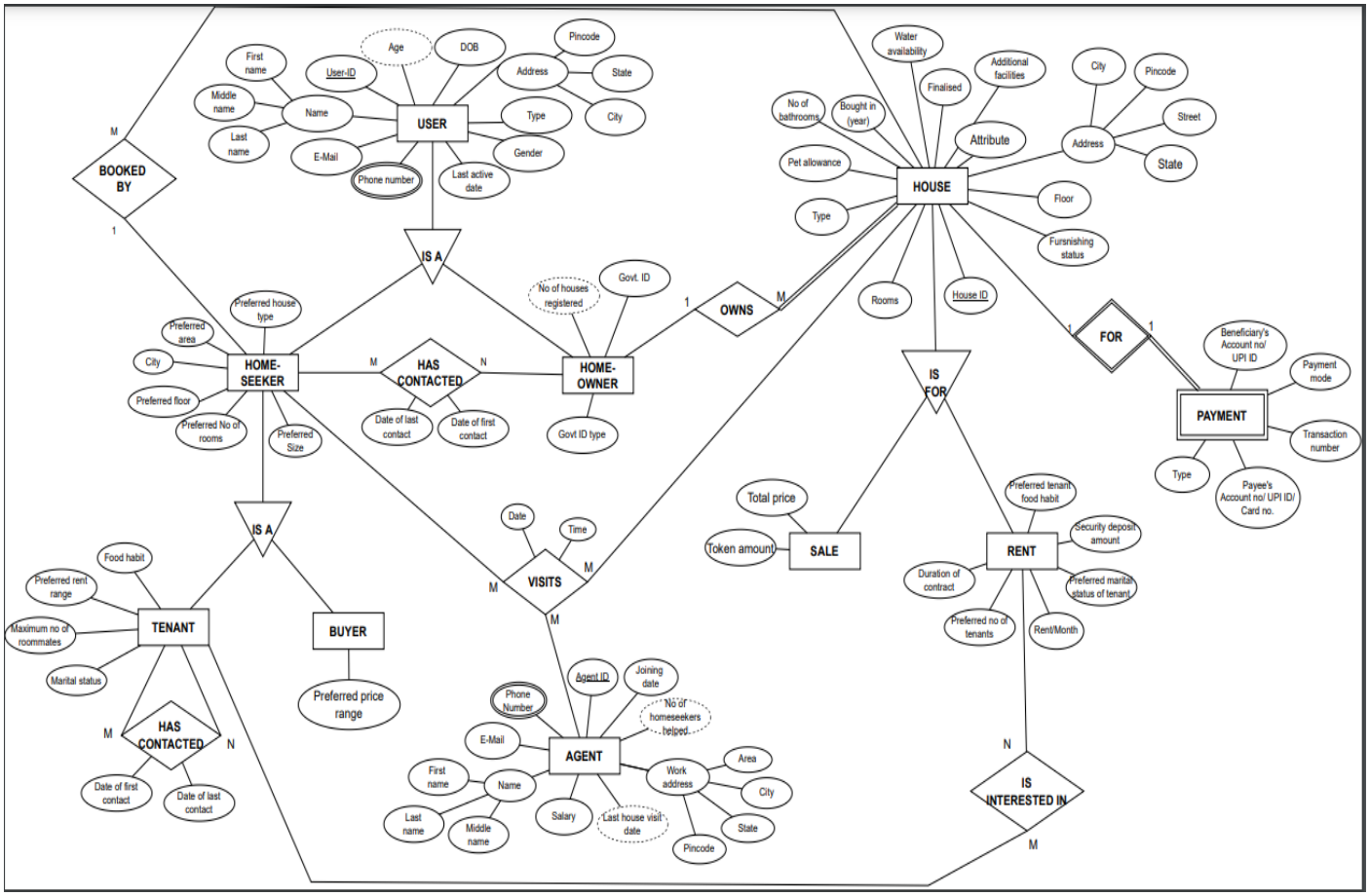
- The company also needs to keep track of each agent's every visit to a house (VISITS). Information to be recorded about the visit includes: ID of the house agent visited (HOUSE_ID), ID of the home seeker on behalf of whom the agent visited the house (SEEKER_ID), date (DATE), time (TIME) of the visit.

- A seeker can contact an owner on the site itself if he/she likes a certain house profile. To keep track of all seekers and owners who have contacted each other through the platform (HAS_CONTACTED), following information needs to be recorded: user ID of the owner (OWNER_ID), user ID of the seeker (SEEKER_ID), date of first contact (FIRST_CONTACT_DATE) and date of last contact (LAST_CONTACT_DATE).

- Each house profile has a list of interested tenants. If a tenant wants to rent a particular house but cannot afford the rent alone and is looking for other tenants who are in the same situation as them, they can register themself under the house's list of interested tenants and can contact other tenants in the list to form groups based on their preferences. To keep track of tenants and houses they're interested in (IS_INTERESTED_IN), user ID of the tenant (TENANT_ID) and ID of the house the user is interested in (HOUSE_ID) is to be recorded for every user who has registered in at least one interested list and every house which has at least one interested tenant. Similarly, to keep track of tenants who have contacted other tenants (T_HAS_CONTACTED_T), user IDs of both tenants (TENANT_ID_1, TENANT_ID_2) and dates of first and last contact (FIRST_CONTACT_DATE and LAST_CONTACT_DATE) are to be recorded.

- Home seekers can pay a security deposit amount or token of confirmation on the platform. To keep track of all payments made on the platform (PAYMENTS), each payment is assigned an auto generated payment ID (PAYMENT_ID) and following information is recorded: transaction number (TRANSACTION_NO), payment mode (i.e. whether paid through debit card, credit card, UPI or net banking) (PAYMENT_MODE), UPI ID or account number or card number of the seeker (UPIID_ACCTNO_CARD), beneficiary's details (BENEFICIARY_UPIID_ACCTNO), date (DATE) and time (TIME) of transaction, whether the transaction was a security deposit payment or a token of confirmation (TYPE), userID of the seeker (BENEFACTOR_ID) and house ID for which the payment was made (HOUSE_ID).

ER MODEL:


USER
First name
Middle name
Last name
Name
User-ID
Age
DOB
Pincode
Address
State
City
Type
Gender
E-Mail
Phone number
Last active date
BOOKED BY
M
1
IS A
HOME-SEEKER
Preferred house type
Preferred area
City
Preferred floor
Preferred No of rooms
Preferred Size
HAS CONTACTED
M
N
Date of last contact
Date of first contact
HOME-OWNER
No of houses registered
Govt. ID
Govt ID type
OWNS
1
M
HOUSE
Water availability
Additional facilities
Finalised
Attribute
No of bathrooms
Bought in (year)
Pet allowance
Type
Floor
Fursnishing status
Address
City
Pincode
Street
State
Rooms
House ID
FOR
1
1
PAYMENT
Beneficiary's Account no/ UPI ID
Payment mode
Transaction number
Type
Payee's Account no/ UPI ID/ Card no.
IS FOR
SALE
Total price
Token amount
RENT
Preferred tenant food habit
Security deposit amount
Duration of contract
Preferred marital status of tenant
Preferred no of tenants
Rent/Month
IS INTERESTED IN
N
M
IS A
TENANT
Food habit
Preferred rent range
Maximum no of roommates
Marital status
HAS CONTACTED
M
N
Date of first contact
Date of last contact
BUYER
Preferred price range
VISITS
Date
Time
M
M
M
AGENT
Agent ID
Joining date
Phone Number
No of homeseekers helped
E-Mail
First name
Name
Last name
Middle name
Salary
Last house visit date
Work address
Area
City
State
Pincode

RELATIONAL SCHEMA:


USER
USER_ID
PASSWORD
EMAIL
F_NAME
M_NAME
L_NAME
DOB
GENDER
ACCOUNT CREATION DATE
LAST ACTIVE DATE
STATE
CITY
AREA
PINCODE
HOMESEEKER
USER_ID
PREFERRED CITY
PREFERRED AREA
PREFERRED HOUSE TYPE
PREFERRED FLOOR
PREFERRED SIZE
PREFERS PET ALLOWANCE
NO OF ROOMS
PREFERRED FURNISHING STATUS
NO OF PARKING SPACES
TENANT
USER_ID
RENT RANGE
MAXIMUM NO OF ROOMMATES
MARITAL STATUS
FOOD HABIT
BUYER
USER_ID
PRICE RANGE
HOMEOWNER
USER_ID
GOVT ID TYPE
GOVT ID NUMBER
HOUSE
HOUSE_ID
OWNER_ID
STATE
CITY
AREA
PINCODE
TYPE
BOUGHT IN
NO OF ROOMS
FLOOR
ADDITIONAL FACILITIES
FURNISHING STATUS
NO OF BATHROOMS
PET ALLOWANCE
WATER AVAILABILITY
FINALISED
BOOKED BY
HOUSE SIZE
RENT
HOUSE_ID
MONTHLY RENT
NO OF TENANTS
TENANT MARITAL STATUS
TENANT FOOD HABIT
DURATION OF CONTRACT
SECURITY DEPOSIT AMOUNT
SALE
HOUSE_ID
TOTAL PRICE
TOKEN AMOUNT
AGENT
AGENT_ID
EMAIL ID
F_NAME
M_NAME
L_NAME
SALARY
JOINING DATE
STATE
CITY
AREA
PINCODE
VISITS
AGENT_ID
HOUSE ID
SEEKER_ID
VISIT DATE
VISIT TIME
HAS CONTACTED
SEEKER_ID
OWNER_ID
FIRST CONTACT DATE
LAST CONTACT DATE
PAYMENT
TRANSACTION NUMBER
PAYMENT MODE
UPI ID/ ACCT NO/ CARD NO
BENEFICIARY UPI ID/ ACCT NO
DATE
TIME
TYPE
HOUSE_ID
IS INTERESTED IN
TENANT_ID
HOUSE_ID
T _HAS_CONTACTED_T
TENANT_ID_1
TENANT_ID_2
FIRST CONTACT DATE
LAST CONTACT DATE
AGENT_PHONE_NO
AGENT_ID
PHONE_NO
USER_PHONE_NO
USER_ID
PHONE_NO

TECHNOLOGY:

- In frontend of the project, languages like HTML, CSS and JavaScript will be used. The database creation is done using MySQL.
- HTML provides the basic structure of sites. CSS will be used to control presentation, formatting and layout. JavaScript will be used to control behaviour of different elements.
- Server side management and collection of forms data can be done using PHP

DATABASE CREATION:

- USER:

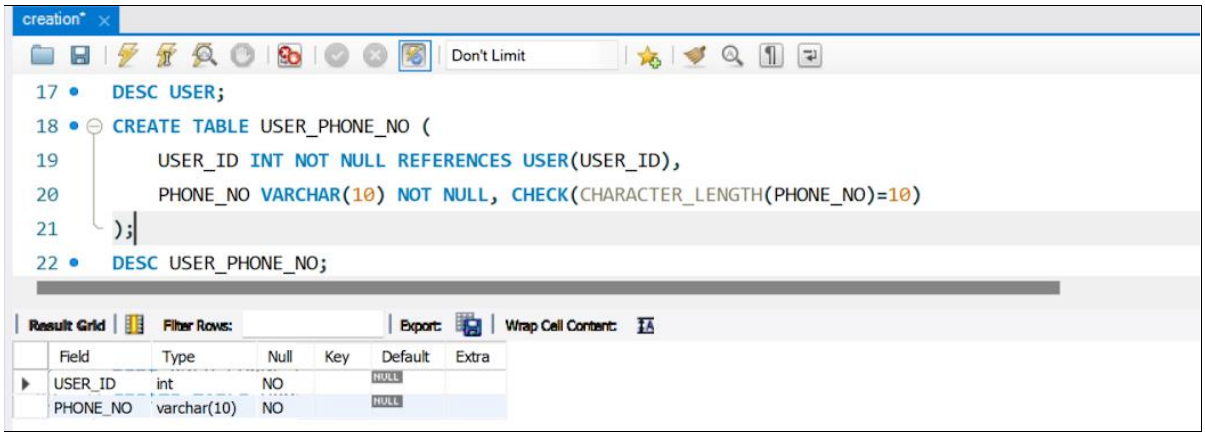
```
creation*
Don't Limit
1  CREATE TABLE USER (
2      USER_ID INT PRIMARY KEY AUTO_INCREMENT,
3      EMAIL VARCHAR(50) UNIQUE NOT NULL,
4      PASSWORD VARCHAR(20) NOT NULL, CHECK(CHARACTER_LENGTH(PASSWORD)>=8),
5      F_NAME VARCHAR(30),
6      M_NAME VARCHAR(30),
7      L_NAME VARCHAR(30),
8      DOB DATE,
9      GENDER ENUM("MALE","FEMALE","OTHER","DON'T WANT TO SPECIFY"),
10     ACCT_CREATION_DATE DATE NOT NULL,
11     LAST_ACTIVE_DATE DATE NOT NULL,
12     STATE VARCHAR(50),
13     CITY VARCHAR(50),
14     AREA VARCHAR(100),
15     PINCODE VARCHAR(6) ,CHECK(CHARACTER_LENGTH(PINCODE)=6)
16 );
17 DESC USER;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
USER_ID	int	NO	PRI	NULL	auto_increment
EMAIL	varchar(50)	NO	UNI	NULL	
PASSWORD	varchar(20)	NO		NULL	
F_NAME	varchar(30)	YES		NULL	
M_NAME	varchar(30)	YES		NULL	
L_NAME	varchar(30)	YES		NULL	
DOB	date	YES		NULL	
GENDER	enum('MALE','FEMALE','OTHER','DON''T WANT TO...	YES		NULL	
ACCT_CREATION_DATE	date	NO		NULL	
LAST_ACTIVE_DATE	date	NO		NULL	
STATE	varchar(50)	YES		NULL	
CITY	varchar(50)	YES		NULL	
AREA	varchar(100)	YES		NULL	
PINCODE	varchar(6)	YES		NULL	

EMAIL and PASSWORD have been made NOT NULL as they are essential for creation of an account. Even though email is not chosen as the primary key, each account can only be associated with one email in our system. USER_ID is the autogenerated primary key and ACTIVE_CREATION_DATE and LAST_ACTIVE_DATE are also NOT NULL as they're automatically recorded by the system. CHECK constraints have been used to ensure that values in the PINCODE column are 6 digits long and those in the password column are at least 8 characters long . As gender can have 4 possible values in our system, ENUM datatype has been used to store GENDER attribute.

- USER_PHONE_NO:



As phone number of a USER is a multivalued attribute, a separate table stores every phone number of every user. A CHECK constraint has been used to ensure that every phone number entered is exactly ten digits long.

- HOMESEEKER:

```
creation*
Don't Limit
22 • CREATE TABLE HOMESEEKER (
23       USER_ID INT PRIMARY KEY REFERENCES USER(USER_ID),
24       PREFERRED_CITY VARCHAR(50),
25       PREFERRED_AREA VARCHAR(100),
26       PREFERRED_HOUSE_TYPE ENUM("FLAT","BUNGALOW"),
27       PREFERRED_FLOOR INT, CHECK(PREFERRED_FLOOR>=0),
28       PREFERRED_SIZE INT, CHECK(PREFERRED_SIZE>=0),
29       PREFERS_PET_ALLOWANCE BOOLEAN,
30       NO_OF_ROOMS INT, CHECK(NO_OF_ROOMS>0),
31       PREFERRED_FURNISHING_STATUS ENUM("FURNISHED","SEMIFURNISHED","UNFURNISHED"),
32       NO_OF_PARKING_SPACES INT, CHECK(NO_OF_PARKING_SPACES>=0)
33   );
34 • DESC HOMESEEKER;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
USER_ID	int	NO	PRI	NULL	
PREFERRED_CITY	varchar(50)	YES		NULL	
PREFERRED_AREA	varchar(100)	YES		NULL	
PREFERRED_HOUSE_TYPE	enum('FLAT','BUNGALOW')	YES		NULL	
PREFERRED_FLOOR	int	YES		NULL	
PREFERRED_SIZE	int	YES		NULL	
PREFERS_PET_ALLOWANCE	tinyint(1)	YES		NULL	
NO_OF_ROOMS	int	YES		NULL	
PREFERRED_FURNISHING_STATUS	enum('FURNISHED','SEMIFURNISHED','UNFURNI...	YES		NULL	
NO_OF_PARKING_SPACES	int	YES		NULL	

ENUM has been used to store house type as there are only to possible values: flat or bungalow. Similarly, PREFERRED_FURNISHING_STATUS has only three possible values. PREFERS_PET_ALLOWANCE can be represented by a Boolean value as it can only be true or false. CHECK constraints have been used on NO_OF_PARKING_SPACES, PREFERRED_FLOOR, PREFERRED_SIZE and NO_OF_ROOMS to ensure that negative values are not entered in any of these columns.

- TENANT:

```
35 • CREATE TABLE TENANT (
36       USER_ID INT PRIMARY KEY REFERENCES HOMESEEKER(USER_ID),
37       MAX_RENT INT, CHECK(MAX_RENT>=0),
38       MAX_NO_OF_ROOMMATES INT, CHECK(MAX_NO_OF_ROOMMATES>=0),
39       MARITAL_STATUS BOOLEAN,
40       FOOD_HABIT BOOLEAN
41   );
42 • DESC TENANT;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
USER_ID	int	NO	PRI	NULL	
MAX_RENT	int	YES		NULL	
MAX_NO_OF_ROOMMATES	int	YES		NULL	
MARITAL_STATUS	tinyint(1)	YES		NULL	
FOOD_HABIT	tinyint(1)	YES		NULL	

MARITAL_STATUS is represented by Boolean as it has only two values (married: 1, unmarried: 0). CHECK constraints have been used on MAX_RENT and MAX_NO_OF_ROOMMATES to ensure that negative values are not entered in any of these columns.

- BUYER:

```
42 • DESC TENANT;
43 • CREATE TABLE BUYER (
44       USER_ID INT PRIMARY KEY REFERENCES HOMESEEKER(USER_ID),
45       MAX_PRICE INT, CHECK(MAX_PRICE>=0)
46   );
47 • DESC BUYER;
48 • CREATE TABLE HOMEOWNER (
49       USER_ID INT PRIMARY KEY REFERENCES USER(USER_ID),
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
USER_ID	int	NO	PRI	NULL	
MAX_PRICE	int	YES		NULL	

A CHECK constraint has been used on MAX_PRICE to disallow entry of a negative value.

- HOMEOWNER:

creation*

```
49 • CREATE TABLE HOMEOWNER (
50       USER_ID INT PRIMARY KEY REFERENCES USER(USER_ID),
51       GOVT_ID_TYPE ENUM("AADHAR","PAN","VOTER ID","PASSPORT","DRIVING LICENSE") NOT NULL,
52       GOVT_ID_NUMBER VARCHAR(20) NOT NULL
53   );
54 • DESC HOMEOWNER;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
USER_ID	int	NO	PRI	NULL	
GOVT_ID_TYPE	enum('AADHAR','PAN','VOTER ID','PASSPORT','DR...	NO		NULL	
GOVT_ID_NUMBER	varchar(20)	NO		NULL	

- HOUSE:

creation*

```
55 • CREATE TABLE HOUSE (
56       HOUSE_ID INT PRIMARY KEY AUTO_INCREMENT,
57       OWNER_ID INT REFERENCES HOMEOWNER(USER_ID),
58       STATE VARCHAR(50),
59       CITY VARCHAR(50),
60       AREA VARCHAR(100),
61       PINCODE VARCHAR(6) ,CHECK(CHARACTER_LENGTH(PINCODE)=6),
62       TYPE ENUM("FLAT","BUNGALOW"),
63       BOUGHT_IN YEAR,
64       NO_OF_ROOMS INT, CHECK(NO_OF_ROOMS>0),
65       FLOOR INT, CHECK(FLOOR>=0),
66       SIZE INT, CHECK(SIZE>=0),
67       FURNISHING_STATUS ENUM("FURNISHED","SEMIFURNISHED","UNFURNISHED"),
68       NO_OF_PARKING_SPACES INT, CHECK(NO_OF_PARKING_SPACES>=0),
69       NO_OF_BATHROOMS INT, CHECK(NO_OF_BATHROOMS>=0),
70       PET_ALLOWANCE BOOLEAN,
71       WATER_AVAILABILITY FLOAT(2), CHECK(WATER_AVAILABILITY>=0 OR WATER_AVAILABILITY<=24),
72       FINALISED BOOLEAN,
73       BOOKED_BY INT REFERENCES HOMESEEKER(USER_ID)
74   );
```

FINALISED attribute has only two possible values, and thus Boolean has been used to represent it (0: no, 1: yes). WATER_AVAILABILITY represents the number of hours per day for which water is available, and thus CHECK constraint has been used to ensure the value under this column is between 0 and 24

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
AREA	varchar(100)	YES		NULL	
PINCODE	char(6)	YES		NULL	
TYPE	enum('FLAT','BUNGALOW')	YES		NULL	
BOUGHT_IN	year	YES		NULL	
NO_OF_ROOMS	int	YES		NULL	
FLOOR	int	YES		NULL	
SIZE	int	YES		NULL	
FURNISHING_S...	enum('FURNISHED','SEMIF...	YES		NULL	
NO_OF_PARKIN...	int	YES		NULL	
NO_OF_BATHR...	int	YES		NULL	
PET_ALLOWANCE	tinyint(1)	YES		NULL	
WATER_AVAILA...	float	YES		NULL	
FINALISED	tinyint(1)	YES		NULL	
BOOKED_BY	int	YES		NULL	

- RENT:

creation*

Don't Limit

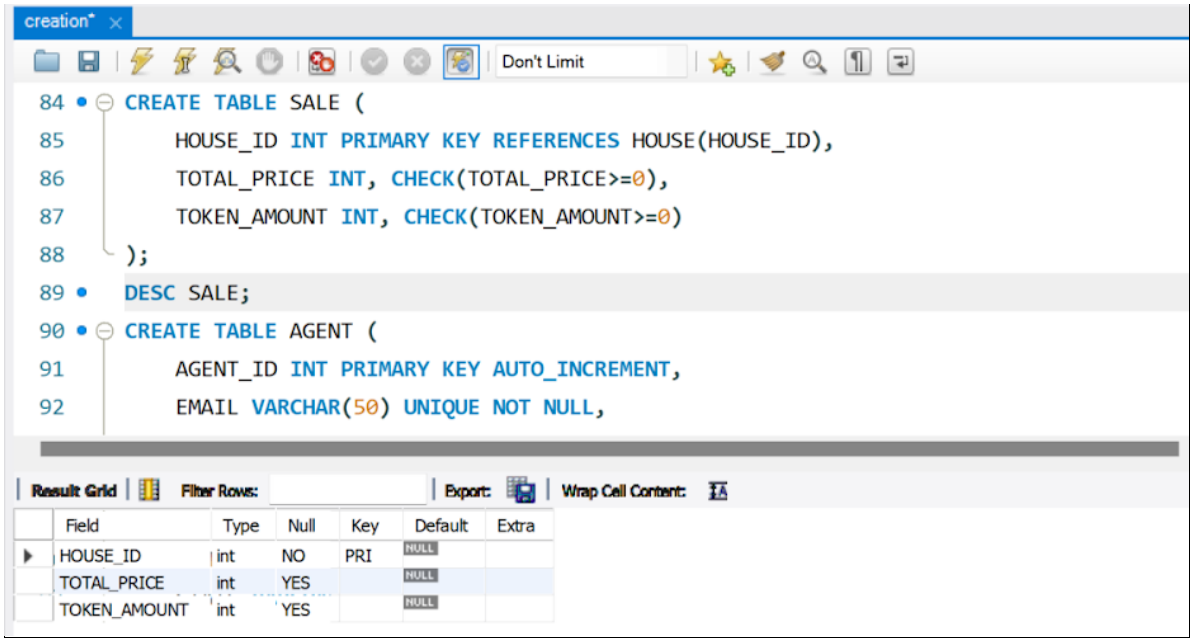
```
75  CREATE TABLE RENT (
76      HOUSE_ID INT PRIMARY KEY REFERENCES HOUSE(HOUSE_ID),
77      MONTHLY_RENT INT NOT NULL, CHECK(MONTHLY_RENT>0),
78      MAX_NO_OF_TENANTS INT, CHECK(MAX_NO_OF_TENANTS>0),
79      TENANT_MARITAL_STATUS BOOLEAN,
80      TENANT_FOOD_HABIT BOOLEAN,
81      SECURITY_DEPOSIT_AMOUNT INT, CHECK(SECURITY_DEPOSIT_AMOUNT>=0)
82  );
83  DESC RENT;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

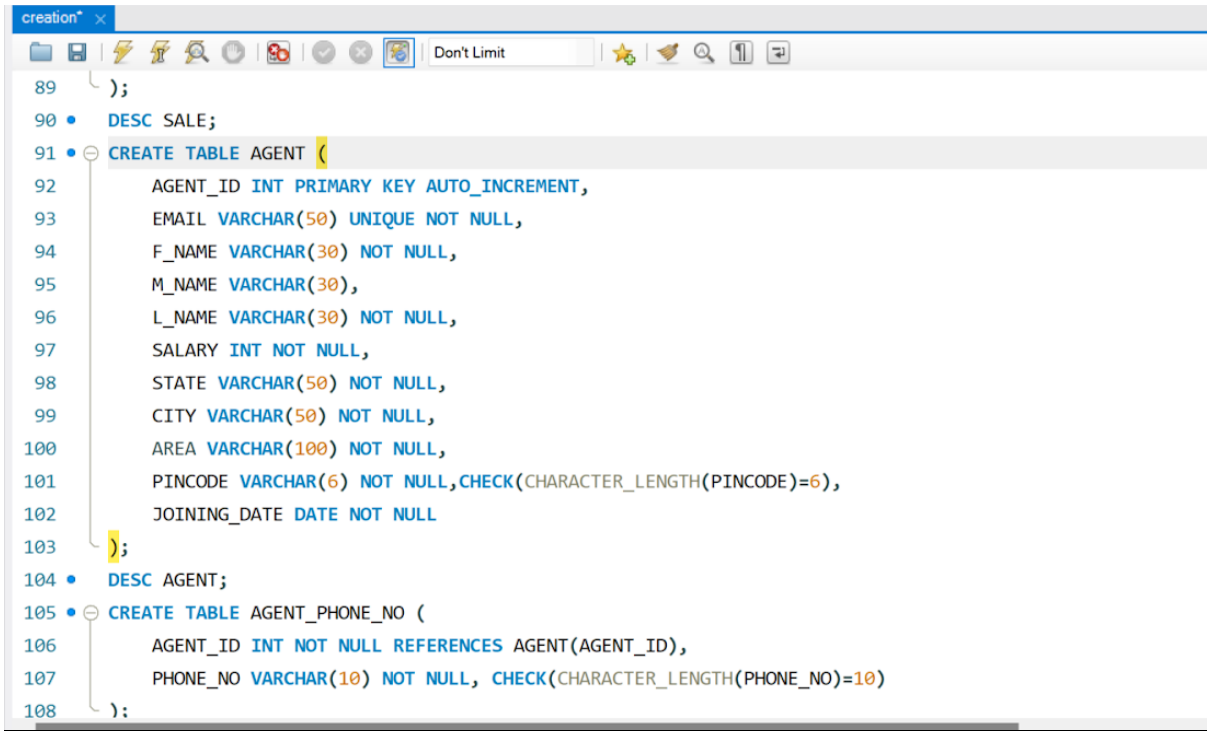
Field	Type	Null	Key	Default	Extra
HOUSE_ID	int	NO	PRI	NULL	
MONTHLY_RENT	int	NO		NULL	
MAX_NO_OF_TENANTS	int	YES		NULL	
TENANT_MARITAL_STATUS	tinyint(1)	YES		NULL	
TENANT_FOOD_HABIT	tinyint(1)	YES		NULL	
SECURITY_DEPOSIT_AMOUNT	int	YES		NULL	

- SALE:

CHECK constraints ensure that negative values are not entered in TOTAL_PRICE and TOKEN_AMOUNT columns.



- AGENT:



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
AGENT_ID	int	NO	PRI	NULL	auto_increment
EMAIL	varchar(50)	NO	UNI	NULL	
F_NAME	varchar(30)	NO		NULL	
M_NAME	varchar(30)	YES		NULL	
L_NAME	varchar(30)	NO		NULL	
SALARY	int	NO		NULL	
STATE	varchar(50)	NO		NULL	
CITY	varchar(50)	NO		NULL	
AREA	varchar(100)	NO		NULL	
PINCODE	varchar(6)	NO		NULL	
JOINING_...	date	NO		NULL	

- AGENT_PHONE_NO:

creation*

Don't Limit

```
103     );
104 •   DESC AGENT;
105 •   CREATE TABLE AGENT_PHONE_NO (
106         AGENT_ID INT NOT NULL REFERENCES AGENT(AGENT_ID),
107         PHONE_NO VARCHAR(10) NOT NULL, CHECK(CHARACTER_LENGTH(PHONE_NO)=10)
108     );
109 •   DESC AGENT_PHONE_NO;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
AGENT_ID	int	NO		NULL	
PHONE_NO	varchar(10)	NO		NULL	

- PAYMENT:

```
creation*
Don't Limit
108 • CREATE TABLE PAYMENT (
109       HOUSE_ID INT NOT NULL REFERENCES HOUSE(HOUSE_ID),
110       TRANSACTION_NUMBER VARCHAR(20) NOT NULL,
111       PAYMENT_MODE ENUM("UPI","NET BANKING","CARD") NOT NULL,
112       UPIID_ACCTNO_CARDNO VARCHAR(30) NOT NULL,
113       BENEDICIARY_UPIID_ACCTNO VARCHAR(30) NOT NULL,
114       P_DATE DATE NOT NULL,
115       P_TIME TIME NOT NULL,
116       TYPE ENUM("SECURITY DEPOSIT","TOKEN OF CONFIRMATION")  NOT NULL
117   );
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
HOUSE_ID	int	NO		NULL	
TRANSACTION_NUMBER	varchar(20)	NO		NULL	
PAYMENT_MODE	enum('UPI','NET BANKING','CARD')	NO		NULL	
UPIID_ACCTNO_CARDNO	varchar(30)	NO		NULL	
BENEDICIARY_UPIID_ACCTNO	varchar(30)	NO		NULL	
P_DATE	date	NO		NULL	
P_TIME	time	NO		NULL	
TYPE	enum('SECURITY DEPOSIT','TOKEN OF CONFIRM...	NO		NULL	

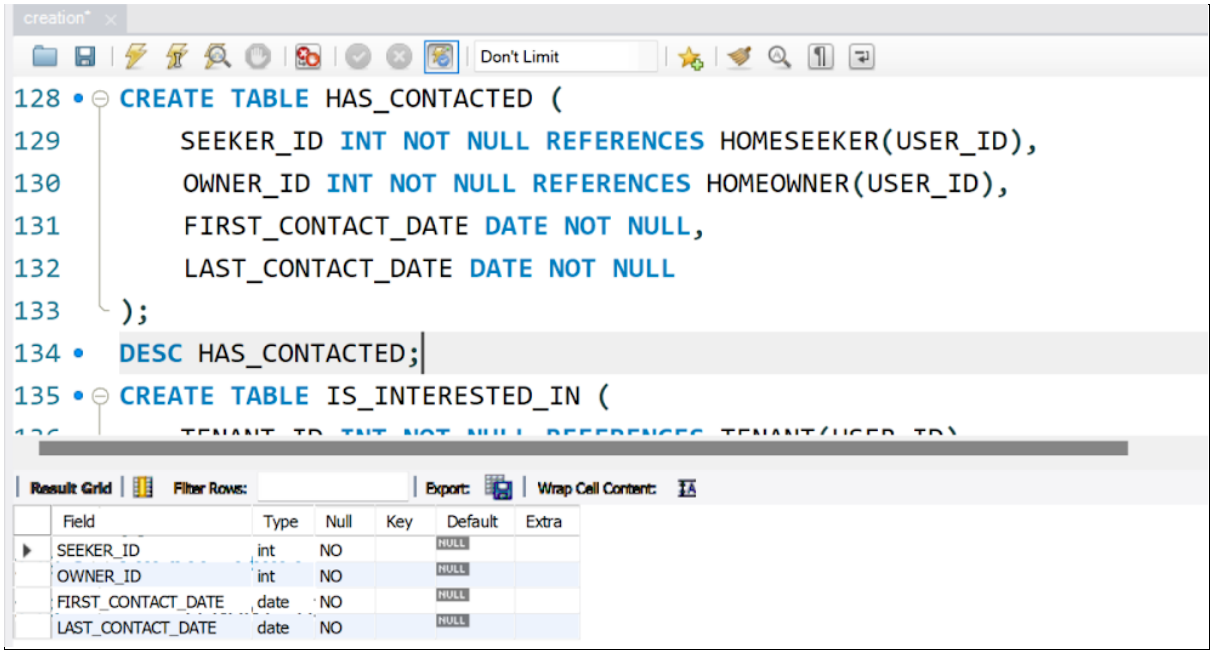
- VISITS:

```
creation*
Don't Limit
120 • CREATE TABLE VISITS (
121       AGENT_ID INT NOT NULL REFERENCES AGENT(AGENT_ID),
122       HOUSE_ID INT NOT NULL REFERENCES HOUSE(HOUSE_ID),
123       SEEKER_ID INT NOT NULL REFERENCES HOMESEEKER(USER_ID),
124       VISIT_DATE DATE NOT NULL,
125       VISIT_TIME TIME NOT NULL
126   );
127 •  DESC VISITS;
```

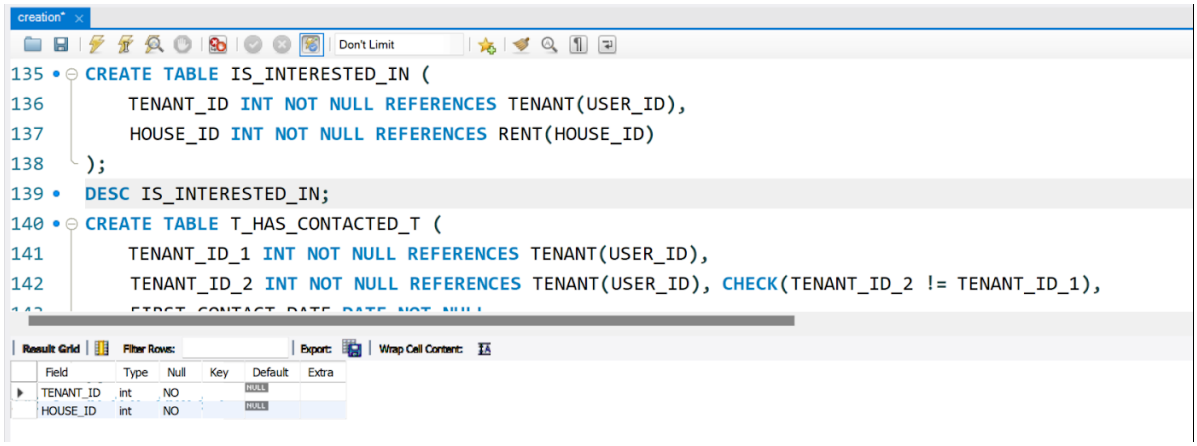
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Field	Type	Null	Key	Default	Extra
AGENT_ID	int	NO		NULL	
HOUSE_ID	int	NO		NULL	
SEEKER_ID	int	NO		NULL	
VISIT_DATE	date	NO		NULL	
VISIT_TIME	time	NO		NULL	

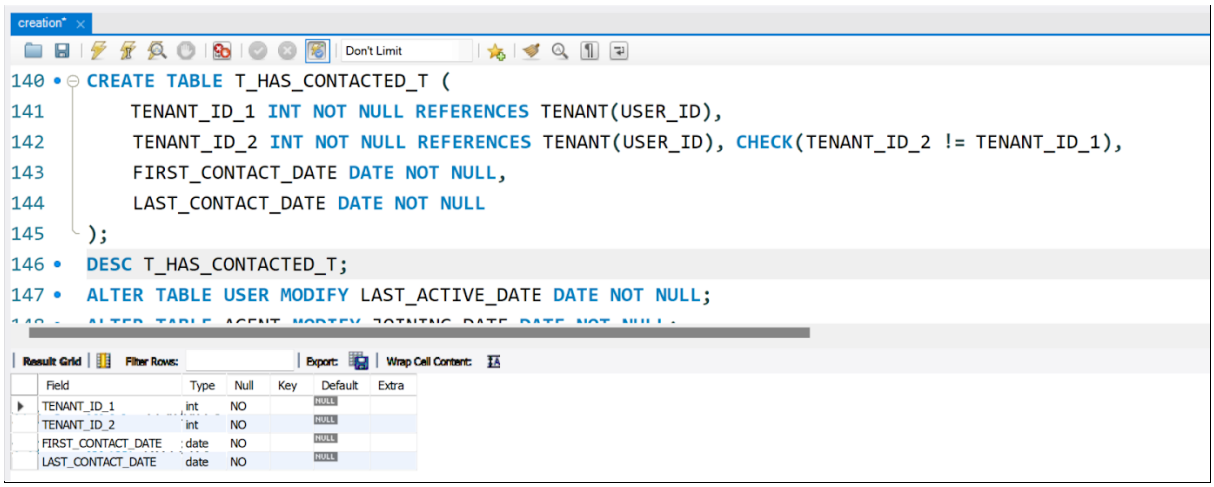
- HAS_CONTACTED:



- IS_INTERESTED_IN:



- TENANT_CONTACTS_TEANANT:



In this way schema of all the tables is created using DDL commands.

DML :

Query 1:
Find the house details with housetype=bunglow

SELECT * FROM HOUSE WHERE TYPE="BUNGALOW";

```
225 • SELECT * FROM HOUSE WHERE TYPE="BUNGALOW";
226
227
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

HOUSE_ID	OWNER_ID	STATE	CITY	AREA	PINCODE	TYPE	BOUGHT_IN	NO_OF_ROOMS	FLOOR	SIZE	FURNISHING_STATUS	NO_OF_PARKING_SPACES	NO_OF_BATHR
2	12	Karnataka	Bangalore	ITPL	400088	BUNGALOW	1985	2	0	410	SEMIFURNISHED	1	1
3	14	Telangana	Hyderabad	Dharmapuri Colony	400010	BUNGALOW	2018	4	0	1060	SEMIFURNISHED	2	2
5	16	Delhi	Delhi	Vinod Nagar West	400064	BUNGALOW	1997	2	1	790	UNFURNISHED	0	0
8	23	Maharashtra	Mumbai	Kohinoor City Phase II, Holly Cross Church Cho...	400009	BUNGALOW	2006	2	1	1050	SEMIFURNISHED	0	0
23	53	Karnataka	Bangalore	Hommadevanahalli	400051	BUNGALOW	2005	4	1	1150	FURNISHED	2	2
26	57	Maharashtra	Mumbai	Hubtown Gardenia, Mira Road	400086	BUNGALOW	2015	1	8	1070	UNFURNISHED	2	2
35	76	Telangana	Hyderabad	Kondapur	400059	BUNGALOW	2001	1	0	1080	UNFURNISHED	3	3
38	82	Telangana	Hyderabad	Narsingi, Outer Ring Road	400013	BUNGALOW	2022	1	9	630	UNFURNISHED	5	5
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Query 2:
To display the house_id, area of houses having no_of_rooms=3

select house_id,area,no_of_rooms from house where NO_OF_ROOMS=3;

```
260
261     -- 14 dml:
262 •   select house_id,area,no_of_rooms from house where NO_OF_ROOMS=3;
263
264     -- 15 dml:
265 •   select l_name,f_name,email,salary from agent where salary>65000;
266
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

house_id	area	no_of_rooms
6	Lajpat Nagar IV	3
13	A Narayanapura	3
16	Tambaram West, Tambaram, Chennai Bypass R...	3
19	Ganguly Bagan	3
20	Periyapalayam	3
29	Kannan Colony, Pazhavanthangal	3

Query 3:
To display the f and l name of agents and email-id having salary>65000

select l_name,f_name,email,salary from agent where salary>65000;

```
263
264     -- 15 dml:
265 •   select l_name,f_name,email,salary from agent where salary>65000;
266
267     -- 16 views:
268 •   create view house_info as (select owner_id, city, area, type, no_of_rooms, floor from
269 •   select * from house_info;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

l_name	f_name	email	salary
Agarwal	Shaan	eqynsjkx@gmail.com	77000

Query 4:
To display user id of tenants having marital status-0

SELECT USER_ID from TENANT where MARITAL_STATUS=0;

```
265
266     -- dml
267 •   SELECT USER_ID from TENANT where MARITAL_STATUS=0;
268
269
270     -- 16 views:
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap C

USER_ID
5
10
20
21
49
71
74
78
81
86
92
96
97
100
NULL

DATE , SET AND ARITHMETIC OPERATIONS:

Query 1:
To decrease the total price of house on sale in city delhi by 2 percent of the actual price – arithmetic.
SELECT HOUSE_ID,TOTAL_PRICE-0.02*TOTAL_PRICE as new_price,AREA,TYPE FROM HOUSE natural join SALE WHERE CITY="Delhi";

```
228     -- 3 arithmetic:
229 •   SELECT HOUSE_ID,TOTAL_PRICE-0.02*TOTAL_PRICE as new_price,AREA,TYPE FROM HOUSE natural join SALE WHERE CITY="Delhi";
230
231     -- 4 join:
232 •   SELECT CITY,MONTHLY_RENT,SIZE,FURNISHING_STATUS FROM HOUSE natural join RENT WHERE BOUGHT_IN=1985;
233
234     -- 5 aggregate (try to use groupby having on one of these):
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

HOUSE_ID	new_price	AREA	TYPE
6	6370000.00	Lajpat Nagar IV	FLAT
36	20482000.00	Jhangir Puri	FLAT

This reduces the prices of house for sale in city Delhi by 2% showing new price. The proce before sale was as shown below:
Old price

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

HOUSE_ID	TOTAL_PRICE	AREA	TYPE
6	6500000	Lajpat Nagar IV	FLAT
36	20900000	Jhangir Puri	FLAT

Query 2:
To increment the salary of agent by 8 percent.
select F_NAME,L_NAME,SALARY+0.08*SALARY AS NEW_SALARY FROM AGENT;

```
260 •  select F_NAME,L_NAME,SALARY+0.08*SALARY AS NEW_SALARY FROM AGENT;
261
262    -- 14 dml:
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

F_NAME	L_NAME	NEW_SALARY
Shaan	Agarwal	83160.00
Deepanshu	Aggarwal	24840.00
Rahil	Ankhad	42120.00
Mohammed	Ansari	70200.00
Utsav	Avaiya	60480.00
Rajas	Baadkar	44280.00
Naman	Badlani	59400.00
Saniya	Bangare	65880.00
Kunal	Bhatia	21600.00
Umang	Bhatt	41040.00
Basuri	Bhujade	39960.00
Eshan	Bhuse	25920.00
Akshat	Biniwale	34560.00
Ayush	Bodade	22680.00
Vaishnavi	Borkar	59400.00

Query 3:
To display the rent house details where monthly rent is 10k ,15k ,20k.

select * from rent where monthly_rent in(10000,15000,20000);

```
257 •  select * from rent where monthly_rent in(10000,15000,20000);
258
259    -- 13 arithmetiv:
260 •  select F_NAME,L_NAME,SALARY+0.08*SALARY AS NEW_SALARY FROM AGENT;
261
262    -- 14 dml:
263 •  select house_id,area,no_of_rooms from house where NO_OF_ROOMS=3;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

HOUSE_ID	MONTHLY_RENT	MAX_NO_OF_TENANTS	TENANT_MARITAL_STATUS	TENANT_FOOD_HABIT	SECURITY_DEPOSIT_AMOUNT
0	20000	2	1	1	61000
5	10000	1	1	1	149000
16	10000	2	0	0	93000
21	10000	1	1	1	17000
32	10000	4	0	1	156000
NULL	NULL	NULL	NULL	NULL	NULL

This displays house on rent details having rent 10k or 20k or 15k using set operation,

Query 4:
To display details of houses on rent in Mumbai given area is Mahim or Cosmopolitan tower,Andheri west.

SELECT AREA,TYPE,NO_OF_ROOMS,MONTHLY_RENT FROM HOUSE NATURAL JOIN RENT WHERE CITY="Mumbai" AND AREA IN("Mahim","Cosmopolis Tower, Andheri West");

```
278    -- set
279 •  SELECT AREA,TYPE,NO_OF_ROOMS,MONTHLY_RENT FROM HOUSE NATURAL JOIN RENT WHERE CITY="Mumbai" AND AREA IN("Mahim","Cosmopolis Tower, Andheri West");
280
281 •  TRUNCATE TABLE USER;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

AREA	TYPE	NO_OF_ROOMS	MONTHLY_RENT
Mahim	FLAT	4	40000
Cosmopolis Tower, Andheri West	FLAT	2	10000

Query 5:
To display system date.

```
247 •  SELECT sysdate() from dual;
248
```

Result Grid | Filter Rows: | Export:

sysdate()
2022-11-30 00:21:35

Query 6:

To display the date after adding 6 months to the joining date of agent where agent city is Mumbai.
SELECT AGENT_ID,F_NAME,M_NAME,L_NAME,DATE_ADD(JOINING_DATE,INTERVAL 6 MONTH) AS NEW_DATE_OF_JOINING FROM AGENT WHERE CITY="Mumbai";

```
245 •  SELECT AGENT_ID,F_NAME,M_NAME,L_NAME,DATE_ADD(JOINING_DATE,INTERVAL 6 MONTH) AS NEW_DATE_OF_JOINING FROM AGENT WHERE CITY="Mumbai";
246
247    -- 10 views:
248 •  create view house_on_rent_in_chennai as (select * from house natural join rent where CITY="Chennai");
249 •  select * from house_on_rent_in_chennai;
```

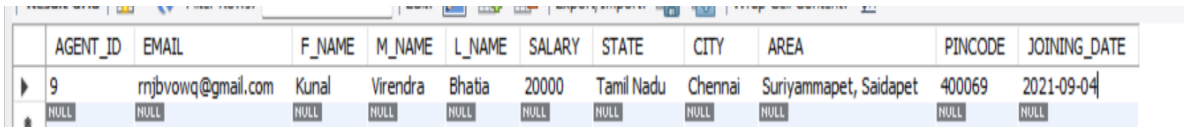
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

AGENT_ID	F_NAME	M_NAME	L_NAME	NEW_DATE_OF_JOINING
3	Rahil	Amit	Ankhad	2020-12-18
5	Utsav	Ghanshyam	Avaiya	2020-02-16
10	Umang	Bhavesh	Bhatt	2023-01-24
14	Ayush	Vivekkumar	Bodade	2021-09-24

This adds six months to joining dates of the agent belonging to city Mumbai.

AGGREGATE AND GROUP BY HAVING CLAUSE:

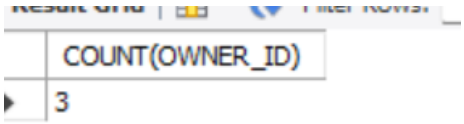
Query 1:
To display agent details with minimum salary.
SELECT * FROM AGENT WHERE SALARY=(SELECT MIN(SALARY) FROM AGENT);



AGENT_ID	EMAIL	F_NAME	M_NAME	L_NAME	SALARY	STATE	CITY	AREA	PINCODE	JOINING_DATE
9	rnjbvowq@gmail.com	Kunal	Virendra	Bhatia	20000	Tamil Nadu	Chennai	Suriyammapet, Saidapet	400069	2021-09-04
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

This shows agent detail with salary 20k which is the minimum among all.

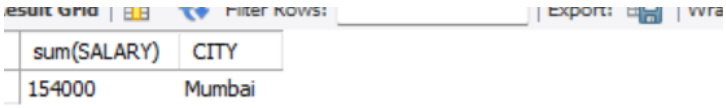
Query 2:
To count the number of houses owned by owner with id=29
SELECT COUNT(OWNER_ID) FROM HOUSE WHERE OWNER_ID=29;



COUNT(OWNER_ID)
3

This means owner with is 29 has registered 3 houses.

Query 3:
To display total salary of all the agents from city Mumbai.
SELECT sum(SALARY),CITY
from AGENT
GROUP BY CITY
having CITY='Mumbai'



sum(SALARY)	CITY
154000	Mumbai

This groups all the tuples with city Mumbai and add the salaries of those agent using group by having clause.(39k+56k+38k+21k=154k)

JOIN OPERATIONS:

Query 1:
To display information of houses on rent.

SELECT * FROM HOUSE NATURAL JOIN (SELECT HOUSE_ID,MONTHLY_RENT,MAX_NO_OF_TENANTS FROM RENT) as A;

```
166     -- 1 join:
167 •   SELECT * FROM HOUSE NATURAL JOIN (SELECT HOUSE_ID,MONTHLY_RENT,MAX_NO_OF_TENANTS FROM RENT) as House_on_rent;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

HOUSE_ID	OWNER_ID	STATE	CITY	AREA	PINCODE	TYPE	BOUGHT_IN	NO_OF_ROOMS	FLOOR	SIZE	FURNISHING_STATUS	NO_OF_PARKING_SPACES	
1	9	Delhi	Delhi	Munirka Vihar, Munirka	400010	FLAT	1994	2	1	810	SEMIFURNISHED	0	
5	16	Delhi	Delhi	Vinod Nagar West	400064	BUNGALOW	1997	2	1	790	UNFURNISHED	0	
7	19	Telangana	Hyderabad	Pragathi Nagar, Moosapet	400070	FLAT	2021	5	0	1250	SEMIFURNISHED	2	
10	29	Tamil Nadu	Chennai	Lake Area, Nungambakkam	400057	FLAT	1982	2	0	950	UNFURNISHED	1	
16	40	Tamil Nadu	Chennai	Tambaram West, Tambaram, Chennai Bypass ...	400057	FLAT	1982	3	6	1110	UNFURNISHED	1	
17	41	Maharashtra	Mumbai	JB Nagar	400051	FLAT	2021	2	1	450	UNFURNISHED	0	
20	45	Tamil Nadu	Chennai	Periyapalayam	400066	FLAT	1980	3	1	100	SEMIFURNISHED	2	
21	47	Tamil Nadu	Chennai	Suriyammapet, Saidapet	400069	FLAT	1983	1	3	560	FURNISHED	1	
25	55	Karnataka	Bangalore	KR Puram	400078	FLAT	1994	2	7	1030	UNFURNISHED	2	
26	57	Maharashtra	Mumbai	Hubtown Gardenia, Mira Road	400086	BUNGALOW	2015	1	8	1070	UNFURNISHED	2	
27	58	West Bengal	Kolkata	East Park	400028	FLAT	1988	2	0	1060	UNFURNISHED	4	
28	61	Tamil Nadu	Chennai	Gowrivakkam	400052	FLAT	1984	2	12	900	UNFURNISHED	1	
29	63	Tamil Nadu	Chennai	Kannan Colony, Pazhavanthangal	400027	FLAT	1983	3	3	600	FURNISHED	4	
30	64	Maharashtra	Mumbai	Mahim	400004	FLAT	1985	4	5	740	UNFURNISHED	1	
31	67	Tamil Nadu	Chennai	Nanganallur	400051	FLAT	2001	2	0	510	UNFURNISHED	2	
32	69	Maharashtra	Mumbai	Cosmopolis Tower, Andheri West	400011	FLAT	2006	2	3	950	UNFURNISHED	2	

This joins the HOUSE and RENT table since both have HOUSE_ID as the common attribute.

Query 2:
To find city, rent, size and furnishing status of houses on rent bought in=1985 year

SELECT CITY,MONTHLY_RENT,SIZE,FURNISHING_STATUS FROM HOUSE natural join RENT WHERE BOUGHT_IN=1985;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CITY	MONTHLY_RENT	SIZE	FURNISHING_STATUS
Mumbai	40000	740	UNFURNISHED

Query 3:
To display homeowner information having properties in city Mumbai.
SELECT * from USER natural join HOMEOWNER where city='Mumbai'

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

USER_ID	EMAIL	PASSWORD	F_NAME	M_NAME	L_NAME	DOB	GENDER	ACCT_CREATION_DATE	LAST_ACTIVE_DATE	STATE	CITY	AREA	PIN
1	ibnsqrei@gmail.com	NI4"B6jn#pr\35	Anushka	Santosh	Acharya	1989-06-02	FEMALE	2017-12-16	2019-12-18	Maharashtra	Mumbai	Mazgaon Dock	4000
6	vnaofvgf@gmail.com	&P"!yjB5GCq/g	Darshit	Vikram	Bhagtani	1993-05-12	MALE	2021-10-10	2022-07-12	Maharashtra	Mumbai	International Airport	4000
7	emglewjl@gmail.com	2xwpd6kK#e:	Asmi	Shailesh	Bhanushali	1962-01-10	FEMALE	2019-03-07	2020-07-19	Maharashtra	Mumbai	Kurla	4000
9	kgkbrxpc@gmail.com	3P^muE9z	Aditi	Nilesh	Bhutada	1986-06-03	FEMALE	2017-06-28	2022-04-25	Maharashtra	Mumbai	Mulund West	4000
12	yptskvbe@gmail.com	iWxStKw8v7	Rahul	Chandrashekhar	Chalwadi	1992-10-25	MALE	2021-08-23	2022-08-26	Maharashtra	Mumbai	Bhawani Shankar Rd	4000
14	xylkkylp@gmail.com	j|QoI/@Z	Om	Chhaganlal	Chandra	1976-09-07	MALE	2018-04-14	2020-09-06	Maharashtra	Mumbai	Princess Dock	4000
15	ecwlftly@gmail.com	7AGXK6lDWEfA	Sujal	Mahendra	Chordia	1993-07-26	MALE	2017-07-22	2017-09-07	Maharashtra	Mumbai	Goregaon East	4000
16	qlnovdcd@gmail.com	P&sPE:#I?|^j	Vedant	Tushar	Dapolikar	2004-02-24	MALE	2017-04-16	2021-08-13	Maharashtra	Mumbai	Vileparle Railway Station	4000
18	nxlygkxo@gmail.com	EFw$"UFR*y@a	Omkar	Ravikant	Deshmukh	1965-02-22	MALE	2022-09-17	2022-10-20	Maharashtra	Mumbai	V P Road	4000
19	zfvtovju@gmail.com	SDuuGDaQ	Reshma	Lahanu	Dhadwad	1962-09-13	FEMALE	2019-05-03	2020-11-10	Maharashtra	Mumbai	Santacruz P&t Colony	4000
23	qgfuoned@gmail.com	59|dD0L&Jj1!	Aditya	Sushil	Gangal	1971-08-31	MALE	2018-10-17	2021-07-05	Maharashtra	Mumbai	Audit Bhavan	4000
24	zfrgltff@gmail.com	SA00/RWkAkDvz9	Vikiraj	Barku	Ghanghav	2003-01-26	MALE	2020-10-23	2021-02-09	Maharashtra	Mumbai	Vihar Road	4000
29	fbvdvaup@gmail.com	Ix5bQ%ubh	Bhushan	Sadashiv	Jadhav	2002-04-24	MALE	2020-09-04	2022-07-17	Maharashtra	Mumbai	V J B Udyan	4000
31	hafrwbsj@gmail.com	$nhZrbH!edQ^	Rohan	Vimal	Jaiswal	1975-07-28	MALE	2019-12-11	2021-11-06	Maharashtra	Mumbai	B N Bhavan	4000
33	svowzwao@gmail.com	Nxn8q2PDVJ:Y	Tanmay	Vivek	Kulkarni	1989-06-03	MALE	2022-08-03	2022-10-20	Maharashtra	Mumbai	Shivaji Park Mumbai	4000
37	yuxuqjjm@gmail.com	JMm*?HSi\/38!	Sanimar	Singh	Manghera	1971-06-04	MALE	2017-05-30	2020-07-19	Maharashtra	Mumbai	Chunabhatti	4000

Result 24

SUBQUERY:

Query 1:
To display agent details with maximum salary.
SELECT * FROM AGENT WHERE SALARY=(SELECT MAX(SALARY) FROM AGENT);

AGENT_ID	EMAIL	F_NAME	M_NAME	L_NAME	SALARY	STATE	CITY	AREA	PINCODE	JOINING_DATE
1	eqynsjkx@gmail.com	Shaan		Agarwal	77000	Karnataka	Bangalore	A Narayanapura	400037	2019-07-20
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Query 2:
To display agent details with salary greater than average salary.

```
89 •  SELECT * FROM AGENT WHERE SALARY>(SELECT AVG(SALARY) FROM AGENT);
90    -- 8 aggregate:
91 •  SELECT COUNT(OWNER_ID) FROM HOUSE WHERE OWNER_ID=29;
92
```

esult Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

AGENT_ID	EMAIL	F_NAME	M_NAME	L_NAME	SALARY	STATE	CITY	AREA	PINCODE	JOINING_D
1	eqynsjkx@gmail.com	Shaan		Agarwal	77000	Karnataka	Bangalore	A Narayanapura	400037	2019-07-20
4	mwthtqvu@gmail.com	Mohammed	Shanouf Valijan	Ansari	65000	Tamil Nadu	Chennai	Tambaram West, Tambaram, Chennai Bypass R...	400057	2019-04-11
5	xmiexfbp@gmail.com	Utsav	Ghanshyam	Avaiya	56000	Maharashtra	Mumbai	JB Nagar	400051	2019-08-16
7	mndrtrka@gmail.com	Naman	Ashok	Badlani	55000	West Bengal	Kolkata	Ganguly Bagan	400029	2021-05-29
8	frowcwic@gmail.com	Saniya	Sadhuram	Bangare	61000	Tamil Nadu	Chennai	Periyapalayam	400066	2020-02-25
15	wgopgqqk@gmail.com	Vaishnavi	Bhagawan	Borkar	55000	West Bengal	Kolkata	East Park	400028	2019-11-04
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Query 3:
To display house details with city, state, area, number of rooms ,furnishing status with monthly rent less than average rent.

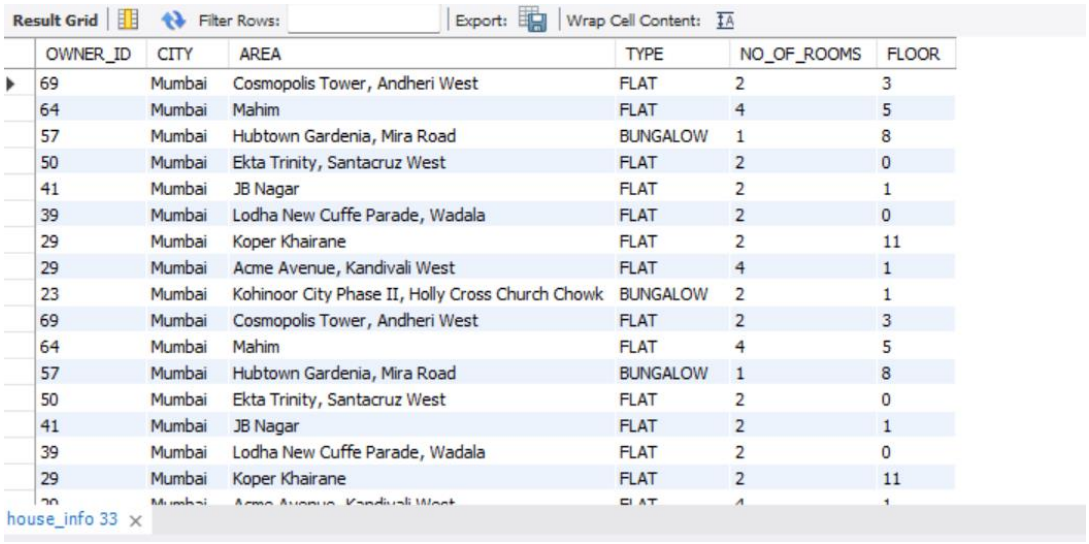
select HOUSE_ID,OWNER_ID,STATE,CITY,AREA,FLOOR,FURNISHING_STATUS,MONTHLY_RENT,NO_OF_ROOMS from house_on_rent where MONTHLY_RENT<=(select avg(MONTHLY_RENT) from RENT) and TYPE='FLAT'

esult Grid | Filter Rows: | Export: | Wrap Cell Content:

HOUSE_ID	OWNER_ID	STATE	CITY	AREA	FLOOR	FURNISHING_STATUS	MONTHLY_RENT	NO_OF_ROOMS
7	19	Telangana	Hyderabad	Pragathi Nagar, Moosapet	0	SEMIFURNISHED	7500	5
10	29	Tamil Nadu	Chennai	Lake Area, Nungambakkam	0	UNFURNISHED	7000	2
16	40	Tamil Nadu	Chennai	Tambaram West, Tambaram, Chennai Bypass R...	6	UNFURNISHED	10000	3
17	41	Maharashtra	Mumbai	JB Nagar	1	UNFURNISHED	5000	2
21	47	Tamil Nadu	Chennai	Suriyammapet, Saidapet	3	FURNISHED	10000	1
27	58	West Bengal	Kolkata	East Park	0	UNFURNISHED	6500	2
28	61	Tamil Nadu	Chennai	Gowrivakkam	12	UNFURNISHED	5500	2
29	63	Tamil Nadu	Chennai	Kannan Colony, Pazhavanthangal	3	FURNISHED	8500	3
31	67	Tamil Nadu	Chennai	Nanganallur	0	UNFURNISHED	6000	2
32	69	Maharashtra	Mumbai	Cosmopolis Tower, Andheri West	3	UNFURNISHED	10000	2
33	70	Karnataka	Bangalore	indiranagar near bda complex	0	FURNISHED	11000	4
37	79	West Bengal	Kolkata	Birati	1	SEMIFURNISHED	6000	4
39	84	Telangana	Hyderabad	Secunderabad	4	FURNISHED	7900	2

VIEW:

Query 1:
To create a view for displaying house info from city Mumbai.
create view house_info
as (select OWNER_ID,CITY,AREA,TYPE,NO_OF_ROOMS,FLOOR from HOUSE natural join HOMEOWNER where CITY='Mumbai');
select *from house_info;



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

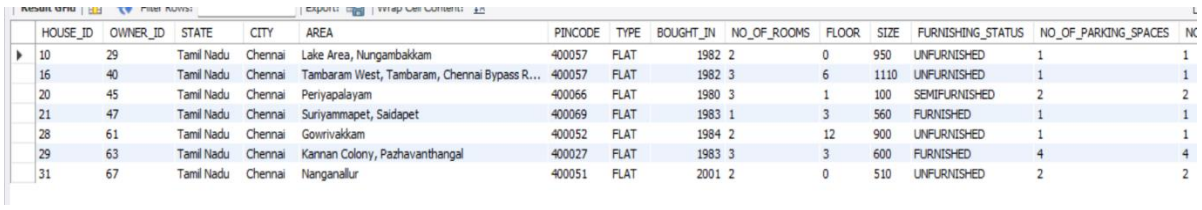
OWNER_ID	CITY	AREA	TYPE	NO_OF_ROOMS	FLOOR
69	Mumbai	Cosmopolis Tower, Andheri West	FLAT	2	3
64	Mumbai	Mahim	FLAT	4	5
57	Mumbai	Hubtown Gardenia, Mira Road	BUNGALOW	1	8
50	Mumbai	Ekta Trinity, Santacruz West	FLAT	2	0
41	Mumbai	JB Nagar	FLAT	2	1
39	Mumbai	Lodha New Cuffe Parade, Wadala	FLAT	2	0
29	Mumbai	Koper Khairane	FLAT	2	11
29	Mumbai	Acme Avenue, Kandivali West	FLAT	4	1
23	Mumbai	Kohinoor City Phase II, Holly Cross Church Chowk	BUNGALOW	2	1
69	Mumbai	Cosmopolis Tower, Andheri West	FLAT	2	3
64	Mumbai	Mahim	FLAT	4	5
57	Mumbai	Hubtown Gardenia, Mira Road	BUNGALOW	1	8
50	Mumbai	Ekta Trinity, Santacruz West	FLAT	2	0
41	Mumbai	JB Nagar	FLAT	2	1
39	Mumbai	Lodha New Cuffe Parade, Wadala	FLAT	2	0
29	Mumbai	Koper Khairane	FLAT	2	11

house_info 33 ×

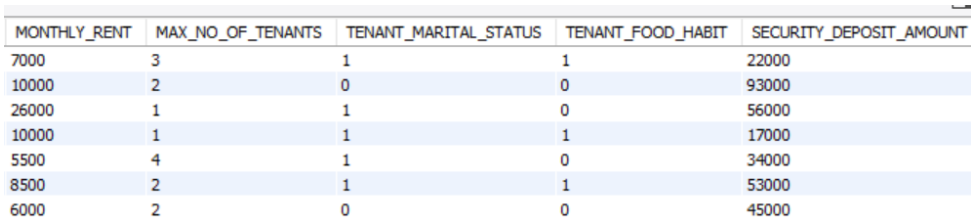
Query 2:

To display details of house on rent with city=Mumbai and security deposit<80000

create view house_rent as (select * from house natural join rent);
select * from house_rent where SECURITY_DEPOSIT_AMOUNT<100000 and city='Chennai';



HOUSE_ID	OWNER_ID	STATE	CITY	AREA	PINCODE	TYPE	BOUGHT_IN	NO_OF_ROOMS	FLOOR	SIZE	FURNISHING_STATUS	NO_OF_PARKING_SPACES	NO
10	29	Tamil Nadu	Chennai	Lake Area, Nungambakkam	400057	FLAT	1982	2	0	950	UNFURNISHED	1	1
16	40	Tamil Nadu	Chennai	Tambaram West, Tambaram, Chennai Bypass R...	400057	FLAT	1982	3	6	1110	UNFURNISHED	1	1
20	45	Tamil Nadu	Chennai	Periyapalayam	400066	FLAT	1980	3	1	100	SEMIFURNISHED	2	2
21	47	Tamil Nadu	Chennai	Suriyammapet, Saidapet	400069	FLAT	1983	1	3	560	FURNISHED	1	1
28	61	Tamil Nadu	Chennai	Gowrivakkam	400052	FLAT	1984	2	12	900	UNFURNISHED	1	1
29	63	Tamil Nadu	Chennai	Kannan Colony, Pazhavanthangal	400027	FLAT	1983	3	3	600	FURNISHED	4	4
31	67	Tamil Nadu	Chennai	Nanganallur	400051	FLAT	2001	2	0	510	UNFURNISHED	2	2

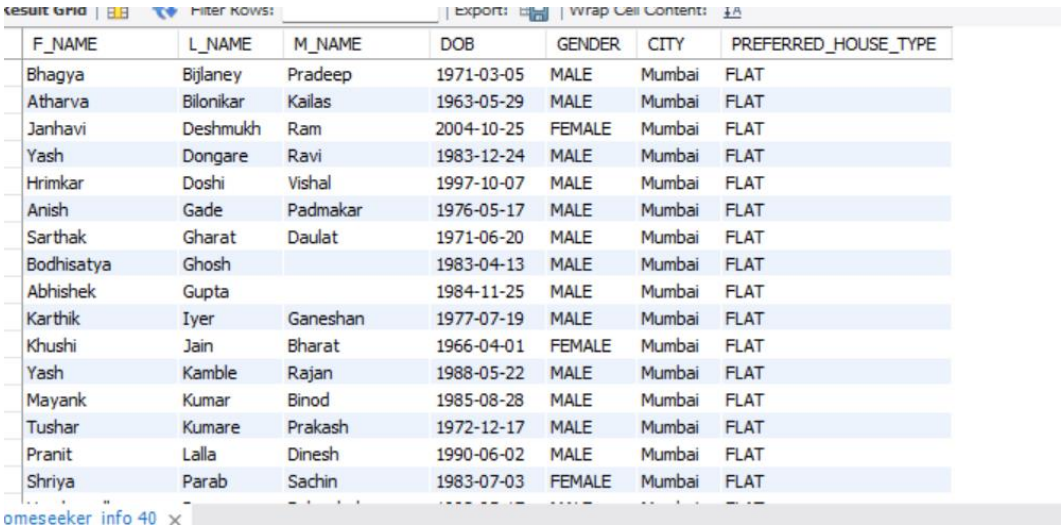


MONTHLY_RENT	MAX_NO_OF_TENANTS	TENANT_MARITAL_STATUS	TENANT_FOOD_HABIT	SECURITY_DEPOSIT_AMOUNT
7000	3	1	1	22000
10000	2	0	0	93000
26000	1	1	0	56000
10000	1	1	1	17000
5500	4	1	0	34000
8500	2	1	1	53000
6000	2	0	0	45000

Creating view helps by not using join command repetitively and helps to create a table house_on_rent on which all queries can be performed.

Query 3:
To display information of user who is looking for home in city Mumbai with preferred type as BUNGALOW.

create view homeseeker_info as (select *from USER natural join HOMESEEKER);
select F_NAME,L_NAME,M_NAME,DOB,GENDER,CITY,PREFERRED_HOUSE_TYPE from homeseeker_info where city='Mumbai' and PREFERRED_HOUSE_TYPE='FLAT';



F_NAME	L_NAME	M_NAME	DOB	GENDER	CITY	PREFERRED_HOUSE_TYPE
Bhagya	Bijlaney	Pradeep	1971-03-05	MALE	Mumbai	FLAT
Atharva	Bilonikar	Kailas	1963-05-29	MALE	Mumbai	FLAT
Janhavi	Deshmukh	Ram	2004-10-25	FEMALE	Mumbai	FLAT
Yash	Dongare	Ravi	1983-12-24	MALE	Mumbai	FLAT
Hrimkar	Doshi	Vishal	1997-10-07	MALE	Mumbai	FLAT
Anish	Gade	Padmakar	1976-05-17	MALE	Mumbai	FLAT
Sarthak	Gharat	Daulat	1971-06-20	MALE	Mumbai	FLAT
Bodhisatya	Ghosh		1983-04-13	MALE	Mumbai	FLAT
Abhishek	Gupta		1984-11-25	MALE	Mumbai	FLAT
Karthik	Iyer	Ganeshan	1977-07-19	MALE	Mumbai	FLAT
Khushi	Jain	Bharat	1966-04-01	FEMALE	Mumbai	FLAT
Yash	Kamble	Rajan	1988-05-22	MALE	Mumbai	FLAT
Mayank	Kumar	Binod	1985-08-28	MALE	Mumbai	FLAT
Tushar	Kumare	Prakash	1972-12-17	MALE	Mumbai	FLAT
Pranit	Lalla	Dinesh	1990-06-02	MALE	Mumbai	FLAT
Shriya	Parab	Sachin	1983-07-03	FEMALE	Mumbai	FLAT

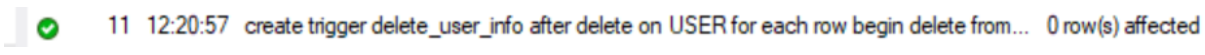
omeseeker info 40 ×

TRIGGERS:

Query 1:
To create a trigger such that when a record from user table is deleted ,the corresponding user data from homeseeker and tenant table should also get deleted automatically to keep the data consistent.

```
delimiter $$
create trigger delete_user_info
after delete
on USER
for each row
begin
delete from HOMESEEKER where HOMESEEKER.USER_ID=Old.USER_id;
delete from TENANT where TENANT.USER_ID=OLD.USER_ID;

end$$
delimiter ;
```



Trigger is created.

Then delete command is performed.
delete from USER where USER_ID in(101,105);

Deleted from USER table.

```
select * from user where user_id>100;
select *from tenant
select *from homeseeker

delimiter $$
create trigger delete_user_info
after delete
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

USER_ID	EMAIL	PASSWORD	F_NAME	M_NAME	L_NAME	DOB	GENDER	ACCT_CREATION_DATE	LAST_ACTIVE_DATE	STATE	CITY	AREA
103	abc@gmail.com	Hello@123	NULL	NULL	NULL	NULL	NULL	2022-11-30	2022-11-30	NULL	NULL	NULL
104	abc@abcd	ABCabc12!	NULL	NULL	NULL	NULL	NULL	2022-11-30	2022-11-30	NULL	NULL	NULL
106	abcd@abcde	Akshi@123	Janhavi	NULL	Deshmukh	2012-12-12	FEMALE	2022-11-30	2022-11-30	Maharashtra	Ghatkopar	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Automatically deleted from TENANT table.

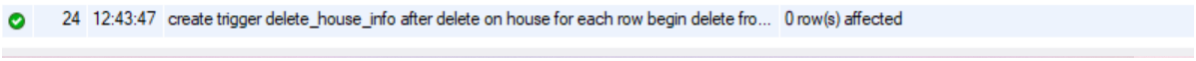
USER_ID	MAX_RENT	MAX_NO_OF_ROOMMATES	MARITAL_STATUS	FOOD_HABIT
81	9000	4	0	0
86	4500	0	0	0
92	18900	0	0	0
96	3390	2	0	1
97	19500	0	0	1
99	26400	1	1	0
100	21000	4	0	0
NULL	NULL	NULL	NULL	NULL

Automatically deleted from HOMESEEKER table as well.

USER_ID	PREFERRED_CITY	PREFERRED_AREA	PREFERRED_HOUSE_TYPE	PREFERRED_FLOOR	PREFERRED_SIZE	PREFERS_PET_ALLOWANCE	NO_(
89	Hyderabad	Pragathi Nagar, Moosapet	FLAT	3	740	0	1
92	Bangalore	AGS Layout	FLAT	2	970	0	2
96	Hyderabad	Allwyn Colony	FLAT	11	850	0	5
97	Mumbai	Shree Sai Usha Complex, Bhandup West	FLAT	1	1190	0	3
99	Mumbai	Station Pada	BUNGALOW	0	770	1	2
100	Mumbai	Shiv Ganga Apartment, Malad West	FLAT	3	1250	0	4
106	Mumbai	Versova	NULL	NULL	NULL	NULL	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Query 2:
To create a trigger such that when house information is deleted from the HOUSE database ,its corresponding information in the RENT or SALE table also gets deleted automatically.

```
delimiter $$
create trigger delete_house_info
after delete
on house
for each row
begin
delete from RENT where RENT.HOUSE_ID_ID=Old.HOUSE_id;
delete from SALE where SALE.HOUSE_ID=OLD.HOUSE_ID;
end $$
delimiter ;
```



Trigger is created successfully.

Then delete command is performed.

Three rows deleted from HOUSE table.

HOUSE_ID	OWNER_ID	STATE	CITY	AREA	PINCODE	TYPE	BOUGHT_IN	NO_OF_ROOMS	FLOOR	SIZE	FURNISHING
25	55	Karnataka	Bangalore	KR Puram	400078	FLAT	1994	2	7	1030	UNFURNISHE
26	57	Maharashtra	Mumbai	Hubtown Gardenia, Mira Road	400086	BUNGALOW	2015	1	8	1070	UNFURNISHE
27	58	West Bengal	Kolkata	East Park	400028	FLAT	1988	2	0	1060	UNFURNISHE
28	61	Tamil Nadu	Chennai	Gowrivakkam	400052	FLAT	1984	2	12	900	UNFURNISHE
29	63	Tamil Nadu	Chennai	Kannan Colony, Pazhavanthangal	400027	FLAT	1983	3	3	600	FURNISHED
30	64	Maharashtra	Mumbai	Mahim	400004	FLAT	1985	4	5	740	UNFURNISHE
31	67	Tamil Nadu	Chennai	Nanganallur	400051	FLAT	2001	2	0	510	UNFURNISHE
32	69	Maharashtra	Mumbai	Cosmopolis Tower, Andheri West	400011	FLAT	2006	2	3	950	UNFURNISHE
36	77	Delhi	Delhi	Jhangir Puri	400037	FLAT	1980	2	1	700	UNFURNISHE
37	79	West Bengal	Kolkata	Birati	400022	FLAT	2000	4	1	800	SEMIFURNISH
38	82	Telangana	Hyderabad	Narsingi, Outer Ring Road	400013	BUNGALOW	2022	1	9	630	UNFURNISHE

Automatically row from SALE table also got deleted.

Result Grid | Filter Rows: | Edit:

HOUSE_ID	TOTAL_PRICE	TOKEN_AMOUNT
12	15400000	800000
13	19500000	800000
14	14200000	800000
15	17300000	500000
18	24800000	900000
19	7900000	700000
22	23800000	1000000
23	7500000	700000
24	11200000	900000
36	20900000	400000
38	20800000	600000
NULL	NULL	NULL

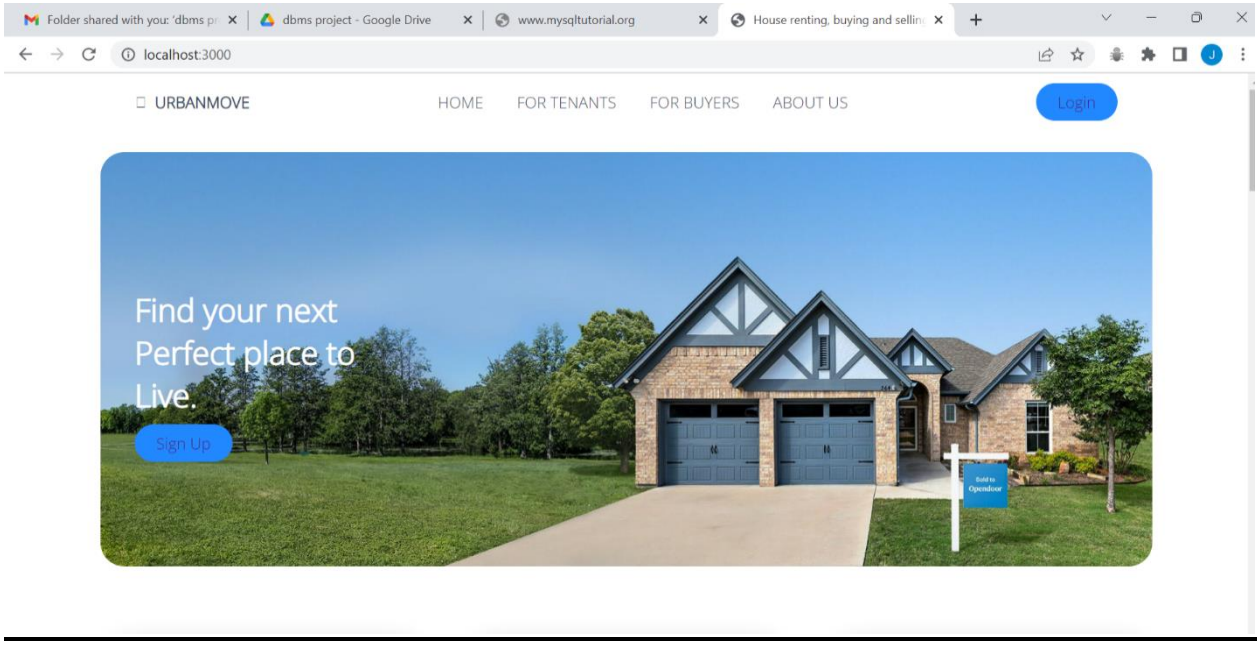
SALE 14 ×

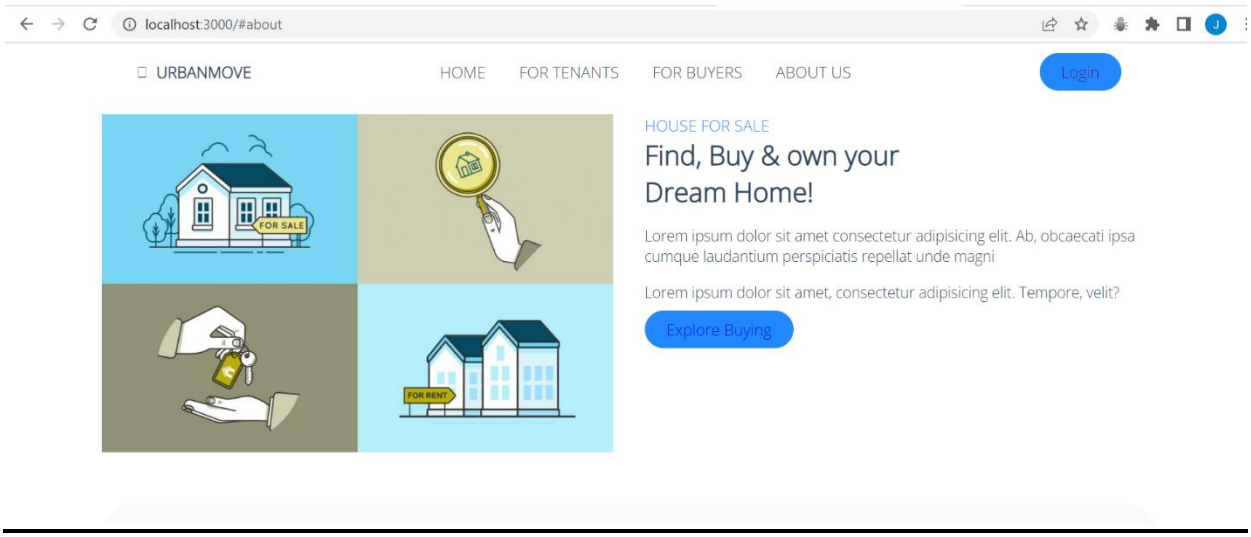
Automatically corresponding house from RENT also got deleted.

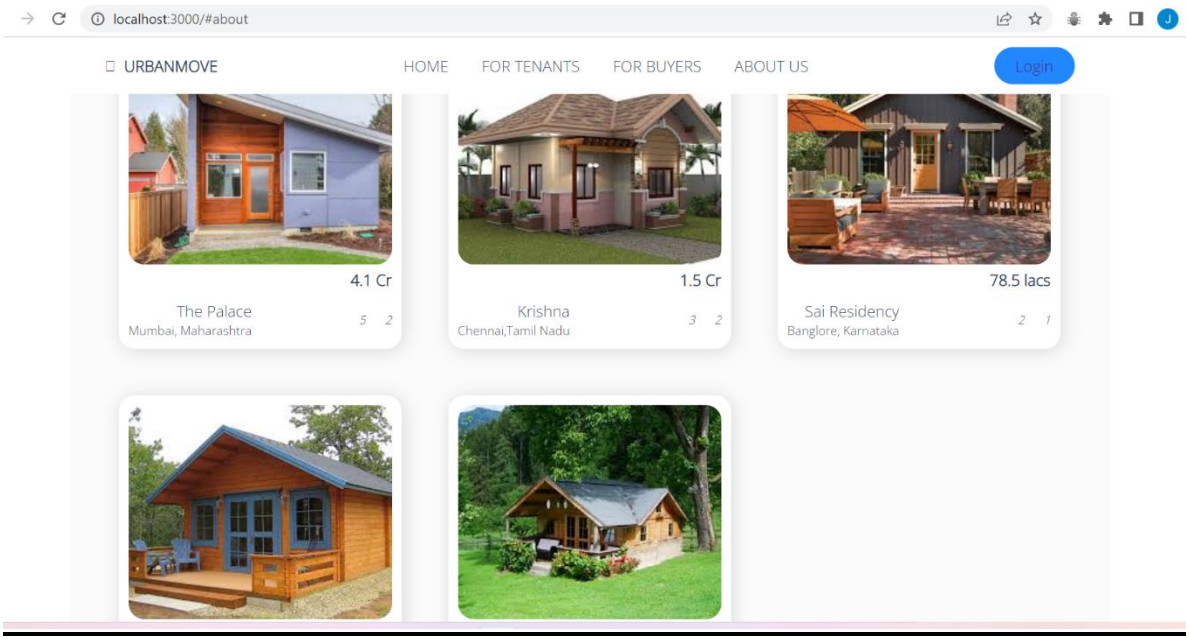
HOUSE_ID	MONTHLY_RENT	MAX_NO_OF_TENANTS	TENANT_MARITAL_STATUS	TENANT_FOOD_HABIT	SECURITY_DEPOSIT_AMOUNT
21	10000	1	1	1	17000
25	25000	3	1	0	162000
26	5000	3	1	0	127000
27	6500	3	0	1	51000
28	5500	4	1	0	34000
29	8500	2	1	1	53000
30	40000	3	1	0	125000
31	6000	2	0	0	45000
32	10000	4	0	1	156000
37	6000	4	0	0	148000
39	7900	1	0	0	107000
NULL	NULL	NULL	NULL	NULL	NULL

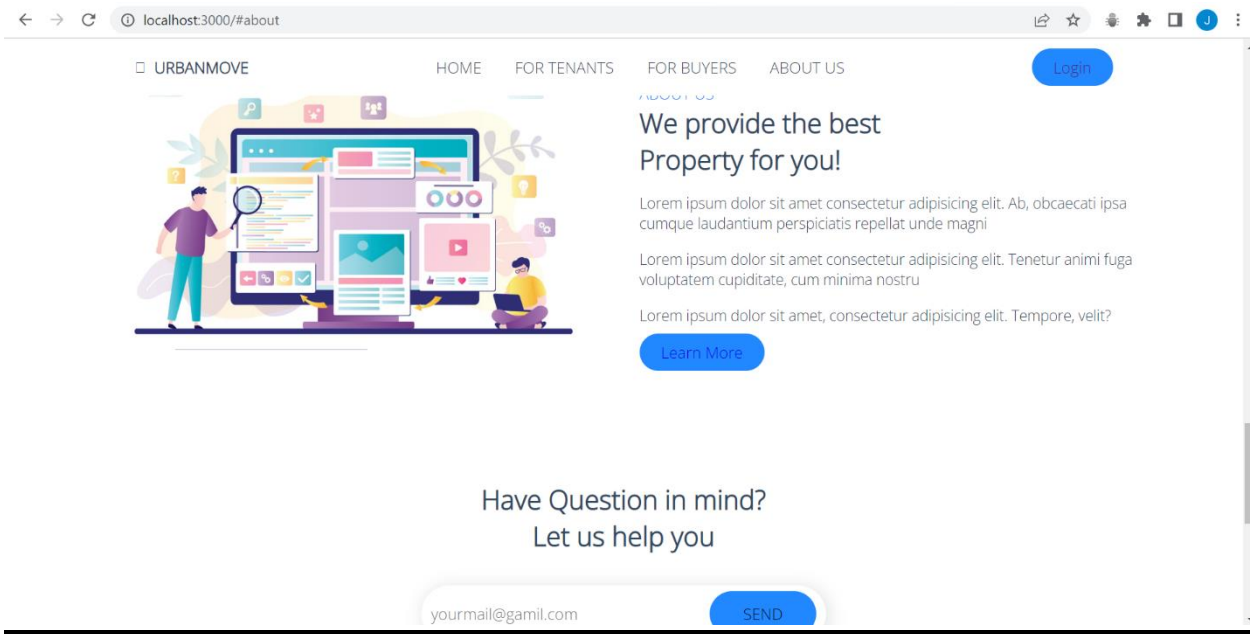
RENT 15 ×

FRONTEND:



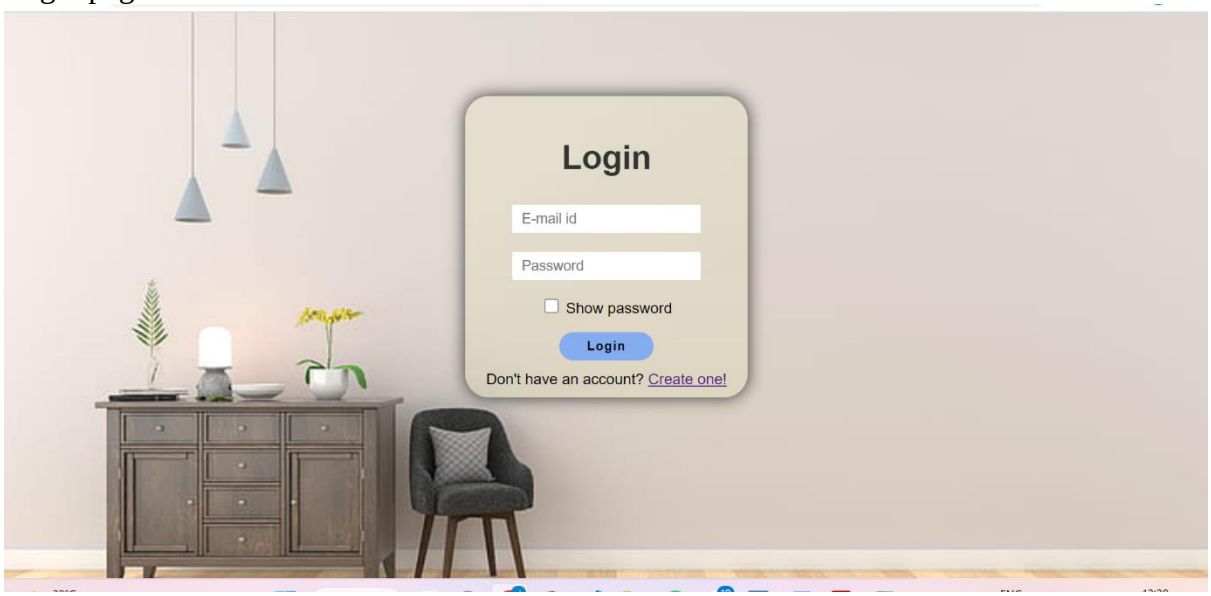

localhost:3000/#about
URBANMOVE
HOME
FOR TENANTS
FOR BUYERS
ABOUT US
Login
FOR SALE
FOR RENT
HOUSE FOR SALE
Find, Buy & own your
Dream Home!
Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab, obcaecati ipsa cumque laudantium perspiciatis repellat unde magni
Lorem ipsum dolor sit amet, consectetur adipisicing elit. Tempore, velit?
Explore Buying


localhost:3000/#about
URBANMOVE
HOME
FOR TENANTS
FOR BUYERS
ABOUT US
Login
4.1 Cr
The Palace
Mumbai, Maharashtra
5 2
1.5 Cr
Krishna
Chennai,Tamil Nadu
3 2
78.5 lacs
Sai Residency
Banglore, Karnataka
2 1

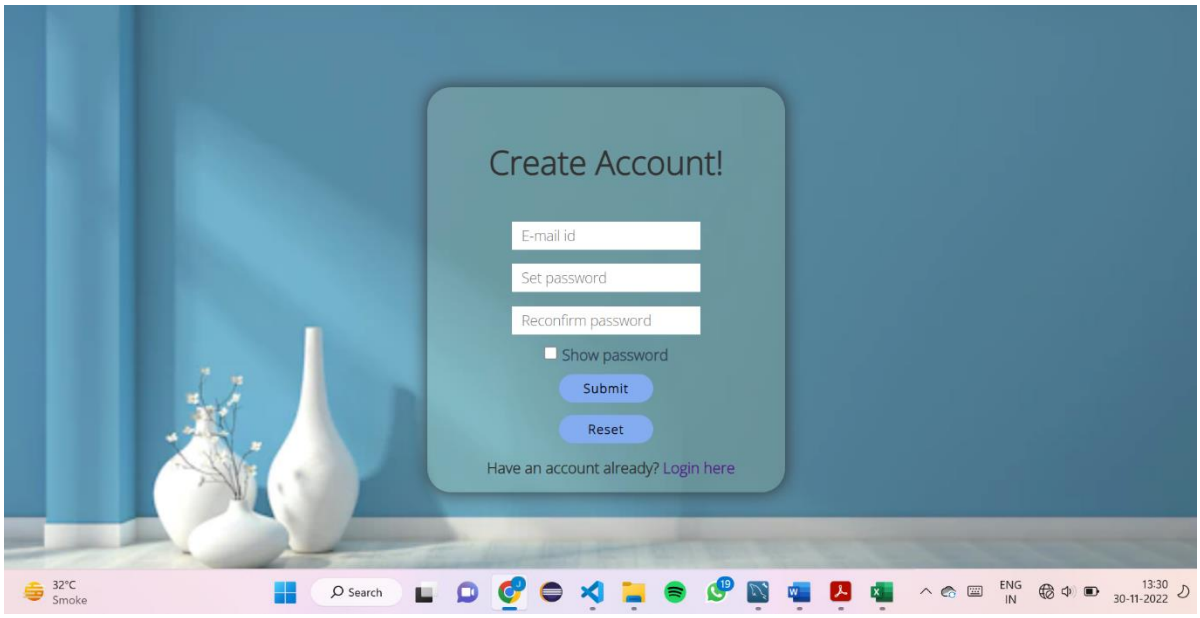


The above are home page designs.

Login page.



Create account



User info registration page.