```
select * from host_athens_df

update host_athens_df
set host_acceptance_rate = 0 ,host_response_rate = 0
where host_acceptance_rate is null or host_response_rate is null

select * from listing_athens_df

update listing_athens_df
set bedrooms = 1 ,beds = 1, review_scores_rating = 0, review_scores_accuracy = 0,
review_scores_cleanliness = 0, review_scores_checkin = 0,
review_scores_communication = 0,review_scores_location = 0,review_scores_value = 0
where bedrooms is null or beds is null or review_scores_rating is null or
review_scores_accuracy is null or
review_scores_cleanliness is null or review_scores_checkin is null or
review_scores_communication is null or review_scores_location is null or
review_scores_value is null


select
instant_bookable,avg(host_acceptance_rate),count(host_athens_df.host_id),sum(host_li
stings_count),max(host_listings_count)
from host_athens_df join listing_athens_df on listing_athens_df.host_id =
host_athens_df.host_id
where host_is_superhost = 1 group by instant_bookable


select host_is_superhost,avg(host_acceptance_rate) as
Avg_Accept_Rate,count(host_athens_df.host_id) as "Total_Hosts",
sum(host_listings_count) as Total_listings from host_athens_df join
listing_athens_df on listing_athens_df.host_id = host_athens_df.host_id
where host_is_superhost is not null or host_has_profile_pic = 1 or
host_identity_verified = 1
group by host_is_superhost

select * from df_thessaloniki_availability

select host_is_superhost,avg(host_acceptance_rate) as
Avg_Accept_Rate,count(host_athens_df.host_id) as "Total_Hosts",
sum(host_listings_count) as Total_listings from host_athens_df join
listing_athens_df on listing_athens_df.host_id = host_athens_df.host_id
where host_is_superhost is not null or host_has_profile_pic = 1 or
host_identity_verified = 1
group by host_is_superhost;

--select * from review_thessaloniki_df

with cte as(
select host_is_superhost, case
when host_has_profile_pic = 'TRUE' then count('TRUE')
```

```
when host_has_profile_pic = 'FALSE' then count('FALSE') end as num_profile_pic
from host_athens_df join listing_athens_df on listing_athens_df.host_id =
host_athens_df.host_id
group by host_is_superhost,host_has_profile_pic)

select host_is_superhost,
case when host_is_superhost=1 then sum(num_profile_pic)
when host_is_superhost=0 then sum(num_profile_pic) end as total_profile_pic
from cte
group by host_is_superhost;


SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
           SUM(l.instant_bookable)
           AS number_of_instand_bookable,
       SUM(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
           min(bedrooms) as Min_Bedrooms,
           min(price) as Min_Price
FROM host_athens_df h
JOIN listing_athens_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost


alter table host_athens_df
alter column host_has_profile_pic int
alter table host_athens_df
alter column host_identity_verified int
alter table listing_athens_df
alter column instant_bookable int



with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified
FROM host_athens_df h
JOIN listing_athens_df l ON h.host_id = l.host_id
```

```
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;

--EXPORT SELECT * FROM results TO '/path/to/results.csv'


SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
           SUM(l.instant_bookable)
           AS number_of_instand_bookable,
       count(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
           min(bedrooms) as Min_Bedrooms,
           min(price) as Min_Price
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost

select * from host_thessaloniki_df

alter table host_thessaloniki_df
alter column host_has_profile_pic int
alter table host_thessaloniki_df
alter column host_identity_verified int
alter table listing_thessaloniki_df
alter column instant_bookable int

select REPLACE(host_has_profile_pic,'True',1) from host_thessaloniki_df

with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)
```

```
select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;



-----------------------------------------------------



SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
JOIN review_thessaloniki_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost



 SELECT h.host_is_superhost, count(r.comments) as number_of__comments
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
JOIN review_thessaloniki_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost

 SELECT h.host_is_superhost, count(property_type) as cnt
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
JOIN review_thessaloniki_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost


select * from host_thessaloniki_df
select * from listing_thessaloniki_df
select * from review_thessaloniki_df


select * from host_rome_df
select * from listing_rome_df

select * from df_venice_availability
```

```
select * from host_venice_df
select * from listing_venice_df
select * from review_venice_df

----CONTRY 2 --

SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
       count(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
           min(bedrooms) as Min_Bedrooms,
           min(price) as Min_Price
FROM host_rome_df h
JOIN listing_rome_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost


alter table host_rome_df
alter column host_has_profile_pic int
alter table host_rome_df
alter column host_identity_verified int
alter table host_rome_df
alter column instant_bookable int


with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating
FROM host_rome_df h
JOIN listing_rome_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;
```

```
--------------------------------------------------------------------------------
----------------------------

SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
       count(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
           min(bedrooms) as Min_Bedrooms,
           AVG(price) as AVG_Price
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost


alter table host_venice_df
alter column host_has_profile_pic int
alter table host_venice_df
alter column host_identity_verified int
alter table listing_venice_df
alter column instant_bookable int


with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;



SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
```

```
JOIN review_venice_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost



 SELECT h.host_is_superhost, count(r.comments) as number_of__comments
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
JOIN review_venice_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost

 SELECT h.host_is_superhost, count(property_type) as cnt
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
JOIN review_venice_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost




select * from host_athens_df

update host_athens_df
set host_acceptance_rate = 0 ,host_response_rate = 0
where host_acceptance_rate is null or host_response_rate is null

select * from listing_athens_df

update listing_athens_df
set bedrooms = 1 ,beds = 1, review_scores_rating = 0, review_scores_accuracy = 0,
review_scores_cleanliness = 0, review_scores_checkin = 0,
review_scores_communication = 0,review_scores_location = 0,review_scores_value = 0
where bedrooms is null or beds is null or review_scores_rating is null or
review_scores_accuracy is null or
review_scores_cleanliness is null or review_scores_checkin is null or
review_scores_communication is null or review_scores_location is null or
review_scores_value is null


select
instant_bookable,avg(host_acceptance_rate),count(host_athens_df.host_id),sum(host_li
stings_count),max(host_listings_count)
from host_athens_df join listing_athens_df on listing_athens_df.host_id =
host_athens_df.host_id
where host_is_superhost = 1 group by instant_bookable
```

```
select host_is_superhost,avg(host_acceptance_rate) as
Avg_Accept_Rate,count(host_athens_df.host_id) as "Total_Hosts",
sum(host_listings_count) as Total_listings from host_athens_df join
listing_athens_df on listing_athens_df.host_id = host_athens_df.host_id
where host_is_superhost is not null or host_has_profile_pic = 1 or
host_identity_verified = 1
group by host_is_superhost

select * from df_thessaloniki_availability

select host_is_superhost,avg(host_acceptance_rate) as
Avg_Accept_Rate,count(host_athens_df.host_id) as "Total_Hosts",
sum(host_listings_count) as Total_listings from host_athens_df join
listing_athens_df on listing_athens_df.host_id = host_athens_df.host_id
where host_is_superhost is not null or host_has_profile_pic = 1 or
host_identity_verified = 1
group by host_is_superhost;

--select * from review_thessaloniki_df

with cte as(
select host_is_superhost, case
when host_has_profile_pic = 'TRUE' then count('TRUE')
when host_has_profile_pic = 'FALSE' then count('FALSE') end as num_profile_pic
from host_athens_df join listing_athens_df on listing_athens_df.host_id =
host_athens_df.host_id
group by host_is_superhost,host_has_profile_pic)

select host_is_superhost,
case when host_is_superhost=1 then sum(num_profile_pic)
when host_is_superhost=0 then sum(num_profile_pic) end as total_profile_pic
from cte
group by host_is_superhost;


SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
           SUM(l.instant_bookable)
           AS number_of_instand_bookable,
       SUM(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
           min(bedrooms) as Min_Bedrooms,
```

```
           min(price) as Min_Price
FROM host_athens_df h
JOIN listing_athens_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost


alter table host_athens_df
alter column host_has_profile_pic int
alter table host_athens_df
alter column host_identity_verified int
alter table listing_athens_df
alter column instant_bookable int



with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified
FROM host_athens_df h
JOIN listing_athens_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;

--EXPORT SELECT * FROM results TO '/path/to/results.csv'


SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
           SUM(l.instant_bookable)
           AS number_of_instand_bookable,
       count(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
```

```
           min(bedrooms) as Min_Bedrooms,
           min(price) as Min_Price
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost

select * from host_thessaloniki_df

alter table host_thessaloniki_df
alter column host_has_profile_pic int
alter table host_thessaloniki_df
alter column host_identity_verified int
alter table listing_thessaloniki_df
alter column instant_bookable int

select REPLACE(host_has_profile_pic,'True',1) from host_thessaloniki_df

with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;



----------------------------------------------------



SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
JOIN review_thessaloniki_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost
```

```
 SELECT h.host_is_superhost, count(r.comments) as number_of__comments
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
JOIN review_thessaloniki_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost

 SELECT h.host_is_superhost, count(property_type) as cnt
FROM host_thessaloniki_df h
JOIN listing_thessaloniki_df l ON h.host_id = l.host_id
JOIN review_thessaloniki_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost


select * from host_thessaloniki_df
select * from listing_thessaloniki_df
select * from review_thessaloniki_df


select * from host_rome_df
select * from listing_rome_df

select * from df_venice_availability
select * from host_venice_df
select * from listing_venice_df
select * from review_venice_df

----CONTRY 2 --

SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
       count(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
           min(bedrooms) as Min_Bedrooms,
           min(price) as Min_Price
FROM host_rome_df h
JOIN listing_rome_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost


alter table host_rome_df
```

```
alter column host_has_profile_pic int
alter table host_rome_df
alter column host_identity_verified int
alter table host_rome_df
alter column instant_bookable int


with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating
FROM host_rome_df h
JOIN listing_rome_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;



-------------------------------------------------------------------------------
-----------------------------

SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
       count(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified,
           count(h.host_id) as "Total_Hosts",
           sum(host_listings_count) as Total_listings,
           avg(accommodates) as Avg_Accomodates,
           min(bedrooms) as Min_Bedrooms,
           AVG(price) as AVG_Price
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
where host_is_superhost is not null
GROUP BY host_is_superhost


alter table host_venice_df
alter column host_has_profile_pic int
alter table host_venice_df
alter column host_identity_verified int
```

```
alter table listing_venice_df
alter column instant_bookable int


with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;



SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
JOIN review_venice_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost



 SELECT h.host_is_superhost, count(r.comments) as number_of__comments
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
JOIN review_venice_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost

 SELECT h.host_is_superhost, count(property_type) as cnt
FROM host_venice_df h
JOIN listing_venice_df l ON h.host_id = l.host_id
JOIN review_venice_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost
```

```
--CANADA


--a. Analyze different metrics to draw the distinction between Super Host and Other
Hosts:
--To achieve this, you can use the following metrics and explore a few yourself as
well.
--Acceptance rate, response rate, instant booking, profile picture, identity
verified, review scores, average no of bookings per month, etc.

----Tornto

select top 10 * from host_toronto_df
select top 10 * from review_toronto_df
select top 10 * from listing_toronto_df

update host_toronto_df
set host_acceptance_rate=0, host_is_superhost=0,host_has_profile_pic='TRUE'
where host_acceptance_rate is null or host_is_superhost is null or
host_has_profile_pic is null ;

update host_toronto_df
set host_has_profile_pic=1
where host_has_profile_pic='TRUE'

update host_toronto_df
set host_has_profile_pic=0
where host_has_profile_pic='FALSE'



alter table host_toronto_df
alter column host_has_profile_pic int
alter table host_toronto_df
alter column host_identity_verified int
alter table listing_toronto_df
alter column instant_bookable int

SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified,
count(h.host_id) as "Number_of_Hosts", sum(host_listings_count) as Total_listings,
avg(accommodates) as Avg_Accomodates, max(bedrooms) as Max_Bedrooms,
min(price) as Min_Price
FROM host_toronto_df h
JOIN listing_toronto_df l ON h.host_id = l.host_id
```

```
GROUP BY host_is_superhost;

--vancouver


update host_vancouver_df
set host_response_rate=0, host_acceptance_rate=0, host_is_superhost=0,
host_listings_count=0, host_has_profile_pic=0, host_identity_verified=0
where host_response_rate is null or host_acceptance_rate is null or
host_is_superhost is null or host_listings_count is null or host_has_profile_pic is
null or host_identity_verified is null

alter table host_vancouver_df
alter column host_has_profile_pic int
alter table host_vancouver_df
alter column host_identity_verified int
alter table listing_vancouver_df
alter column instant_bookable int

SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate,AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS number_identity_verified,
count(h.host_id) as "Number_of_Hosts", sum(host_listings_count) as Total_listings,
avg(accommodates) as Avg_Accomodates, max(bedrooms) as Max_Bedrooms,
min(price) as Min_Price
FROM host_vancouver_df h
JOIN listing_vancouver_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost;

--b.Using the above analysis, identify the top 3 crucial metrics one needs to
maintain to become a Super Host and also, find their average values.

--Toronto

---top 3 matrics to maintain a become superhost are:-

--1.Acceptance rate: It is important to have a high acceptance rate to show that you
are reliable and responsive to booking requests.

--2.Response rate: A high response rate demonstrates that you are available and easy
to communicate with, which can help build trust with potential guests.

--3.Review scores: High review scores show that you are offering a high-quality
experience and are likely to attract more bookings.

with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
```

```
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified
FROM host_toronto_df h
JOIN listing_toronto_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;

----vancouver

with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified
FROM host_vancouver_df h
JOIN listing_vancouver_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;

--c.Analyze how the comments of reviewers vary for listings of Super Hosts vs Other
Hosts(Extract words from the comments provided by the reviewers)

---Toronto

SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_toronto_df h
JOIN listing_toronto_df l ON h.host_id = l.host_id
JOIN review_toronto_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost
```

```
SELECT h.host_is_superhost, count(r.comments) as number_of_negative_comments
FROM host_toronto_df h
JOIN listing_toronto_df l ON h.host_id = l.host_id
JOIN review_toronto_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost

--Vancouver

SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_vancouver_df h
JOIN listing_vancouver_df l ON h.host_id = l.host_id
JOIN review_vancouver_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost



SELECT h.host_is_superhost, count(r.comments) as number_of_negative_comments
FROM host_vancouver_df h
JOIN listing_vancouver_df l ON h.host_id = l.host_id
JOIN review_vancouver_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost


--d.Analyze do Super Hosts tend to have large property types as compared to Other
Hosts

--Toronto

SELECT h.host_is_superhost, count(property_type) as number_large_property
FROM host_toronto_df h
JOIN listing_toronto_df l ON h.host_id = l.host_id
JOIN review_toronto_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost

--Vancouver

SELECT h.host_is_superhost, count(property_type) as number_large_property
FROM host_vancouver_df h
JOIN listing_vancouver_df l ON h.host_id = l.host_id
JOIN review_vancouver_df r ON l.id = r.listing_id
```

```
where  property_type like '%entire%'
group by h.host_is_superhost



--CHINA


--a. Analyze different metrics to draw the distinction between Super Host and Other
Hosts:
--To achieve this, you can use the following metrics and explore a few yourself as
well.
--Acceptance rate, response rate, instant booking, profile picture, identity
verified, review scores, average no of bookings per month, etc.

----Beijing

select top 10 * from host_beijing_df
select top 10 * from review_beijing_df
select top 10 * from listing_beijing_df

update host_beijing_df
set host_response_rate=0 ,host_acceptance_rate=0,
host_is_superhost=0,host_has_profile_pic='TRUE', host_identity_verified='True'
where host_response_rate is null or host_acceptance_rate is null or
host_is_superhost is null or host_has_profile_pic is null
or host_identity_verified is null ;

update host_beijing_df
set host_has_profile_pic=1
where host_has_profile_pic='TRUE'

update host_beijing_df
set host_has_profile_pic=0
where host_has_profile_pic='FALSE'

update host_beijing_df
set host_identity_verified=1
where host_identity_verified='TRUE'

update host_beijing_df
set host_identity_verified=0
where host_identity_verified='FALSE'



alter table host_beijing_df
alter column host_has_profile_pic int
alter table host_beijing_df
```

```
alter column host_identity_verified int
alter table listing_beijing_df
alter column instant_bookable int

SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified,
count(h.host_id) as "Number_of_Hosts", sum(host_listings_count) as Total_listings,
avg(accommodates) as Avg_Accomodates, max(bedrooms) as Max_Bedrooms,
min(price) as Min_Price
FROM host_beijing_df h
JOIN listing_beijing_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost;

--Shanghai

select top 10 * from host_shanghai_df
select top 10 * from review_shanghai_df
select top 10 * from listing_shanghai_df

update host_shanghai_df
set host_response_rate=0 ,host_acceptance_rate=0,
host_is_superhost=0,host_has_profile_pic='TRUE'
where host_response_rate is null or host_acceptance_rate is null or
host_is_superhost is null or host_has_profile_pic is null  ;

update host_shanghai_df
set host_has_profile_pic=1
where host_has_profile_pic='TRUE'

update host_shanghai_df
set host_has_profile_pic=0
where host_has_profile_pic='FALSE'

alter table host_shanghai_df
alter column host_has_profile_pic int
alter table host_shanghai_df
alter column host_identity_verified int
alter table listing_shanghai_df
alter column instant_bookable int

SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate,AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS number_identity_verified,
```

```
count(h.host_id) as "Number_of_Hosts", sum(host_listings_count) as Total_listings,
avg(accommodates) as Avg_Accomodates, max(bedrooms) as Max_Bedrooms,
min(price) as Min_Price
FROM host_shanghai_df h
JOIN listing_shanghai_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost;

--b.Using the above analysis, identify the top 3 crucial metrics one needs to
maintain to become a Super Host and also, find their average values.

--Beijing

---top 3 matrics to maintain a become superhost are:-

--1.Acceptance rate: It is important to have a high acceptance rate to show that you
are reliable and responsive to booking requests.

--2.Response rate: A high response rate demonstrates that you are available and easy
to communicate with, which can help build trust with potential guests.

--3.Review scores: High review scores show that you are offering a high-quality
experience and are likely to attract more bookings.

with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified,
count(h.host_id) as "Number_of_Hosts", sum(host_listings_count) as Total_listings,
avg(accommodates) as Avg_Accomodates, max(bedrooms) as Max_Bedrooms,
min(price) as Min_Price
FROM host_beijing_df h
JOIN listing_beijing_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;

----Shanghai

with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate,AVG(l.review_scores_rating) as
```

```
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS number_identity_verified,
count(h.host_id) as "Number_of_Hosts", sum(host_listings_count) as Total_listings,
avg(accommodates) as Avg_Accomodates, max(bedrooms) as Max_Bedrooms,
min(price) as Min_Price
FROM host_shanghai_df h
JOIN listing_shanghai_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate ,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost is not null
group by
host_is_superhost,avg_acceptance_rate,avg_review_score_rating,avg_response_rate;

--c.Analyze how the comments of reviewers vary for listings of Super Hosts vs Other
Hosts(Extract words from the comments provided by the reviewers)

---Beijing

SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_beijing_df h
JOIN listing_beijing_df l ON h.host_id = l.host_id
JOIN review_beijing_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost



SELECT h.host_is_superhost, count(r.comments) as number_of_negative_comments
FROM host_beijing_df h
JOIN listing_beijing_df l ON h.host_id = l.host_id
JOIN review_beijing_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost

--Shanghai

SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_shanghai_df h
JOIN listing_shanghai_df l ON h.host_id = l.host_id
JOIN review_shanghai_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
```

```
group by h.host_is_superhost



SELECT h.host_is_superhost, count(r.comments) as number_of_negative_comments
FROM host_shanghai_df h
JOIN listing_shanghai_df l ON h.host_id = l.host_id
JOIN review_shanghai_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by  h.host_is_superhost


--d.Analyze do Super Hosts tend to have large property types as compared to Other
Hosts

--Beijing

SELECT h.host_is_superhost, count(property_type) as number_large_property
FROM host_beijing_df h
JOIN listing_beijing_df l ON h.host_id = l.host_id
JOIN review_beijing_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost

--Shanghai

SELECT h.host_is_superhost, count(property_type) as number_large_property
FROM host_shanghai_df h
JOIN listing_shanghai_df l ON h.host_id = l.host_id
JOIN review_shanghai_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost




select * from review_austin_df
select * from review_dallas_df
select * from listing_austin_df
select * from listing_dallas_df
select * from host_austin_df
select * from host_dallas_df
select * from df_dallas_availability

SELECT *
FROM host_dallas_df h
JOIN listing_dallas_df l ON h.host_id = l.host_id
JOIN review_dallas_df r ON l.id = r.listing_id
```

```
JOIN df_dallas_availability a on a.id = l.id



SELECT *
FROM host_austin_df h
JOIN listing_austin_df l ON h.host_id = l.host_id
JOIN review_austin_df r ON l.id = r.listing_id



alter table host_austin_df
alter column host_has_profile_pic int
alter table host_austin_df
alter column host_identity_verified int
alter table listing_austin_df
alter column instant_bookable int

----------
UPDATE  host_austin_df
SET host_is_superhost = 0,host_acceptance_rate = 0,host_has_profile_pic = 0
,host_response_rate = 0
WHERE host_is_superhost IS NULL OR host_acceptance_rate IS NULL OR
host_has_profile_pic IS NULL OR host_response_rate IS NULL

UPDATE listing_austin_df
SET instant_bookable = 0
WHERE instant_bookable IS NULL




--AUSTIN CITY
------
--a. Analyze different metrics to draw the distinction between Super Host and Other
Hosts:
--To achieve this, you can use the following metrics and explore a few yourself as
well.
--Acceptance rate, response rate, instant booking, profile picture, identity
verified,
--review scores, average no of bookings per month, etc.


SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
       AVG(h.host_response_rate) AS avg_response_rate,
           SUM(l.instant_bookable)
           AS number_of_instand_pic,
       SUM(h.host_has_profile_pic)
           AS number_of_profile_pic,
```

```
           SUM(h.host_identity_verified)
           AS number_identity_verified
FROM host_austin_df h
JOIN listing_austin_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost


 --b. Using the above analysis, identify the top 3 crucial metrics one needs to
maintain
 --to become a Super Host and also, find their average values.

 with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified
FROM host_austin_df h
JOIN listing_austin_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select host_is_superhost,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost=1
group by host_is_superhost;




 --c. Analyze how the comments of reviewers vary for listings of Super Hosts vs
Other Hosts(Extract words
 --from the comments provided by the reviewers)



SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_austin_df h
JOIN listing_austin_df l ON h.host_id = l.host_id
JOIN review_austin_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost



 SELECT h.host_is_superhost, count(r.comments) as number_of__comments
FROM host_austin_df h
JOIN listing_austin_df l ON h.host_id = l.host_id
```

```
JOIN review_austin_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost



--d. Analyze do Super Hosts tend to have large property types as compared to Other
Hosts

 SELECT h.host_is_superhost, count(property_type) as cnt
FROM host_austin_df h
JOIN listing_austin_df l ON h.host_id = l.host_id
JOIN review_austin_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost



--DALLAS CITY
alter table host_dallas_df
alter column host_has_profile_pic int
alter table host_dallas_df
alter column host_identity_verified int
alter table listing_dallas_df
alter column instant_bookable int
------

UPDATE  host_dallas_df
SET host_is_superhost = 0,host_acceptance_rate = 0,host_has_profile_pic = 0
,host_response_rate = 0
WHERE host_is_superhost IS NULL OR host_acceptance_rate IS NULL OR
host_has_profile_pic IS NULL OR host_response_rate IS NULL

UPDATE listing_dallas_df
SET instant_bookable = 0
WHERE instant_bookable IS NULL

--a. Analyze different metrics to draw the distinction between Super Host and Other
Hosts:
--To achieve this, you can use the following metrics and explore a few yourself as
well.
--Acceptance rate, response rate, instant booking, profile picture, identity
verified,
--review scores, average no of bookings per month, etc.


SELECT h.host_is_superhost,
           AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
```

```
       AVG(h.host_response_rate) AS avg_response_rate,
           sum(l.instant_bookable)
           AS number_of_instand_pic,
       SUM(h.host_has_profile_pic)
           AS number_of_profile_pic,
           SUM(h.host_identity_verified)
           AS number_identity_verified
FROM host_dallas_df h
JOIN listing_dallas_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost


 --b. Using the above analysis, identify the top 3 crucial metrics one needs to
maintain
 --to become a Super Host and also, find their average values.

 with cte as
(SELECT h.host_is_superhost,AVG(h.host_acceptance_rate) AS avg_acceptance_rate,
AVG(h.host_response_rate) AS avg_response_rate, AVG(l.review_scores_rating) as
avg_review_score_rating,
SUM(l.instant_bookable) AS total_instand_bookable,
SUM(h.host_has_profile_pic) AS total_profile_pic,
SUM(h.host_identity_verified) AS Total_identity_verified
FROM host_dallas_df h
JOIN listing_dallas_df l ON h.host_id = l.host_id
GROUP BY host_is_superhost)

select host_is_superhost,
avg(avg_acceptance_rate+avg_response_rate+avg_review_score_rating) as
avg_crucial_matrics from cte
where host_is_superhost=1
group by host_is_superhost;




 --c. Analyze how the comments of reviewers vary for listings of Super Hosts vs
Other Hosts(Extract words
 --from the comments provided by the reviewers)



SELECT h.host_is_superhost, count(r.comments) as number_of_postive_comments
FROM host_dallas_df h
JOIN listing_dallas_df l ON h.host_id = l.host_id
JOIN review_dallas_df r ON l.id = r.listing_id
where r.comments like '%like%' or r.comments like '%good%' or r.comments like
'%love%'
group by h.host_is_superhost
```

```
 SELECT h.host_is_superhost, count(r.comments) as number_of__comments
FROM host_dallas_df h
JOIN listing_dallas_df l ON h.host_id = l.host_id
JOIN review_dallas_df r ON l.id = r.listing_id
where r.comments like '%bad%' or r.comments like '%not good%' or r.comments like
'%improve%'
group by   h.host_is_superhost
```

```
--d. Analyze do Super Hosts tend to have large property types as compared to Other
Hosts

 SELECT h.host_is_superhost, count(property_type) as cnt
FROM host_dallas_df h
JOIN listing_dallas_df l ON h.host_id = l.host_id
JOIN review_dallas_df r ON l.id = r.listing_id
where  property_type like '%entire%'
group by h.host_is_superhost
```