Table of Contents

DMDD Final project: Your Rental Buddy

Project Scope and description:

This project comprises designing and building a database to assist students in finding off-campus accommodations by bringing them into an existing network of students who are searching for roommates, subletting a spot, or planning to renew leases with new roommates based on student predilections such as proximity to the university, transit connectivity, budget, etc.

We will be developing an application that will provide students access to the database which will have the following features:

1. Creating a profile(Both Renter & Rentee)
2. To add/remove/update the spot(Renter)
3. To find a spot/roommate(Rentee)
4. Filtering preferences(Both Renter & Rentee)

Team Members:

Neha Bhutkar- (email: bhutkar.n@northeastern.edu)
Github: https://github.com/NehaBhutkar/Your-Rental-BuddyLinks to an external site.
Amey Parange (email: parange.a@northeastern.edu) Github: https://github.com/amey379/Your-Rental-BuddyLinks to an external site.

Ruchi Kapadiwala (email: kapadiwala.r@northeastern.edu) Github: https://github.com/ruchi-kapadiwala/Your-Rental-Buddy

Twitter Scraping:

SQL QUERIES:

```
CREATE SCHEMA `rentalbuddy` ;
```

----------table Tweets----------

```
CREATE TABLE `rentalbuddy`.`Tweets` ( `tweet_id` VARCHAR(50), `Twitter_handle` VARCHAR(50), `tweet_text` VARCHAR(500), `country` VARCHAR(20), `city` VARCHAR(20), `retweet_count` INT, `favorite_count` INT, `created_at` DATETIME, PRIMARY KEY (`tweet_id`) );
```

----------table User----------

```
CREATE TABLE `rentalbuddy`.`User` ( `Twitter_handle` VARCHAR(50), `name` VARCHAR(50), `description` VARCHAR(500), `followers_count` INT, `following_count` INT, `location` VARCHAR(50), `created_at` DATETIME, PRIMARY KEY (`Twitter_handle`) );
```

----------table Tag----------

```
CREATE TABLE `rentalbuddy`.`Tags` ( `tweet_id` VARCHAR(50), `tweet_tags` VARCHAR(500), PRIMARY KEY (`tweet_id`) );
```

----------table Mentions----------

```
CREATE TABLE `Tweet_Mentions` ( `tweet_id` VARCHAR(50), `source_user` VARCHAR(50), `target_user` VARCHAR(500), PRIMARY KEY (`tweet_id`) );
```

----------table Accomodation----------

```
CREATE TABLE `Accomodation` ( `tweet_id` VARCHAR(50), `location` VARCHAR(50), PRIMARY KEY (`tweet_id`) );
```

—----------table Rent—----------

CREATE TABLE `Rent` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `Range` VARCHAR(50), -- high or low PRIMARY KEY (`tweet_id`));

—----------table Transportation—----------

CREATE TABLE `Transportation` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `opinion` VARCHAR(50), -- positive // negative ///neutral PRIMARY KEY (`tweet_id`));

—----------table Propertrymgmt —----------

CREATE TABLE `Propertrymgmt` (`tweet_id` VARCHAR(50), `name` VARCHAR(50), `location` VARCHAR(50), `rating` VARCHAR(50), -- 1-5 PRIMARY KEY (`tweet_id`));

—----------table SocialIssues—----------

CREATE TABLE `SocialIssues` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `intensity` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Entertainment —----------

CREATE TABLE `Entertainment` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `intensity` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Crimes—----------

CREATE TABLE `Crimes` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `rate` VARCHAR(10), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Educational—----------

CREATE TABLE `Educational` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `opinion` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Health—----------

CREATE TABLE `Health` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `opinion` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Sports—----------

CREATE TABLE `Sports` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `opinion` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Job—----------

CREATE TABLE `Job` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `opinion` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Weather—----------

CREATE TABLE `Weather` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `level` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Food—----------

CREATE TABLE `Food` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `rating` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------table Groceries—----------

CREATE TABLE `Groceries` (`tweet_id` VARCHAR(50), `location` VARCHAR(50), `rating` VARCHAR(50), -- -- high or low or neutral PRIMARY KEY (`tweet_id`));

—----------ALTER Statements—----------

table Groceries—---------- ALTER TABLE `Groceries` ADD CONSTRAINT `Groceries_fk1` FOREIGN KEY (`tweet_id`) REFERENCES `Tweets`(`tweet_id`);

—----------table Transportation—----------

ALTER TABLE `Transportation` ADD CONSTRAINT `Transportation_fk1` FOREIGN KEY (`tweet_id`); REFERENCES `Tweets`(`tweet_id`);

—----------table Entertainment—----------

ALTER TABLE `Entertainment` ADD CONSTRAINT `Entertainment_fk1` FOREIGN KEY (`tweet_id`); REFERENCES `Tweets`(`tweet_id`);

—----------table Food—----------

ALTER TABLE `Food` ADD CONSTRAINT `Food_fk1` FOREIGN KEY (`tweet_id`); REFERENCES `Tweets`(`tweet_id`);

—----------table Crimes—----------

ALTER TABLE `Crimes` ADD CONSTRAINT `Crimes_fk1` FOREIGN KEY (`tweet_id`); REFERENCES `Tweets`(`tweet_id`);

—----------table Health—----------

ALTER TABLE `Health` ADD CONSTRAINT `Health_fk1` FOREIGN KEY (`tweet_id`); REFERENCES `Tweets`(`tweet_id`);

—----------table SocialIssues—----------

ALTER TABLE `SocialIssues` ADD CONSTRAINT `SocialIssues_fk1` FOREIGN KEY (`tweet_id`) REFERENCES `Tweets`(`tweet_id`);

—----------INSERT Statements----------

—----------table Tweets—----------

INSERT INTO`Tweets`(`tweet_id`,`Twitter_handle`,`tweet_text`, `country`,`city`,`retweet_count`,`favorite_count`,`created_at`) VALUES('1','bostonpolice','Crime at Huntingtun Avee','US','BOSTON',4,5,sysdate());

—----------table Entertainment—----------

INSERT INTO `Entertainment`(`tweet_id`,`location`,`intensity`) VALUES ('1890','Boston','High'());

—----------table Job—----------

INSERT INTO `Job`(`tweet_id`,`location`,`opinion`) VALUES ('2908','BOSTON','ample of jobs available');

—----------table Groceries—----------

INSERT INTO `Groceries`(`tweet_id`,`location`,`Rating`) VALUES ('2908','BOSTON','4');

—----------table TweetMentions —----------

INSERT INTO `TweetMentions`(`tweet_id`,`source_user`,`target_user`) VALUES ('8774','@Avanti','@Preksha');

—----------table Weather —----------

INSERT INTO `Weather`(`tweet_id`,`location`,`level`) VALUES ('7654','Alaska','Extreme');

—----------table Rent —----------

INSERT INTO `Rent`(`tweet_id`,`location`,`Range`) VALUES ('7654','Lynn','low');

Relational Queries:

1.

```
Select name, location, AVG(rating)
from Propertrymgmt
GROUP BY name, location
ORDER BY avg(rating)
```

$$_{\text{name, location}}\ \Im\ [\ \pi_{\text{name, location, AVG(rating) ORDER BY AVG(rating)}}(\text{Propertrymgmt})\ ]$$

2.

```
Select t.tweet, r.location
from Rent r
JOIN Tweet t
ON r.tweet_id = t.tweet_id
WHERE r.rating = 'low'
```

$$\pi_{\text{t.tweet, r.location}}(\ \sigma_{\text{t.tweet\_id = r.tweet\_id ^ r.rating='low'}}\ )\ (\ \rho_{r}\,(\text{Rent})\ \times\ \rho_{t}\,(\text{Tweet})\ )$$

3.

```
Select t.tweet, s.location, s.intensity
from SocialIssues s
JOIN Tweet t
ON r.tweet_id = t.tweet_id AND r.location = t.city
```

$\pi_{t.tweet,\ s.location,\ s.intensity}\ (\ \sigma_{t.tweet\_id\ =\ s.tweet\_id\ \wedge\ s.location\ =\ t.city}\ )\ (\ \rho_s\ (SocialIssues) \times\ \rho_t\ (tweet)\ )$

4.

```
Select tweet_id, location, rating
from Crimes
ORDER BY rating DESC
LIMIT 1
```

$\pi_{tweet\_id,\ location,\ rating\ ORDER\ BY\ rating\ DESC\ LIMIT\ 1}\ (Crimes)$

5.

```
Select tweet_id, location
from Weather
WHERE level = neutral
LIMIT 1
```

$\pi_{tweet\_id,\ location}\ (\ \sigma_{level\ =\ 'neutral'\ LIMIT\ 1}\ Weather\ )$

Use cases for Scraping Twitter:

Use Case 1: Register/log in for an account in Your Rental Buddy Application

Description: The user registers/logs in for an account in the Your Rental Buddy application.

Actor: User

Precondition: When a user wants to look for rental houses using our application, he will first have to register/login to his existing account.

Steps:

Actor action: The user enters his/her information and registers/logs in to the application.

System Responses: The system validates the user information, if found correct, it registers/logs the user into the system.

Post Condition: Customer successfully registered/logged in.

Alternate Path: If the system is not successful in validating the information, it throws an error.

Error: User information is incorrect, registration/login unsuccessful.

Use Case 2: Get information regarding housing options on the basis of locality in the application

Description: When the user searches for living accommodation, the Twitter bot displays a list of relevant tweets for the search as per the associated filter(s).

Actor: User

Precondition: User must have a unique Twitter handle to look for tweets.

Steps:

Actor action: The user searches for accommodation using hashtags, keywords and handles.

System Responses: The system provides the user with all the tweets that mention accommodation on the basis of locality made by other Twitter users.

Post Condition: Customer gets the information for the accommodation relative to the filters he/she used.

Use Case 3: View housing options that talk about affordable housing options.

Description: The user can view the tweets that mention housing options according to budget.

Actor: User

Precondition: The User must have a unique Twitter handle to look for tweets.

Steps:

Actor action: The user searches for tweets using budget hashtags, keywords, and handles.

System Responses: Tweets mentioning accommodations per the user's budget are displayed to the user.

Post Condition: The system displays the list of houses for the given condition.

Use Case 4: Acquire people's views on public transportation in a particular city/locality.

Description: The user can get a public opinion of the transportation facilities for the accommodation he wants to choose.

Actor: User

Precondition: The user should mention particular transports and location hashtags.

Steps:

Actor action: The user searches for tweets using transport hashtags, keywords, and handles.

System Responses: Tweets mentioning transportation and their experiences.

Post Condition: The system displays the tweets made by people regarding transportation garnering overall public sentiment about specific transportation.

Use Case 5: The user looks for the top property management agencies.

Description: The user searches for the best property management agencies based on tweets.

Actor: User

Precondition: The user should mention the property management hashtags and keywords.

Steps:

Actor action: The user searches for tweets using the property management agency's hashtags and keywords.

System Responses: The system displays the tweets made by people regarding management agencies citing their experiences.

Use Case 6: The user wants to support Social Issues related to accommodation.

Description: The user wants to make himself/herself aware of the social movements associated with finding accommodation.

Actor: User

Precondition: The user should mention the correct hashtags and keywords regarding the current social issues.

Steps:

Actor action: The user searches for tweets using hashtags and keywords regarding to accommodation.

System Responses: The system displays tweets mentioning social movements regarding accommodation.

Use Case 7: The user wants to know if the locality has places for entertainment

Description: The user wants to know if the locality has clubs, gardens, museums, and theaters.

Actor: User

Precondition: The user should mention the hashtags for the type of entertainment place and keywords regarding the locality/city he's looking for.

Steps:

Actor action: The user searches for tweets with hashtags and keywords of entertainment places and keywords regarding the locality/city he's looking for.

System Responses: The system displays tweets mentioning places of entertainment in a specific city.

Use Case 8: The user wants to know the safety quotient of the city/locality.

Description: The user wants to know about the crimes/kind of crimes that took place in the city/locality.

Actor: User

Precondition: The user should mention the hashtags and mentions for crimes and keywords regarding the locality/city he's looking for.

Steps:

Actor action: The user searches for tweets with crime hashtags and keywords regarding the locality/city he's looking for.

System Responses: The system displays tweets mentioning crimes that happened in a specific city.

Use Case 9: The user wants to find the best universities/colleges in a specific city.

Description: The user wants to learn about universities and colleges in a specific city and read about student experiences about the same.

Actor: User

Precondition: The user should mention the hashtags and keywords for the universities and colleges he is looking for.

Steps:

Actor action: The user searches for tweets with hashtags and keywords for the universities and colleges he is looking for.

System Responses: The system displays tweets mentioning the universities and colleges based on the user's preferences.

Use Case 10: The user wants to look for the best grocery stores in a particular locality/city.

Description: The user searches the grocery stores nearby his accommodation and looks for the grocery stores with the highest ratings.

Actor: User

Precondition: The user should mention the hashtags and keywords for the grocery stores and cities.

Steps:

Actor action: The user searches for tweets with hashtags and keywords for the grocery stores in a particular place.

System Responses: The system displays tweets mentioning the grocery stores at a particular place.

Use Case 11: The user wants to find the best-rated Hospital/Pharmacy in a city/locality.

Description: The user searches for the pharmacies and hospitals nearby his accommodation and finds the best-rated Hospital/Pharmacy based on mass opinion.

Actor: User

Precondition: The user should mention the hashtags and keywords for the hospitals and pharmacies and cities/localities.

Steps:

Actor action: The user searches for tweets with hashtags and keywords for hospitals and pharmacies along with correct mentions.

System Responses: The system displays tweets mentioning the hospitals and pharmacies citing people's experiences.

Use Case 12: The user wants to know about popular restaurants in a city/locality.

Description: The user searches for well-known restaurants in a particular city/locality based on public opinion.

Actor: User

Precondition: The user should mention the hashtags and keywords for the popular restaurants in a particular city/locality.

Steps:

Actor action: The user searches for tweets with hashtags and keywords for popular restaurants in a particular city/locality.

System Responses: The system displays tweets mentioning the popular restaurants in a particular city/locality based on people's experiences.

Use Case 13: The user wants to know about the availability of jobs in a particular city.

Description: The user searches for his preferred jobs(like Software Testing, Software Developer, Product Manager, etc.) in a particular city.

Actor: User

Precondition: The user should mention the hashtags and keywords for the type of job, new openings, and companies.

Steps:

Actor action: The user searches for tweets with hashtags and keywords for the type of job, new openings, and companies.

System Responses: The system displays tweets mentioning the jobs preferred by the user.

Use Case 14: User searches for sports in a city.

Description: The user wants to know if his favorite sport is played in the city and all the available information regarding the same.

Actor: User

Precondition: The user should mention the hashtags and keywords for the type of job, new openings, and companies.

Steps:

Actor action: The user searches for tweets with hashtags and keywords for the type of sport he plays along with the city name.

System Responses: The system displays tweets containing all the information about the sport user searched for.

Use Case 15: The user searches for weather conditions in a city.

Description: The user wants to know the weather conditions in a particular city.

Actor: User

Precondition: The user should mention the hashtags and keywords for the weather conditions and city names along with weather forecasting mentions.

Steps:

Actor action: The user searches for tweets with hashtags and keywords for the weather conditions and city names along with weather forecasting mentions.

System Responses: The system displays tweets containing all the information about weather conditions in a particular city.

Gathering, Scraping, Munging and Cleaning Data:

----------README FILE----------

Your Rental Buddy: Gathering, Scraping, Munging, and Cleaning Data

Estimate of total data: around 2000 records

Step 1: Found Sources of data to scrape from
JSON, HTML websites

Step 2: Gathered real-world data for multiple databases which will aid international students with finding accommodation
Data Sources:
*Scraped data from the JuneHomes website
*Google forms for real-time updates about the needs of students when it comes to finding accommodation

Step 3: Cleaned the data
Data cleaning methods used:
*Added data to data frames
*Removed null values as per the percentage
*Filled the remaining null values with mean or median
*Removed outliers
*Removed redundant data
*Normalized the data

Step 4: Created tables corresponding to the data found

Step 5: Created use cases and joined the relevant tables for retrieving required information for the rentee/renter

----------INSERT QUERIES----------

```
INSERT INTO JuneApartments(
id,
Apt_id ,
url ,
address,
Beds ,
Bath ,
Price ,
BedArea ,
Availablefrom ,
Availabletill ,
Description
)
VALUES
        ( '1001',
```

```
    '78',
    'https://junehomes.com/residences/boston-ma/mission-hill/1068-mission-hill/3141',
    'Park Drive',
                '2',
                '2',
                '1100',
                '69',
                '2022-12-05',
                '2023-01-04',
'This comfy, cute and charming room is available'
                    );
```

```
INSERT INTO JuneAmenities(
id,
Amenities
)
VALUES
        (
   '6811',
        'Wifi'
                    );

INSERT INTO Junetransport(
id,
Trans_id,
stations,
color,
walktime,
description)
VALUES
        (
                    '9',
                     '2',
                      'Blue Line',
                     '#2040AA',
                     '10',
                    'Boston Landing'
                );
```

```
INSERT INTO SubleaseSpot(

 `Name`,
 `PhoneNumber`,
 `Email`,
 `Gender`,
 `Address`,
 `ProxToUni`,
 `Brokerage`,
```

```
 `LeaseSpotType`,
 `BedroomCount`,
 `BathroomCount`,
 `Rent`,
 `DietaryPref`,
 `GenderPref`,
 `Amenities`,
 `AvailSpot`,
 `PrefMoveInDate`,
 `AvailSpotNum`
 )
VALUES

                    (

                    'Vaishali Mhatre',
                     '8573286790',
                     'veenam45@gmail.com',
                                                        'Female',
                     'Park Drive',
```

```
                                                    '0.5',
                    '700',
          'OnLease',
                    '2',
          '3',
                                                    '1100',
         'Vegetarian',
         'Female',
         'Dishwasher',
         '1',
         '2023-01-01',
'1'
          );
```

```
INSERT INTO SubleaseRoommate(
 `Name`,
 `Gender`,
  `PhoneNumber`,
 `Email`,
 `Budget`,
 `RoommateDietaryPref`,
 `RoommateGenderPref`,
 `Amenities`,
 `PrefModeofTravel`,
 `TypeOfSpot`,
 `PrefMoveInDate`,
```

```
 `NoOfRoommates`
 )
VALUES

                    (
                    'Shalini Pawar',
                    'Female',
                     '8573286790',
                     'veenam45@gmail.com',
                                                        '700',
                     'Vegetarian',
                              'Female',
                     'Wifi',
                                                      'Green Line',
                   'OnLease',
                   '2023-01-01',
                   '2'
                    );
```

```
INSERT INTO TemporarySpot(
 `Name`,
 `PhoneNumber`,
   `Email`,
    `Gender`,
 `Address`,
 `ProximityToUni`,
 `BedroomCount`,
 `BathroomCount`,
 `TempRent`,
 `DietaryPref`,
 `GenderPref`,
 `Amenities`,
 `AvailableSpot`,
 `PrefMoveInDate`,
 `PrefMoveOutDate`,
 `AvailSpotNum`
 )
VALUES

                    (
                     'Sayak Hande',
                    '8573286790',
                    'veenam45@gmail.com',
                     'Male',
                     'J Vue at the LMA',
                                                          '0.5',
                     '2',
                     '2',
```

```
'10',
'Vegetarian',
'Female',
'INHouseLoundry',
'2',
'2023-01-01',
'2023-02-05',
'2'

);
```

```
INSERT INTO TemporaryRoommate(`RoommateId`,
 `Name`,
 `Gender`,
  `PhoneNumber`,
 `Email`,
 `Budget`,
 `DietaryPref`,
 `GenderPref`,
 `Amenities`,
 `PrefModeofTravel`,
 `TypeOfSpot`,
 `PrefMoveInDate`,
 `NoOfRoommates`
 )
VALUES

                    ( '1',
                    'Shalini Pawar',
                    'Female',
                     '8573286790',
                     'veenam45@gmail.com',
                                                          '700',
                     'Vegetarian',
                              'Female',
                     'Wifi',
                                                        'Green Line',
                    'OnLease',
                    '2023-01-01',
                    '2'
                     );
```

----------CREATE QUERIES----------

```
CREATE TABLE JuneApartments
(
```

```
id int not NULL PRIMARY KEY,
Apt_Id int,
url varchar(300),
address varchar(100),
Beds decimal,
Bath decimal,
Price decimal,
BedArea int,
Availablefrom date,
Availabletill date,
Description varchar(500));
```

```
CREATE TABLE JuneAmenities
(
id int not NULL ,
Amenities varchar(100));


CREATE TABLE Junetransport
(
id int not NULL ,
Trans_id int,
stations varchar(100),
color varchar(100),
walktime int,
description varchar(100));

CREATE TABLE `SubleaseSpot` (
  `SpotID` INT NOT NULL AUTO_INCREMENT,
  `Name` VARCHAR(45),
  `PhoneNumber` VARCHAR(10),
  `Email` VARCHAR(45),
  `Gender` VARCHAR(45) ,
  `Address` VARCHAR(200),
  `ProxToUni` FLOAT,
  `Brokerage` INT,
  `LeaseSpotType` VARCHAR(100),
  `BedroomCount` INT,
  `BathroomCount` INT,
  `Rent` INT,
  `DietaryPref` VARCHAR(45),
  `GenderPref` VARCHAR(45),
  `Amenities` VARCHAR(200),
  `AvailSpot` VARCHAR(45),
  `PrefMoveInDate` DATE,
  `AvailSpotNum` INT,
  PRIMARY KEY (`SpotID`),
```

```
UNIQUE INDEX `SpotID_UNIQUE` (`SpotID` ASC) VISIBLE);


CREATE TABLE `SubleaseRoommate` (
`RoommateID` INT NOT NULL AUTO_INCREMENT,
`Name` VARCHAR(45) ,
`Gender` VARCHAR(45),
`PhoneNumber` varchar(10) ,
`Email` VARCHAR(45),
`Budget` INT,
`RoommateDietaryPref` VARCHAR(45),
`RoommateGenderPref` VARCHAR(45),
`Amenities` VARCHAR(45),
`PrefModeofTravel` VARCHAR(45),
`TypeOfSpot` VARCHAR(45),
`PrefMoveInDate` DATE,
`NoOfRoommates` INT,
PRIMARY KEY (`RoommateID`),
UNIQUE INDEX `RoommateID_UNIQUE` (`RoommateID` ASC) VISIBLE);

CREATE TABLE TemporarySpot
(SpotID int NOT NULL PRIMARY KEY,
Name VARCHAR(45),
PhoneNumber int(10),
Email VARCHAR(45),
Gender VARCHAR(45),
Address VARCHAR(200),
ProximityToUni float,
BedroomCount int,
BathroomCount int,
TempRent int,
DietaryPref VARCHAR(45),
GenderPref VARCHAR(45),
Amenities VARCHAR(45),
AvailableSpot VARCHAR(45),
PrefMoveInDate date,
PrefMoveOutDate date,
AvailSpotNum int);
```

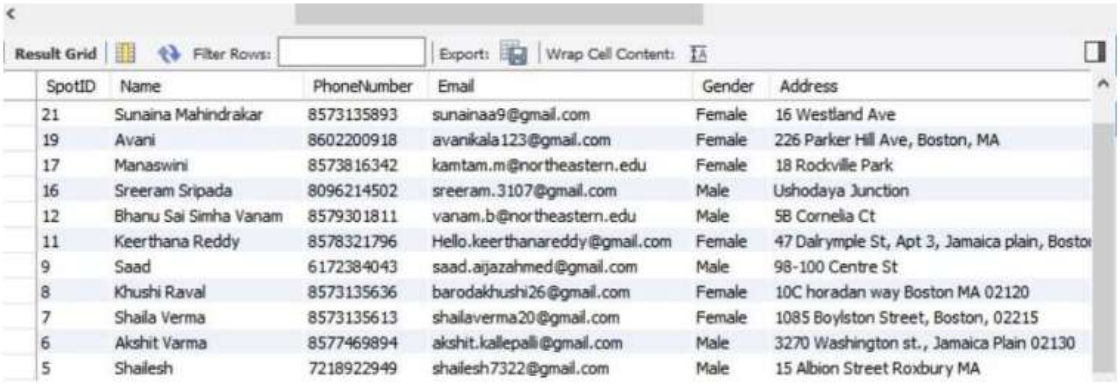
```
CREATE TABLE TemporaryRoommate
(RoommateId int NOT NULL PRIMARY KEY,
Name varchar(45),
Gender varchar(45),
PhoneNumber varchar(10),
Email varchar(45),
Budget varchar(45),
DietaryPref varchar(45),
GenderPref varchar(45),
```

```
Amenities varchar(45),
PrefModeofTravel varchar(45),
TypeOfSpot varchar(45),
PrefMoveInDate date,
NoOfRoommates int);
```

Use Cases with Join Queries:

1. Rentee is looking for accommodation with the following amenities: Wifi(paid), Laundry

```
select ss.* from SubleaseSpot ss, SubleaseRoommate sr
where sr.name='Ritik Bhandari'
and s.Amenities like "%Wifi%" AND ss.Amenities like
"%Laundry%" ;
```



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

SpotID	Name	PhoneNumber	Email	Gender	Address
21	Sunaina Mahindrakar	8573135893	sunainaa9@gmail.com	Female	16 Westland Ave
19	Avani	8602200918	avanikala123@gmail.com	Female	226 Parker Hill Ave, Boston, MA
17	Manaswini	8573816342	kamtam.m@northeastern.edu	Female	18 Rockville Park
16	Sreeram Sripada	8096214502	sreeram.3107@gmail.com	Male	Ushodaya Junction
12	Bhanu Sai Simha Vanam	8579301811	vanam.b@northeastern.edu	Male	5B Cornelia Ct
11	Keerthana Reddy	8578321796	Hello.keerthanareddy@gmail.com	Female	47 Dalrymple St, Apt 3, Jamaica plain, Bosto
9	Saad	6172384043	saad.aijazahmed@gmail.com	Male	98-100 Centre St
8	Khushi Raval	8573135636	barodakhushi26@gmail.com	Female	10C horadan way Boston MA 02120
7	Shaila Verma	8573135613	shailaverma20@gmail.com	Female	1085 Boylston Street, Boston, 02215
6	Akshit Varma	8577469894	akshit.kallepalli@gmail.com	Male	3270 Washington st., Jamaica Plain 02130
5	Shailesh	7218922949	shailesh7322@gmail.com	Male	15 Albion Street Roxbury MA

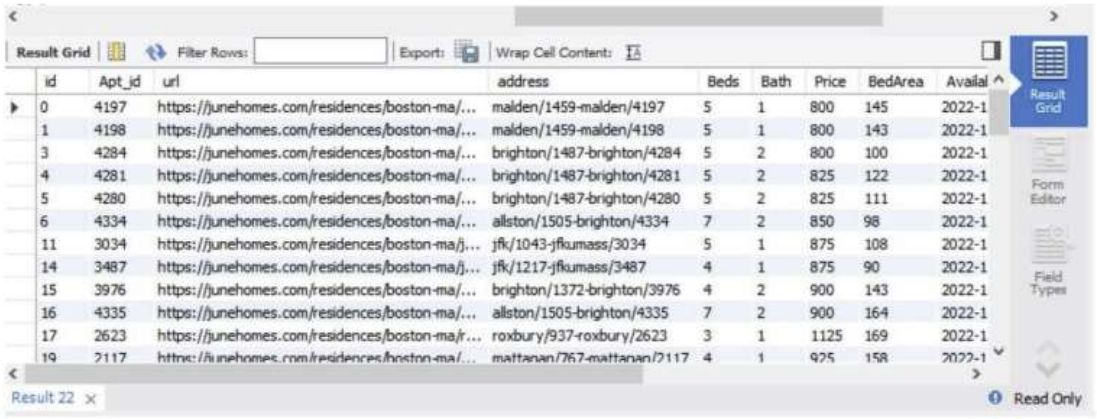
2. Rentee is looking for accommodations on the JuneHomes.com website at Harvard Avenue a min distance from the nearest transportation.

```
select j.address,ja.* from JuneApartments j join
Junetransport ja on j.id= ja.id

join SubleaseRoommate sr where sr.name='Parixit Sanghani'
and address like "%harvard%"

and ja.walktime= (select walktime from Junetransport order
by walktime limit 1);
```

address	id	Trans_id	stations	color	walktime	description
harvard-square/443-harvard-square/1169	607	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1164	621	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1166	924	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1163	937	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1168	942	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1167	946	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1165	955	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave

3. Rentee is looking for accommodations on the JuneHomes.com website with amenities like paid Wifi

```
select j.* from JuneApartments j join JuneAmenities ja on
j.id= ja.id

join SubleaseRoommate sr where sr.name='Ritik Bhandari'
and ja.Amenities like "%Wifi%";
```



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

id	Apt_id	url	address	Beds	Bath	Price	BedArea	Availal
0	4197	https://junehomes.com/residences/boston-ma/...	malden/1459-malden/4197	5	1	800	145	2022-1
1	4198	https://junehomes.com/residences/boston-ma/...	malden/1459-malden/4198	5	1	800	143	2022-1
3	4284	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4284	5	2	800	100	2022-1
4	4281	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4281	5	2	825	122	2022-1
5	4280	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4280	5	2	825	111	2022-1
6	4334	https://junehomes.com/residences/boston-ma/...	allston/1505-brighton/4334	7	2	850	98	2022-1
11	3034	https://junehomes.com/residences/boston-ma/j...	jfk/1043-jfkumass/3034	5	1	875	108	2022-1
14	3487	https://junehomes.com/residences/boston-ma/j...	jfk/1217-jfkumass/3487	4	1	875	90	2022-1
15	3976	https://junehomes.com/residences/boston-ma/...	brighton/1372-brighton/3976	4	2	900	143	2022-1
16	4335	https://junehomes.com/residences/boston-ma/...	allston/1505-brighton/4335	7	2	900	164	2022-1
17	2623	https://junehomes.com/residences/boston-ma/r...	roxbury/937-roxbury/2623	3	1	1125	169	2022-1
19	2117	https://junehomes.com/residences/boston-ma/...	mattapan/767-mattapan/2117	4	1	925	158	2022-1

Result Grid | Form Editor | Field Types

Result 22 | Read Only

4. The rentee is looking for accommodations on the JuneHomes.com website with rent according to his Budget per month

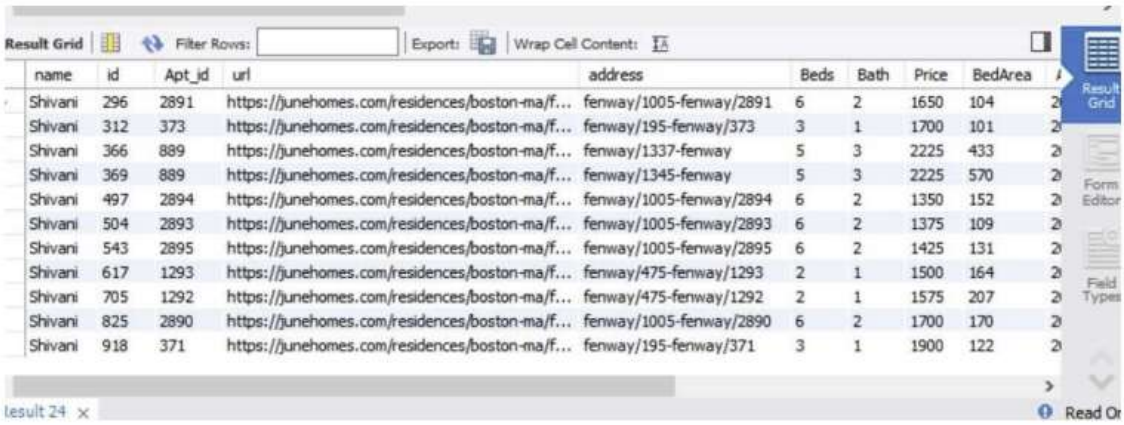
```
select sr.name,j.* from JuneApartments j join TemporaryRoommate sr where sr.name='Sai Uttam Gadde' and j.Price <sr.budget;
```

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

name	id	Apt_id	url	address	Beds	Bath	Price
Sai Uttam Gadde	0	4197	https://junehomes.com/residences/boston-ma/...	malden/1459-malden/4197	5	1	800
Sai Uttam Gadde	1	4198	https://junehomes.com/residences/boston-ma/...	malden/1459-malden/4198	5	1	800
Sai Uttam Gadde	2	4404	https://junehomes.com/residences/boston-ma/...	brighton/1524-brighton/4404	5	2	800
Sai Uttam Gadde	3	4284	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4284	5	2	800
Sai Uttam Gadde	4	4281	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4281	5	2	825
Sai Uttam Gadde	5	4280	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4280	5	2	825
Sai Uttam Gadde	6	4334	https://junehomes.com/residences/boston-ma/...	allston/1505-brighton/4334	7	2	850
Sai Uttam Gadde	7	4592	https://junehomes.com/residences/boston-ma/...	somerville/1594-somerville/4592	7	1	850
Sai Uttam Gadde	8	4403	https://junehomes.com/residences/boston-ma/...	brighton/1524-brighton/4403	5	2	850
Sai Uttam Gadde	9	4590	https://junehomes.com/residences/boston-ma/...	somerville/1594-somerville/4590	7	1	850
Sai Uttam Gadde	10	4594	https://junehomes.com/residences/boston-ma/...	somerville/1594-somerville/4594	7	1	850

5. The rentee is looking for accommodations on the JuneHomes.com website near Fenway

```
select sr.name, j.* from JuneApartments j
join SubleaseRoommate sr
on sr.name='Shivani' and j.address like "%fenway%";
```



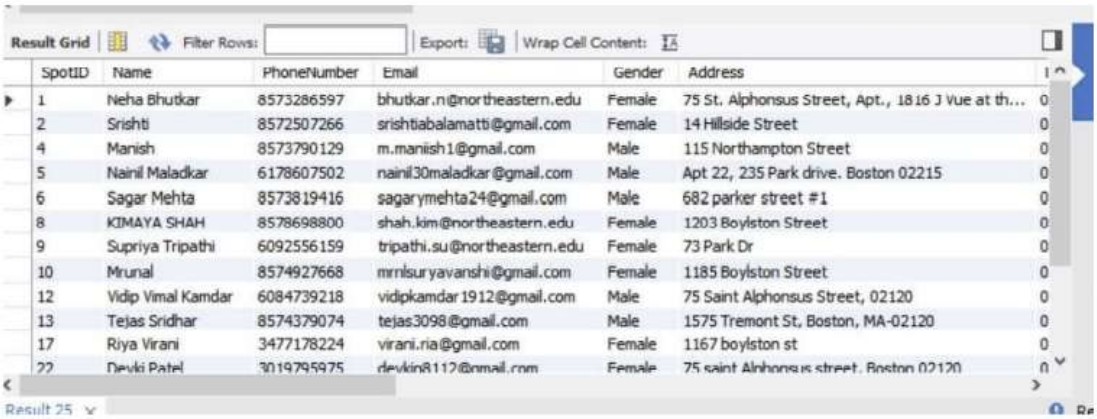
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	id	Apt_id	url	address	Beds	Bath	Price	BedArea
Shivani	296	2891	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2891	6	2	1650	104
Shivani	312	373	https://junehomes.com/residences/boston-ma/f...	fenway/195-fenway/373	3	1	1700	101
Shivani	366	889	https://junehomes.com/residences/boston-ma/f...	fenway/1337-fenway	5	3	2225	433
Shivani	369	889	https://junehomes.com/residences/boston-ma/f...	fenway/1345-fenway	5	3	2225	570
Shivani	497	2894	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2894	6	2	1350	152
Shivani	504	2893	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2893	6	2	1375	109
Shivani	543	2895	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2895	6	2	1425	131
Shivani	617	1293	https://junehomes.com/residences/boston-ma/f...	fenway/475-fenway/1293	2	1	1500	164
Shivani	705	1292	https://junehomes.com/residences/boston-ma/f...	fenway/475-fenway/1292	2	1	1575	207
Shivani	825	2890	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2890	6	2	1700	170
Shivani	918	371	https://junehomes.com/residences/boston-ma/f...	fenway/195-fenway/371	3	1	1900	122

Result 24

6. Rentee is looking for temporary accommodation at walking distance (1 mile)from the university

```
select ts.* from temporaryspot ts ,
temporaryroommate tr
where ProximityToUni<1
and tr.Name="Bharath Chandra Bottu" ;
```



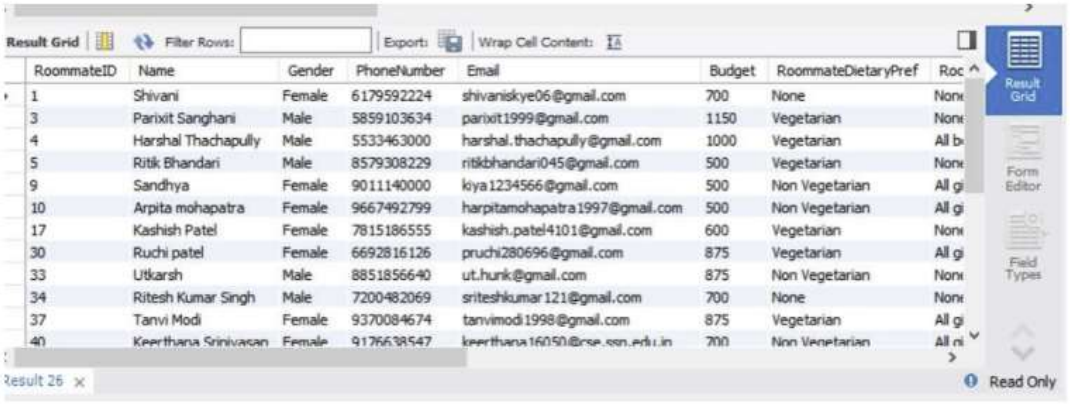
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SpotID	Name	PhoneNumber	Email	Gender	Address	
1	Neha Bhutkar	8573286597	bhutkar.n@northeastern.edu	Female	75 St. Alphonsus Street, Apt., 1816 J Vue at th...	0
2	Srishti	8572507266	srishtiabalamatti@gmail.com	Female	14 Hillside Street	0
4	Manish	8573790129	m.maniish1@gmail.com	Male	115 Northampton Street	0
5	Nainil Maladkar	6178607502	nainil30maladkar@gmail.com	Male	Apt 22, 235 Park drive. Boston 02215	0
6	Sagar Mehta	8573819416	sagarymehta24@gmail.com	Male	682 parker street #1	0
8	KIMAYA SHAH	8578698800	shah.kim@northeastern.edu	Female	1203 Boylston Street	0
9	Supriya Tripathi	6092556159	tripathi.su@northeastern.edu	Female	73 Park Dr	0
10	Mrunal	8574927668	mrnlsuryavanshi@gmail.com	Female	1185 Boylston Street	0
12	Vidip Vimal Kamdar	6084739218	vidipkamdar1912@gmail.com	Male	75 Saint Alphonsus Street, 02120	0
13	Tejas Sridhar	8574379074	tejas3098@gmail.com	Male	1575 Tremont St, Boston, MA-02120	0
17	Riya Virani	3477178224	virani.ria@gmail.com	Female	1167 boylston st	0
22	Devki Patel	3019795975	devkin8112@gmail.com	Female	75 saint Alphonsus street, Boston 02120	0

Result 25

7. Renter is looking for a list of people willing to stay in a private bedroom

```
select sr.* from subleasespot ss, subleaseroommate sr
where sr.TypeOfSpot like "%Private Bedroom%"
and ss.Name="Haseeb";
```



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

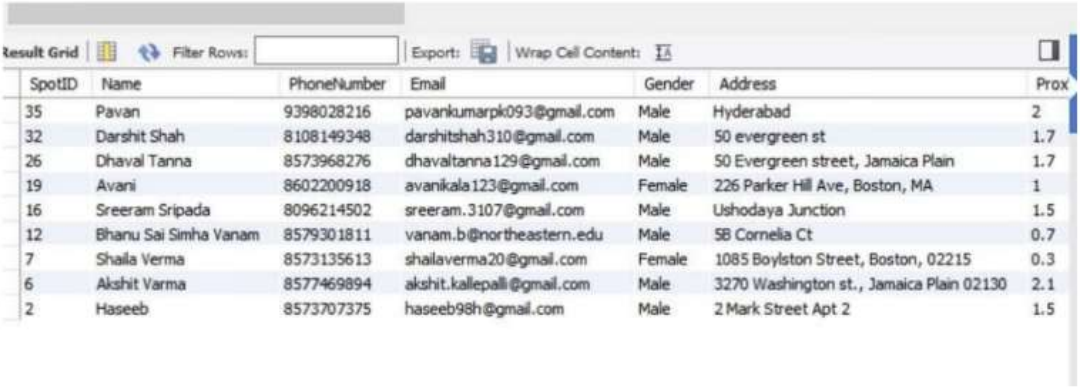
RoommateID	Name	Gender	PhoneNumber	Email	Budget	RoommateDietaryPref	Roc
1	Shivani	Female	6179592224	shivaniskye06@gmail.com	700	None	Non
3	Parixit Sanghani	Male	5859103634	parixit1999@gmail.com	1150	Vegetarian	Non
4	Harshal Thachapully	Male	5533463000	harshal.thachapully@gmail.com	1000	Vegetarian	All b
5	Ritik Bhandari	Male	8579308229	ritikbhandari045@gmail.com	500	Vegetarian	Non
9	Sandhya	Female	9011140000	kiya1234566@gmail.com	500	Non Vegetarian	All gi
10	Arpita mohapatra	Female	9667492799	harpitamohapatra1997@gmail.com	500	Non Vegetarian	All gi
17	Kashish Patel	Female	7815186555	kashish.patel4101@gmail.com	600	Vegetarian	Non
30	Ruchi patel	Female	6692816126	pruchi280696@gmail.com	875	Vegetarian	All gi
33	Utkarsh	Male	885185664O	ut.hunk@gmail.com	875	Non Vegetarian	Non
34	Ritesh Kumar Singh	Male	7200482069	sriteshkumar121@gmail.com	700	None	Non
37	Tanvi Modi	Female	9370084674	tanvimodi1998@gmail.com	875	Vegetarian	All gi
40	Keerthana Srinivasan	Female	9176638547	keerthana16050@cse.ssn.edu.in	700	Non Vegetarian	All gi

Result 26 ×

Result Grid | Form Editor | Field Types | Read Only

8. Female Rentee is looking for a Mixed Gender permanent spot

```
select ss.* from subleasespot ss,
subleaseroommate sr
where sr.Name="Shreya Sharma"
and ss.GenderPref like "%Mixed%";
```

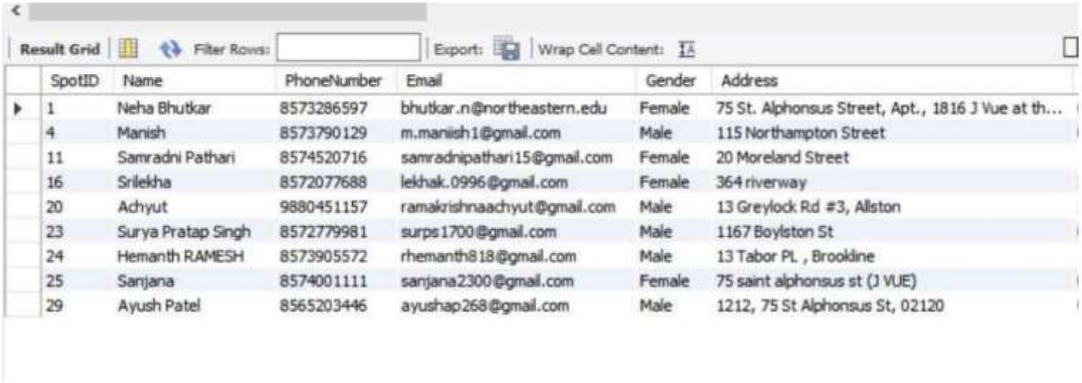


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SpotID	Name	PhoneNumber	Email	Gender	Address	Prox
35	Pavan	9398028216	pavankumarpk093@gmail.com	Male	Hyderabad	2
32	Darshit Shah	8108149348	darshitshah310@gmail.com	Male	50 evergreen st	1.7
26	Dhaval Tanna	8573968276	dhavaltanna129@gmail.com	Male	50 Evergreen street, Jamaica Plain	1.7
19	Avani	8602200918	avanikala123@gmail.com	Female	226 Parker Hill Ave, Boston, MA	1
16	Sreeram Sripada	8096214502	sreeram.3107@gmail.com	Male	Ushodaya Junction	1.5
12	Bhanu Sai Simha Vanam	8579301811	vanam.b@northeastern.edu	Male	5B Cornelia Ct	0.7
7	Shaila Verma	8573135613	shailaverma20@gmail.com	Female	1085 Boylston Street, Boston, 02215	0.3
6	Akshit Varma	8577469894	akshit.kallepalli@gmail.com	Male	3270 Washington st., Jamaica Plain 02130	2.1
2	Haseeb	8573707375	haseeb98h@gmail.com	Male	2 Mark Street Apt 2	1.5

9. The rentee is looking for a temporary hall spot with rent lesser than 20 USD per day

```
select ts.* from temporaryspot ts ,
temporaryroommate tr
where TempRent<20
and tr.Name="Bharath Chandra Bottu";
```

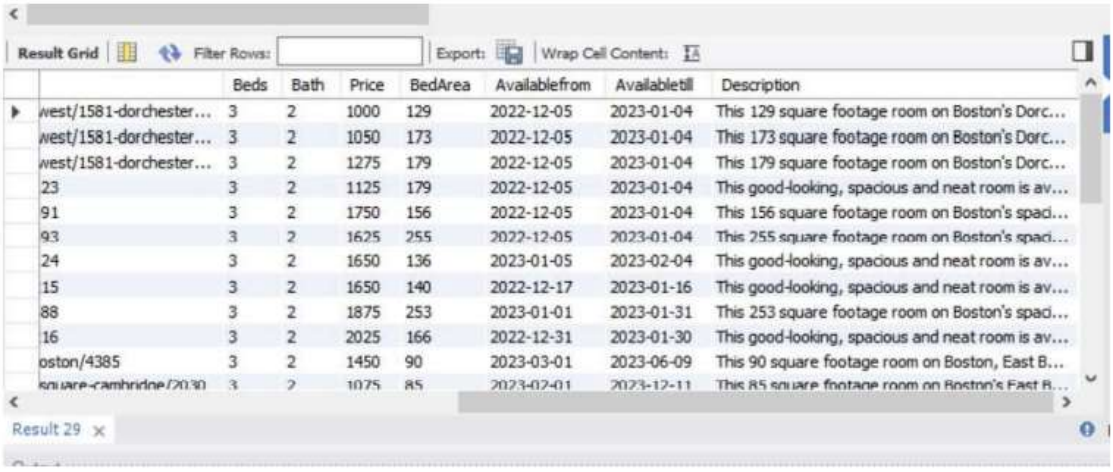


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SpotID	Name	PhoneNumber	Email	Gender	Address
1	Neha Bhutkar	8573286597	bhutkar.n@northeastern.edu	Female	75 St. Alphonsus Street, Apt., 1816 J Vue at th...
4	Manish	8573790129	m.maniish1@gmail.com	Male	115 Northampton Street
11	Samradni Pathari	8574520716	samradnipathari15@gmail.com	Female	20 Moreland Street
16	Srilekha	8572077688	lekhak.0996@gmail.com	Female	364 riverway
20	Achyut	9880451157	ramakrishnaachyut@gmail.com	Male	13 Greylock Rd #3, Allston
23	Surya Pratap Singh	8572779981	surps1700@gmail.com	Male	1167 Boylston St
24	Hemanth RAMESH	8573905572	rhemanth818@gmail.com	Male	13 Tabor PL , Brookline
25	Sanjana	8574001111	sanjana2300@gmail.com	Female	75 saint alphonsus st (J VUE)
29	Ayush Patel	8565203446	ayushap268@gmail.com	Male	1212, 75 St Alphonsus St, 02120

10. The rentee is looking for accommodations on the JuneHomes.com website with 3 bedrooms and 2 bathrooms

```
select sr.name,  j.* from JuneApartments j

join SubleaseRoommate sr

on sr.name='Shivani' and Beds=3 and Bath=2;
```



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

	Beds	Bath	Price	BedArea	Availablefrom	Availabletill	Description
west/1581-dorchester...	3	2	1000	129	2022-12-05	2023-01-04	This 129 square footage room on Boston's Dorc...
west/1581-dorchester...	3	2	1050	173	2022-12-05	2023-01-04	This 173 square footage room on Boston's Dorc...
west/1581-dorchester...	3	2	1275	179	2022-12-05	2023-01-04	This 179 square footage room on Boston's Dorc...
23	3	2	1125	179	2022-12-05	2023-01-04	This good-looking, spacious and neat room is av...
91	3	2	1750	156	2022-12-05	2023-01-04	This 156 square footage room on Boston's spaci...
93	3	2	1625	255	2022-12-05	2023-01-04	This 255 square footage room on Boston's spaci...
24	3	2	1650	136	2023-01-05	2023-02-04	This good-looking, spacious and neat room is av...
15	3	2	1650	140	2022-12-17	2023-01-16	This good-looking, spacious and neat room is av...
88	3	2	1875	253	2023-01-01	2023-01-31	This 253 square footage room on Boston's spaci...
16	3	2	2025	166	2022-12-31	2023-01-30	This good-looking, spacious and neat room is av...
oston/4385	3	2	1450	90	2023-03-01	2023-06-09	This 90 square footage room on Boston, East B...
square-cambridge/2030	3	2	1075	85	2023-02-01	2023-12-11	This 85 square footage room on Boston's East B...

Result 29

11. Rentee is looking for On lease accommodation

```
select ja.* from SubleaseSpot ja,
SubleaseRoommate sr
where sr.name='Ritik Bhandari'
and LeaseSpotType ='On Lease' ;
```

Email	Gender	Address	ProxToUni	Brokerage	LeaseSpotType	Bedroom(
pavankumarpk093@gmail.com	Male	Hyderabad	2	0	On Lease	2
pillai.p@northeastern.edu	Female	892 Huntington Ave	1.1	624	On Lease	2
bhavik.bhosale21@gmail.com	Male	30 woodbine st	1.3	0	On Lease	3
divyamyneni333@gmail.com	Female	75 St.Alphonsus street Roxbury crossing	1.1	630	On Lease	1
abhignareddy721@gmail.com	Female	Apt 3E, 123 Northampton street, Boston	0.4	0	On Lease	2
dhavaltanna129@gmail.com	Male	50 Evergreen street, Jamaica Plain	1.7	383	On Lease	1
sunainaa9@gmail.com	Female	16 Westland Ave	0.3	675	On Lease	2
nainil30maladkar@gmail.com	Male	Apt 22, 235 Park Drive	0.6	0	On Lease	1
kamtam.m@northeastern.edu	Female	18 Rockville Park	1.3	550	On Lease	3
sreeram.3107@gmail.com	Male	Ushodaya Junction	1.5	0	On Lease	3
gauravibendre25@gmail.com	Female	85 Park Drive	0.5	0	On Lease	2
rrithvik18@gmail.com	Male	9 Pompeii street unit 1 Roxbury	1.2	0	On Lease	2

12. Renter wants to find a roommate based on his/her Dietary preferences

```
select sr.* from SubleaseSpot s
join SubleaseRoommate sr
on (s.DietaryPref like "Non%"
and sr.RoommateDietaryPref like "Non%")
or (s.DietaryPref not like "Non%"
and sr.RoommateDietaryPref not like "Non%")
where s.name='Haseeb';
```

RoommateID	Name	Gender	PhoneNumber	Email	Budget	RoommateDieta
1	Shivani	Female	6179592224	shivaniskye06@gmail.com	700	None
2	Shreemoy Nanda	Male	8118075705	moy.nanda97@gmail.com	600	None
6	Shreya Sharma	Female	8573135980	shreya1319.ss@gmail.com	500	Non Vegetarian
8	Murtaza Gheewala	Male	8573132618	murtazag0123@gmail.com	500	Non Vegetarian
9	Sandhya	Female	9011140000	kiya1234566@gmail.com	500	Non Vegetarian
10	Arpita mohapatra	Female	9667492799	harpitamohapatra1997@gmail.com	500	Non Vegetarian
11	Mohan Raj	Male	8573135569	addlurumohanraj@gmail.com	500	Non Vegetarian
13	Kalyani Ramachandra Murthy	Female	8971030195	kalyaniramachandramurthy@gmail.com	600	Non Vegetarian
14	Sreeram Sripada	Male	8096214502	sreeram.3107@gmail.com	500	Non Vegetarian
16	Aditya Nikhil	Male	9949724445	aditya.digala@gmail.com	500	None
18	Mohit Barhate	Male	8424812441	mohitbarhate99@gmail.com	500	Non Vegetarian
19	Srinija	Female	8639771873	srinijapogula@gmail.com	500	None

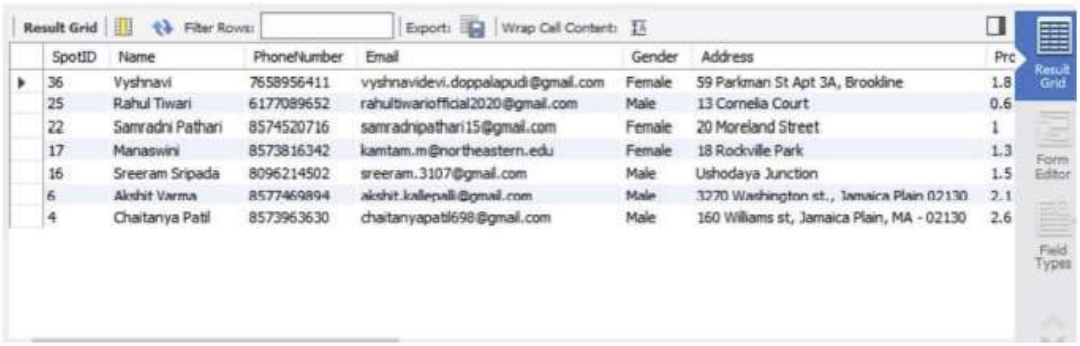
13. Rentee wants to find a house based on his/her gender preferences

```
select s.name, sr.name from SubleaseSpot s

join SubleaseRoommate sr

on s.Gender like "%"

|| sr.RoommateGenderPref

|| "%"

where sr.name='Shreya Sharma';
```

RoommateID	Name	Gender	PhoneNumber	Email	Budget	RoommateDieta
1	Shivani	Female	6179592224	shivaniskye06@gmail.com	700	None
2	Shreemoy Nanda	Male	8118075705	moy.nanda97@gmail.com	600	None
6	Shreya Sharma	Female	8573135980	shreya1319.ss@gmail.com	500	Non Vegetarian
8	Murtaza Gheewala	Male	8573132618	murtazag0123@gmail.com	500	Non Vegetarian
9	Sandhya	Female	9011140000	kiya1234566@gmail.com	500	Non Vegetarian
10	Arpita mohapatra	Female	9667492799	harpitamohapatra1997@gmail.com	500	Non Vegetarian
11	Mohan Raj	Male	8573135569	addlurumohanraj@gmail.com	500	Non Vegetarian
13	Kalyani Ramachandra Murthy	Female	8971030195	kalyaniramachandramurthy@gmail.com	600	Non Vegetarian
14	Sreeram Sripada	Male	8096214502	sreeram.3107@gmail.com	500	Non Vegetarian
16	Aditya Nikhil	Male	9949724445	aditya.digala@gmail.com	500	None
18	Mohit Barhate	Male	8424812441	mohitbarhate99@gmail.com	500	Non Vegetarian
19	Srinija	Female	8639771873	srinijapogula@gmail.com	500	None

14. Rentee wants to find accommodation within the budget between 500 USD-600 USD per month rent

```
select s.* from SubleaseSpot s

join SubleaseRoommate sr

on rent>=500 and rent<=550

where sr.name='Shreya Sharma';
```



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

SpotID	Name	PhoneNumber	Email	Gender	Address	Prc
36	Vyshnavi	7658956411	vyshnavidevi.doppalapudi@gmail.com	Female	59 Parkman St Apt 3A, Brookline	1.8
25	Rahul Tiwari	6177089652	rahultiwariofficial2020@gmail.com	Male	13 Cornelia Court	0.6
22	Samradni Pathari	8574520716	samradnipathari15@gmail.com	Female	20 Moreland Street	1
17	Manaswini	8573816342	kamtam.m@northeastern.edu	Female	18 Rockville Park	1.3
16	Sreeram Sripada	8096214502	sreeram.3107@gmail.com	Male	Ushodaya Junction	1.5
6	Akshit Varma	8577469894	akshit.kallepalli@gmail.com	Male	3270 Washington st., Jamaica Plain 02130	2.1
4	Chaitanya Patil	8573963630	chaitanyapatil698@gmail.com	Male	160 Williams st, Jamaica Plain, MA - 02130	2.6

Result Grid | Form Editor | Field Types

15. The renter is looking for people who need 2 vacant spots in the house

```
select sr.* from subleasespot ss, subleaseroommate sr

where sr.NoOfRoommates >2

and ss.Name="Haseeb";
```

RoommateID	Name	Gender	PhoneNumber	Email	Budget	RoommateDietaryPref	Room
1	Shivani	Female	6179592224	shivaniskye06@gmail.com	700	None	None/I
4	Harshal Thachapully	Male	5533463000	harshal.thachapully@gmail.com	1000	Vegetarian	All boy
5	Ritik Bhandari	Male	8579308229	ritikbhandari045@gmail.com	500	Vegetarian	None/I
6	Shreya Sharma	Female	8573135980	shreya1319.ss@gmail.com	500	Non Vegetarian	None/I
7	Nithyasri	Female	8220231099	nithyasriravi.official@gmail.com	600	Vegetarian	None/I
8	Murtaza Gheewala	Male	8573132618	murtazag0123@gmail.com	500	Non Vegetarian	None/I
9	Sandhya	Female	9011140000	kiya1234566@gmail.com	500	Non Vegetarian	All girls
11	Mohan Raj	Male	8573135569	addlurumohanraj@gmail.com	500	Non Vegetarian	None/I
12	Manogna Pallapothu	Female	8577776288	pallapothu.m@northeastern.edu	500	Vegetarian	All girls
15	Anushka Bhadra	Female	7021780348	anushkabhadra85@gmail.com	600	Vegetarian	All girls
18	Mohit Barhate	Male	8424812441	mohitbarhate99@gmail.com	500	Non Vegetarian	None/I
20	Rajeswari Gampa	Female	9494144343	rajeswarishivrai1@gmail.com	500	Non Vegetarian	All girls

Normalization:

SQL QUERIES:

```
CREATE TABLE `roommate` (
  `roommateid` INT NOT NULL AUTO_INCREMENT,
  `roommatetype` VARCHAR(45), --renter or rentee
  `Name` VARCHAR(45),
  `PhoneNumber` VARCHAR(10),
  `Email` VARCHAR(45),
  `Gender` VARCHAR(45),
  `add_id` INT,
  PRIMARY KEY (`RoommateID`));


  ---AMey ver
   CREATE TABLE `address` (
  `add_id` INT NOT NULL AUTO_INCREMENT,
  `Street` VARCHAR(45),
  `Unit` VARCHAR(45),
  `City` VARCHAR(10),
  `State` VARCHAR(45),
  `Zipcode` VARCHAR(45),
  PRIMARY KEY (`add_id`));



   CREATE TABLE `RoommatePreference` (
        `roommateprefid` INT NOT NULL AUTO_INCREMENT,
        `roommateid` INT NOT NULL,
        `dietpref` VARCHAR(45),
```

```
        `GenderPref` VARCHAR(45),
        `PrefMoveInDate` DATE
  PRIMARY KEY (`preferenceid`));

                CREATE TABLE `SpotPreference` (
        `spotprefid` INT NOT NULL AUTO_INCREMENT,
        `roommateid` INT NOT NULL,
        `accomodation_type` VARCHAR(45), ----------Leased or Temporary
        `Budget` INT,
        `PrefModeofTravel` VARCHAR(45),
        `PrefMoveInDate` DATE,
        `availspot` INT,
  PRIMARY KEY (`preferenceid`));

        CREATE TABLE `Spotamenities` (
        `spotamenitiesid` INT NOT NULL AUTO_INCREMENT,
        `roommateid` INT NOT NULL,
        `Wifi` VARCHAR(10),
        `Wifi` VARCHAR(10),
        `Wifi` VARCHAR(10),
        `PrefMoveInDate` DATE,
        `availspot` INT,
  PRIMARY KEY (`preferenceid`));
```

Use cases and views:

1. The renter is looking for people who need 2 vacant spots in the house.

Create view `People looking for 2 vacancies in a house` as select sr.* from subleasespot ss, subleaseroommate sr where sr.NoOfRoommates >2 and ss.Name="Haseeb";

RoommateID	Name	Gender	PhoneNumber	Email	Budget	RoommateDietaryPref	Room
1	Shivani	Female	6179592224	shivaniskye06@gmail.com	700	None	None/I
4	Harshal Thachapully	Male	5533463000	harshal.thachapully@gmail.com	1000	Vegetarian	All boy
5	Ritik Bhandari	Male	8579308229	ritikbhandari045@gmail.com	500	Vegetarian	None/I
6	Shreya Sharma	Female	8573135980	shreya1319.ss@gmail.com	500	Non Vegetarian	None/I
7	Nithyasri	Female	8220231099	nithyasriravi.official@gmail.com	600	Vegetarian	None/I
8	Murtaza Gheewala	Male	8573132618	murtazag0123@gmail.com	500	Non Vegetarian	None/I
9	Sandhya	Female	9011140000	kiya1234566@gmail.com	500	Non Vegetarian	All girls
11	Mohan Raj	Male	8573135569	addlurumohanraj@gmail.com	500	Non Vegetarian	None/I
12	Manogna Pallapothu	Female	8577776288	pallapothu.m@northeastern.edu	500	Vegetarian	All girls
15	Anushka Bhadra	Female	7021780348	anushkabhadra85@gmail.com	600	Vegetarian	All girls
18	Mohit Barhate	Male	8424812441	mohitbarhate99@gmail.com	500	Non Vegetarian	None/I
20	Rajeswari Gampa	Female	9494144343	rajeswarishivraj1@gmail.com	500	Non Vegetarian	All girls

2. Rentee wants to find accommodation within the budget between 500 USD-600 USD per month rent.

   Create view `People looking for accommodations with monthly rents between 500 and 600 USD` as select s.* from SubleaseSpot s join SubleaseRoommate sr on rent>=500 and rent<=550 where sr.name='Shreya Sharma'

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

SpotID	Name	PhoneNumber	Email	Gender	Address	Pric
36	Vyshnavi	7658956411	vyshnavidevi.doppalapudi@gmail.com	Female	59 Parkman St Apt 3A, Brookline	1.8
25	Rahul Tiwari	6177089652	rahultiwariofficial2020@gmail.com	Male	13 Cornelia Court	0.6
22	Samradni Pathari	8574520716	samradnipathari15@gmail.com	Female	20 Moreland Street	1
17	Manaswini	8573816342	kamtam.m@northeastern.edu	Female	18 Rockville Park	1.3
16	Sreeram Sripada	8096214502	sreeram.3107@gmail.com	Male	Ushodaya Junction	1.5
6	Akshit Varma	8577469894	akshit.kallepalli@gmail.com	Male	3270 Washington st., Jamaica Plain 02130	2.1
4	Chaitanya Patil	8573963630	chaitanyapatil698@gmail.com	Male	160 Williams st, Jamaica Plain, MA - 02130	2.6

Result Grid | Form Editor | Field Types

3. Rentee wants to find a house based on his/her gender preferences.

   Create view `People looking for roommates as per their Gender preferences` as select s.name, sr.name from SubleaseSpot s join SubleaseRoommate sr on s.Gender like "%" || sr.RoommateGenderPref || "%" where sr.name='Shreya Sharma';

RoommateID	Name	Gender	PhoneNumber	Email	Budget	RoommateDieta
1	Shivani	Female	6179592224	shivaniskye06@gmail.com	700	None
2	Shreemoy Nanda	Male	8118075705	moy.nanda97@gmail.com	600	None
6	Shreya Sharma	Female	8573135980	shreya1319.ss@gmail.com	500	Non Vegetarian
8	Murtaza Gheewala	Male	8573132618	murtazag0123@gmail.com	500	Non Vegetarian
9	Sandhya	Female	9011140000	kiya1234566@gmail.com	500	Non Vegetarian
10	Arpita mohapatra	Female	9667492799	harpitamohapatra1997@gmail.com	500	Non Vegetarian
11	Mohan Raj	Male	8573135569	addlurumohanraj@gmail.com	500	Non Vegetarian
13	Kalyani Ramachandra Murthy	Female	8971030195	kalyaniramachandramurthy@gmail.com	600	Non Vegetarian
14	Sreeram Sripada	Male	8096214502	sreeram.3107@gmail.com	500	Non Vegetarian
16	Aditya Nikhil	Male	9949724445	aditya.digala@gmail.com	500	None
18	Mohit Barhate	Male	8424812441	mohitbarhate99@gmail.com	500	Non Vegetarian
19	Srinija	Female	8639771873	srinijapogula@gmail.com	500	None

4. Renter wants to find a roommate based on his/her Dietary preferences.

   Create view `People looking for roommates as per their Dietary preferences` as select sr.* from SubleaseSpot s join SubleaseRoommate sr on (s.DietaryPref like "Non%" and sr.RoommateDietaryPref like "Non%") or (s.DietaryPref not like "Non%" and sr.RoommateDietaryPref not like "Non%") where s.name='Haseeb';

RoommateID	Name	Gender	PhoneNumber	Email	Budget	RoommateDieta
1	Shivani	Female	6179592224	shivaniskye06@gmail.com	700	None
2	Shreemoy Nanda	Male	8118075705	moy.nanda97@gmail.com	600	None
6	Shreya Sharma	Female	8573135980	shreya1319.ss@gmail.com	500	Non Vegetarian
8	Murtaza Gheewala	Male	8573132618	murtazag0123@gmail.com	500	Non Vegetarian
9	Sandhya	Female	9011140000	kiya1234566@gmail.com	500	Non Vegetarian
10	Arpita mohapatra	Female	9667492799	harpitamohapatra1997@gmail.com	500	Non Vegetarian
11	Mohan Raj	Male	8573135569	addlurumohanraj@gmail.com	500	Non Vegetarian
13	Kalyani Ramachandra Murthy	Female	8971030195	kalyaniramachandramurthy@gmail.com	600	Non Vegetarian
14	Sreeram Sripada	Male	8096214502	sreeram.3107@gmail.com	500	Non Vegetarian
16	Aditya Nikhil	Male	9949724445	aditya.digala@gmail.com	500	None
18	Mohit Barhate	Male	8424812441	mohitbarhate99@gmail.com	500	Non Vegetarian
19	Srinija	Female	8639771873	srinijapogula@gmail.com	500	None

5. Rentee is looking for On lease accommodation

   Create view `People looking for On lease accommodations` as select ja.* from SubleaseSpot ja, SubleaseRoommate sr where sr.name='Ritik Bhandari' and LeaseSpotType ='On Lease' ;

Email	Gender	Address	ProxToUni	Brokerage	LeaseSpotType	Bedroom(
pavankumarpk093@gmail.com	Male	Hyderabad	2	0	On Lease	2
pillai.p@northeastern.edu	Female	892 Huntington Ave	1.1	624	On Lease	2
bhavik.bhosale21@gmail.com	Male	30 woodbine st	1.3	0	On Lease	3
divyamyneni333@gmail.com	Female	75 St.Alphonsus street Roxbury crossing	1.1	630	On Lease	1
abhignareddy721@gmail.com	Female	Apt 3E, 123 Northampton street, Boston	0.4	0	On Lease	2
dhavaltanna129@gmail.com	Male	50 Evergreen street, Jamaica Plain	1.7	383	On Lease	1
sunainaa9@gmail.com	Female	16 Westland Ave	0.3	675	On Lease	2
nainil30maladkar@gmail.com	Male	Apt 22, 235 Park Drive	0.6	0	On Lease	1
kamtam.m@northeastern.edu	Female	18 Rockville Park	1.3	550	On Lease	3
sreeram.3107@gmail.com	Male	Ushodaya Junction	1.5	0	On Lease	3
gauravibendre25@gmail.com	Female	85 Park Drive	0.5	0	On Lease	2
rrithvik18@gmail.com	Male	9 Pompeii street unit 1 Roxbury	1.2	0	On Lease	2

6. Rentee is looking for accommodation with the following amenities: Wifi(paid), Laundry.
   Create view `Accommodation with WiFi and Laundry` as
   select ss.* from SubleaseSpot ss, SubleaseRoommate sr where ss.Amenities like "%Wifi%" AND ss.Amenities like "%Laundry%"

address	id	Trans_id	stations	color	walktime	description
harvard-square/443-harvard-square/1169	607	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1164	621	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1166	924	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1163	937	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1168	942	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1167	946	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1165	955	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave

7. Rentee is looking for accommodations on the JuneHomes.com website at Harvard Avenue a min distance from the nearest transportation.

Create view `Accommodation at Harvard Avenue closest to the nearest transportation` as

select j.address,ja.* from JuneApartments j join Junetransport ja on j.id= ja.id

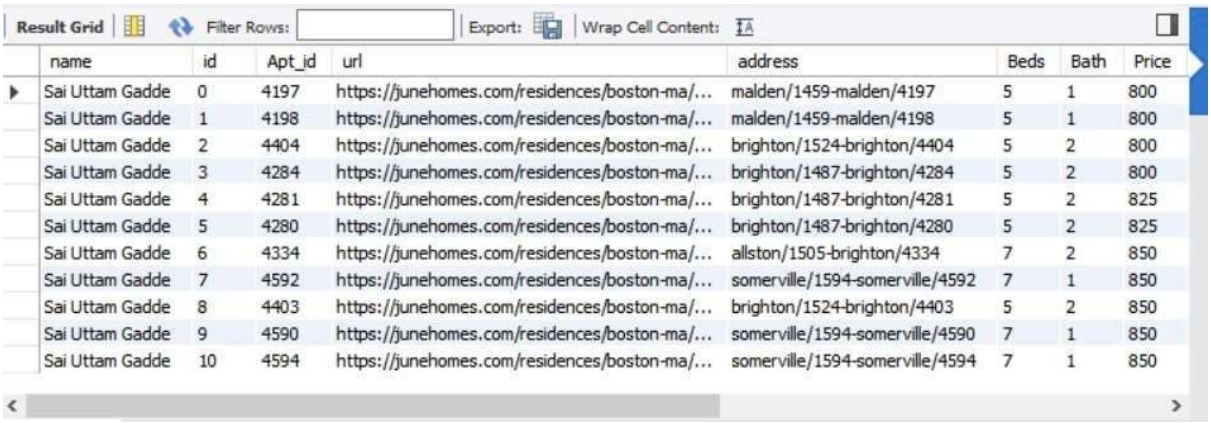
join SubleaseRoommate sr where sr.name='Parixit Sanghani' and address like "%harvard%"

and ja.walktime= (select walktime from Junetransport order by walktime limit 1);

address	id	Trans_id	stations	color	walktime	description
harvard-square/443-harvard-square/1169	607	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1164	621	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1166	924	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1163	937	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1168	942	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1167	946	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave
harvard-square/443-harvard-square/1165	955	1	,1	#F5C00E	1	Mt Auburn St @ Putnam Ave

8. The rentee is looking for accommodations on the JuneHomes.com website with rent according to his Budget per month

Create view `Accommodations with rent according to rentee's Budget` as select sr.name,j.* from JuneApartments j join TemporaryRoommate sr where sr.name='Sai Uttam Gadde' and j.Price <sr.budget;

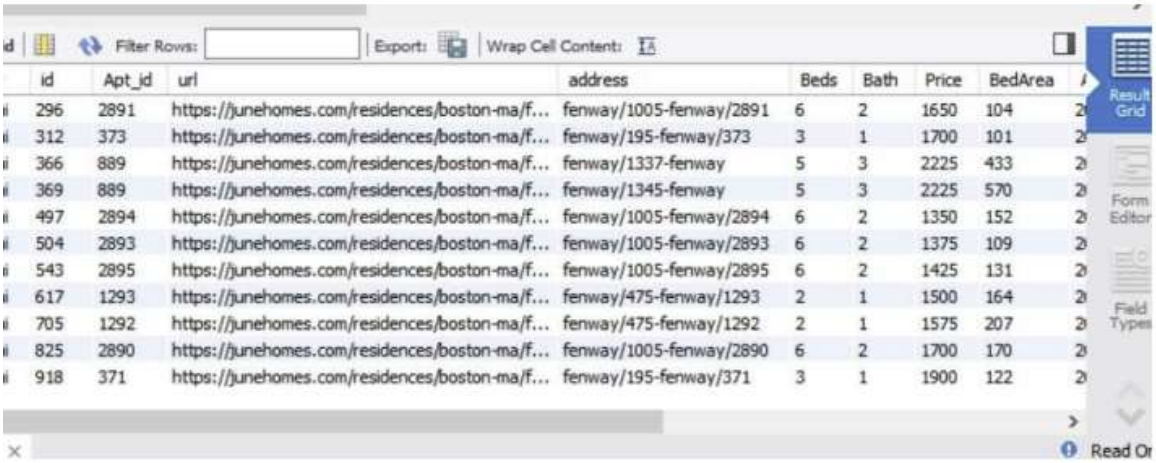


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	id	Apt_id	url	address	Beds	Bath	Price
Sai Uttam Gadde	0	4197	https://junehomes.com/residences/boston-ma/...	malden/1459-malden/4197	5	1	800
Sai Uttam Gadde	1	4198	https://junehomes.com/residences/boston-ma/...	malden/1459-malden/4198	5	1	800
Sai Uttam Gadde	2	4404	https://junehomes.com/residences/boston-ma/...	brighton/1524-brighton/4404	5	2	800
Sai Uttam Gadde	3	4284	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4284	5	2	800
Sai Uttam Gadde	4	4281	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4281	5	2	825
Sai Uttam Gadde	5	4280	https://junehomes.com/residences/boston-ma/...	brighton/1487-brighton/4280	5	2	825
Sai Uttam Gadde	6	4334	https://junehomes.com/residences/boston-ma/...	allston/1505-brighton/4334	7	2	850
Sai Uttam Gadde	7	4592	https://junehomes.com/residences/boston-ma/...	somerville/1594-somerville/4592	7	1	850
Sai Uttam Gadde	8	4403	https://junehomes.com/residences/boston-ma/...	brighton/1524-brighton/4403	5	2	850
Sai Uttam Gadde	9	4590	https://junehomes.com/residences/boston-ma/...	somerville/1594-somerville/4590	7	1	850
Sai Uttam Gadde	10	4594	https://junehomes.com/residences/boston-ma/...	somerville/1594-somerville/4594	7	1	850

. The rentee is looking for accommodations on the JuneHomes.com website near Fenway

```
Create view `Accommodations near Fenway` as

select sr.name, j.* from JuneApartments j

join SubleaseRoommate sr

on sr.name='Shivani' and j.address like "%fenway%";
```



id	Apt_id	url	address	Beds	Bath	Price	BedArea
296	2891	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2891	6	2	1650	104
312	373	https://junehomes.com/residences/boston-ma/f...	fenway/195-fenway/373	3	1	1700	101
366	889	https://junehomes.com/residences/boston-ma/f...	fenway/1337-fenway	5	3	2225	433
369	889	https://junehomes.com/residences/boston-ma/f...	fenway/1345-fenway	5	3	2225	570
497	2894	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2894	6	2	1350	152
504	2893	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2893	6	2	1375	109
543	2895	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2895	6	2	1425	131
617	1293	https://junehomes.com/residences/boston-ma/f...	fenway/475-fenway/1293	2	1	1500	164
705	1292	https://junehomes.com/residences/boston-ma/f...	fenway/475-fenway/1292	2	1	1575	207
825	2890	https://junehomes.com/residences/boston-ma/f...	fenway/1005-fenway/2890	6	2	1700	170
918	371	https://junehomes.com/residences/boston-ma/f...	fenway/195-fenway/371	3	1	1900	122

9.

10.

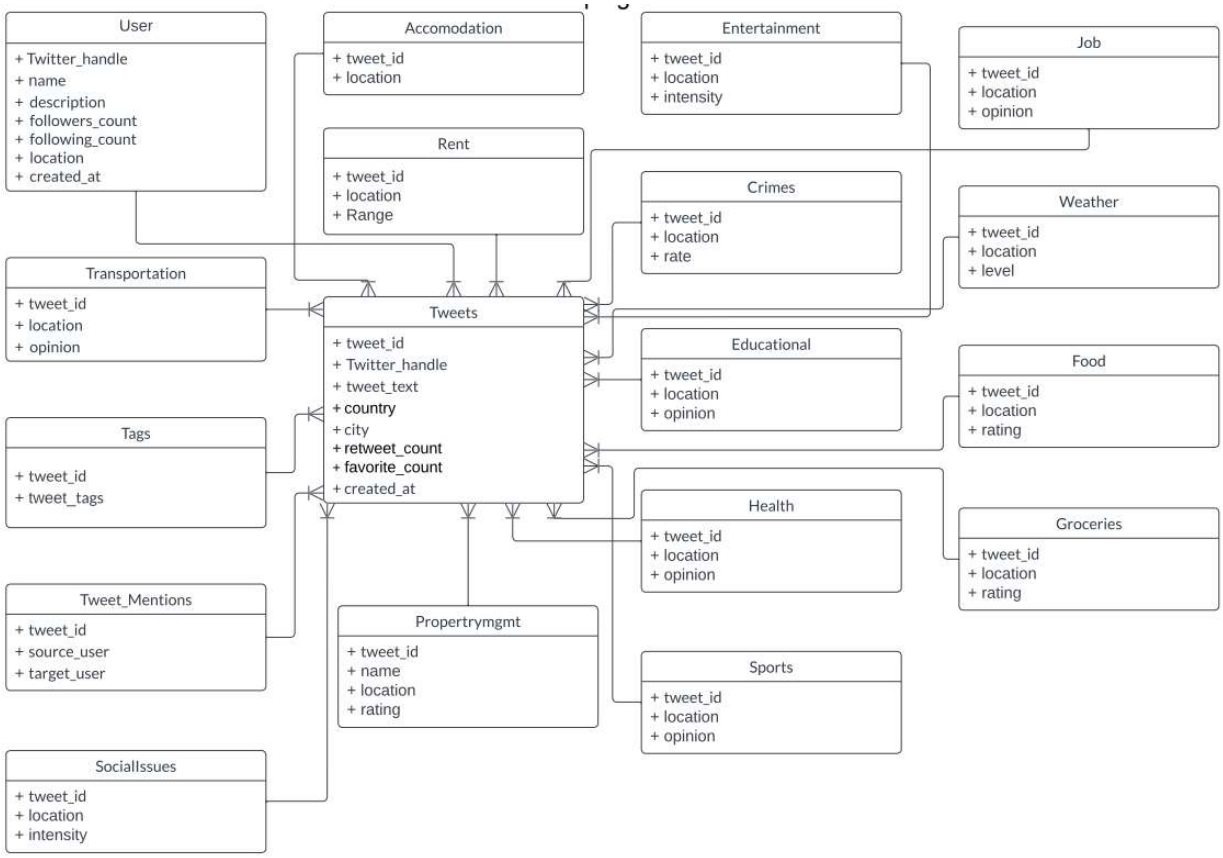
11.

12.
13.
14.

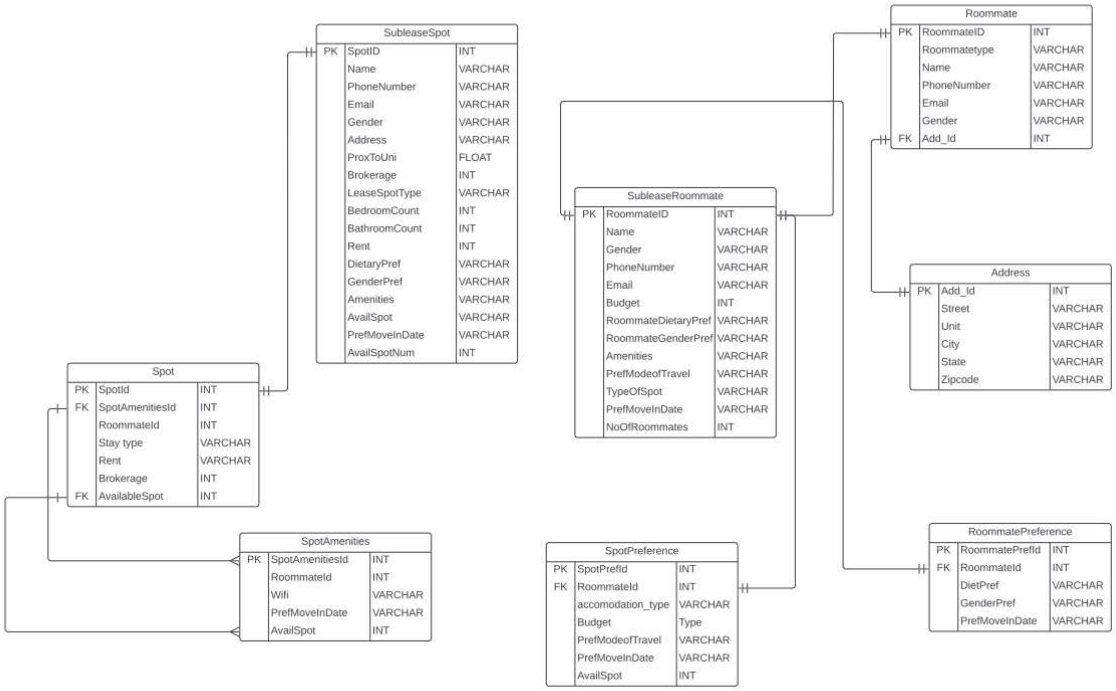
15.

All Entity Relationship Diagrams:

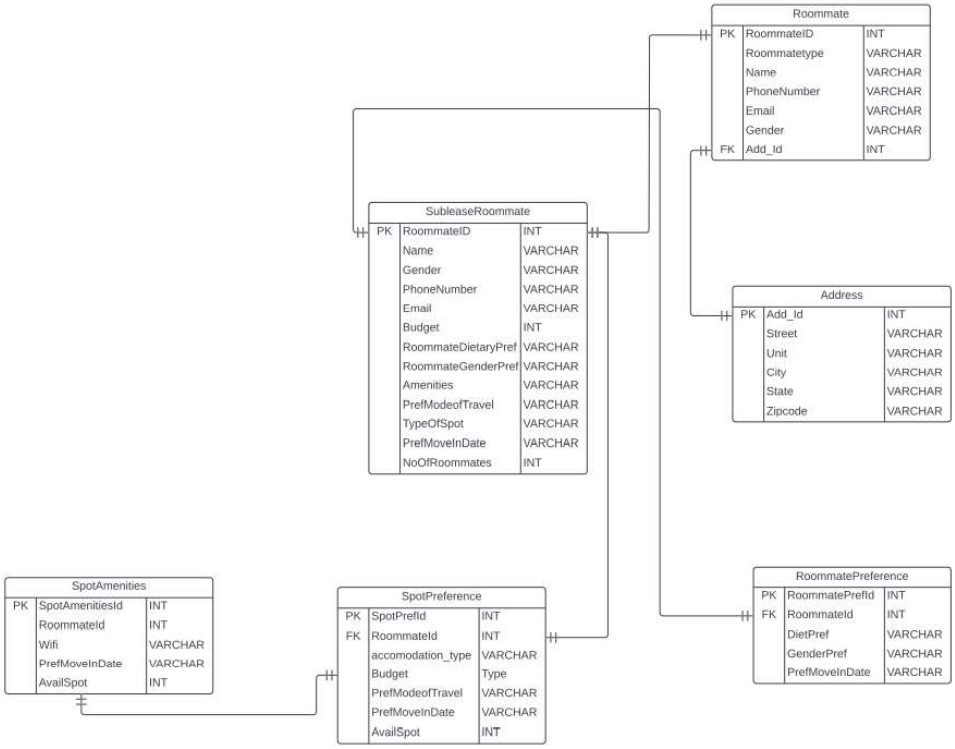
ERD for scraping Twitter:



ERD Before Normalization:



ERD after normalization:



Steps followed with description:

1. Discussed scope and gathered the information for our project. Decided on sources for gathering live data(datasets) as we needed real-time updates about the housing needs of students, like google forms (rental data) and web scraping from JSON and HTML websites as it was a very relevant approach for our project.
2. Created google forms, distributed them across various WhatsApp groups and discords, and collected live data through them.

https://docs.google.com/forms/d/e/1FAIpQLScT6WggMohjiiwYymDhDYA3vVh9kvUCqoTnGbZgxqrfWFX0Jw/viewform?usp=sharing

Roommate/Accommodation Finder (google.com)

Temporary Accommodation Finder (google.com)

Sublet your Spot (google.com)

3. Scraped data from JuneHomes website.
   Boston Room For Rent in Malden - Starting at $800 (junehomes.com)

4. Cleaned the data using following data cleaning methods:
   i) Added data to data frames.
   ii) Removed null values as per the percentage
   iii) Filled the remaining null values with mean/median.
   iv) Removed outliners.
   v) Removed redundant data
   vi) Normalized the data
5. Created tables corresponding to the data found. Populated data to the tables by executing python scripts.
6. Developed use cases considering rental needs in mind and performed joins on tables for retrieving relevant information.
7. Normalized the tables in First, second, and third normal form with python scripting.