DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Databases Project – Spring 2019

Name: Yong Hao, Zeng Yanxi, Zhang Yuehan Team No: 49

Contents

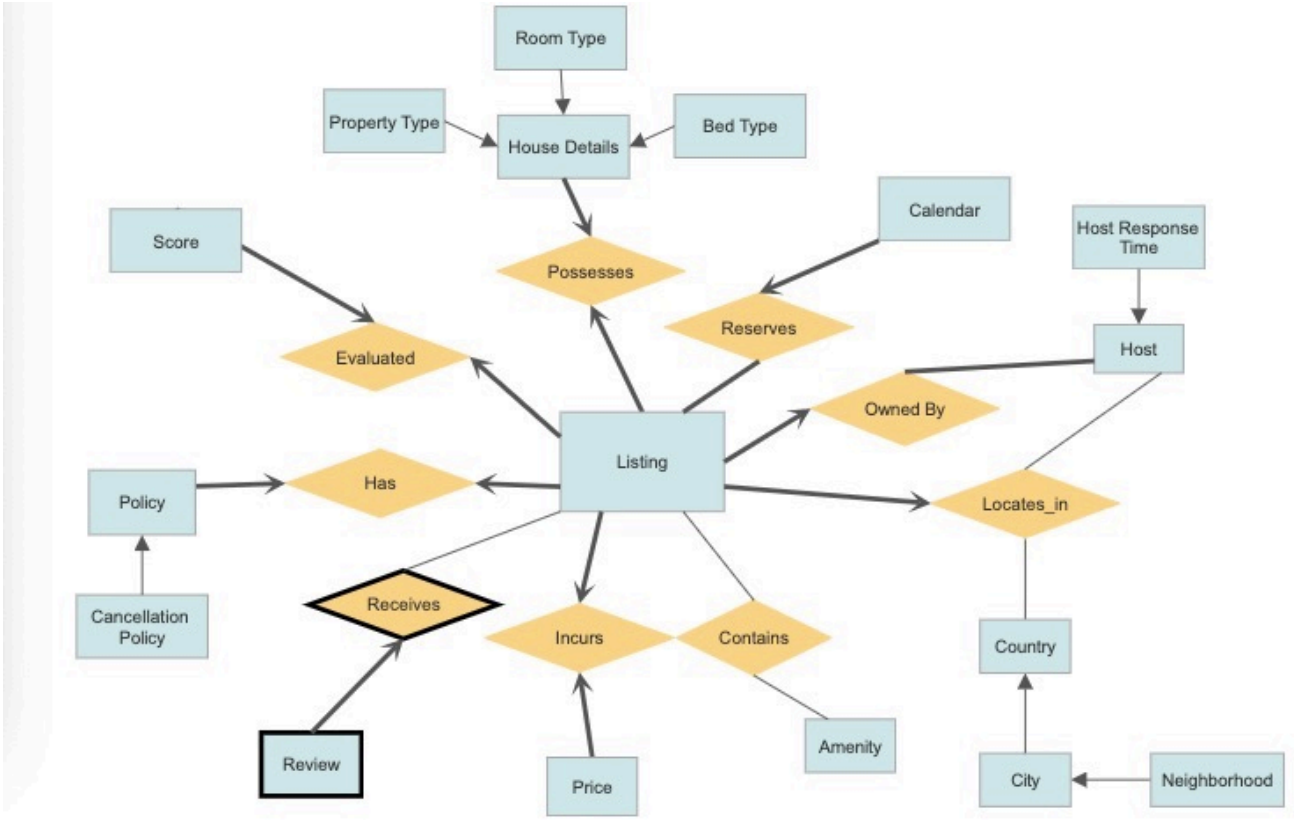
DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Deliverable 1

Entity Relationship Schema

Schema



DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Description

<Describe all the choices you made for Entities and Relationships>

Justification of the design choices & Description of the data constraints

Relationships:

- Possesses: an association among two entities *Listing* and *House Details.* It represents that house(House refers to listing item for all following contents) possesses its house details. *Listing* and *House Details* both have a key constraint and total participation, i.e. exactly one relationship. Every house (listing) possesses exactly one set of house details, and every set of house details is possessed by exactly one house.

- Owned by: an association among entities *Listing* and *Host*. This relationship represents that host owns house(s). *Listing* has a key constraint and total participation, i.e. exactly one relationship, while *Host* has total participation, which makes *Host* and *Listing* a one-to-many relationship. This means every house must have and only can have one host, and every host must have at least one house to make them a host in this system.

- Locates_in: an association among entities Listing and Country (City and neighbourhood as child entity in a hierarchy), Host and Country (City and neighbourhood as child entity in a hierarchy). It represents that house (in this part house refers to both house in listing as well as the host's own) locates at certain country, city, neighbourhood. Listing and Host has a key constraint as well as total participation, i.e. exactly one relationship. Every house locates at exactly one country, city, neighbourhood, which means house must and can only reside in one position, which is obviously true. Country has no constraints such that a country/city/neighbourhood can be not located at by any house, or be located at by one to many houses. These make Country and Listing/Host a one-to-many relationship.

- Incurs: an association among entities *Listing* and *Price*. This represents a relationship that listing(house) incurs price in this Airbnb system. *Listing* and *Price* both have a key constraint and total participation, i.e. exactly one relationship. Every house must incur and can only have one price, and a certain price must and can only be incurred by one house.

- Reserves: an association among entities *Listing* and *Calendar*. This represents the relationship that house is reserved on certain date shown on calendar. *Listing* and *Calendar* both have a total participation, while Calendar also has a key constraint, which

makes *Listing* and *Calendar* a one-to-many relationship, and each instance in calendar corresponds to exactly one instance in listing. Every house must have information about its availability and price on at least one date on the calendar. Every instance in Calendar must indicate the availability and price for one corresponding Listing instance for the purpose of reservation.

- Receives: an association among entities *Listing* and *Review*. It represents that house receives review from tenants. *Review* has a key constraint as well as total participation, i.e. exactly one relationship, while *Listing* has no constraint. This makes *House* and *Review* a one-to-many relationship. Every house can attain no review, or can attains one to many reviews. Since review is a weak entity of house, if there exists a review, there must have a house and only one house for it to review.

- Has: an association among entities *Listing* and *Policy*. This represents a relationship that house has policy for tenants to obey. *Listing* and *Policy* both have a key constraint and total participation, i.e. exactly one relationship. Every house has exactly one policy for its tenant, and every policy must and only can be owned by one house.

- Evaluated: an association among entities *Listing* and *Score*. It represents that house be evaluated by tenants and receives a score. *Listing* and *Score* both have a key constraint and total participation, i.e. exactly one relationship. Every house is evaluated exactly once to achieve a score. Every score must and can only be given to one house after tenants' evaluation.

- Contains: an association among entities *Listing* and *Amenity*. This relationship represents that house contains amenities inside. There is no constraint for both entities, which makes *Listing* and *Amenity* a many-to-many relationship. Every house can contain no amenity at all, also can have one or many amenities inside. Every kind of amenity can be not contained by any house or can be contained by one to many houses.

Relational Schema

ER schema to Relational schema

Please refer to the description under the corresponding DDL code.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

DDL

```
CREATE TABLE Listing
        (
        listing_id INTEGER(10),
         listing_url VARCHAR(255) NOT NULL,
         name   VARCHAR(255),
         summary VARCHAR(255),
         space VARCHAR(255),
         description VARCHAR(255),
         neighborhood_overview VARCHAR(255),
         notes VARCHAR(255),
         transit VARCHAR(255),
         access VARCHAR(255),
         interaction VARCHAR(255),
         house_rules VARCHAR(255),
         pricture_url VARCHAR(255),
        minimum_nights INTEGER(10),
        maximum_nights INTEGER(10),
        host_id INTEGER(10) NOT NULL,
         neighborhood VARCHAR(255) ,
         city_id INTEGER(10) NOT NULL,
         country_id INTEGER(10) NOT NULL,
         latitude DOUBLE(20),
         longtitude DOUBLE(20),
         PRIMARY KEY(listing_id),
         FOREIGN KEY(host_id) REFERENCES Host(host_id),
        FOREIGN KEY(city_id) REFERENCES Venue(city_id),
        FOREIGN KEY(country_id) REFERENCES Venue(country_id),
        UNIQUE(listing_url)
        )
```

The Listing Entity is translated into a table with one primary key, listing_id, and three foreign keys, host_id, city_id and country_id. With the three foreign keys, we combined the relationship Owned_by, Located_in, Located_at with Listing since each listing will have one unique host, one city and one country. As a result, these columns have NOT NULL constraint on their entry values.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Also, the listing_url is set to be unique for all instances to ensure proper display of the listing items on the website.

```
CREATE TABLE Host
        (
        host_id INTEGER(10),
         host_url VARCHAR(255) NOT NULL,
         host_name VARCHAR(255) NOT NULL,
         host_since DATE,
         host_about VARCHAR(255),
         host_response_rate FLOAT(10),
         host_response_time VARCHAR(255),
         host_thumbnail_url VARCHAR(255),
         host_neighborhood VARCHAR(255),
         host_verifications VARCHAR(255),
         PRIMARY KEY(host_id),
         UNIQUE(host_url),
        )


CREATE TABLE City
        (
        city_id INTEGER(10),
        city_name VARCHAR(255),
         PRIMARY KEY(city_id)
        )


CREATE TABLE Country
        (
        country_id INTEGER(10),
        country_name VARCHAR(255),
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/


```
 PRIMARY KEY(country_id)
)

CREATE TABLE Review
(
review_id INTERGER(10),
listing_id INTEGER(10) NOT NULL,
date DATE NOT NULL,
reviewer_id INTERGER(10),
reviewer_name VARCHAR(255),
comments VARCHAR(255),
PRIMARY KEY(review_id, listing_id),
FOREIGN KEY (listing_id) REFERENCES Listing(listing_id)
ON DELETE CASCADE
)
```

The relationship Receives is combined with the Reviews table as each Reviews corresponds to a unique listing item. As Reviews is designed to be a weak entity of Listing, when one instance of Listing is deleted, the corresponding Reviews instances will be deleted as well.

```
CREATE TABLE Score
(
score_id INTERGER(10),
listing_id INTEGER(10) NOT NULL,
review_scores_accuracy INTEGER(10),
review_scores_clean INTEGER(10),
reciew_scores_checkin INTERGER(10),
review_scores_communication INTERGER(10),
review_scores_location INTERGER(10),
review_scores_value INTERGER(10),
PRIMARY KEY(score_id),
FOREIGN KEY (listing_id) REFERENCES Listing(listing_id)
)
```

The relationship Evaluated is combined with the Score table as each Score corresponds to a unique listing item.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

```
CREATE TABLE Policy
        (
        policy_id INTEGER(10),
        is_business_travel_ready BIT,
        cancellation_policy VARCHAR(255),
        require_guest_profile_picture BIT,
        require_guest_phone_verification BIT,
        listing_id INTEGER(10) NOT NULL,

        PRIMARY KEY(policy_id),
        FOREIGN KEY (listing_id) REFERENCES Listing(listing_id)
        )
```

The relationship Has is combined with the Policy table as each Policy instance corresponds to a unique Listing instance.

```
CREATE TABLE Price
        (
        price_id INTEGER(10),
        price FLOAT(10),
        weekly_price FLOAT(10),
        monthly_price FLOAT(10),
        security_deposit FLOAT(10),
        cleaning_fee FLOAT(10),
        guests_included INTEGER(10),
        extra_people INTERGER(10),
        listing_id INTEGER(10) NOT NULL,
        PRIMARY KEY(price_id),
        FOREIGN KEY (listing_id) REFERENCES Listing(listing_id)
        )
```

The relationship Incurs is combined with the Price table as each Price instance corresponds to a unique Listing instance.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

```
CREATE TABLE House_Details
        (
        detail_id INTEGER(10),
        property_type VARCHAR(255),
        room-type VARCHAR(255),
        accommodates VARCHAR(255),
        bathrooms INTEGER(10),
        bedrooms INTEGER(10),
        beds INTERGER(10),
        bed_type VARCHAR(255),
        amenities VARCHAR(255),
        square_feet INTEGER(10),
        listing_id INTEGER(10) NOT NULL,
        PRIMARY KEY(detail_id),
        FOREIGN KEY (listing_id) REFERENCES Listing(listing_id)
        )
```

The relationship Pocesses is combined with the House_Details table as each House_Details instance corresponds to a unique Listing instance.

```
CREATE TABLE Amenities
        (
        amenity_id INTERGER(10),
        amenity_name varchar(255),
        PRIMARY KEY(amenity_id)
        )
```

```
CREATE TABLE Contains
        (
        listing_id INTEGER(10),
        amenitty_id INTEGER(10),
        PRIMARY KEY(amenity_id),
        FOREIGN KEY(listing_id)  REFERENCES Listing(listing_id)
        )
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

The table Contains is created to store the many-to-many relationship between Amenities and Listing.

CREATE TABLE Calender

```
{
listing_id INTEGER(10),
date DATE,
available BIT,
price FLOAT(10),
PRIMARY KEY(listing_id, date),
FOREIGN KEY(listing_id)  REFERENCES Listing(listing_id)
}
```

The relationship Reserves is combined with the Calendar table as each Calendar instance corresponds to exactly one Listing instance.

General Comments

<In this section write general comments about your deliverable (comments and work allocation between team members>

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

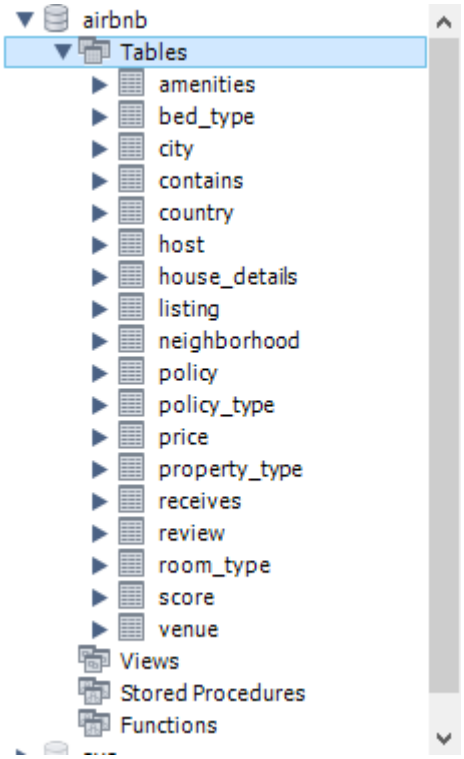
Deliverable 2

Assumptions

<In this section write down the assumptions you made about the data. Write a sentence for each assumption you made>

Data Loading

We have hosted a local database server and created a database named “airbnb”. We’ve created the tables corresponding to the ER models we’ve designed. The tables schemas are coped with the design as well.



We’ve pre-processed the csv data files and organized them with respect to the tables to be created.

To load the data, we adopted the following template of SQL code:

LOAD DATA INFILE

'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/data_file.csv' (This version requires the loaded file to be only in the secure position designated by the software)

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

INTO TABLE [*table_name*]

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

([fields with respect to the columns of the csv files])

(IGNORE LINE 1) (if the first row is the schema)

The data was loaded by executing the commands for all the csv data files.

The screenshot of a portion of the loaded data is as following:

listing_id	listing_url	listing_name	summary
0	listing_url	name	summary
2015	https://www.airbnb.com/rooms/2015	Berlin-Mitte Value! Quiet courtyard/very central	Great location! 30 of 75 sq meters. This wood floored/high ceiling typical Berlin "Altbau" section of an apartment consists of 1 simple large room, a small kitchen and a bathroc
3176	https://www.airbnb.com/rooms/3176	Fabulous Flat in great Location	This beautiful first floor apartment is situated at Kollwitzplatz. Its ideal for 2 but can comfortably accommodate 4 with the extra double bed in the living room. PLEASE INQUI
3309	https://www.airbnb.com/rooms/3309	BerlinSpot Schöneberg near KaDeWe	First of all: I prefer short-notice bookings. Requests 1-2 weeks before arrival are perfect. Longer plans in advance are sometimes a bit difficult for me. But please just try! M
7071	https://www.airbnb.com/rooms/7071	BrightRoom with sunny greenview!	Cozy and large room in the beautiful district Prenzlauer Berg.
9991	https://www.airbnb.com/rooms/9991	Geourgeous flat - outstanding views	4 bedroom with very large windows and outstanding views of Mitte is great for families or groups. Walking distance from Kollwitz (relaxed - farmers market TH/SA) & Alexande
14325	https://www.airbnb.com/rooms/14325	Apartment in Prenzlauer Berg	The apartment is located on the upper second floor in a typical Berlin-style building from the turn of the 19th century in Prenzlauer Berg. It is tastefully furnished with a mix of
16401	https://www.airbnb.com/rooms/16401	APARTMENT TO RENT	
16644	https://www.airbnb.com/rooms/16644	In the Heart of Berlin - Kreuzberg	Light and sunny 2-Room-turn of the century-flat in a typical Berliner front house in the best part of Kreuzberg.
17409	https://www.airbnb.com/rooms/17409	Downtown Above The Roofs In Berlin	The 25 sqm room is located in the heart of Berlin. It is part of our private apartment, but you have your own entrance, your own bathroom and your privacy. Towels and she
18666	https://www.airbnb.com/rooms/18666	Flat with Sunny Terrace	Apartment located near the "Plaza de las Glorias" and the second-hand market (Encants). The accommodation is also close to the National Theatre of Catalunya and the Agba
18674	https://www.airbnb.com/rooms/18674	Huge flat for 8 people close to Sagrada Familia	110m2 apartment to rent in Barcelona. Located in the Eixample district, near the Sagrada Familia. It has a small balcony where you can see the temple of Gaudi. Capacity for
19864	https://www.airbnb.com/rooms/19864	PLAZA MAYOR (wifi, air conditioning)	Cosy small studio with a nice balcony close to Plaza Mayor and the typical Madrilean tapas bars area.
20858	https://www.airbnb.com/rooms/20858	Designer Loft in Berlin Mitte	Bright and sunny condo with two balconies in a stunning location in Berlin Mitte/Prenzelberg!
21605	https://www.airbnb.com/rooms/21605	Nice and sunny duble room	The flat is in Poblenou district, and the room is a double room with a double bed, a wardrobe, a table, TV, wifi, heating and wood floor. Beautiful and charmy.
21853	https://www.airbnb.com/rooms/21853	Bright and airy room	We have a quiet and sunny room with a good view in our 3 bedroom flat. In a welcoming atmosphere, our clean and respectful flatshare is located in a leafy neighbourhood w
21869	https://www.airbnb.com/rooms/21869	Studio in the Heart of Kreuzberg	Light and sunny 1-Room-turn of the century-flat in a typical Berliner front house in the best part of Kreuzberg.
22415	https://www.airbnb.com/rooms/22415	Stylishly furnished 3 room flat	Artsy, light, spacious and cozy appartment in the heart of bohemian kreuzberg.
22677	https://www.airbnb.com/rooms/22677	Prenzel garden with leafy terrace	Comfortable, cozy and quiet 70m2 apartment plus garden and terrace, perfect to relax after your trips around the city. Centrally located in trendy Kastanienallee, it is just r
23834	https://www.airbnb.com/rooms/23834	Apartment in the heart of Berlin	Beautiful and calm apartment in close vicinity of the Viktoria Park, Bergmannstrasse and the western centre of Berlin.
24569	https://www.airbnb.com/rooms/24569	Sunny & Wheelchair accessible	
24805	https://www.airbnb.com/rooms/24805	Gran Via Studio Madrid	Studio located 50 meters from Gran Via, next to the Plaza de Callao. In the best shopping and cinemas and theaters. Comfortable apartment for 3 people. 1 bed and the sof
24836	https://www.airbnb.com/rooms/24836	Select the Madrid more "cool".	Apartamento céntrico junto a Gran Vía (VT-296) con 2 dormitorios. Exterior y en la segunda planta. Situación ideal para turismo por la ciudad, visitas a museos y palacios o zo
25786	https://www.airbnb.com/rooms/25786	NICE ROOM AVAILABLE IN THE HEART OF GR...	JUST GO THROUGH THE MANY REVIEWS I GOT THROUGH THE YEARS, NO BETTER FEEDBACK THAN THAT. WELCOME.
26543	https://www.airbnb.com/rooms/26543	Helmholtzplatz Bright&Spacious Apt.	This flat has permission to rent under Licence Nr: 03/z/za/ (Phone number hidden by Airbnb) Genehmigungsbescheid vom (Phone number hidden by Airbnb) , It has 2 bathroc
26823	https://www.airbnb.com/rooms/26823	Room for two people	

Query Implementation

Query 1: What is the average price for a listing with 8 bedrooms?

Description of logic:

Select houses that have 8 bedrooms and then obtain the average price of the houses.

SQL statement

```
SELECT AVG(P.price)
FROM House_Details H, Price P
WHERE H.listing_id = P.listing_id AND H.bedrooms = 8
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Query 2: What is the average cleaning review score for listings with TV?

Description of logic:

Get the amenity id of "TV" and select houses with this amenity id. Then return the average cleaning review score for the houses.

SQL statement

```
SELECT AVG(S.review_scores_clean )
FROM Score S, Amenities A, Contains C
WHERE A.amenity_name = "TV" AND A.amenity_id = C.amenity_id AND C.listing_id = S.listing_id
```

Query 3: Print all the hosts who have an available property between date 03.2019 and 09.2019.

Description of logic:

Find the houses which are available any date between 2019-03-01 to 2019-09-30, and then find the host of the house. Use UNIQUE command to remove duplicate.

SQL statement

```
SELECT UNIQUE H.host_id, H.host_name
FROM Host H, Calendar C, Listing L
WHERE  H.host_id = L.host_id AND L.listing_id = C.listing_id AND C.date <=2019-09-30 AND C.date>=2019-03-01 AND C.available = 1
```

Query 4: Print how many listing items exist that are posted by two different hosts but the hosts have the same name.

Description of logic:

Apply self-join to Host table, get the pairs of listing id for hosts who have the same name but different id, that is two different people. Then count the number of listing ids which are found in the union of two pairs set.

SQL statement

```
SELECT COUNT (DISTINCT L.listing_id)
FROM Listing L
WHERE L.listing_id IN (
                SELECT L1.listing_id
                FROM Host H1, Host H2, Listing L1
                WHERE H1.host_name = H2.host_name AND
                H1.host_id <> H2.host_id AND
                L1.host_id = H1.host_id
                UNION
                SELECT L2.listing_id
                FROM Host H1, Host H2, Listing L2
                WHERE H1.host_name = H2.host_name AND
                H1.host_id <> H2.host_id AND
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

L2.host_id = H2.host_id)

Query 5: Print all the dates that 'Viajes Eco' has available accommodations for rent.

Description of logic:

Find the houses under Viajes Eco and then find the UNIQUE dates that any of his houses is available.

SQL statement

```
SELECT UNIQUE C.date
FROM Calendar C, Host H, Listing L
WHERE H.host_name = 'Viajes Eco' AND H.host_id = L.host_id AND L.listing_id = C.listing_id AND C.available = 1
```

Query 6: Find all the hosts (host_ids, host_names) that have only one listing.

Description of logic:

Join host and listing tables together with the same host id, and group by host id. Find that for each group of same host id, the number of members, which is the listing owned by the host. The host has only one listing if the number is one.

SQL statement

```
SELECT H.host_name, H.host_id
FROM Host H, Listing L
WHERE H.host_id = L.host_id
GROUP BY L.host_id
HAVING COUNT (*) = 1
```

Query 7: What is the difference in the average price of listings with and without Wifi.

Description of logic:

First find the average price of houses with wifi, the the average price of house without wifi. Return the difference of the two values.

SQL statement

```
(
SELECT AVG(P.price)
FROM Price P, Contain C, Amenities A
WHERE A.amenity_name = "Wifi" AND A.amenity_id = C.amenity_id AND C.listing_id = P.listing_id
)
- (
SELECT AVG(P.price)
FROM Price P
WHERE NOT EXIST (       SELECT C.listing_id
                        FROM Contains C, Amenities A
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

```
                    WHERE A.amenity_name = "Wifi" AND A.amenity_id = C.amenity_id AND C.listing_id =
            P.listing_id)
)
```

Query 8: How much more (or less) costly to rent a room with 8 beds in Berlin compared to Madrid on average?

Description of logic:

First find the average price of house with 8 beds for each city by joining house detail table, listing table, price table and city table. Then use the average price of Berlin to minus that of Madrid.

SQL statement

```
(
SELECT AVG (P1.price)
FROM Price P1, HouseDetail H_d, Listing L, City C
WHERE C.city_name ='Berlin'AND C.city_id = L.city_id AND H_d.beds = 8
AND H_d.listing_id = L.listing_id AND H_d.listing_id = P1.listing_id
    )
-(
SELECT AVG (P2.price)
FROM Price P2, HouseDetail H_d, Listing L, City C
WHERE C.city_name ='Madrid'AND C.city_id = L.city_id AND H_d.beds = 8
AND H_d.listing_id = L.listing_id AND H_d.listing_id = P2.listing_id
    )
```

Query 9: Find the top-10 (in terms of the number of listings) hosts (host_ids, host_names) in Spain.

Description of logic:

Group the houses by host_id, and count the number of houses each host that locates in Spain. Rank them according to count in descend order and select the top ten results. Return the corresponding host id and host name.

SQL statement

```
SELECT H.host_id, H.host_name
FROM Host H.
WHERE EXISTS (
        SELECT TOP 10 H.host_id, COUNT(
                SELECT L.listing_id
                FROM Listing L, Country C
                WHERE C.country_name = "Spain" AND C.country_id = L.country_id AND  L.host_id =
        H.host_id) as Count
        GROUP BY H.host_id
        ORDER BY Count DESC)
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Query 10: Find the top-10 rated (review_score_rating) apartments (id,name) in Barcelona.

Description of logic:

Group the houses by review score rating, and find the top 10 apartments which are lecated in Barcelona, through the joining of listing, review and city table. Rank them according to count in descend order and select the top ten results. Return the corresponding listing id and listing name.

SQL statement

```
SELECT L.listing_id, L.listing_name
FROM Listing L
WHERE EXISTS (
                SELECT TOP 10 L.listing_id, L.listing_name
                FROM Listing L, Review R, City C
                        WHERE C.city_name =‘Barcelona’ AND
 C.listing_id = L.listing_id AND L.listing_id = R.listing_id
 GROUP BY R.review_score_rating
 ORDER BY R.review_score_rating DESC
)
```

Interface

Design logic Description

<Describe the general logic of your design as well as the technology you decided to use>

Screenshots

<Provide some initial screen shots of your interface>

General Comments

<here we briefly explain the changes/improvements we have done from deliverable 1 to deliverable 2>

1. Categorical variables are extracted out from the original entities they belongs to and changed into entities themselves instead of attributes, e.g. Neighborhood, Property_Type, Room_Type, Bed_Type, Cancellation_policy. As a result, the implementation performance is further optimized since the manipulation of long string is reduced while manipulate integers instead.
2. Relationship between Country, City, Neighborhood together with listing is improved by adding hierarchy(has a relationship) between Country&City, City&Neighborhood. And the new tables created accordingly.
3. ER Model as well as the DDL in deliverable 1 will be further optimized accordingly.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Deliverable 3

Assumptions

<In this section write down the assumptions you made about the data. Write a sentence for each assumption you made>

Query Implementation

<For each query>

Query a:

Description of logic:

<What does the query do and how do I decide to solve it>

SQL statement

<The SQL statement>

Query Analysis

Selected Queries (and why)

Query 1

<Initial Running time:
Optimized Running time:
Explain the improvement:
Initial plan
Improved plan>

Query 2

<Initial Running time:
Optimized Running time:
Explain the improvement:
Initial plan
Improved plan>

Query 3

<Initial Running time:
Optimized Running time:
Explain the improvement:
Initial plan
Improved plan>

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Interface

Design logic Description

<Describe the general logic of your design as well as the technology you decided to use>

Screenshots

<Provide some initial screen shots of your interface>

General Comments

<In this section write general comments about your deliverable (comments and work allocation between team members>