

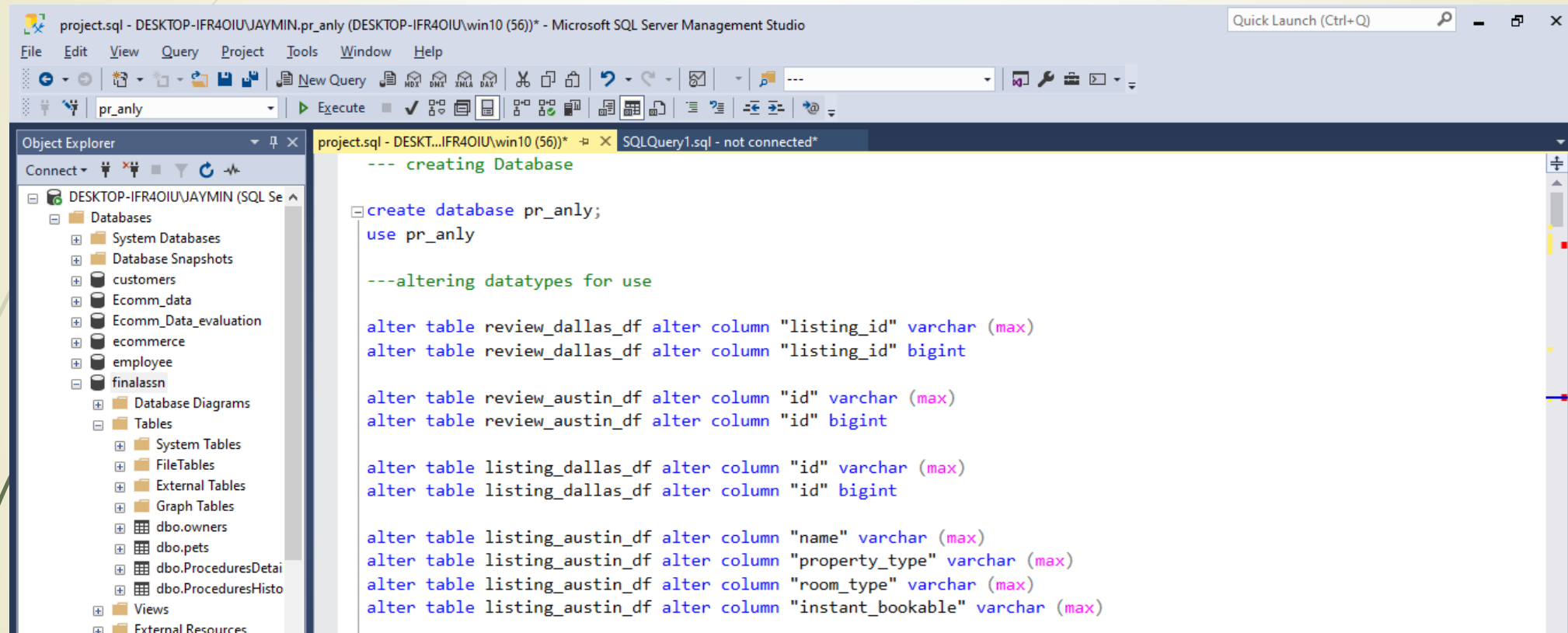
PROPERTY LISTING ANALYSIS FOR PROPERTY RENTAL COMPANY



SQL PROJECT

By: Jaymin Parekh (Code:289)
Sachin Gupta (Code:339)

Beginning with creating DB and altering fields



a. Analyze different metrics to draw the distinction between the different types of property along with their price listings(bucketize them within 3-4 categories basis your understanding):
To achieve this, you can use the following metrics and explore a few yourself as well.

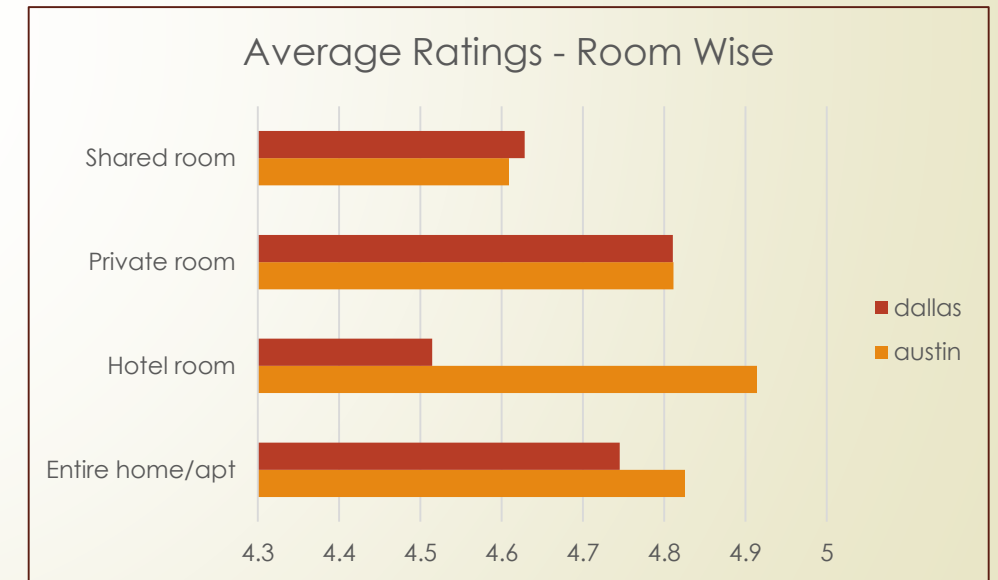
Ans)
Showing SQL Query And Excel Charts

```
project.sql - DESKTOP...IFR4OIU\win10 (56))* - SQLQuery1.sql - not connected*
---(i) based on average ratings (individual cities, years combined)
select room_type,
avg ((review_scores_rating + review_scores_accuracy + review_scores_checkin + review_scores_cleanliness
+ review_scores_communication + review_scores_location + review_scores_value) / 7)
as average_ratings from listing_austin_df
group by room_type

select room_type,
avg ((review_scores_rating + review_scores_accuracy + review_scores_checkin + review_scores_cleanliness
+ review_scores_communication + review_scores_location + review_scores_value) / 7)
as average_ratings from listing_dallas_df
group by room_type
```

room_type	average_ratings
Shared room	4.60917461069803
Entire home/apt	4.82575707738973
Hotel room	4.91428571428571
Private room	4.81139500005329

room_type	average_ratings
Shared room	4.62853002128641
Entire home/apt	4.7454597077984
Hotel room	4.51476190203712
Private room	4.81073978881176



--based on accomodation (Combined for 2 years)

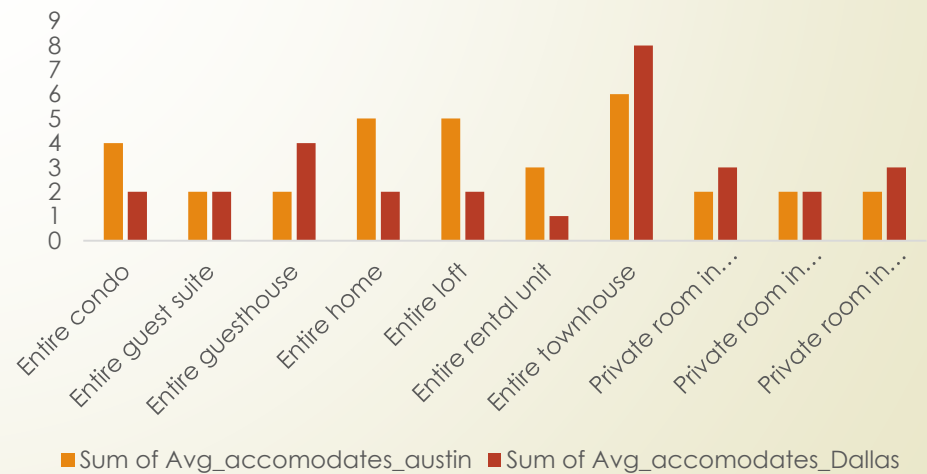
```
select A.*,B.Avg_accomodates_Dallas from
(select A.property_type, Avg(accommodates) as Avg_accomodates_austin from listing_austin_df as A
inner join df_austin_availability as B
on A.id=b.id group by property_type) as A inner join
(select A.property_type, Avg(accommodates) as Avg_accomodates_Dallas from listing_dallas_df as A
inner join df_dallas_availability as B
on A.id=b.id group by property_type) as B
on A.property_type=B.property_type
```

120 %

Results Messages

	property_type	Avg_accomodates_austin	Avg_accomodates_Dallas
1	Entire condo	4	2
2	Entire guest suite	2	2
3	Entire guesthouse	2	4
4	Entire home	5	2
5	Entire loft	5	2
6	Entire rental unit	3	1
7	Entire townhouse	6	8
8	Private room in guest suite	2	3
9	Private room in home	2	2
10	Private room in townhouse	2	3

Accommodation Available per property



b. Study the trends of the different categories and provide insights on same

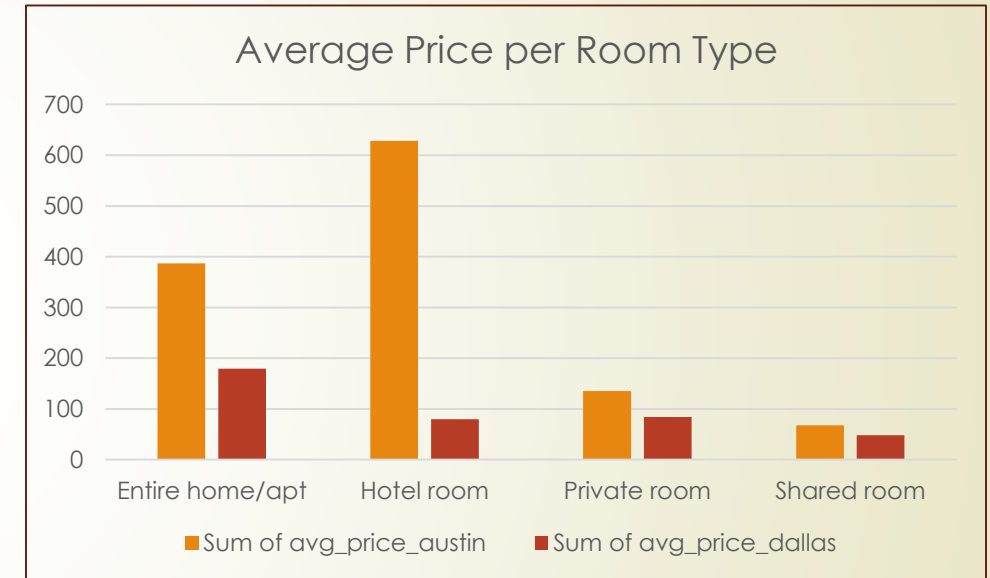
---b. Study the trends of the different categories and provide insights on same


```
SELECT room_type, avg(price) as avg_price_austin
from listing_austin_df group by room_type

SELECT room_type, avg(price) as avg_price_dallas
from listing_dallas_df group by room_type
```

	room_type	avg_price_austin
1	Shared room	67.6068376068376
2	Entire home/apt	386.597701149425
3	Hotel room	628.3
4	Private room	134.958291956306

	room_type	avg_price_dallas
1	Shared room	48.2542372881356
2	Entire home/apt	179.232146709817
3	Hotel room	79.6666666666667
4	Private room	83.8422619047619






```
--- based on property type (individual)
- select property_type, room_type, avg(price) as average_price
  from listing_austin_df
  group by property_type, room_type order by avg(price) desc;

- select property_type, room_type, avg(price) as average_price
  from listing_dallas_df
  group by property_type, room_type order by avg(price) desc;

-- max price top 10
- select top 10 property_type, room_type, max(price) as maximum_price
  from listing_austin_df
  group by property_type, room_type order by max(price) desc;

--- minimum price top 10
- select top 10 property_type, room_type, min(price) as maximum_price
  from listing_austin_df
  group by property_type, room_type order by min(price) desc;
```



c. Using the above analysis, identify top 2 crucial metrics which makes different property types along their listing price that stand ahead of other categories

Ans-

Average Number of Bookings

It signifies that how many repetitive times a particular property is being booked on a particular length of time.

If the number is high it signifies better facilities with proper value, prime location, good services etc. which implies a direct relation with the price of a property.

Availability

It can be also called as the Acceptance rate which represents the share of occupied property during a certain time period. This Directly affects the demand of the property and price of the property.

d. Analyze how does the comments of reviewers vary for listings of distinct categories
(Extract words from the comments provided by the reviewers)

Ans) Both Cities analysed individually

```
--- INDIVIDUAL CITY - DALLAS
with CTE1 as
(select A.property_type, Count (B.Comments) as Beautiful from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Beautiful%'
group by a.property_type),
CTE2 as
(select A.property_type, Count (B.Comments) as great from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%great%'
group by A.property_type),
CTE3 AS
(select A.property_type, Count (B.Comments) as Awsome from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Awsome%'
group by A.property_type),
CTE4 AS
(select A.property_type, Count (B.Comments) as Perfect from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Perfect%'
group by A.property_type),
CTE5 AS
(select A.property_type, Count (B.Comments) as Fantastic from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Fantastic%'
group by A.property_type),
```

Continued...


```
CTE6 AS
(select A.property_type, Count (B.Comments) as Bad from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Bad%'
group by A.property_type),
CTE7 AS
(select A.property_type, Count (B.Comments) as negative from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%negative%'
group by A.property_type),
CTE8 AS
(select A.property_type, Count (B.Comments) as worst from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%worst%'
group by A.property_type),
CTE9 AS
(select A.property_type, Count (B.Comments) as Late from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Late%'
group by A.property_type),
CTE10 AS
(select A.property_type, Count (B.Comments) as Horrible from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Horrible%'
group by A.property_type)
```

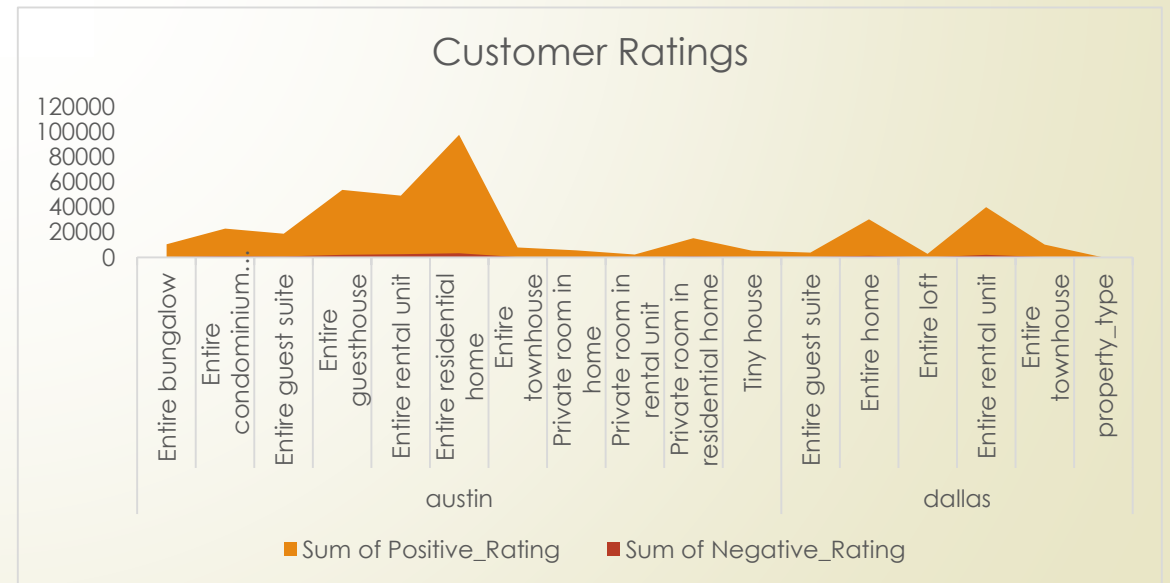
Continued...

CTE10 AS

```
(select A.property_type, Count (B.Comments) as Horrible from listing_dallas_df as A
Inner join review_dallas_df as B on A.id=B.listing_id
where B.Comments like '%Horrible%'
group by A.property_type)
Select A.property_type, (Beautiful + Great + Awsome + Perfect + Fantastic) as Positive_Rating,
(Bad + Negative + worst + late + horrible) as Negative_Rating
from CTE1 as A
inner join CTE2 as B on A.property_type = B.Property_type
inner join CTE3 as C on B.property_type = C.Property_type
inner join CTE4 as D on C.property_type = D.Property_type
inner join CTE5 as E on D.property_type = E.Property_type
inner join CTE6 as F on E.property_type = F.Property_type
inner join CTE7 as G on F.property_type = G.Property_type
inner join CTE8 as H on G.property_type = H.Property_type
inner join CTE9 as I on H.property_type = I.Property_type
inner join CTE10 as J on I.property_type = J.Property_type
```

Output with visualisation

	property_type	Positive_Rating	Negative_Rating
1	Entire guest suite	3849	157
2	Entire home	30350	1308
3	Entire loft	2906	123
4	Entire rental unit	39955	2141
5	Entire townhouse	10253	353



e. Analyze if there is any correlation between property type and their availability across the months

Ans) I have used correlation between property type availability, room type and monthly availability

```
---based on availability (combined years)
select property_type, count (case when available = 'true' then 1 end) as Available,
count (case when Available = 'false' then 1 end) as Not_available
from (select a.id, a.name, a.property_type, b.available from (
(select * from listing_austin_df union select * from listing_dallas_df) as A
inner join (select * from df_austin_availability union select * from df_dallas_availability) as B
on A.id=B.id)) AA
Group by property_type

--- based on years

select room_type, month (date) as Month, year (date) as year, Count (case when Available= 'true' then 1 end) as Available,
Count (case when Available= 'false' then 1 end) as Not_Available from
(select * from df_austin_availability union select * from df_dallas_availability) as A inner join
(select * from listing_austin_df union select * from listing_dallas_df) as B
on A.id=b.id
group by room_type, month (date), year (date)
order by room_type, month (date), Not_Available
```

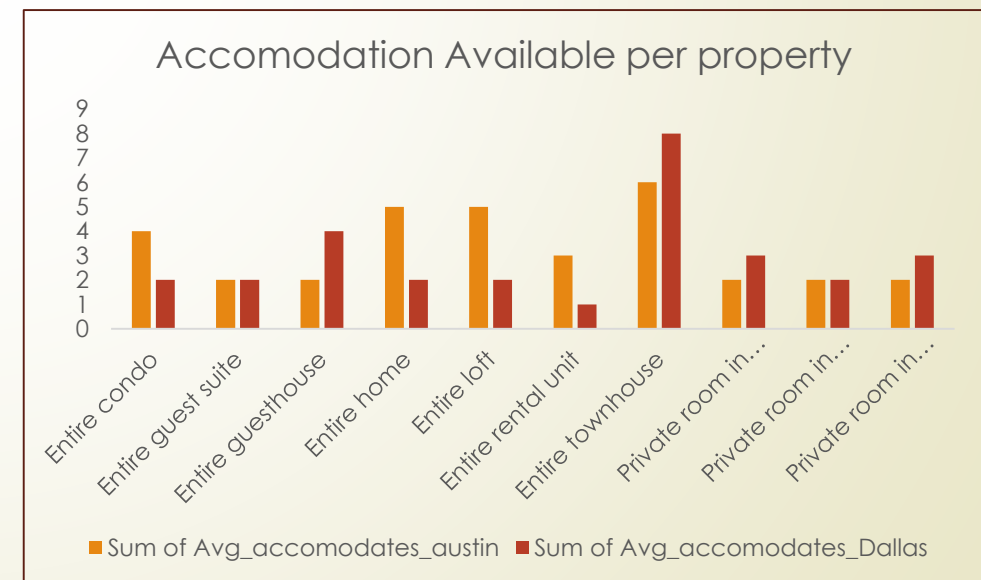
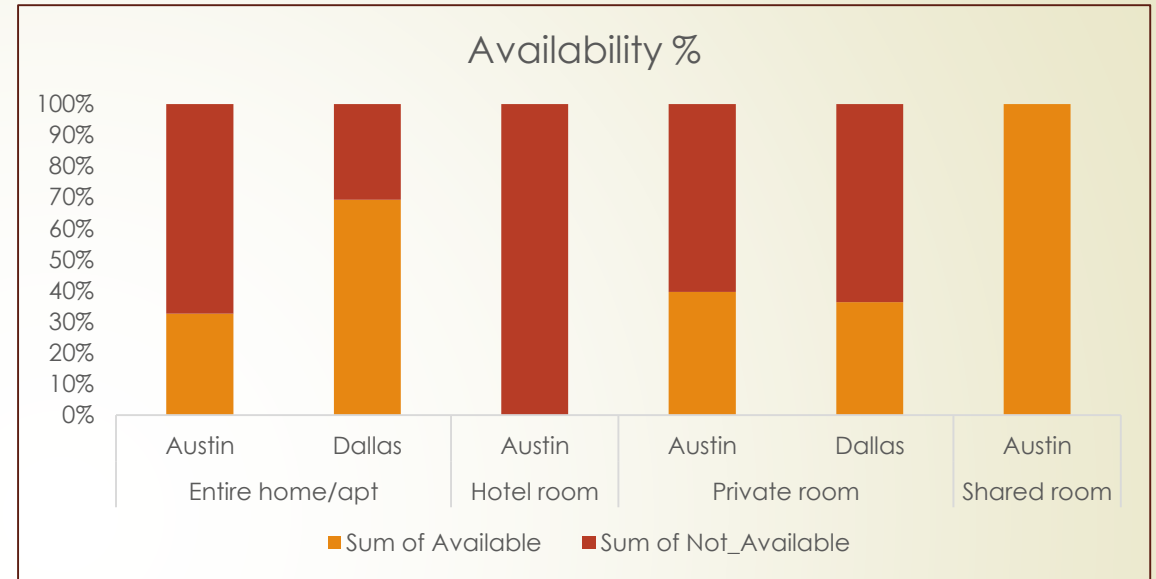
Output with visualisation

Results

Messages

	property_type	Available	Not_available
1	Boat	0	1
2	Camper/RV	4	0
3	Campsite	3	1
4	Entire bungalow	13	43
5	Entire cabin	1	1
6	Entire condo	8	10
7	Entire condom...	23	27
8	Entire cottage	6	2
9	Entire guest s...	15	27
10	Entire guestho...	53	82
11	Entire home	88	107

	room_type	Month	year	Available	Not_Available
1	Entire home/apt	1	2023	45	65
2	Entire home/apt	2	2023	33	49
3	Entire home/apt	3	2022	5	18
4	Entire home/apt	3	2023	30	64
5	Entire home/apt	4	2023	25	8
6	Entire home/apt	4	2022	22	41
7	Entire home/apt	5	2023	11	10
8	Entire home/apt	5	2022	34	45
9	Entire home/apt	6	2022	41	47
10	Entire home/apt	7	2022	43	61
11	Entire home/apt	8	2022	43	45
12	Entire home/apt	9	2022	53	56

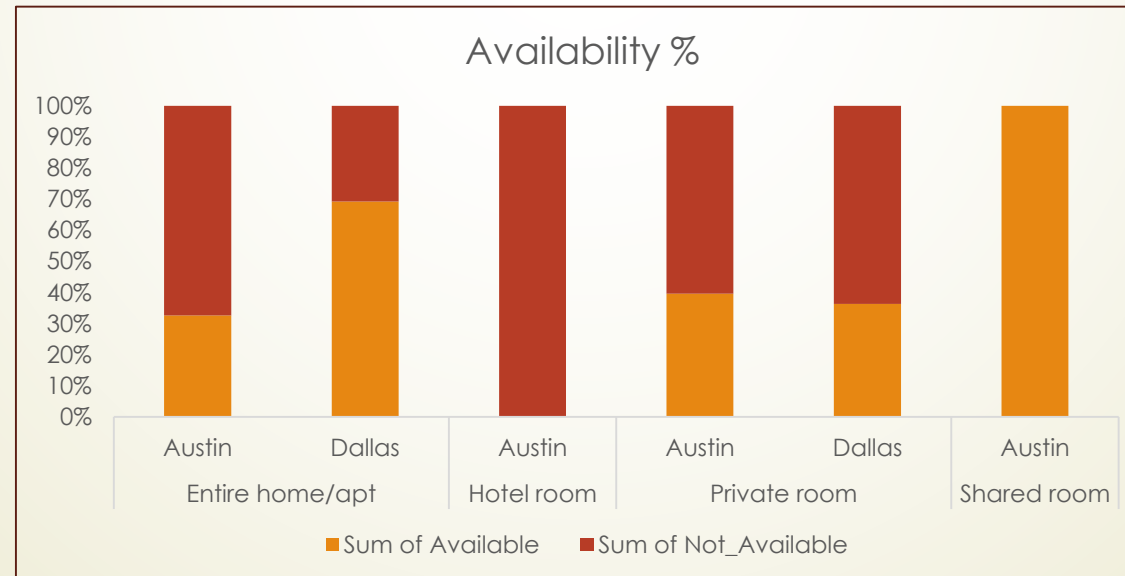


```

---austin
select room_type, month (date) as Month, year (date) as year, Count (case when Available= 'true' then 1 end) as Available,
Count (case when Available= 'false' then 1 end) as Not_Available from
(df_austin_availability as A inner join
listing_austin_df as B
on A.id=b.id)
group by room_type, month (date), year (date)
order by room_type, month (date), Not_Available

---dallas
select room_type, month (date) as Month, year (date) as year, Count (case when Available= 'true' then 1 end) as Available,
Count (case when Available= 'false' then 1 end) as Not_Available from
(df_dallas_availability as A inner join
listing_dallas_df as B
on A.id=b.id)
group by room_type, month (date), year (date)
order by room_type, month (date), Not_Available

```



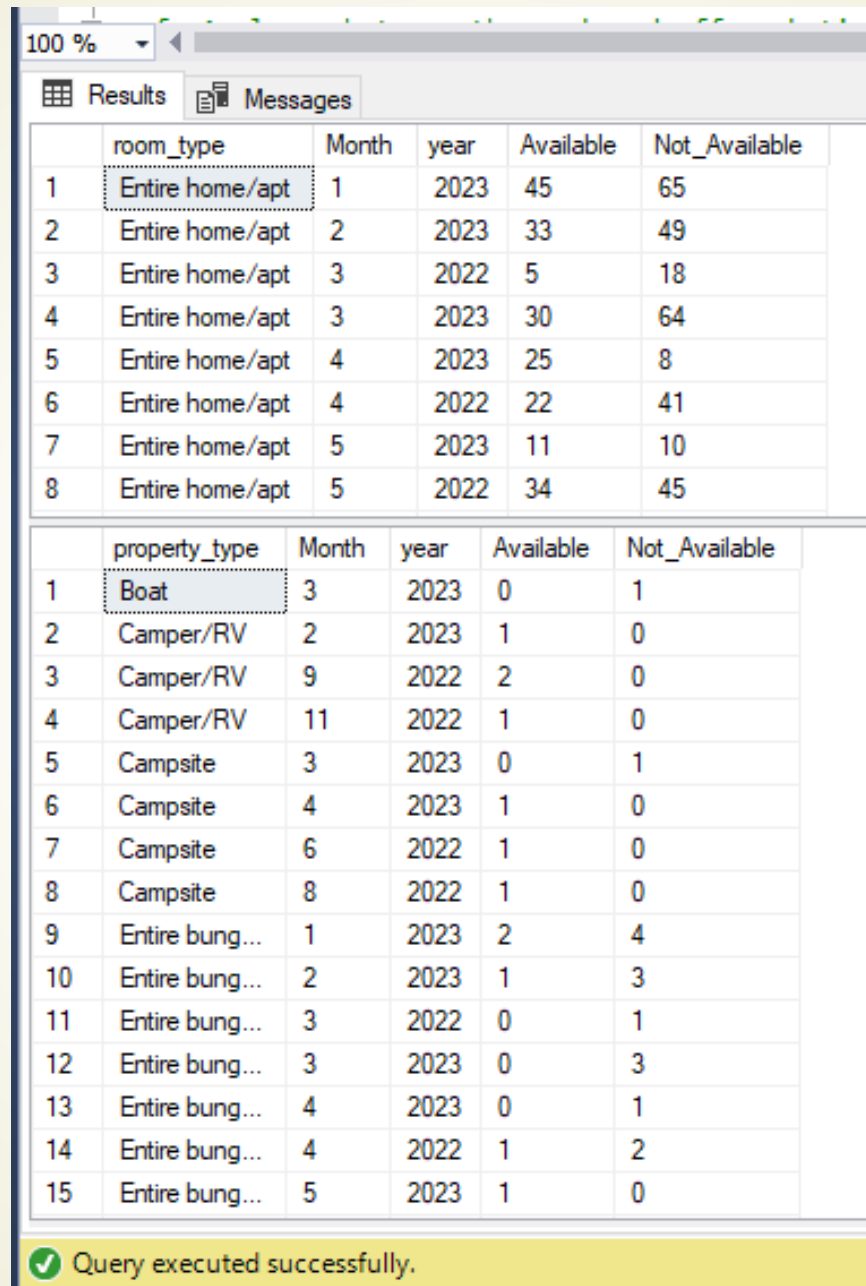
f. Analyze what are the peak and off-peak time for the different categories of property type and their listings. Do we see some commonalities in the trend or is it dependent on the category

```
---f. Analyze what are the peak and off-peak time for the different categories of property type and their listings.
-- Do we see some commonalities in the trend or is it dependent on the category

---room type
select room_type, month (date) as Month, year (date) as year, Count (case when Available= 'true' then 1 end) as Available,
Count (case when Available= 'false' then 1 end) as Not_Available from
(select * from df_austin_availability union select * from df_dallas_availability) as A inner join
(select * from listing_austin_df union select * from listing_dallas_df) as B
on A.id=b.id
group by room_type, month (date), year (date)
order by room_type, month (date), Not_Available

---property type
select property_type, month (date) as Month, year (date) as year, Count (case when Available= 'true' then 1 end) as Available,
Count (case when Available= 'false' then 1 end) as Not_Available from
(select * from df_austin_availability union select * from df_dallas_availability) as A inner join
(select * from listing_austin_df union select * from listing_dallas_df) as B
on A.id=b.id
group by property_type, month (date), year (date)
order by property_type, month (date), Not_Available
```


Output:




The screenshot shows a database query results window with a zoom level of 100%. It contains two tables of data. The first table, titled 'room_type', has 8 rows. The second table, titled 'property_type', has 15 rows. A status bar at the bottom indicates 'Query executed successfully.'

	room_type	Month	year	Available	Not_Available
1	Entire home/apt	1	2023	45	65
2	Entire home/apt	2	2023	33	49
3	Entire home/apt	3	2022	5	18
4	Entire home/apt	3	2023	30	64
5	Entire home/apt	4	2023	25	8
6	Entire home/apt	4	2022	22	41
7	Entire home/apt	5	2023	11	10
8	Entire home/apt	5	2022	34	45

	property_type	Month	year	Available	Not_Available
1	Boat	3	2023	0	1
2	Camper/RV	2	2023	1	0
3	Camper/RV	9	2022	2	0
4	Camper/RV	11	2022	1	0
5	Campsite	3	2023	0	1
6	Campsite	4	2023	1	0
7	Campsite	6	2022	1	0
8	Campsite	8	2022	1	0
9	Entire bung...	1	2023	2	4
10	Entire bung...	2	2023	1	3
11	Entire bung...	3	2022	0	1
12	Entire bung...	3	2023	0	3
13	Entire bung...	4	2023	0	1
14	Entire bung...	4	2022	1	2
15	Entire bung...	5	2023	1	0

✓ Query executed successfully.



g. Using the above analysis, suggest what is the best performing category for the company

Following analysis were observed:

- ❖ Combining both the countries, Entire Home / Apartment are most demanded in room category
- ❖ Entire building is the most booked property, also it is most pre-booked for upcoming year (2023)
- ❖ In Austin Hotel Rooms are most expensive and in Dallas Entire Home are most expensive

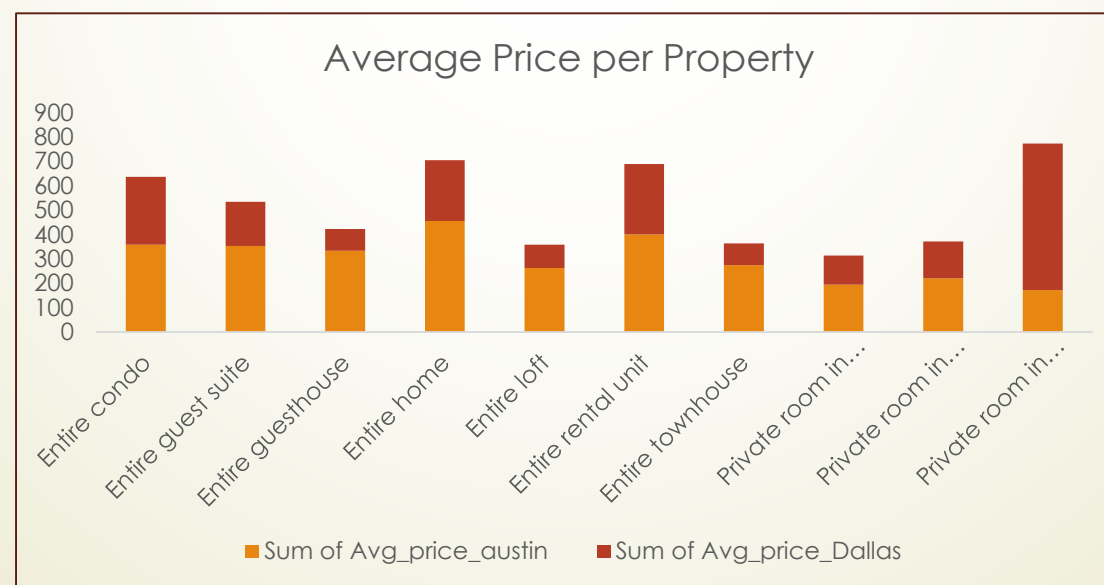
h. Analyze the above trends for the two cities for which data has been provided and provide insights on comparison

Ans) For Avg prices

```
--- avg prices
select A.*,B.Avg_price_Dallas from
(select A.property_type, Avg(B.Price) as Avg_price_austin from listing_austin_df as A
inner join df_austin_availability as B
on A.id=b.id group by property_type) as A inner join
(select A.property_type, Avg(B.Price) as Avg_price_Dallas from listing_dallas_df as A
inner join df_dallas_availability as B
on A.id=b.id group by property_type) as B
on A.property_type=B.property_type
```

Output with visualisation

	property_type	Avg_price_austin	Avg_price_Dallas
1	Entire condo	358.571428571429	277.5
2	Entire guest suite	352.913043478261	181
3	Entire guesthouse	333.467532467532	89
4	Entire home	454.825757575758	250
5	Entire loft	263.142857142857	95
6	Entire rental unit	399.795081967213	288.666666666667
7	Entire townhouse	273.941176470588	90
8	Private room in guest suite	193.75	120
9	Private room in home	220.288888888889	150.666666666667
10	Private room in townhouse	172.25	600



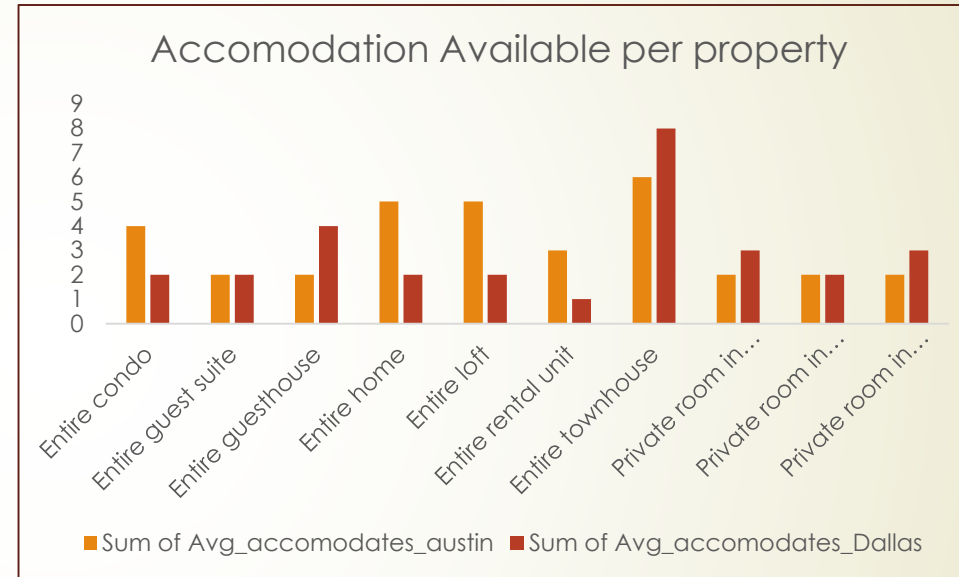
Accommodations

```
--- accommodates
```

```
select A.*,B.Avg_Acc_Dallas from  
(select property_type, Round(Avg(accommodates),2) as Avg_Acc_austin from listing_austin_df  
group by property_type) as A  
Inner Join  
(select property_type, Round(Avg(accommodates),2) as Avg_Acc_Dallas from listing_dallas_df  
group by property_type) as B  
on A.property_type=B.property_type|
```

Output with visualisation

	property_type	Avg_Acc_austin	Avg_Acc_Dallas
1	Camper/RV	3	3
2	Entire bungalow	4	4
3	Entire cabin	5	6
4	Entire condo	3	3
5	Entire cottage	3	3
6	Entire guest suite	2	3
7	Entire guesthouse	2	2
8	Entire home	6	8
9	Entire loft	4	3
10	Entire place	4	7
11	Entire rental unit	3	3
12	Entire serviced apartment	3	3
13	Entire townhouse	5	7
14	Entire vacation home	8	8
15	Entire villa	11	10
16	Private room in bed and breakfast	2	2
17	Private room in bungalow	2	1
18	Private room in casa particular	2	2
19	Private room in condo	2	1



DASHBOARD FOR PROPERTY RENTAL ANALYSIS

city

Austin

dallas

year

2022

2023

property_type

Entire condo

Entire guest suite

Entire guesthouse

Entire home

Entire loft

Entire rental unit

Entire townhouse

Private room in quarte...

room_type

Entire home/apt

Hotel room

Private room

Shared room

Month

1

2

3

4

5

6

7

8

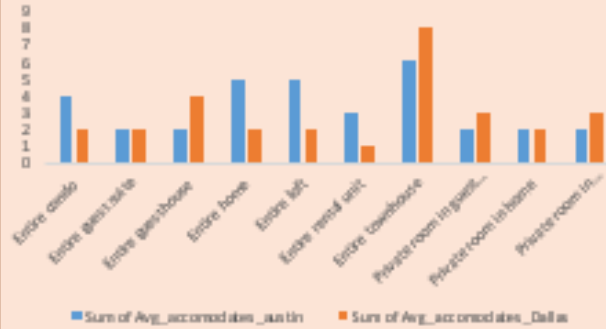
9

10

11

12

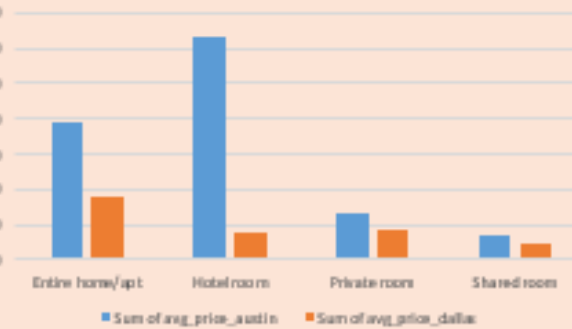
Accommodation Available per property



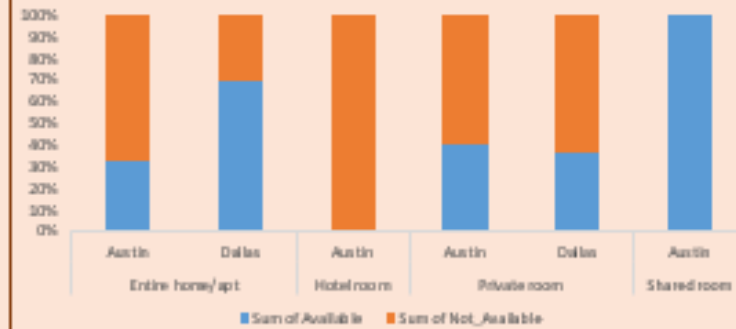
Average Price per Property



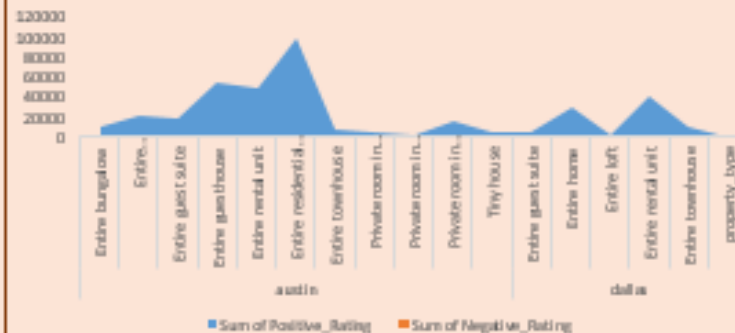
Average Price per Room Type



Availability %



Customer Ratings



Average Ratings - Room Wise

