



SQL-Mongo Project – Seattle Airbnb

BUAN 6320.SW1

Group Members

Group 3:
Arul Chakravarthy
Raghava Sridar

Activity	Arul Chakravarthy	Raghav Sridhar
Prepared Data Model	x	
Data Cleaning and Loaded Data into Database	x	x
Wrote SQL Queries		
Prepared Mongo Database		
Loaded data into Mongo DB		
Wrote Mongo Queries		
Prepared Report		
Reviewed Report		

Contents

Relational Data Model	4
Assumptions/Notes About Data Entities and Relationships	Error! Bookmark not defined.
Entity-Relationship Diagram	10
Physical MySQL Database	12
Assumptions/Notes About Data Set	12
Screen shot of Physical Database objects.....	12
Data in the Database.....	16
SQL Queries.....	Error! Bookmark not defined.
SQL Query 1	Error! Bookmark not defined.
Question.....	Error! Bookmark not defined.
Notes/Comments About SQL Query and Results (Include # of Rows in Result)...	Error! Bookmark not defined.
Translation	Error! Bookmark not defined.
Screen Shot of SQL Query and Results.....	Error! Bookmark not defined.
SQL Query 2	Error! Bookmark not defined.
Question.....	Error! Bookmark not defined.
Notes/Comments About SQL Query and Results (Include # of Rows in Result)...	Error! Bookmark not defined.
Translation	Error! Bookmark not defined.
Screen Shot of SQL Query and Results.....	Error! Bookmark not defined.
SQL Query 3	Error! Bookmark not defined.
Question.....	Error! Bookmark not defined.
Notes/Comments About SQL Query and Results (Include # of Rows in Result)...	Error! Bookmark not defined.
Translation	Error! Bookmark not defined.
Screen Shot of SQL Query and Results.....	Error! Bookmark not defined.
SQL Query 4	Error! Bookmark not defined.
Question.....	Error! Bookmark not defined.
Notes/Comments About SQL Query and Results (Include # of Rows in Result)...	Error! Bookmark not defined.
Translation	Error! Bookmark not defined.
Screen Shot of SQL Query and Results.....	Error! Bookmark not defined.
SQL Query 5	Error! Bookmark not defined.

Question.....	Error! Bookmark not defined.
Notes/Comments About SQL Query and Results (Include # of Rows in Result)...	Error! Bookmark not defined.
Translation	Error! Bookmark not defined.
Screen Shot of SQL Query and Results.....	Error! Bookmark not defined.
SQL Query 6	Error! Bookmark not defined.
Question.....	Error! Bookmark not defined.
Notes/Comments About SQL Query and Results (Include # of Rows in Result)...	Error! Bookmark not defined.
Translation	Error! Bookmark not defined.
Screen Shot of SQL Query and Results.....	Error! Bookmark not defined.
Data Review for MongoDB.....	16
Assumptions/Notes About Data Collections, Attributes and Relationships between Collections	16
Physical Mongo Database	24
Assumptions/Notes About Data Set	24
Screen shot of Physical Database objects (Database, Collections and Attributes)	24
Data in the Database.....	24
MongoDB Queries/Code	25
Mongo Query 1	25
Question.....	25
Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) ...	25
Translation	25
Screen Shot of MongoDB Query/Code and Results.....	25
Mongo Query 2	26
Question.....	26
Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) ...	26
Translation	26
Screen Shot of MongoDB Query/Code and Results.....	26
Mongo Query 3	27
Question.....	27
Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) ...	27
Translation	27
Screen Shot of MongoDB Query/Code and Results.....	27

Scenario

Design a database using good practices for database design and normalization. Use the data provided here. The business scenario is as follows.

Airbnb is an American vacation rental online marketplace offers arrangement for lodging, primarily homestays, or tourism experiences. Since 2008, guests and hosts have used Airbnb to travel in a more unique, personalized way. As part of the Airbnb Inside initiative, dataset (from kaggle.com) describes the listing activity of homestays in Seattle, WA.

Combining city-level data on Airbnb availability, property configurations, host details, amenities can help us identify which factors play a crucial role in engaging more customers and servicing the Airbnb customers in a better way.

Relational Data Model

Assumptions

The columns in listing file is collected from various other sources and because of uncertainty of source columns in listings table, all the columns except for host subset are considered independent to achieve 3NF.

Reasons why the data model is in 3NF?

- All the tables of the data model are in second normal form and there is no transitive dependency.
- Considering the assumption-1 mentioned earlier, every non-key attribute of each table of the data model is functionally dependent only on the complete primary key and every attribute contains only a single atomic value.
- Considering the assumption-1, in addition, every non-key attribute is independent of any other non-key attribute.

Data requirements for Seattle Airbnb Data:

1. Listings

The 'listing_id' which uniquely identifies each listing and its listing equivalents in Seattle, United States of Airbnb is the primary key. Listings contains the all the high level properties of the Seattle Airbnb listings The foreign keys are host_id, property_type, room_type and bed_type.

listings
listing_id INT
name VARCHAR(60)
summary VARCHAR(300)
scrape_id INT
last_scraped DATE
space VARCHAR(1005)
description VARCHAR(1005)
experiences_offered VARCHAR(45)
notes VARCHAR(1005)
transit VARCHAR(1005)
host_id INT
requires_license TINYINT
license VARCHAR(25)
jurisdiction_names VARCHAR(25)
instant_bookable TINYINT
cancellation_policy VARCHAR(45)
calculated_host_listings_count INT
accommodates INT
bathrooms DECIMAL
bedrooms INT
beds INT
square_feet INT
street VARCHAR(45)
neighbourhood VARCHAR(45)
neighbourhood_cleansed VARCH...
50 more...
Indexes
PRIMARY
fk_listings_host1_idx
fk_listings_bed_type1_idx
fk_listings_room_type1_idx
fk_listings_property_type1_idx

2. Host

Airbnb hosts can list entire homes/apartments, private or shared rooms. In order to keep a note of the host details. Some Airbnb hosts have multiple listings. A host may list separate rooms in the same apartment, or multiple apartments or homes available in their entity. The primary key is 'host_id' and no foreign key as such.

host	
host_id	INT
host_url	VARCHAR(100)
host_name	VARCHAR(45)
host_since	DATE
host_city	VARCHAR(45)
host_state	VARCHAR(45)
host_country	VARCHAR(45)
host_about	VARCHAR(1005)
host_response_time	VARCHAR(45)
host_response_rate	INT
host_acceptance_rate	INT
host_is_superhost	TINYINT
host_thumbnail_url	VARCHAR(100)
host_picture_url	VARCHAR(100)
host_neighbourhood	VARCHAR(45)
host_listings_count	INT
host_total_listings_count	INT
host_verifications	INT
host_has_profile_pic	TINYINT
host_identity_verified	TINYINT
Indexes	
PRIMARY	

3. Verifications

Verifications table act as the linking table between hosts table and host_verifications table. It consists of the foreign keys ie host_id from the hosts table and host_verifications_id from host_verifications table

verifications
host_host_id INT
host_verifications_host_verifications_id INT
Indexes
PRIMARY
fk_verifications_host1_idx
fk_verifications_host_verifications1_idx

4. host_verifications

All the different mediums used for host verifications like email, linkedin etc are maintained in this table and the primary key is host_verifications_id and there is no foreign key

host_verifications
host_verifications_id INT
host_verifications_name VARCHAR(45)
Indexes
PRIMARY

5. reviews

Airbnb guests may leave a review after their stay, and these can be used as an indicator of airbnb activity. The minimum stay, price and number of reviews have been used to estimate the occupancy rate, the number of nights per year and the income per month for each listing. Listing_id along with date acts as the composite primary key and listing_id, reviewer_id is the foreign key

reviews
listings_listing_id INT
date DATE
reviewers_reviewer_id INT
comments VARCHAR(1005)
2 more...
Indexes
PRIMARY
fk_reviews_listings1_idx
fk_reviews_reviewers1_idx

6. reviewers

Reviewers as part of the reviews table and their details are maintained in a separate table wherein reviewer_id acts as the primary_key and there is no foreign key

reviewers
reviewer_id INT
reviewer_name VARCHAR(45)
Indexes
PRIMARY

7. availability

Each listings availability in a datewise fashion is maintained inside the availability table and the listings id along with date acts as composite primary key and listing_id is the foreign key

availability
listings_listing_id INT
date DATE
available TINYINT
price (USD) DECIMAL(10,2)
Indexes
PRIMARY
fk_availability_listings1_idx

8. property_amenities

property_amenities table act as the linking table between listings table and property_amenities table. It consists of the foreign keys ie listings_id from the listings table and amenities_id from property_amenities table

property_amenities
listings_listing_id INT
amenities_amenities_id INT
Indexes
PRIMARY
fk_amenities_listings_amenitie...
fk_property_amenities_listings...

9. amenities

Each listing in Airbnb can host a multiple array of amenities and the unique set of amenities are maintained inside this table where amenities_id acts as the primary key and there are no foreign keys

amenities	
amenities_id	INT
amenities_name	VARCHAR(45)
Indexes	
PRIMARY	

10. bed_type

This table consists of the all the unique bed types made available by Airbnb listings. bed_type_id acts as the primary key and there is no foreign key

bed_type	
bed_type_id	INT
bed_type_name	VARCHAR(45)
Indexes	
PRIMARY	

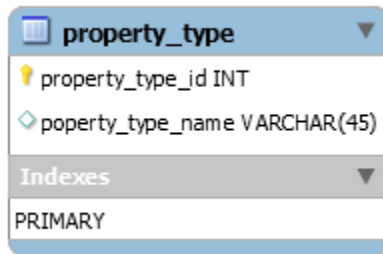
11. room_type

This table consists of the all the unique bed types made available by Airbnb listings. room_type_id acts as the primary key and there is no foreign key

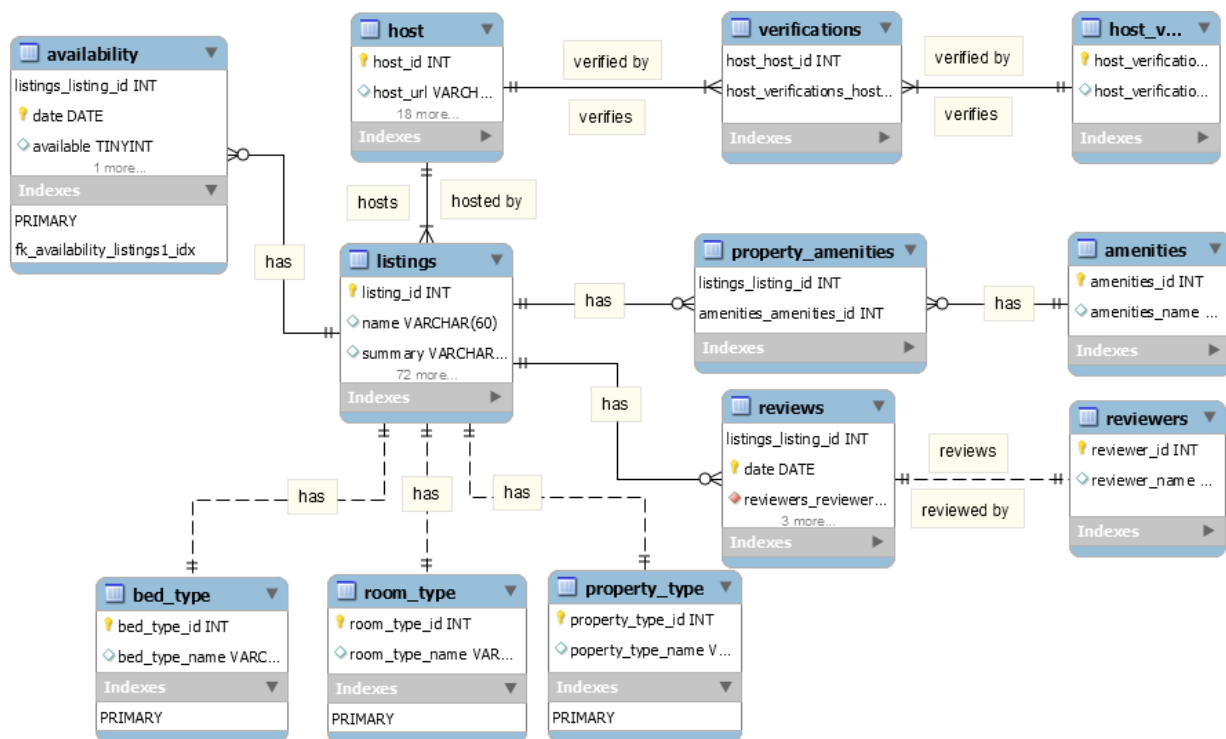
room_type	
room_type_id	INT
room_type_name	VARCHAR(45)
Indexes	
PRIMARY	

12. property_type

This table consists of all the unique bed types made available by Airbnb listings. `property_type_id` acts as the primary key and there is no foreign key.



Entity-Relationship Diagram



Data Entities and Relationships

1. The listings and availability are in 0:m relationship. Each listing corresponds to unique property listing in Seattle, hence listing_id acts as the primary key in listings table. 1 listing can have 0 to many records in availability table in a datewise fashion. Forming a composite primary key (CPK) with "listing_id" and "date" columns in availability table help achieve this relation.
2. The listings and reviews are in 0:m relationship. Each listing (unique listing id) can have 0 to many reviews. Forming a composite primary key (CPK) with "listing_id" and "date" columns in reviews table help to achieve this relation.
3. The listings and reviewers are in 1:1 relationship. Each review (unique review id) would have been reviewed by 1 reviewer. Forming a primary and foreign key (PK and FK) with "reviewer_id" column helps to achieve this relation.
4. The host and listings are in 1:m relationship. Each host (unique host id) can have many listings in airbnb. Forming a primary and foreign key (PK and FK) with "host_id" column helps to achieve this relation.
5. The listings and amenities are in **m:m** relationship. Each listing (unique listing id) can have several amenities and vice versa. In order to resolve this property_amenities **linking table** is used which has foreign keys namely listing_id and amenities_id from each of the above tables to achieve this relation.
6. The host and host_verifications are in **m:m** relationship. Each host (unique host_id) can have several verifications mediums and vice versa. In order to resolve this verifications **linking table** is used which has foreign keys namely host_id and verifications_id from each of the above tables to achieve this relation.
7. Listings has 1:1 relationship with bed_type, room_type and property_type tables each. Each listing (unique listing id) has an associated bed_type, room_type and property_type whose enumeration is maintained in above mentioned type tables. Forming a non-identifying relationship using primary and foreign key (PK and FK) with "bed_type_id", "room_type_id" and "property_type_id" columns helps to achieve this relation.

Notes:

Since there are too many columns in the data model to be fitted in this document, we have made an .xlsx file consisting more details on column names of each table with their datatypes along with primary and foreign keys.



Seattle_AirBnb_Colu
mns.xlsx

Physical MySQL Database

Assumptions/Notes About Data Set

Assumptions:

The missing observations in the dataset are replaced with NULL values. In future if we need to consider all the observations to calculate an aggregation function for a specific column. We calculate using the sum (all values) and count(rows).

Notes:

There were close to 4000 listings at Seattle Airbnb, that contains data about each property, host details, pricing details, reviews of each listings. Columns like price, weekly_price, monthly_price, security_deposit, cleaning_fee, street, extra_people, host_is_superhost, host_location, first_review, last_review had data cleaning to be done. Removed excess strings in the data like '%','\$', limited the float characters. Street column had the details that are redundant like city and state which has been removed.

Screen shot of Physical Database objects

Bed_type

```
9 • select * from bed_type;
```

bed_type_id	bed_type_name
1	Real Bed
2	Futon
3	Pull-out Sofa
4	Airbed
5	Couch

Host

```
16 • select * from host ;
```

host_id	host_url	host_name	host_since	host_city	host_about	host_response_time	host_response_rate
4193	https://www.airbnb.com/users/show/4193	Jessica	2008-11-10	Seattle	Hello! I am an avid gardener, community organi...	within a few hours	88
6207	https://www.airbnb.com/users/show/6207	Cory & Amanda	2009-01-08	Seattle	Hi! We are Cory and Amanda. A married couple ...	within an hour	100
8021	https://www.airbnb.com/users/show/8021	Becky	2009-02-16	Boston	I am an apparel designer and an artist. just ov...	within a few hours	75
8993	https://www.airbnb.com/users/show/8993	Maddy	2009-03-03	Seattle	Hello my name is Maddy. I enjoy meeting and h...	within an hour	100
11775	https://www.airbnb.com/users/show/11775	Andrew	2009-03-30	Seattle	I like to play tennis	within a few hours	100
14942	https://www.airbnb.com/users/show/14942	Joyce	2009-04-26	Seattle	I am a therapist/innkeeper.I know my city well a...	within a few hours	90
19425	https://www.airbnb.com/users/show/19425	Shireen	2009-05-30	Seattle	hi, i'm a native of the east coast but have loved...	NULL	NULL
20721	https://www.airbnb.com/users/show/20721	Til	2009-06-03	Seattle	I'm originally from in Utah, Korea, and I grew up i...	within an hour	100

Listing

18 • `select * from listing ;`

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

	listing_id	scrape_id	name	summary	last_scraped	space	des
▶	3335	20160104002432	Sweet Seattle Urban Homestead 2 Bdr	Welcome! If you stay here you will be living in a...	2016-01-04	Welcome! Come enjoy your time in Seattle at a...	Wel
	4291	20160104002432	Sunrise in Seattle Master Suite	Located in Seattle, this is a spacious, clean and...	2016-01-04	Located in Seattle, this is a spacious, clean and...	Loc
	5682	20160104002432	Cozy Studio, min. to downtown -WIFI	The Cozy Studio is a perfect launchpad for your...	2016-01-04	Hello fellow travelers, Save some money and ha...	The
	6606	20160104002432	Fab, private seattle urban cottage!	Soo centrally located, this is a little house all yo...	2016-01-04	Soo centrally located, this is a little house all yo...	Soo
	7369	20160104002432	launchingpad/landingpad	contemporary condo on the western edge of pi...	2016-01-04	spacious condo with all the amenities	cont
	9419	20160104002432	Golden Sun vintage warm/sunny	This beautiful double room features a magical sl...	2016-01-04	Our new Sunny space has a private room from t...	This
	9460	20160104002432	Downtown Convention Ctr B&B - Nice!	Great location, 98% walk score, next to the Co...	2016-01-04	Greetings from Seattle! Thanks for considering ...	Grei
	9531	20160104002432	The Adorable Sweet Grass Cottage	The Sweet Grass is a delightful and cozy 2...	2016-01-04	The Sweet Grass invites you to stay and play...	The

Property_Amenities

19 • `select * from Property_Ameniti`

Result Grid | Filter Rows: |

	amenities_id	id
▶	1	241032
	2	953595
	2	2493658
	3	3308979
	4	7421966
	5	278830
	6	5956968
	7	1909058
	8	855555

Property_Type

20 • `select * from Property_type ;`

Result Grid | Filter Rows: |

	property_type_id	property_type_name
▶	1	Apartment
	2	House
	3	2 bin
	4	Condominium
	5	Camper/RV
	6	Bungalow
	7	Townhouse
	8	Loft
	9	Boat

Review

21 • `select * from Review ;`

<

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

	review_id	listing_id	reviewer_id	comments
▶	38917982	7202016	28943674	Cute and cozy place. Perfect location to everything!
	39087409	7202016	32440555	Kelly has a great room in a very central location. Beautiful building , architecture and a style that we r...
	3982001	39087409	37722850	Very spacious apartment, and in a great neighborhood. This is the kind of apartment I wish I had! Di...
	40813543	7202016	33671805	Close to Seattle Center and all it has to offer - ballet, theater, museum, Space Needle, restaurants of ...
	41986501	7202016	34959538	Kelly was a great host and very accommodating in a great neighborhood. She has some great coffee ...
	43979139	7202016	1154501	Kelly was great, place was great, just what I was looking for- clean, simple, well kept place. 5 min walk...
	45265631	7202016	37853266	Kelly was great! Very nice and the neighborhood and place to stay was expected and comfortable. Ov...
	46749120	7202016	24445447	hola all bnb erz - Just left Seattle where I had a simply fantastic time for the weekend , no small part b...
	4732246	7202016	240582	Kelly's place is conveniently located on a quiet street in Lower Queen Anne which is an easy walk or by

Reviewer

22 • `select * from reviewer;`

<

Result Grid | Filter Rows:

	reviewer_id	reviewer_name
▶	28943674	Bianca
	32440555	Frank
	37722850	Ian
	33671805	George
	34959538	Ming
	1154501	Barent
	37853266	Kevin
	24445447	Rick
	240582	Todd

Room_type

23 • `select * from Room_type ;`

<

Result Grid | Filter Rows:

	room_type_id	room_type_name
▶	1	Entire home/apt
	2	Private room
	3	Shared room

Verifications

24 • `select * from verifications;`

Result Grid | Filter Rows:

	host_verifications_id	host_id
▶	1	956 956883
	2	5177328
	3	16708587
	4	9851441
	5	1452570
	6	326758
	7	2497928
	8	4016632
	9	2166277

Count of all tables in the database

amenities
bed_type
calendar
host
host_verifications
listing
property_amenities
property_type
review
reviewer
room_type
verifications

3 • `select count(*) from amenities union all select count(*) from bed_type union all select count(*) from host`
4 `union all select count(*) from Host_verifications union all select count(*) from listing`
5 `union all select count(*) from Property_Amenities`
6 `union all select count(*) from Property_type union all select count(*) from Review union all select count(*) from reviewer`
7 `union all select count(*) from Room_type union all select count(*) from verifications;`
8

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

	count(*)
▶	3284
	5
	2751
	2751
	3818
	3818
	17
	84849
	75730
	3
	2751

Data in the database

Table Name	Primary Key	Foreign Key	# of Rows in Table
Amenities	Amenities_Id	-	3284
Bed_type	bed_type_id	-	5
Host	host_Id	-	2751
Host_verifications	host_Verifications_id	-	2751
listing	Listing_Id	Host_id, bed_type_id, room_type_id, property_type_id	3818
Property_Amenities	Listing_id	Listing_id, amenities_id	3818
Property_type	Property_type_id	-	17
Reviews	Listing_id, date (Composite Primary Key)	Listing_id, reviewer_id	84849
reviewers	Reviewer_id	-	75730
Room_type	Room_type_id	-	3
verifications	Host_id	Host_id, host_verifications_id	2751

Data Review for MongoDB

Assumptions/Notes About Data Collections, Attributes and Relationships between Collections

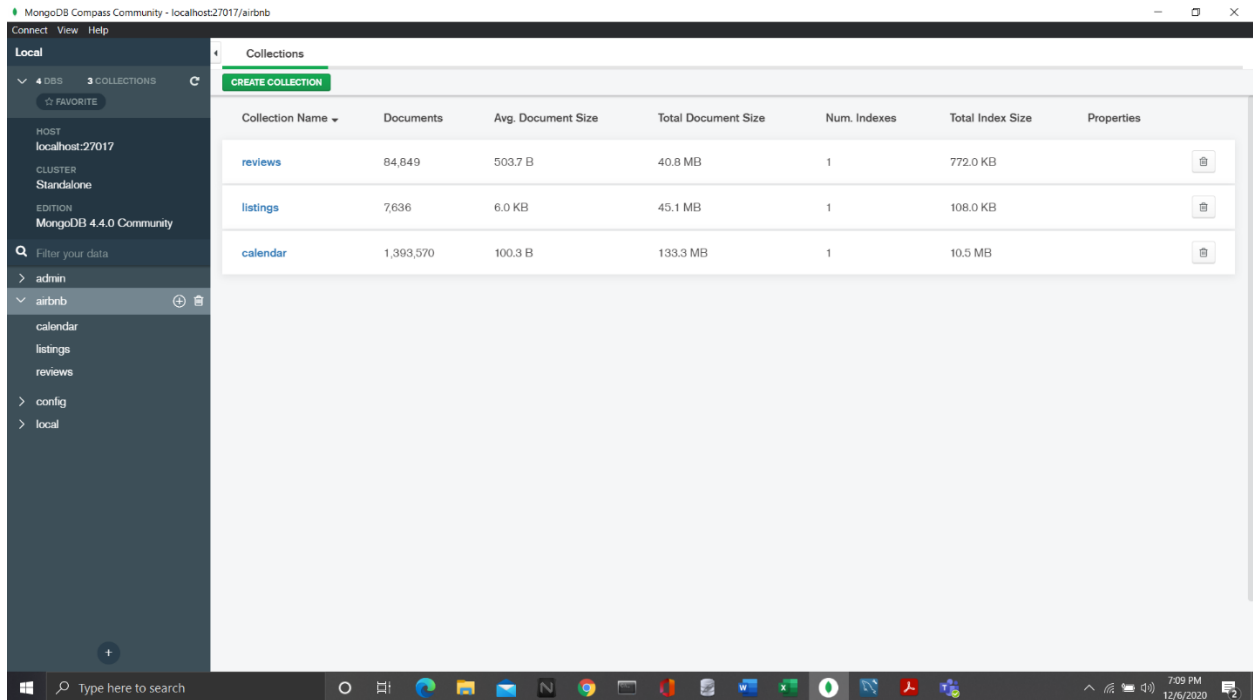
1. There is no relationship between any collections in MongoDB. Hence, we tried to load raw data files into the MongoDB community.
2. To develop the query as instructed, we have loaded the given raw files as individual collections into the MongoDB and used for querying.

Physical Mongo Database

Assumptions/Notes About Data Set

1. Data is loaded into the Mongo through Compass and visualized using Compass and Mongo Shell.
2. We created a database names Airbnb and loaded the required collection for the listings, reviews and calendar for querying.

Screen shot of Physical Database objects (Database, Collections and Attributes)



Data in the Database

Collection Name	Relationships With Other Collections (if any)	# of Documents in Collection
reviews	NA	84849
Listings	NA	7636
calendar	NA	1393570

MongoDB Queries/Code

Pick 3 SQL queries and write them in MongoDB

Mongo Query 1

Question Daily prices can be higher for properties with more bathrooms

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

This statement proves true for most properties, although properties like Dorm have many bathrooms, yet their prices are low and properties like townhouses have single bathrooms but have high prices. But as we see in general, there can be high prices for properties with more bathrooms

Translation

Group by bathrooms and got average daily price all bathrooms, projected the above fields. Sorted based on descending prices.

Screen Shot of MongoDB Query/Code and Results

```
[{
  $group: {
    _id: {
      bathroom: '$bathrooms'
    },
    price: {
      $avg: '$price'
    },
  },
}, {
  $project: {
    'bathrooms': 1,
    'price': 1
  }
}, {
  $sort: {
    price: -1
  }
}]
```

MongoDB Compass Community - localhost:27017

airbnb.listings

Documents 3.8k TOTAL SIZE 22.5MB AVG. SIZE 6.0KB INDEXES 1 TOTAL SIZE 4.0KB AVG. SIZE 4.0KB

Aggregations

Documents Aggregations Explain Plan Indexes

3818 Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

Preview of Documents in the Collection

1. {

2. \$group

3. {

4. _id: {

5. bathrooms: '\$bathrooms'

6. },

7. price: {

8. \$avg: '\$price'

9. },

10. }

Output after \$group stage (Sample of 13 documents)

1. {

2. _id: {

3. bathrooms: 1.5

4. },

5. price: 130.46648187096774

6. }

7. {

8. _id: {

9. bathrooms: 4

10. },

11. price: 463.75

12. }

13. {

14. _id: {

15. bathrooms: 275

16. },

17. price: 463.75

18. }

MongoDB Compass Community - localhost:27017

airbnb.listings

Documents 3.8k TOTAL SIZE 22.5MB AVG. SIZE 6.0KB INDEXES 1 TOTAL SIZE 4.0KB AVG. SIZE 4.0KB

Aggregations

Documents Aggregations Explain Plan Indexes

\$project

1. {

2. 'bathrooms': 1,

3. 'price': 1

4. }

5. }

Output after \$project stage (Sample of 13 documents)

1. {

2. _id: {

3. bathrooms: 3

4. },

5. price: 275.453125

6. }

7. {

8. _id: {

9. bathrooms: 2

10. },

11. price: 191.7372854155496

12. }

13. {

14. _id: {

15. bathrooms: 463

16. },

17. price: 463.75

18. }

\$sort

1. {

2. price: -1

3. }

4. }

Output after \$sort stage (Sample of 13 documents)

1. {

2. _id: {

3. bathrooms: 4.5

4. },

5. price: 406.6666666666667

6. }

7. {

8. _id: {

9. bathrooms: 4

10. },

11. price: 463.75

12. }

13. {

14. _id: {

15. bathrooms: 308

16. },

17. price: 463.75

18. }

Mongo Query 2

Question Weekly prices can be lower for properties with lesser bedrooms

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

This above statement is clearly an invalid statement as we could see there are same weekly prices for all the properties regardless of number of bedrooms. Although daily prices have their differences, the weekly price seems to be normalized across all the properties in Seattle.

Translation

Group by bedrooms, find average price, project all the above variables, sort based on weekly price descending

Screen Shot of MongoDB Query/Code and Results

```
[[
  $group: {
    _id: {
      bedrooms: '$bedrooms'
    },
    weekly_price: {
      $avg: '$weekly_price'
    },
  },
]
}, {
  $project: {
    'bedrooms': 1,
    'weekly_price': 1
  }
}, {
  $sort: {
    weekly_price: -1
  }
}]
```

MongoDB Compass Community - localhost:27017

Connect View Collection Help

Local

4 DBS 3 COLLECTIONS

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.4.0 Community

Filter your data

admin

airbnb

listings

reviews

config

local

airbnb.listings

Aggregations

Documents Aggregations Explain Plan Indexes

COLLATION Untitled Modified SAVE

SAMPLE MODE AUTO PREVIEW

\$group

Output after \$group stage (Sample of 9 documents)

```
1 {
2   "_id": {
3     "bedrooms": "$bedrooms"
4   },
5   "weekly_price": {
6     "$avg": "$weekly_price"
7   },
8 }
9
10
```

Output after \$group stage (Sample of 9 documents)

```
1 { "_id": { "bedrooms": 1, "weekly_price": 1002.3333333333334 } }
2
3 { "_id": { "bedrooms": 2, "weekly_price": 954.375 } }
4
5 { "_id": { "bedrooms": 3, "weekly_price": 1002.3333333333334 } }
6
7 { "_id": { "bedrooms": 4, "weekly_price": 954.375 } }
8
9 { "_id": { "bedrooms": 5, "weekly_price": 1002.3333333333334 } }
10
```

\$project

Output after \$project stage (Sample of 9 documents)

```
1 {
2   "bedrooms": 1,
3   "weekly_price": 1
4 }
5
```

Output after \$project stage (Sample of 9 documents)

```
1 { "_id": { "bedrooms": 1, "weekly_price": 1002.3333333333334 } }
2
3 { "_id": { "bedrooms": 2, "weekly_price": 954.375 } }
4
5 { "_id": { "bedrooms": 3, "weekly_price": 1002.3333333333334 } }
6
7 { "_id": { "bedrooms": 4, "weekly_price": 954.375 } }
8
9 { "_id": { "bedrooms": 5, "weekly_price": 1002.3333333333334 } }
10
```

\$sort

Output after \$sort stage (Sample of 9 documents)

```
1 {
2   "weekly_price": 1
3 }
4
```

MongoDB Compass Community - localhost:27017

Connect View Collection Help

Local

4 DBS 3 COLLECTIONS

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.4.0 Community

Filter your data

admin

airbnb

listings

reviews

config

local

airbnb.listings

Aggregations

Documents Aggregations Explain Plan Indexes

COLLATION Untitled Modified SAVE

SAMPLE MODE AUTO PREVIEW

\$project

Output after \$project stage (Sample of 9 documents)

```
1 {
2   "bedrooms": 1,
3   "weekly_price": 1
4 }
5
```

Output after \$project stage (Sample of 9 documents)

```
1 { "_id": { "bedrooms": 1, "weekly_price": 1002.3333333333334 } }
2
3 { "_id": { "bedrooms": 2, "weekly_price": 954.375 } }
4
5 { "_id": { "bedrooms": 3, "weekly_price": 1002.3333333333334 } }
6
7 { "_id": { "bedrooms": 4, "weekly_price": 954.375 } }
8
9 { "_id": { "bedrooms": 5, "weekly_price": 1002.3333333333334 } }
10
```

\$sort

Output after \$sort stage (Sample of 9 documents)

```
1 {
2   "weekly_price": -1
3 }
4
```

ADD STAGE

Mongo Query 3

Question Strict cancellation policies are best for properties with high review scores

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

Cancellation policies although seem to be a factor initially based on the statement, while we analyzed the data, we could see that the average review scores based on the cancellation policies are lying really close to each other, and properties moderate cancellation policy has slightly higher review scores. So, we can say that cancellation policy is not a strong feature to look at best properties.

Translation

Grouped by cancellation policies, found out average review scores based on cancellation policies, projected all the above columns, sorted based on descending review scores

Screen Shot of MongoDB Query/Code and Results

```
{
  $group: {
    _id: {
      cancellation_policy: '$cancellation_policy'
    },
    review_scores_value: {
      $avg: '$review_scores_value'
    }
  }
}, {
  $project: {
    'cancellation_policy': 1,
    'review_scores_value': 1
  }
}, {
  $sort: {
    review_scores_value: -1
  }
}
```


MongoDB Compass Community - localhost:27017

Connect View Collection Help

Local

4 DBS 3 COLLECTIONS

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.4.0 Community

Filter your data

admin

airbnb

listings

reviews

config

local

airbnb.listings Aggregations

Documents Aggregations Explain Plan Indexes

COLLATION Untitled Modified SAVE

SAMPLE MODE AUTO PREVIEW

7636 Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

Preview of Documents in the Collection

1 * {

2 _id: {

3 cancellation_policy: '\$cancellation_policy'

4 },

5 review_scores_value: {

6 \$avg: '\$review_scores_value'

7 }

8 }

Output after \$group stage (Sample of 3 documents)

1 * {

2 _id: {

3 cancellation_policy: '\$cancellation_policy'

4 },

5 review_scores_value: {

6 \$avg: '\$review_scores_value'

7 }

8 }

Output after \$project stage (Sample of 3 documents)

1 * {

2 _id: {

3 cancellation_policy: '\$cancellation_policy'

4 },

5 review_scores_value: {

6 \$avg: '\$review_scores_value'

7 }

8 }

11:33 AM 12/6/2020

MongoDB Compass Community - localhost:27017

Connect View Collection Help

Local

4 DBS 3 COLLECTIONS

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.4.0 Community

Filter your data

admin

airbnb

listings

reviews

config

local

airbnb.listings Aggregations

Documents Aggregations Explain Plan Indexes

COLLATION Untitled Modified SAVE

SAMPLE MODE AUTO PREVIEW

7636 Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

Preview of Documents in the Collection

1 * {

2 _id: {

3 cancellation_policy: '\$cancellation_policy'

4 },

5 review_scores_value: {

6 \$avg: '\$review_scores_value'

7 }

8 }

Output after \$group stage (Sample of 3 documents)

1 * {

2 _id: {

3 cancellation_policy: '\$cancellation_policy'

4 },

5 review_scores_value: {

6 \$avg: '\$review_scores_value'

7 }

8 }

Output after \$project stage (Sample of 3 documents)

1 * {

2 _id: {

3 cancellation_policy: '\$cancellation_policy'

4 },

5 review_scores_value: {

6 \$avg: '\$review_scores_value'

7 }

8 }

11:33 AM 12/6/2020

Physical Mongo Database

Assumptions/Notes About Data Set

Screen shot of Physical Database objects (Database, Collections and Attributes)

Data in the Database

Collection Name	Relationships With Other Collections (if any)	# of Documents in Collection

MongoDB Queries/Code

Pick 3 SQL queries and write them in MongoDB

Mongo Query 1

Question

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

Translation

Screen Shot of MongoDB Query/Code and Results

Mongo Query 2

Question

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

Translation

Screen Shot of MongoDB Query/Code and Results

Mongo Query 3

Question

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

Translation

Screen Shot of MongoDB Query/Code and Results