FIT5137, Advanced Database Technology

Group Assignment - Sem 2/2019

Submitted by

Manali Choudhary

Rohan Nishchal

GROUP ASSIGNMENT COVER SHEET

Student ID Number	Surname	Given Names
30151198	*Choudhary*	**Manali Prakash**
30383633	*Nischal*	**Rohan**

* Please include the names of all other group members.

Unit name and code	**FIT5137, Advanced Database Technology**	
Title of assignment	**Group Assignment - Sem 2/2019**	
Lecturer/tutor	Chaluka	
Tutorial day and time	18th September, 2019 11.55 pm	**Campus Caulfield**
Is this an authorised group assignment? □ **Yes** □ **No**		
Has any part of this assignment been previously submitted as part of another unit/course? □ **Yes** □ **No**		
Due Date *18th September, 2019*		**Date submitted** *18th September, 2019*

All work must be submitted by the due date. If an extension of work is granted this must be specified with the signature of the lecturer/tutor.

Extension granted until (date) .... Signature of lecturer/tutor ....

Please note that it is your responsibility to retain copies of your assessments.

Intentional plagiarism or collusion amounts to cheating under Part 7 of the Monash University (Council) Regulations

Plagiarism: Plagiarism means taking and using another person's ideas or manner of expressing them and passing them off as one's own. For example, by failing to give appropriate acknowledgement. The material used can be from any source (staff, students or the internet, published and unpublished works).

Collusion: Collusion means unauthorised collaboration with another person on assessable written, oral or practical work and includes paying another person to complete all or part of the work.

Where there are reasonable grounds for believing that intentional plagiarism or collusion has occurred, this will be reported to the Associate Dean (Education) or delegate, who may disallow the work concerned by prohibiting assessment or refer the matter to the Faculty Discipline Panel for a hearing.

Student Statement:

- I have read the university's Student Academic Integrity Policy and Procedures.
- I understand the consequences of engaging in plagiarism and collusion as described in Part 7 of the Monash University (Council) Regulations http://adm.monash.edu/legal/legislation/statutes
- I have taken proper care to safeguard this work and made all reasonable efforts to ensure it could not be copied.
- No part of this assignment has been previously submitted as part of another unit/course.
- I acknowledge and agree that the assessor of this assignment may for the purposes of assessment, reproduce the assignment and:
    i. provide to another member of faculty and any external marker; and/or
    ii. submit it to a text matching software; and/or
    iii. submit it to a text matching software which may then retain a copy of the assignment on its database for the purpose of future plagiarism checking.
- I certify that I have not plagiarised the work of others or participated in unauthorised collaboration when preparing this assignment.

Signature ....Manali Choudhary, Rohan Nischal... *Date………*18th September, 2019 ……………………………

* delete (iii) if not applicable

Signature __ *Manali Choudhary* ________________Date:____ 18th September, 2019 ________________ Signature _________________ *Rohan Nischal* ___________Date:____ 18th September, 2019 ____

Signature __________________Date:__________________ Signature ______________________________Date:_______

Signature __________________Date:____________________ Signature ______________________________Date:_______

Privacy Statement

The information on this form is collected for the primary purpose of assessing your assignment and ensuring the academic integrity requirements of the University are met. Other purposes of collection include recording your plagiarism and collusion declaration, attending to course and administrative matters and statistical analyses. If you choose not to complete all the questions on this form it may not be possible for Monash University to assess your assignment. You have a right to access personal information that Monash University holds about you, subject to any exceptions in relevant legislation. If you wish to seek access to your personal information or inquire about the handling of your personal information, please contact the University Privacy Officer: privacyofficer@adm.monash.edu.au

Contribution Declaration Form
(to be completed by all team members)

Please fill in the form with the contribution from each student towards the assignment.

1 NAME AND CONTRIBUTION DETAILS

Student ID	Student Name	Contribution Percentage
30151198	Manali Choudhary	50%
30383633	Rohan Nischal	50%

2 DECLARATION

Percentage of contribution:

I. Name: Manali, ID: 30151198**, Contribution: 50%**

II. Name: Rohan, ID: 30383633**, Contribution: 50%**

List of parts that each student did:

I. Manali:

□ Task C.1 (MongoDB create collections and insert for Host, Cassandra insert)

□ Task C.2(Q1-3, Q6)

□ Task C.3(Q1-7 queries of Mongodb,Q16-18,indexing)

□ Task C.4(comparison and analysis)

II. Rohan:

□ Task C.1 (MongoDB create collections and insert for Listing, Cassandra create table, Cassandra insert)

□ Task C.2(Q4-5, Q7)

□ Task C.3(Q8-15 queries of Mongodb,Q18-20,extra queries)

□ Task C.4(flowchart and analysis)

We declare that:

- The information we have supplied in or with this form is complete and correct.
- We understand that the information we have provided in this form will be used for individual assessment of the assignment.

3 SIGNATURE

Signatures	Manali Choudhary	

Rohan Nischal

	Day Month Year
Date	18 / 09 / 2019

--C.1. Database Design.--

INSERT for MongoDB

```
use FIT5137_Assign;

db.createCollection("listing");

db.createCollection("host");
```

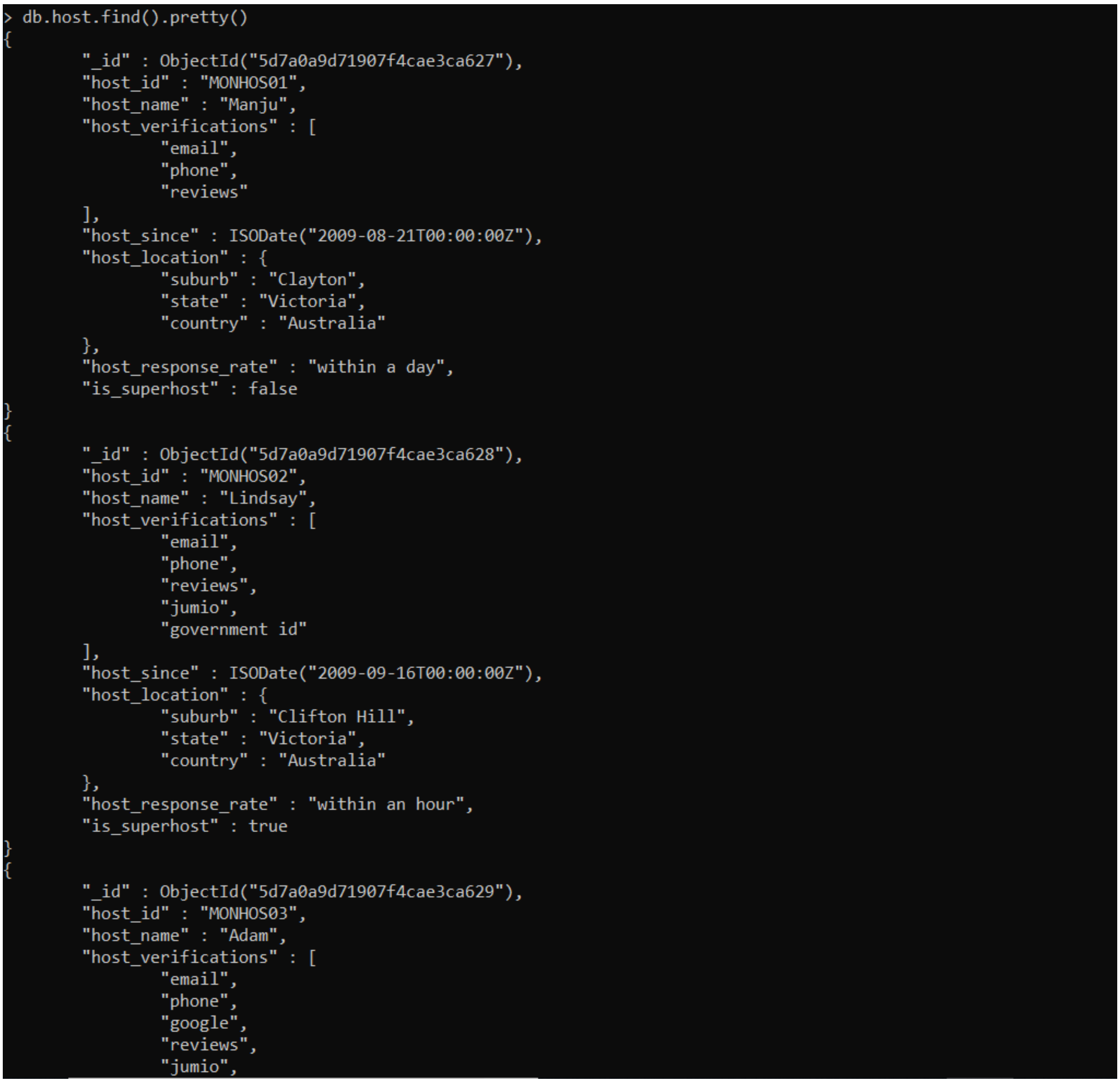
For Host:

```
db.host.insertOne({
host_id : "MONHOS01",
host_name : "Manju",
host_verifications : ["email","phone","reviews"],
host_since : new Date("2009-08-21"),
host_location : { suburb : "Clayton", state : "Victoria", country : "Australia" },
host_response_rate : "within a day",
is_superhost : false
});
```

For Listing:

```
db.listing.insertOne({
listing_id : "MONLST01",
name : "Monash Beautiful House",
host_id : "MONHOS14",
neighbourhood : "Manningham",
address : { suburb : "Clayton", state : "VIC", postcode : "3800" },
latitude : -37.773,
longitude : 145.09213,
room_type : "Entire home",
amenities : ["TV","Wifi","Pets Allowed","Family friendly","24-hour check-in","Self check-in"],
price_per_night : 61,
price_for_extra_people : 22,
min_nights_for_booking : 1,
availability : 365
}
);
```

Note: The other insert queries are similar and are mentioned in the file FIT5137_Assign_C1_MongoDB.txt



```
> db.host.find().pretty()
{
        "_id" : ObjectId("5d7a0a9d71907f4cae3ca627"),
        "host_id" : "MONHOS01",
        "host_name" : "Manju",
        "host_verifications" : [
                "email",
                "phone",
                "reviews"
        ],
        "host_since" : ISODate("2009-08-21T00:00:00Z"),
        "host_location" : {
                "suburb" : "Clayton",
                "state" : "Victoria",
                "country" : "Australia"
        },
        "host_response_rate" : "within a day",
        "is_superhost" : false
}
{
        "_id" : ObjectId("5d7a0a9d71907f4cae3ca628"),
        "host_id" : "MONHOS02",
        "host_name" : "Lindsay",
        "host_verifications" : [
                "email",
                "phone",
                "reviews",
                "jumio",
                "government id"
        ],
        "host_since" : ISODate("2009-09-16T00:00:00Z"),
        "host_location" : {
                "suburb" : "Clifton Hill",
                "state" : "Victoria",
                "country" : "Australia"
        },
        "host_response_rate" : "within an hour",
        "is_superhost" : true
}
{
        "_id" : ObjectId("5d7a0a9d71907f4cae3ca629"),
        "host_id" : "MONHOS03",
        "host_name" : "Adam",
        "host_verifications" : [
                "email",
                "phone",
                "google",
                "reviews",
                "jumio",
```

INSERT for Cassandra

CREATE KEYSPACE FIT5137_Assign

WITH replication = {'class': 'SimpleStrategy',

'replication_factor' : 1};

use FIT5137_Assign;

CREATE TABLE review (listing_id text,

review_id text,

review_date date,

sequence text,

reviewer_id int,

reviewer_name text,

review_scores_rating int,

satisfied_reason set<text>,

comments text,

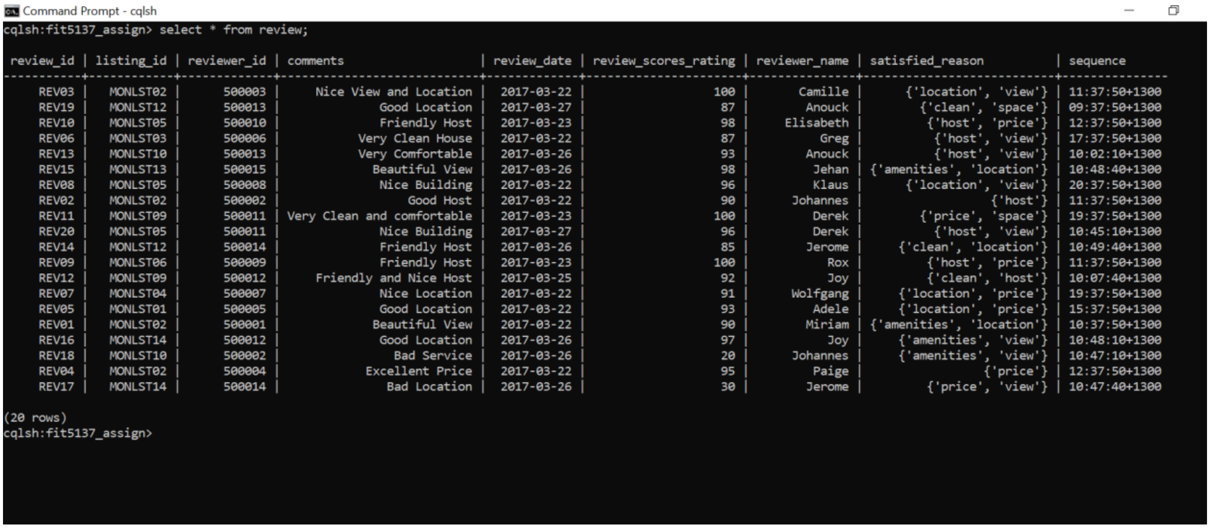
PRIMARY KEY ((review_id),listing_id,reviewer_id)

) ;

INSERT INTO review (listing_id, review_id,review_date,sequence,reviewer_id,reviewer_name,review_scores_rating,satisfied_reason,comments)

VALUES ('MONLST02', 'REV01','2017-03-22','10:37:50+1300',500001,'Miriam',90,{'location','amenities'},'Beautiful View');

Note: The other insert queries are similar and are mentioned in the file FIT5137_Assign_C1_Cassandra.txt



```
cqlsh:fit5137_assign> select * from review;

 review_id | listing_id | reviewer_id | comments                   | review_date | review_scores_rating | reviewer_name | satisfied_reason          | sequence
-----------+------------+-------------+----------------------------+-------------+----------------------+---------------+---------------------------+---------------
     REV03 |   MONLST02 |      500003 |     Nice View and Location |  2017-03-22 |                  100 |       Camille |      {'location', 'view'} | 11:37:50+1300
     REV19 |   MONLST12 |      500013 |              Good Location |  2017-03-27 |                   87 |        Anouck |        {'clean', 'space'} | 09:37:50+1300
     REV10 |   MONLST05 |      500010 |              Friendly Host |  2017-03-23 |                   98 |     Elisabeth |         {'host', 'price'} | 12:37:50+1300
     REV06 |   MONLST03 |      500006 |           Very Clean House |  2017-03-22 |                   87 |          Greg |          {'host', 'view'} | 17:37:50+1300
     REV13 |   MONLST10 |      500013 |           Very Comfortable |  2017-03-26 |                   93 |        Anouck |          {'host', 'view'} | 10:02:10+1300
     REV15 |   MONLST13 |      500015 |             Beautiful View |  2017-03-26 |                   98 |         Jehan | {'amenities', 'location'} | 10:48:40+1300
     REV08 |   MONLST05 |      500008 |              Nice Building |  2017-03-22 |                   96 |         Klaus |      {'location', 'view'} | 20:37:50+1300
     REV02 |   MONLST02 |      500002 |                  Good Host |  2017-03-22 |                   90 |      Johannes |                  {'host'} | 11:37:50+1300
     REV11 |   MONLST09 |      500011 | Very Clean and comfortable |  2017-03-23 |                  100 |         Derek |        {'price', 'space'} | 19:37:50+1300
     REV20 |   MONLST05 |      500011 |              Nice Building |  2017-03-27 |                   96 |         Derek |          {'host', 'view'} | 10:45:10+1300
     REV14 |   MONLST12 |      500014 |              Friendly Host |  2017-03-26 |                   85 |        Jerome |     {'clean', 'location'} | 10:49:40+1300
     REV09 |   MONLST06 |      500009 |              Friendly Host |  2017-03-23 |                  100 |           Rox |         {'host', 'price'} | 11:37:50+1300
     REV12 |   MONLST09 |      500012 |     Friendly and Nice Host |  2017-03-25 |                   92 |           Joy |         {'clean', 'host'} | 10:07:40+1300
     REV07 |   MONLST04 |      500007 |              Nice Location |  2017-03-22 |                   91 |      Wolfgang |     {'location', 'price'} | 19:37:50+1300
     REV05 |   MONLST01 |      500005 |              Good Location |  2017-03-22 |                   93 |         Adele |     {'location', 'price'} | 15:37:50+1300
     REV01 |   MONLST02 |      500001 |             Beautiful View |  2017-03-22 |                   90 |        Miriam | {'amenities', 'location'} | 10:37:50+1300
     REV16 |   MONLST14 |      500012 |              Good Location |  2017-03-26 |                   97 |           Joy |     {'amenities', 'view'} | 10:48:10+1300
     REV18 |   MONLST10 |      500002 |                Bad Service |  2017-03-26 |                   20 |      Johannes |     {'amenities', 'view'} | 10:47:10+1300
     REV04 |   MONLST02 |      500004 |            Excellent Price |  2017-03-22 |                   95 |         Paige |                 {'price'} | 12:37:50+1300
     REV17 |   MONLST14 |      500014 |               Bad Location |  2017-03-26 |                   30 |        Jerome |         {'price', 'view'} | 10:47:40+1300

(20 rows)
cqlsh:fit5137_assign>
```

--C.2. Database Modifications.--

Q1

db.host.updateOne(

{ host_name : "Adam" },

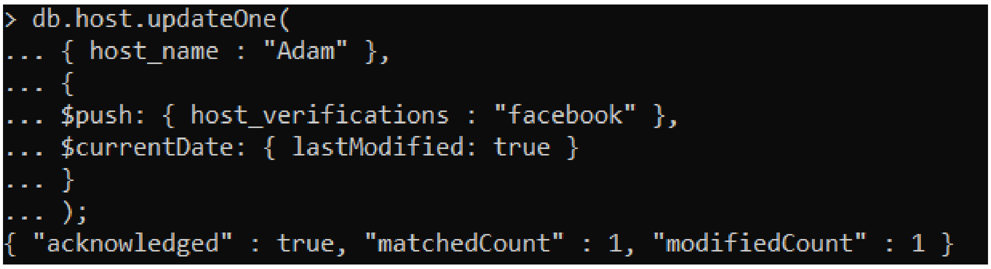
{

$push: { host_verifications : "facebook" },

$currentDate: { lastModified: true }

}

);



```
> db.host.updateOne(
... { host_name : "Adam" },
... {
... $push: { host_verifications : "facebook" },
... $currentDate: { lastModified: true }
... }
... );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
```

Q2

db.host.insertMany([

{

host_id : "MONHOS11",

host_name : "Alison",

host_verifications : ["email","phone","facebook","reviews"],

host_since : new Date("2019-01-09"),

host_location : { suburb : "Caulfield", state : "Victoria", country : "Australia" },

host_response_rate : "within an hour",

is_superhost : false

},

{

host_id : "MONHOS12",

host_name : "Mike",

host_verifications : ["email","phone"],

host_since : new Date("2019-01-09"),

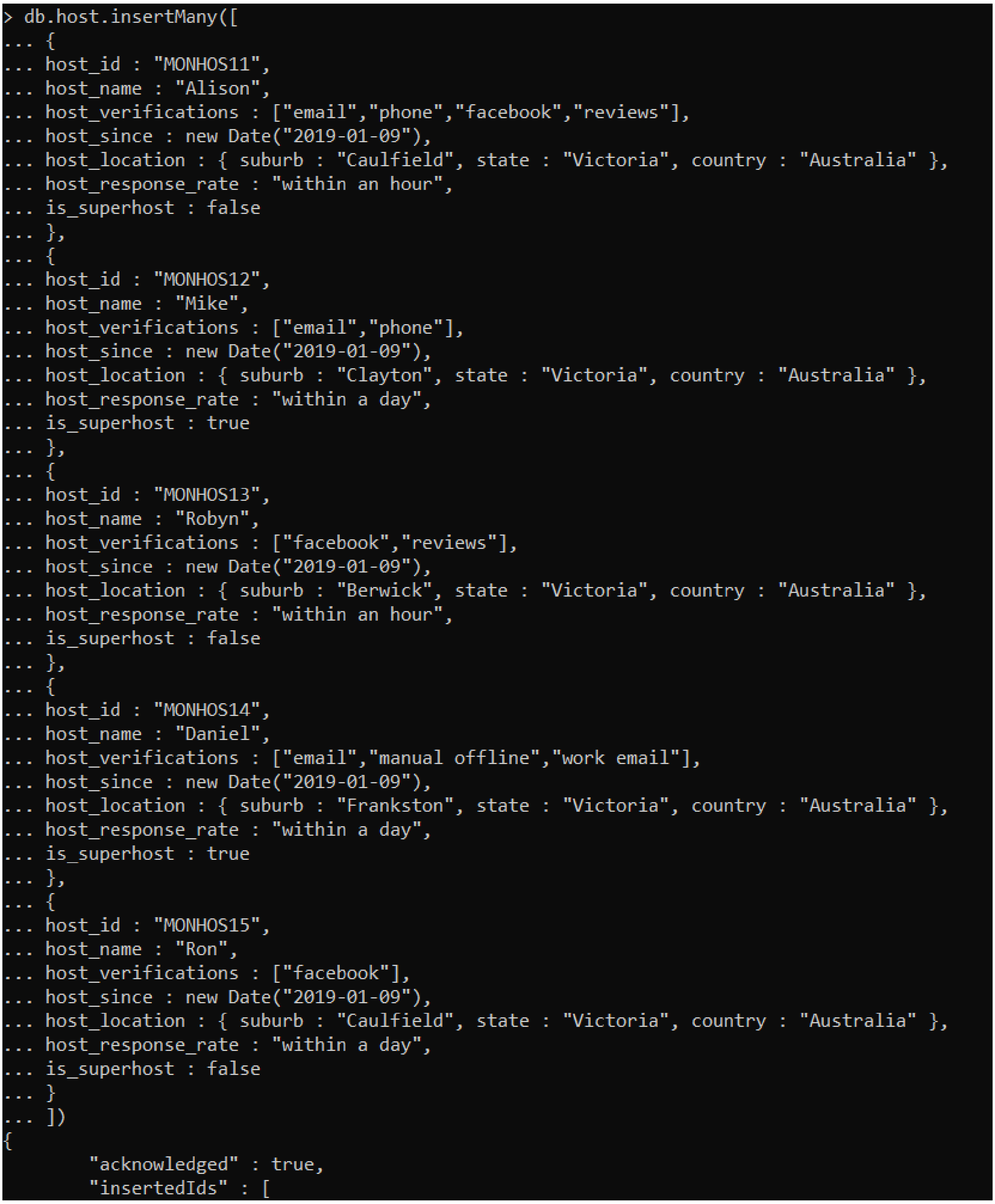
host_location : { suburb : "Clayton", state : "Victoria", country : "Australia" },

host_response_rate : "within a day",

is_superhost : true

},

```
{
host_id : "MONHOS13",
host_name : "Robyn",
host_verifications : ["facebook","reviews"],
host_since : new Date("2019-01-09"),
host_location : { suburb : "Berwick", state : "Victoria", country : "Australia" },
host_response_rate : "within an hour",
is_superhost : false
},
{
host_id : "MONHOS14",
host_name : "Daniel",
host_verifications : ["email","manual offline","work email"],
host_since : new Date("2019-01-09"),
host_location : { suburb : "Frankston", state : "Victoria", country : "Australia" },
host_response_rate : "within a day",
is_superhost : true
},
{
host_id : "MONHOS15",
host_name : "Ron",
host_verifications : ["facebook"],
host_since : new Date("2019-01-09"),
host_location : { suburb : "Caulfield", state : "Victoria", country : "Australia" },
host_response_rate : "within a day",
is_superhost : false
}
])
;
```



```
> db.host.insertMany([
... {
... host_id : "MONHOS11",
... host_name : "Alison",
... host_verifications : ["email","phone","facebook","reviews"],
... host_since : new Date("2019-01-09"),
... host_location : { suburb : "Caulfield", state : "Victoria", country : "Australia" },
... host_response_rate : "within an hour",
... is_superhost : false
... },
... {
... host_id : "MONHOS12",
... host_name : "Mike",
... host_verifications : ["email","phone"],
... host_since : new Date("2019-01-09"),
... host_location : { suburb : "Clayton", state : "Victoria", country : "Australia" },
... host_response_rate : "within a day",
... is_superhost : true
... },
... {
... host_id : "MONHOS13",
... host_name : "Robyn",
... host_verifications : ["facebook","reviews"],
... host_since : new Date("2019-01-09"),
... host_location : { suburb : "Berwick", state : "Victoria", country : "Australia" },
... host_response_rate : "within an hour",
... is_superhost : false
... },
... {
... host_id : "MONHOS14",
... host_name : "Daniel",
... host_verifications : ["email","manual offline","work email"],
... host_since : new Date("2019-01-09"),
... host_location : { suburb : "Frankston", state : "Victoria", country : "Australia" },
... host_response_rate : "within a day",
... is_superhost : true
... },
... {
... host_id : "MONHOS15",
... host_name : "Ron",
... host_verifications : ["facebook"],
... host_since : new Date("2019-01-09"),
... host_location : { suburb : "Caulfield", state : "Victoria", country : "Australia" },
... host_response_rate : "within a day",
... is_superhost : false
... }
... ])
{
        "acknowledged" : true,
        "insertedIds" : [
```

Q3

```
db.host.updateOne(

{host_response_rate:"within an hour",is_superhost:false},

{

$set: { is_superhost : true },
```

$currentDate: { lastModified: true }

}

);

Q4

db.listing.deleteMany({ availability : 0 });

Q5

db.listing.updateMany(

{neighbourhood:"Monash"},

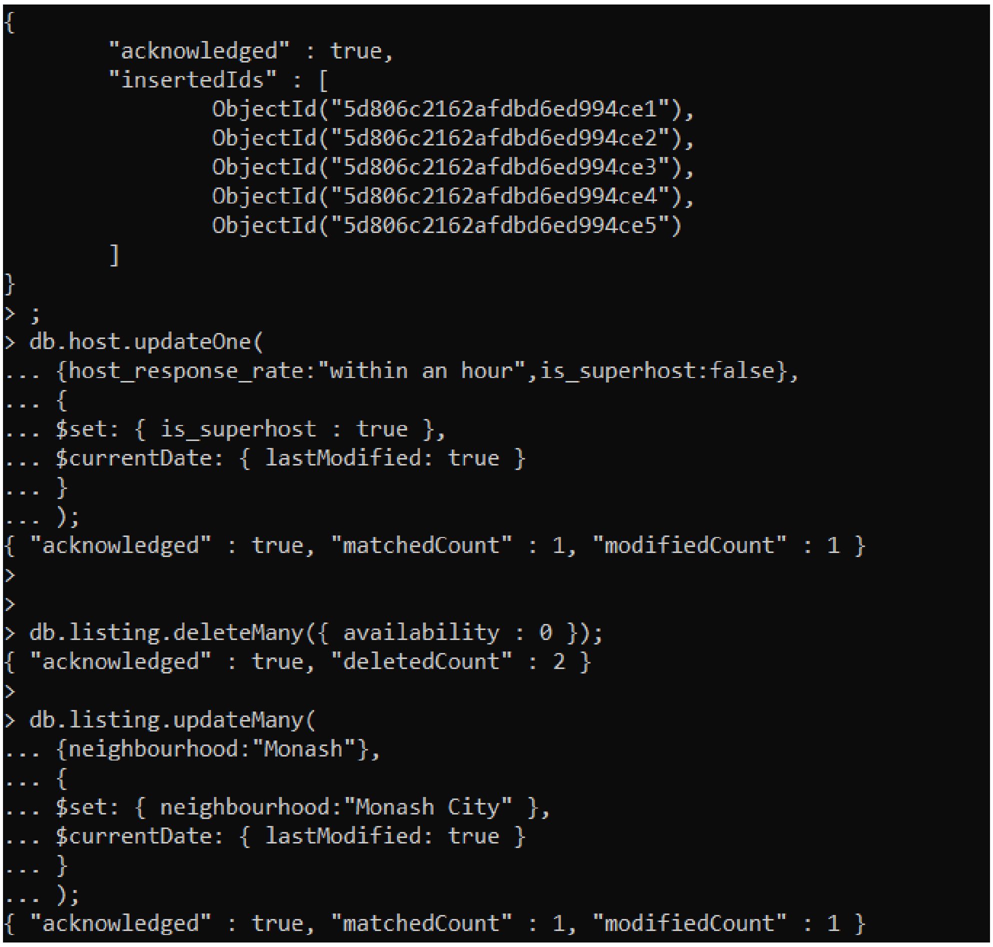
{

$set: { neighbourhood:"Monash City" },

$currentDate: { lastModified: true }

}

);



```
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("5d806c2162afdbd6ed994ce1"),
                ObjectId("5d806c2162afdbd6ed994ce2"),
                ObjectId("5d806c2162afdbd6ed994ce3"),
                ObjectId("5d806c2162afdbd6ed994ce4"),
                ObjectId("5d806c2162afdbd6ed994ce5")
        ]
}
> ;
> db.host.updateOne(
... {host_response_rate:"within an hour",is_superhost:false},
... {
... $set: { is_superhost : true },
... $currentDate: { lastModified: true }
... }
... );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
>
>
> db.listing.deleteMany({ availability : 0 });
{ "acknowledged" : true, "deletedCount" : 2 }
>
> db.listing.updateMany(
... {neighbourhood:"Monash"},
... {
... $set: { neighbourhood:"Monash City" },
... $currentDate: { lastModified: true }
... }
... );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
```

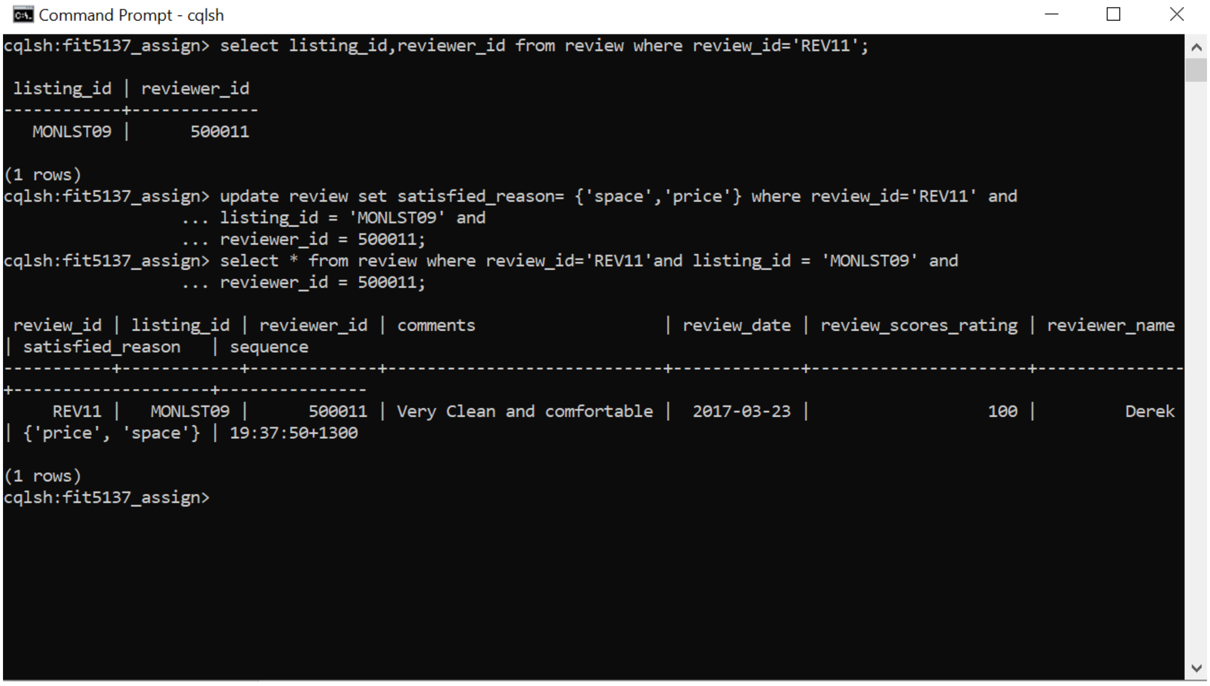
Q6

select listing_id,reviewer_id from review where review_id='REV11';

update review set satisfied_reason= {'space','price'} where review_id='REV11' and

listing_id = 'MONLST09' and

reviewer_id = 500011;



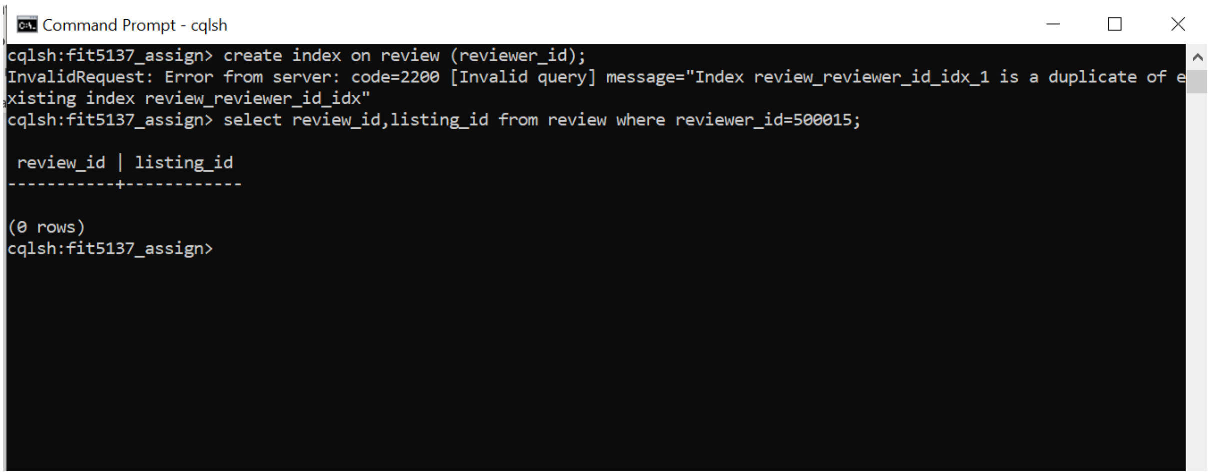
```
cqlsh:fit5137_assign> select listing_id,reviewer_id from review where review_id='REV11';

 listing_id | reviewer_id
------------+-------------
   MONLST09 |      500011

(1 rows)
cqlsh:fit5137_assign> update review set satisfied_reason= {'space','price'} where review_id='REV11' and
                  ... listing_id = 'MONLST09' and
                  ... reviewer_id = 500011;
cqlsh:fit5137_assign> select * from review where review_id='REV11'and listing_id = 'MONLST09' and
                  ... reviewer_id = 500011;

 review_id | listing_id | reviewer_id | comments                     | review_date | review_scores_rating | reviewer_name
| satisfied_reason   | sequence
-----------+------------+-------------+-----------------------------+-------------+----------------------+---------------
+--------------------+--------------
     REV11 |   MONLST09 |      500011 | Very Clean and comfortable |  2017-03-23 |                  100 |         Derek
| {'price', 'space'} | 19:37:50+1300

(1 rows)
cqlsh:fit5137_assign>
```

Q7

create index on review (reviewer_id);

select review_id,listing_id from review where reviewer_id=500015;

delete from review where reviewer_id = 500015 and review_id= 'REV15' and listing_id= 'MONLST13';



```
cqlsh:fit5137_assign> create index on review (reviewer_id);
InvalidRequest: Error from server: code=2200 [Invalid query] message="Index review_reviewer_id_idx_1 is a duplicate of e
xisting index review_reviewer_id_idx"
cqlsh:fit5137_assign> select review_id,listing_id from review where reviewer_id=500015;

 review_id | listing_id
-----------+------------

(0 rows)
cqlsh:fit5137_assign>
```

--C.3. Queries.--

Importing JSON files for MongoDB

```
mongoimport --db FIT5137_Assign_C3 --collection host --file D:\assignment_data\host.json

mongoimport --db FIT5137_Assign_C3 --collection listing --file D:\assignment_data\listing.json
```

Importing CSV file for Cassandra

```
CREATE KEYSPACE FIT5137_Assign_C3

WITH replication = {'class': 'SimpleStrategy',

'replication_factor' : 1};


use FIT5137_Assign_C3;


CREATE TABLE review (listing_id int,

id int,

date date,

reviewer_id int,

reviewer_name text,

review_scores_rating int,

comments text,

PRIMARY KEY ((listing_id),id,reviewer_id)

) ;


COPY FIT5137_Assign_C3.review (listing_id,id,date,reviewer_id,reviewer_name,review_scores_rating,comments)

FROM 'C:\Users\Manali Choudhary\Documents\Sem2\adb\assignment_data\assignment_data\review.csv' WITH HEADER=TRUE;


select count(*) from review;
```

//Windows OS

Create embedding model

```
db.listing.aggregate([
 { $lookup : {
 from : "host",
 localField : "host_id",
 foreignField : "host_id",
 as : "host_info"
 } },
 {
 $out:"listing_host"
 }
]);
```

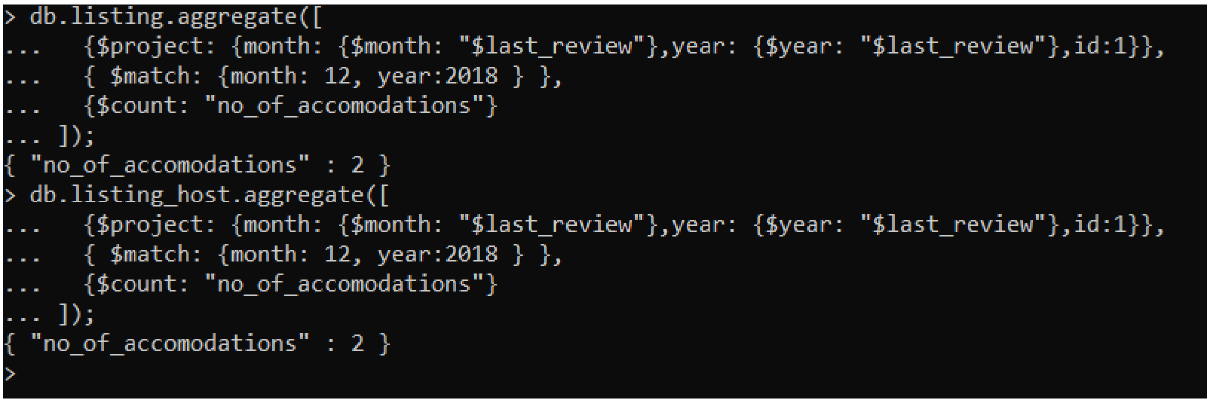
Q1

Reference

```
db.listing.aggregate([
  {$project: {month: {$month: "$last_review"},year: {$year: "$last_review"},id:1}},
  { $match: {month: 12, year:2018 } },
  {$count: "no_of_accomodations"}
]);
```

Embedded

```
db.listing_host.aggregate([
  {$project: {month: {$month: "$last_review"},year: {$year: "$last_review"},id:1}},
  { $match: {month: 12, year:2018 } },
  {$count: "no_of_accomodations"}
]);
```



```
> db.listing.aggregate([
...   {$project: {month: {$month: "$last_review"},year: {$year: "$last_review"},id:1}},
...   { $match: {month: 12, year:2018 } },
...   {$count: "no_of_accomodations"}
... ]);
{ "no_of_accomodations" : 2 }
> db.listing_host.aggregate([
...   {$project: {month: {$month: "$last_review"},year: {$year: "$last_review"},id:1}},
...   { $match: {month: 12, year:2018 } },
...   {$count: "no_of_accomodations"}
... ]);
{ "no_of_accomodations" : 2 }
>
```

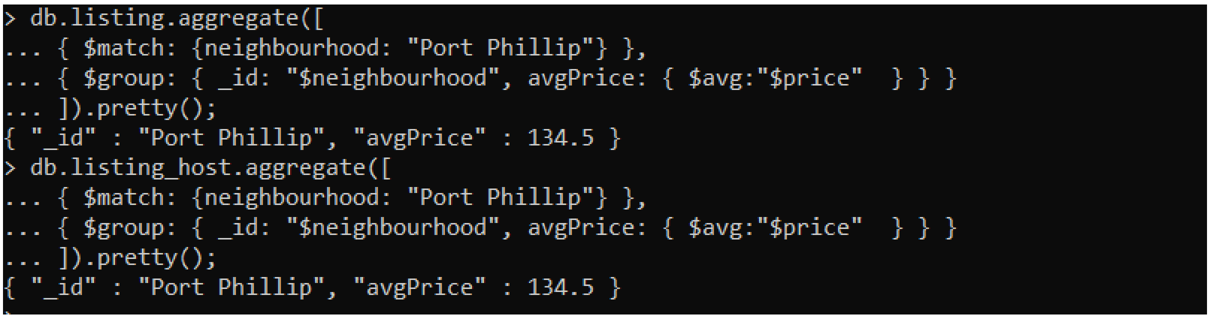
Q2

Reference

```
db.listing.aggregate([
{ $match: {neighbourhood: "Port Phillip"} },
{ $group: { _id: "$neighbourhood", avgPrice: { $avg:"$price"  } } }
]).pretty();
```

Embedded

```
db.listing_host.aggregate([
{ $match: {neighbourhood: "Port Phillip"} },
{ $group: { _id: "$neighbourhood", avgPrice: { $avg:"$price"  } } }
]).pretty();
```



```
> db.listing.aggregate([
... { $match: {neighbourhood: "Port Phillip"} },
... { $group: { _id: "$neighbourhood", avgPrice: { $avg:"$price"  } } }
... ]).pretty();
{ "_id" : "Port Phillip", "avgPrice" : 134.5 }
> db.listing_host.aggregate([
... { $match: {neighbourhood: "Port Phillip"} },
... { $group: { _id: "$neighbourhood", avgPrice: { $avg:"$price"  } } }
... ]).pretty();
{ "_id" : "Port Phillip", "avgPrice" : 134.5 }
```

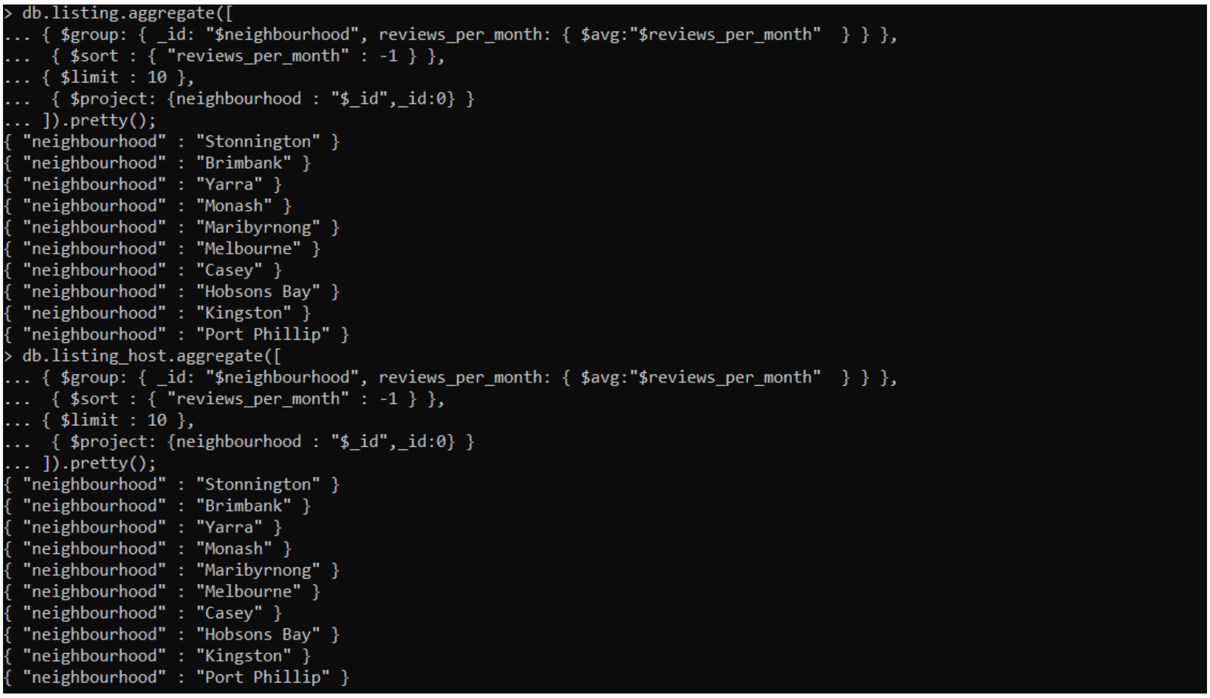
Q3

Reference

```
db.listing.aggregate([
{ $group: { _id: "$neighbourhood", reviews_per_month: { $avg:"$reviews_per_month"  } } },

 { $sort : { "reviews_per_month" : -1 } },

{ $limit : 10 },

 { $project: {neighbourhood : "$_id",_id:0} }

]).pretty();
```

Embedded

```
db.listing_host.aggregate([

{ $group: { _id: "$neighbourhood", reviews_per_month: { $avg:"$reviews_per_month"  } } },

 { $sort : { "reviews_per_month" : -1 } },

{ $limit : 10 },

 { $project: {neighbourhood : "$_id",_id:0} }

]).pretty();
```



```
> db.listing.aggregate([
... { $group: { _id: "$neighbourhood", reviews_per_month: { $avg:"$reviews_per_month"  } } },
...  { $sort : { "reviews_per_month" : -1 } },
... { $limit : 10 },
...  { $project: {neighbourhood : "$_id",_id:0} }
... ]).pretty();
{ "neighbourhood" : "Stonnington" }
{ "neighbourhood" : "Brimbank" }
{ "neighbourhood" : "Yarra" }
{ "neighbourhood" : "Monash" }
{ "neighbourhood" : "Maribyrnong" }
{ "neighbourhood" : "Melbourne" }
{ "neighbourhood" : "Casey" }
{ "neighbourhood" : "Hobsons Bay" }
{ "neighbourhood" : "Kingston" }
{ "neighbourhood" : "Port Phillip" }
> db.listing_host.aggregate([
... { $group: { _id: "$neighbourhood", reviews_per_month: { $avg:"$reviews_per_month"  } } },
...  { $sort : { "reviews_per_month" : -1 } },
... { $limit : 10 },
...  { $project: {neighbourhood : "$_id",_id:0} }
... ]).pretty();
{ "neighbourhood" : "Stonnington" }
{ "neighbourhood" : "Brimbank" }
{ "neighbourhood" : "Yarra" }
{ "neighbourhood" : "Monash" }
{ "neighbourhood" : "Maribyrnong" }
{ "neighbourhood" : "Melbourne" }
{ "neighbourhood" : "Casey" }
{ "neighbourhood" : "Hobsons Bay" }
{ "neighbourhood" : "Kingston" }
{ "neighbourhood" : "Port Phillip" }
```

Q4

Reference

```
db.listing.aggregate([
```

{ $group: { _id: null, maxRange: { $max:"$number_of_reviews" }, minRange: { $min:"$number_of_reviews" } } },

 { $project: { _id:0, minNumberOfReviews : "$minRange", maxNumberOfReviews : "$maxRange" }}

]).pretty();

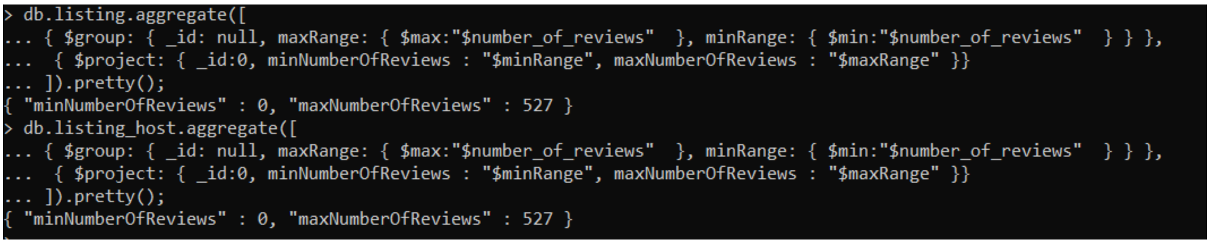
Embedded

db.listing_host.aggregate([

{ $group: { _id: null, maxRange: { $max:"$number_of_reviews" }, minRange: { $min:"$number_of_reviews" } } },

 { $project: { _id:0, minNumberOfReviews : "$minRange", maxNumberOfReviews : "$maxRange" }}

]).pretty();



```
> db.listing.aggregate([
... { $group: { _id: null, maxRange: { $max:"$number_of_reviews"  }, minRange: { $min:"$number_of_reviews"  } } },
...  { $project: { _id:0, minNumberOfReviews : "$minRange", maxNumberOfReviews : "$maxRange" }}
... ]).pretty();
{ "minNumberOfReviews" : 0, "maxNumberOfReviews" : 527 }
> db.listing_host.aggregate([
... { $group: { _id: null, maxRange: { $max:"$number_of_reviews"  }, minRange: { $min:"$number_of_reviews"  } } },
...  { $project: { _id:0, minNumberOfReviews : "$minRange", maxNumberOfReviews : "$maxRange" }}
... ]).pretty();
{ "minNumberOfReviews" : 0, "maxNumberOfReviews" : 527 }
```

Q5

Reference

db.listing.aggregate([

 { $group : { _id : "$room_type", noOfOccurence : { $sum: 1 } } },

 { $sort : { "noOfOccurence" : -1 } },

 {$limit : 1},

 { $project: {room_type : "$_id",_id:0} }

]).pretty();

Embedded

db.listing_host.aggregate([

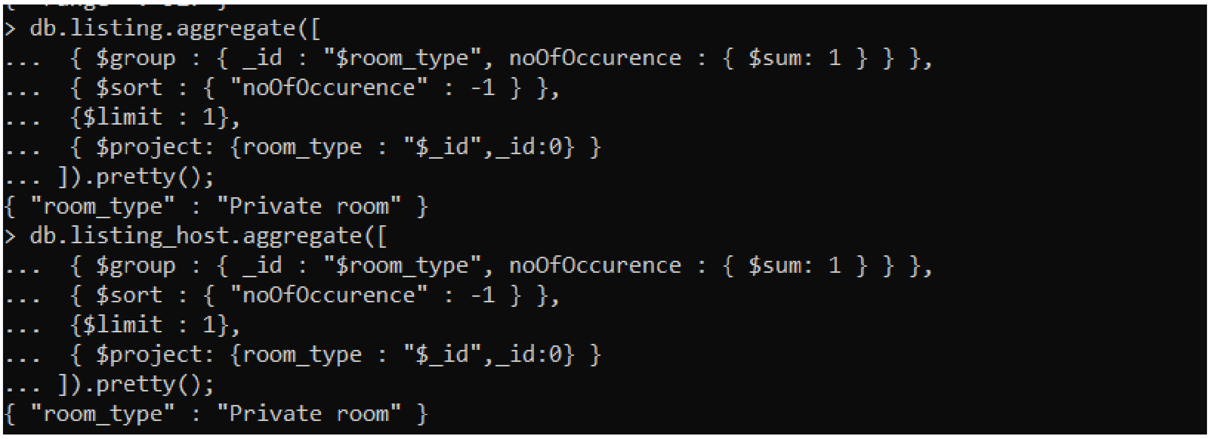
 { $group : { _id : "$room_type", noOfOccurence : { $sum: 1 } } },

 { $sort : { "noOfOccurence" : -1 } },

 {$limit : 1},

 { $project: {room_type : "$_id",_id:0} }

]).pretty();



```
> db.listing.aggregate([
...   { $group : { _id : "$room_type", noOfOccurence : { $sum: 1 } } },
...   { $sort : { "noOfOccurence" : -1 } },
...   {$limit : 1},
...   { $project: {room_type : "$_id",_id:0} }
... ]).pretty();
{ "room_type" : "Private room" }
> db.listing_host.aggregate([
...   { $group : { _id : "$room_type", noOfOccurence : { $sum: 1 } } },
...   { $sort : { "noOfOccurence" : -1 } },
...   {$limit : 1},
...   { $project: {room_type : "$_id",_id:0} }
... ]).pretty();
{ "room_type" : "Private room" }
```

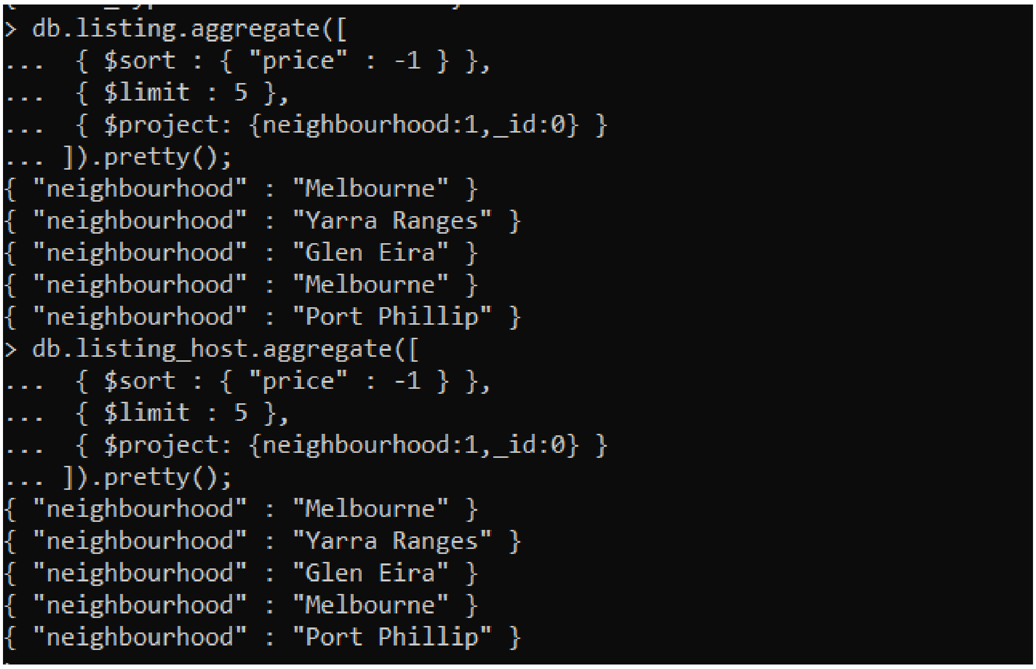
Q6

Reference

```
db.listing.aggregate([
 { $sort : { "price" : -1 } },
 { $limit : 5 },
 { $project: {neighbourhood:1,_id:0} }
]).pretty();
```

Embedded

```
db.listing_host.aggregate([
 { $sort : { "price" : -1 } },
 { $limit : 5 },
 { $project: {neighbourhood:1,_id:0} }
]).pretty();
```

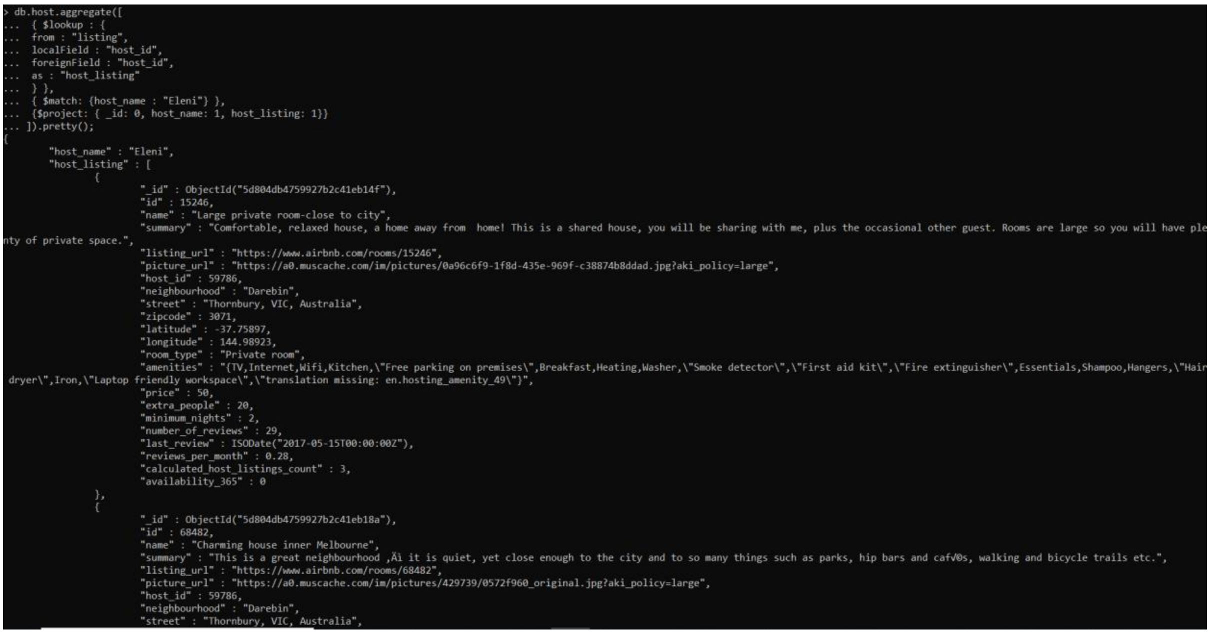


```
> db.listing.aggregate([
...   { $sort : { "price" : -1 } },
...   { $limit : 5 },
...   { $project: {neighbourhood:1,_id:0} }
... ]).pretty();
{ "neighbourhood" : "Melbourne" }
{ "neighbourhood" : "Yarra Ranges" }
{ "neighbourhood" : "Glen Eira" }
{ "neighbourhood" : "Melbourne" }
{ "neighbourhood" : "Port Phillip" }
> db.listing_host.aggregate([
...   { $sort : { "price" : -1 } },
...   { $limit : 5 },
...   { $project: {neighbourhood:1,_id:0} }
... ]).pretty();
{ "neighbourhood" : "Melbourne" }
{ "neighbourhood" : "Yarra Ranges" }
{ "neighbourhood" : "Glen Eira" }
{ "neighbourhood" : "Melbourne" }
{ "neighbourhood" : "Port Phillip" }
```

Q7

Reference

```
db.host.aggregate([
 { $lookup : {
 from : "listing",
 localField : "host_id",
 foreignField : "host_id",
 as : "host_listing"
 } },
 { $match: {host_name : "Eleni"} },
 {$project: { _id: 0, host_name: 1, host_listing: 1}}
]).pretty();
```



```
> db.host.aggregate([
...  { $lookup : {
...  from : "listing",
...  localField : "host_id",
...  foreignField : "host_id",
...  as : "host_listing"
...  } },
...  { $match: {host_name : "Eleni"} },
...  {$project: { _id: 0, host_name: 1, host_listing: 1}}
... ]).pretty();
{
        "host_name" : "Eleni",
        "host_listing" : [
                {
                        "_id" : ObjectId("5d804db4759927b2c41eb14f"),
                        "id" : 15246,
                        "name" : "Large private room-close to city",
                        "summary" : "Comfortable, relaxed house, a home away from  home! This is a shared house, you will be sharing with me, plus the occasional other guest. Rooms are large so you will have ple
nty of private space.",
                        "listing_url" : "https://www.airbnb.com/rooms/15246",
                        "picture_url" : "https://a0.muscache.com/im/pictures/0a96c6f9-1f8d-435e-969f-c38874b8ddad.jpg?aki_policy=large",
                        "host_id" : 59786,
                        "neighbourhood" : "Darebin",
                        "street" : "Thornbury, VIC, Australia",
                        "zipcode" : 3071,
                        "latitude" : -37.75897,
                        "longitude" : 144.98923,
                        "room_type" : "Private room",
                        "amenities" : "{TV,Internet,Wifi,Kitchen,\"Free parking on premises\",Breakfast,Heating,Washer,\"Smoke detector\",\"First aid kit\",\"Fire extinguisher\",Essentials,Shampoo,Hangers,\"Hair
 dryer\",Iron,\"Laptop friendly workspace\",\"translation missing: en.hosting_amenity_49\"}",
                        "price" : 50,
                        "extra_people" : 20,
                        "minimum_nights" : 2,
                        "number_of_reviews" : 29,
                        "last_review" : ISODate("2017-05-15T00:00:00Z"),
                        "reviews_per_month" : 0.28,
                        "calculated_host_listings_count" : 3,
                        "availability_365" : 0
                },
                {
                        "_id" : ObjectId("5d804db4759927b2c41eb18a"),
                        "id" : 68482,
                        "name" : "Charming house inner Melbourne",
                        "summary" : "This is a great neighbourhood ,Äì it is quiet, yet close enough to the city and to so many things such as parks, hip bars and caf√©s, walking and bicycle trails etc.",
                        "listing_url" : "https://www.airbnb.com/rooms/68482",
                        "picture_url" : "https://a0.muscache.com/im/pictures/429739/0572f960_original.jpg?aki_policy=large",
                        "host_id" : 59786,
                        "neighbourhood" : "Darebin",
                        "street" : "Thornbury, VIC, Australia",
```

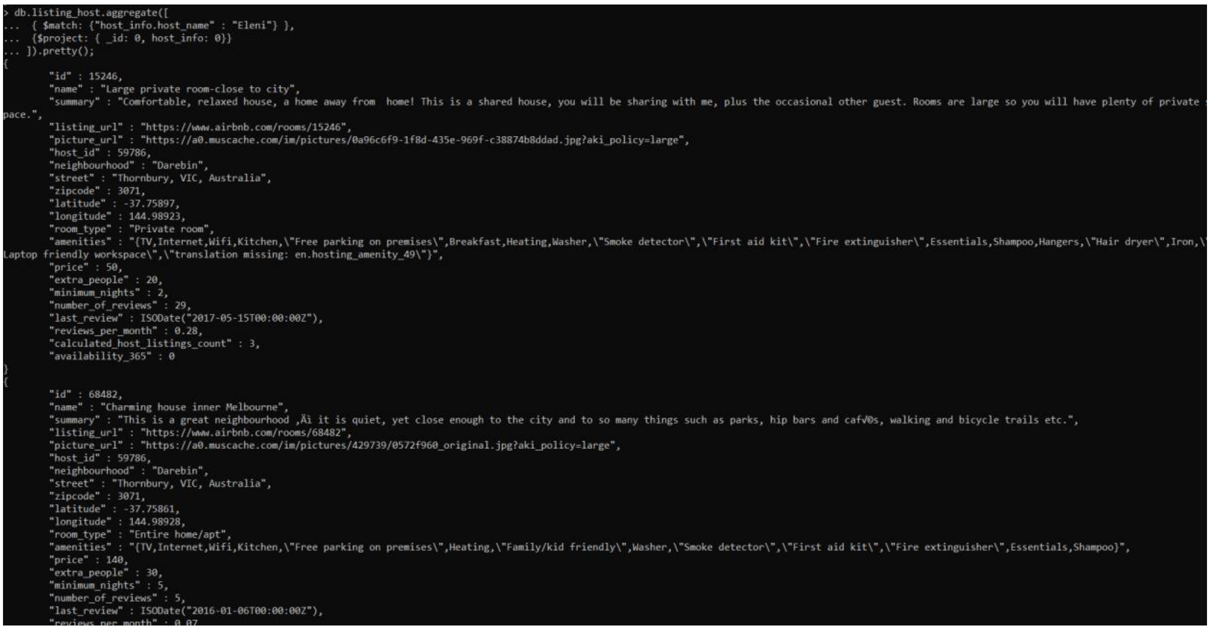
Embedded

db.listing_host.aggregate([

{ $match: {"host_info.host_name" : "Eleni"} },

{$project: { _id: 0, host_info: 0}}

]).pretty();



```
> db.listing_host.aggregate([
...  { $match: {"host_info.host_name" : "Eleni"} },
...  {$project: { _id: 0, host_info: 0}}
... ]).pretty();
{
        "id" : 15246,
        "name" : "Large private room-close to city",
        "summary" : "Comfortable, relaxed house, a home away from  home! This is a shared house, you will be sharing with me, plus the occasional other guest. Rooms are large so you will have plenty of private s
pace.",
        "listing_url" : "https://www.airbnb.com/rooms/15246",
        "picture_url" : "https://a0.muscache.com/im/pictures/0a96c6f9-1f8d-435e-969f-c38874b8ddad.jpg?aki_policy=large",
        "host_id" : 59786,
        "neighbourhood" : "Darebin",
        "street" : "Thornbury, VIC, Australia",
        "zipcode" : 3071,
        "latitude" : -37.75897,
        "longitude" : 144.98923,
        "room_type" : "Private room",
        "amenities" : "{TV,Internet,Wifi,Kitchen,\"Free parking on premises\",Breakfast,Heating,Washer,\"Smoke detector\",\"First aid kit\",\"Fire extinguisher\",Essentials,Shampoo,Hangers,\"Hair dryer\",Iron,\"
Laptop friendly workspace\",\"translation missing: en.hosting_amenity_49\"}",
        "price" : 50,
        "extra_people" : 20,
        "minimum_nights" : 2,
        "number_of_reviews" : 29,
        "last_review" : ISODate("2017-05-15T00:00:00Z"),
        "reviews_per_month" : 0.28,
        "calculated_host_listings_count" : 3,
        "availability_365" : 0
}
{
        "id" : 68482,
        "name" : "Charming house inner Melbourne",
        "summary" : "This is a great neighbourhood ,Äì it is quiet, yet close enough to the city and to so many things such as parks, hip bars and caf√©s, walking and bicycle trails etc.",
        "listing_url" : "https://www.airbnb.com/rooms/68482",
        "picture_url" : "https://a0.muscache.com/im/pictures/429739/0572f960_original.jpg?aki_policy=large",
        "host_id" : 59786,
        "neighbourhood" : "Darebin",
        "street" : "Thornbury, VIC, Australia",
        "zipcode" : 3071,
        "latitude" : -37.75861,
        "longitude" : 144.98928,
        "room_type" : "Entire home/apt",
        "amenities" : "{TV,Internet,Wifi,Kitchen,\"Free parking on premises\",Heating,\"Family/kid friendly\",Washer,\"Smoke detector\",\"First aid kit\",\"Fire extinguisher\",Essentials,Shampoo}",
        "price" : 140,
        "extra_people" : 30,
        "minimum_nights" : 5,
        "number_of_reviews" : 5,
        "last_review" : ISODate("2016-01-06T00:00:00Z"),
        "reviews_per_month" : 0.07
```

Q8

Reference

db.host.aggregate([

{ $lookup : {

from : "listing",

localField : "host_id",

foreignField : "host_id",

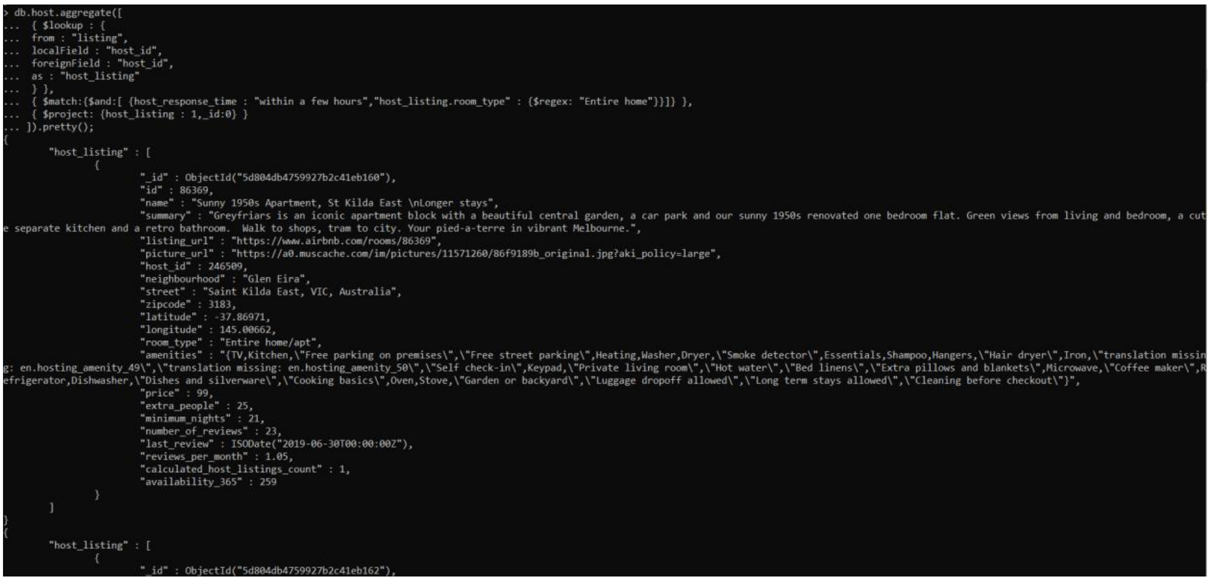
as : "host_listing"

} },

{ $match:{$and:[{host_response_time : "within a few hours","host_listing.room_type" : {$regex: "Entire home"}}]} },

{ $project: {host_listing : 1,_id:0} }

]).pretty();



```
> db.host.aggregate([
...  { $lookup : {
...  from : "listing",
...  localField : "host_id",
...  foreignField : "host_id",
...  as : "host_listing"
...  } },
...  { $match:{$and:[ {host_response_time : "within a few hours","host_listing.room_type" : {$regex: "Entire home"}}]} },
...  { $project: {host_listing : 1,_id:0} }
... ]).pretty();
{
        "host_listing" : [
                {
                        "_id" : ObjectId("5d804db4759927b2c41eb160"),
                        "id" : 86369,
                        "name" : "Sunny 1950s Apartment, St Kilda East \nLonger stays",
                        "summary" : "Greyfriars is an iconic apartment block with a beautiful central garden, a car park and our sunny 1950s renovated one bedroom flat. Green views from living and bedroom, a cut
e separate kitchen and a retro bathroom.  Walk to shops, tram to city. Your pied-a-terre in vibrant Melbourne.",
                        "listing_url" : "https://www.airbnb.com/rooms/86369",
                        "picture_url" : "https://a0.muscache.com/im/pictures/11571260/86f9189b_original.jpg?aki_policy=large",
                        "host_id" : 246509,
                        "neighbourhood" : "Glen Eira",
                        "street" : "Saint Kilda East, VIC, Australia",
                        "zipcode" : 3183,
                        "latitude" : -37.86971,
                        "longitude" : 145.00662,
                        "room_type" : "Entire home/apt",
                        "amenities" : "{TV,Kitchen,\"Free parking on premises\",\"Free street parking\",Heating,Washer,Dryer,\"Smoke detector\",Essentials,Shampoo,Hangers,\"Hair dryer\",Iron,\"translation missin
g: en.hosting_amenity_49\",\"translation missing: en.hosting_amenity_50\",\"Self check-in\",Keypad,\"Private living room\",\"Hot water\",\"Bed linens\",\"Extra pillows and blankets\",Microwave,\"Coffee maker\",R
efrigerator,Dishwasher,\"Dishes and silverware\",\"Cooking basics\",Oven,Stove,\"Garden or backyard\",\"Luggage dropoff allowed\",\"Long term stays allowed\",\"Cleaning before checkout\"}",
                        "price" : 99,
                        "extra_people" : 25,
                        "minimum_nights" : 21,
                        "number_of_reviews" : 23,
                        "last_review" : ISODate("2019-06-30T00:00:00Z"),
                        "reviews_per_month" : 1.05,
                        "calculated_host_listings_count" : 1,
                        "availability_365" : 259
                }
        ]
}
{
        "host_listing" : [
                {
                        "_id" : ObjectId("5d804db4759927b2c41eb162"),
```

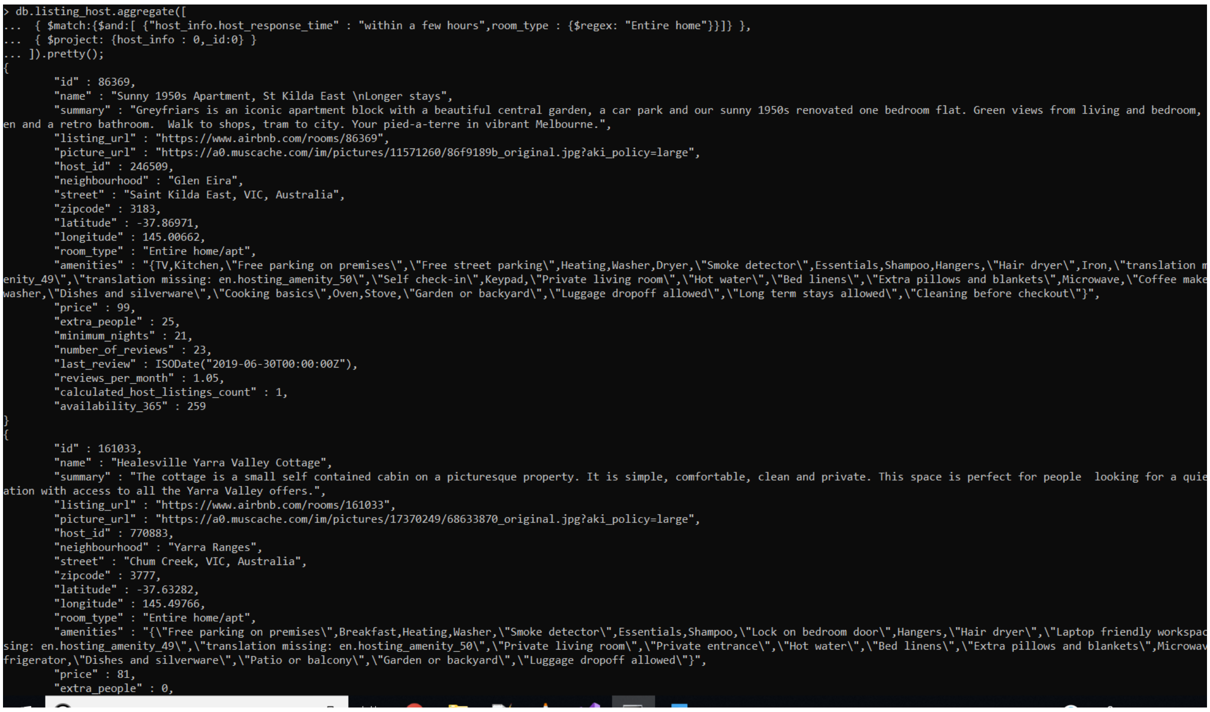
Embedded

db.listing_host.aggregate([

{ $match:{$and:[{"host_info.host_response_time" : "within a few hours",room_type : {$regex: "Entire home"}}]} },

{ $project: {host_info : 0,_id:0} }

]).pretty();



```
> db.listing_host.aggregate([
...  { $match:{$and:[ {"host_info.host_response_time" : "within a few hours",room_type : {$regex: "Entire home"}}]} },
...  { $project: {host_info : 0,_id:0} }
... ]).pretty();
{
        "id" : 86369,
        "name" : "Sunny 1950s Apartment, St Kilda East \nLonger stays",
        "summary" : "Greyfriars is an iconic apartment block with a beautiful central garden, a car park and our sunny 1950s renovated one bedroom flat. Green views from living and bedroom,
en and a retro bathroom.  Walk to shops, tram to city. Your pied-a-terre in vibrant Melbourne.",
        "listing_url" : "https://www.airbnb.com/rooms/86369",
        "picture_url" : "https://a0.muscache.com/im/pictures/11571260/86f9189b_original.jpg?aki_policy=large",
        "host_id" : 246509,
        "neighbourhood" : "Glen Eira",
        "street" : "Saint Kilda East, VIC, Australia",
        "zipcode" : 3183,
        "latitude" : -37.86971,
        "longitude" : 145.00662,
        "room_type" : "Entire home/apt",
        "amenities" : "{TV,Kitchen,\"Free parking on premises\",\"Free street parking\",Heating,Washer,Dryer,\"Smoke detector\",Essentials,Shampoo,Hangers,\"Hair dryer\",Iron,\"translation m
enity_49\",\"translation missing: en.hosting_amenity_50\",\"Self check-in\",Keypad,\"Private living room\",\"Hot water\",\"Bed linens\",\"Extra pillows and blankets\",Microwave,\"Coffee make
washer,\"Dishes and silverware\",\"Cooking basics\",Oven,Stove,\"Garden or backyard\",\"Luggage dropoff allowed\",\"Long term stays allowed\",\"Cleaning before checkout\"}",
        "price" : 99,
        "extra_people" : 25,
        "minimum_nights" : 21,
        "number_of_reviews" : 23,
        "last_review" : ISODate("2019-06-30T00:00:00Z"),
        "reviews_per_month" : 1.05,
        "calculated_host_listings_count" : 1,
        "availability_365" : 259
}
{
        "id" : 161033,
        "name" : "Healesville Yarra Valley Cottage",
        "summary" : "The cottage is a small self contained cabin on a picturesque property. It is simple, comfortable, clean and private. This space is perfect for people  looking for a quie
ation with access to all the Yarra Valley offers.",
        "listing_url" : "https://www.airbnb.com/rooms/161033",
        "picture_url" : "https://a0.muscache.com/im/pictures/17370249/68633870_original.jpg?aki_policy=large",
        "host_id" : 770883,
        "neighbourhood" : "Yarra Ranges",
        "street" : "Chum Creek, VIC, Australia",
        "zipcode" : 3777,
        "latitude" : -37.63282,
        "longitude" : 145.49766,
        "room_type" : "Entire home/apt",
        "amenities" : "{\"Free parking on premises\",Breakfast,Heating,Washer,\"Smoke detector\",Essentials,Shampoo,\"Lock on bedroom door\",Hangers,\"Hair dryer\",\"Laptop friendly workspac
sing: en.hosting_amenity_49\",\"translation missing: en.hosting_amenity_50\",\"Private living room\",\"Private entrance\",\"Hot water\",\"Bed linens\",\"Extra pillows and blankets\",Microwav
frigerator,\"Dishes and silverware\",\"Patio or balcony\",\"Garden or backyard\",\"Luggage dropoff allowed\"}",
        "price" : 81,
        "extra_people" : 0,
```

Q9

Reference

db.host.aggregate([

{ $lookup : {

from : "listing",

localField : "host_id",

foreignField : "host_id",

as : "host_listing"

} },

{ $match: {$and:[{host_name : "Colin", "host_listing.amenities" : {$regex : "Internet"}, "host_listing.amenities" : {$regex : "Gym Access"} }] }},

{ $project: {host_listing : 1,_id:0} }

]).pretty();

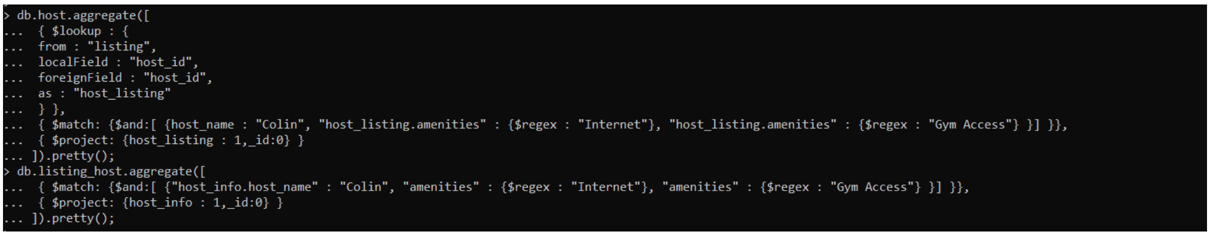
Embedded

db.listing_host.aggregate([

{ $match: {$and:[{"host_info.host_name" : "Colin", "amenities" : {$regex : "Internet"}, "amenities" : {$regex : "Gym Access"} }] }},

{ $project: {host_info : 1,_id:0} }

```
]).pretty();
```



```
> db.host.aggregate([
...  { $lookup : {
...  from : "listing",
...  localField : "host_id",
...  foreignField : "host_id",
...  as : "host_listing"
...  } },
...  { $match: {$and:[ {host_name : "Colin", "host_listing.amenities" : {$regex : "Internet"}, "host_listing.amenities" : {$regex : "Gym Access"} }] }},
...  { $project: {host_listing : 1,_id:0} }
... ]).pretty();
> db.listing_host.aggregate([
...  { $match: {$and:[ {"host_info.host_name" : "Colin", "amenities" : {$regex : "Internet"}, "amenities" : {$regex : "Gym Access"} }] }},
...  { $project: {host_info : 1,_id:0} }
... ]).pretty();
```

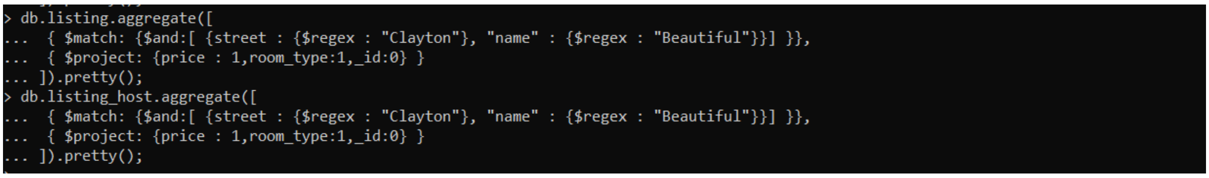
Q10

Reference

```
db.listing.aggregate([
 { $match: {$and:[ {street : {$regex : "Clayton"}, "name" : {$regex : "Beautiful"}}] }},
 { $project: {price : 1,room_type:1,_id:0} }
]).pretty();
```

Embedded

```
db.listing_host.aggregate([
 { $match: {$and:[ {street : {$regex : "Clayton"}, "name" : {$regex : "Beautiful"}}] }},
 { $project: {price : 1,room_type:1,_id:0} }
]).pretty();
```

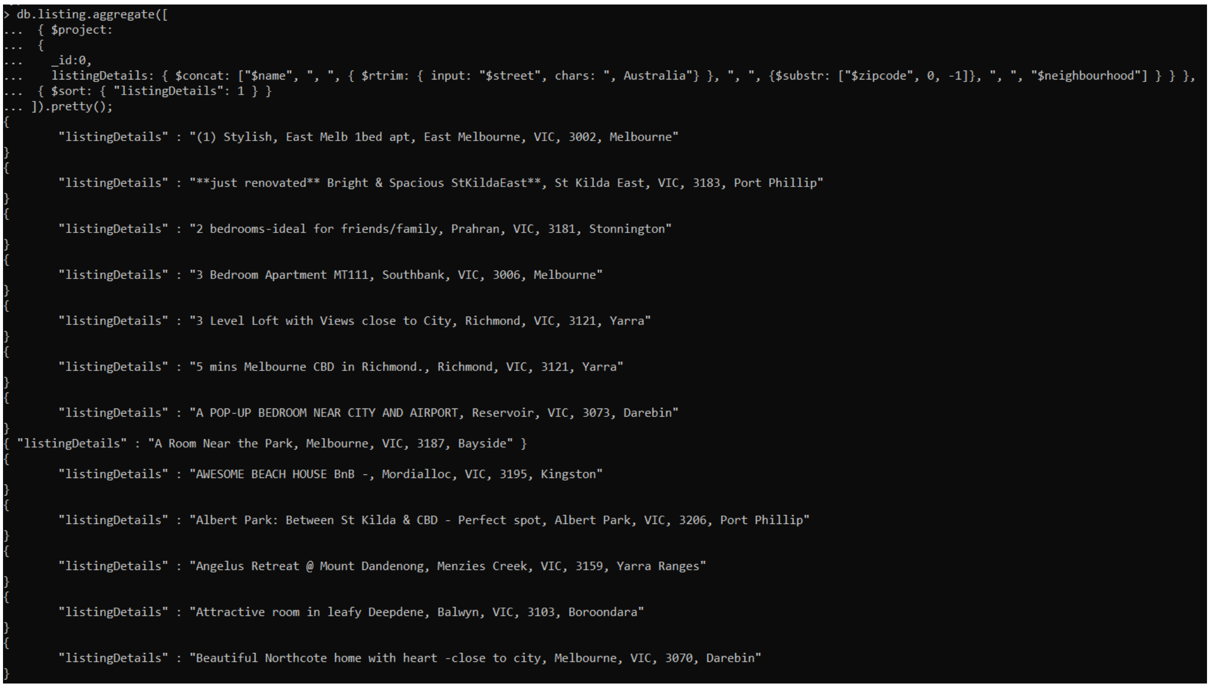


```
> db.listing.aggregate([
...  { $match: {$and:[ {street : {$regex : "Clayton"}, "name" : {$regex : "Beautiful"}}] }},
...  { $project: {price : 1,room_type:1,_id:0} }
... ]).pretty();
> db.listing_host.aggregate([
...  { $match: {$and:[ {street : {$regex : "Clayton"}, "name" : {$regex : "Beautiful"}}] }},
...  { $project: {price : 1,room_type:1,_id:0} }
... ]).pretty();
```

Q11

Reference

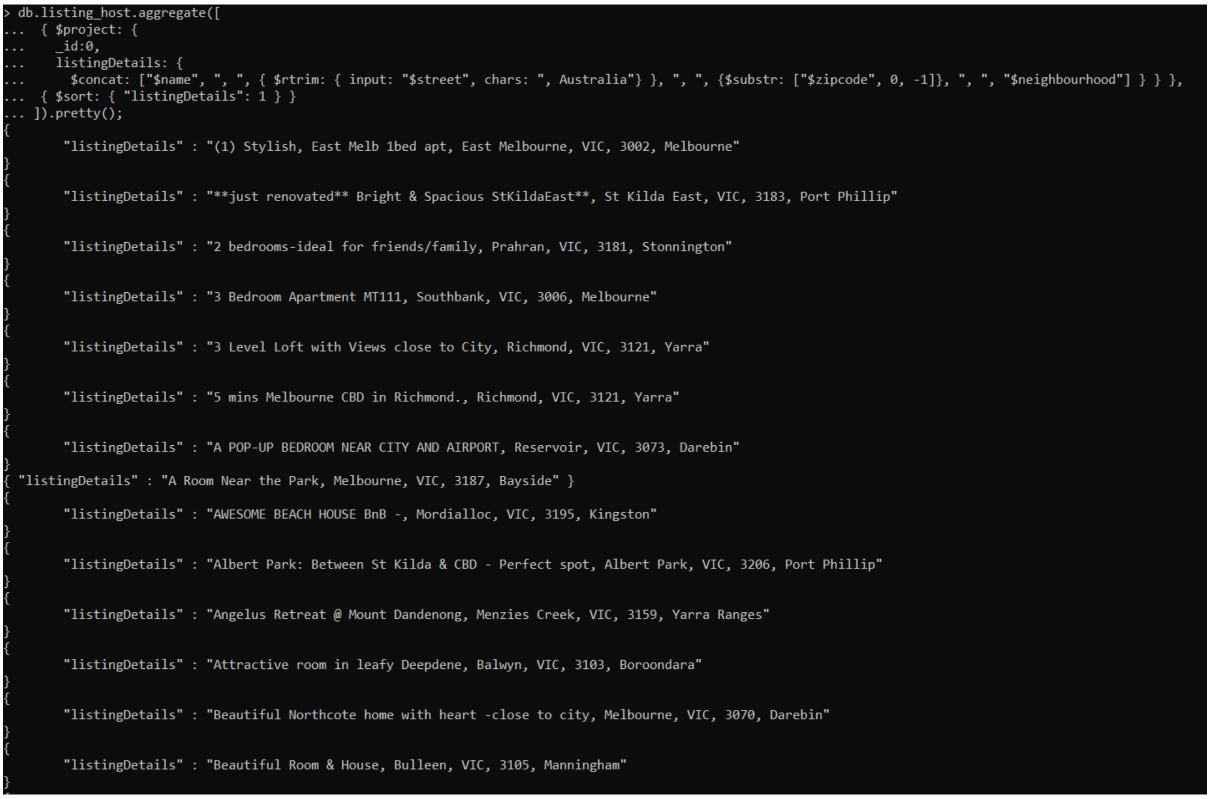
```
db.listing.aggregate([
 { $project:
 {
   _id:0,
   listingDetails: { $concat: ["$name", ", ", { $rtrim: { input: "$street", chars: ", Australia"} }, ", ", {$substr: ["$zipcode", 0, -1]}, ", ", "$neighbourhood"] } } },
 { $sort: { "listingDetails": 1 } }
]).pretty();
```



```
> db.listing.aggregate([
...   { $project:
...   {
...     _id:0,
...     listingDetails: { $concat: ["$name", ", ", { $rtrim: { input: "$street", chars: ", Australia"} }, ", ", {$substr: ["$zipcode", 0, -1]}, ", ", "$neighbourhood"] } } },
...   { $sort: { "listingDetails": 1 } }
... ]).pretty();
{
        "listingDetails" : "(1) Stylish, East Melb 1bed apt, East Melbourne, VIC, 3002, Melbourne"
}
{
        "listingDetails" : "**just renovated** Bright & Spacious StKildaEast**, St Kilda East, VIC, 3183, Port Phillip"
}
{
        "listingDetails" : "2 bedrooms-ideal for friends/family, Prahran, VIC, 3181, Stonnington"
}
{
        "listingDetails" : "3 Bedroom Apartment MT111, Southbank, VIC, 3006, Melbourne"
}
{
        "listingDetails" : "3 Level Loft with Views close to City, Richmond, VIC, 3121, Yarra"
}
{
        "listingDetails" : "5 mins Melbourne CBD in Richmond., Richmond, VIC, 3121, Yarra"
}
{
        "listingDetails" : "A POP-UP BEDROOM NEAR CITY AND AIRPORT, Reservoir, VIC, 3073, Darebin"
}
{ "listingDetails" : "A Room Near the Park, Melbourne, VIC, 3187, Bayside" }
{
        "listingDetails" : "AWESOME BEACH HOUSE BnB -, Mordialloc, VIC, 3195, Kingston"
}
{
        "listingDetails" : "Albert Park: Between St Kilda & CBD - Perfect spot, Albert Park, VIC, 3206, Port Phillip"
}
{
        "listingDetails" : "Angelus Retreat @ Mount Dandenong, Menzies Creek, VIC, 3159, Yarra Ranges"
}
{
        "listingDetails" : "Attractive room in leafy Deepdene, Balwyn, VIC, 3103, Boroondara"
}
{
        "listingDetails" : "Beautiful Northcote home with heart -close to city, Melbourne, VIC, 3070, Darebin"
}
```

Embedded

```
db.listing_host.aggregate([
 { $project:
 {
   _id:0,
   listingDetails: { $concat: ["$name", ", ", { $rtrim: { input: "$street", chars: ", Australia"} }, ", ", {$substr: ["$zipcode", 0, -1]}, ", ", "$neighbourhood"] } } },
 { $sort: { "listingDetails": 1 } }
]).pretty();
```



```
> db.listing_host.aggregate([
...  { $project: {
...    _id:0,
...    listingDetails: {
...      $concat: ["$name", ", ", { $rtrim: { input: "$street", chars: ", Australia"} }, ", ", {$substr: ["$zipcode", 0, -1]}, ", ", "$neighbourhood"] } } },
...  { $sort: { "listingDetails": 1 } }
... ]).pretty();
{

        "listingDetails" : "(1) Stylish, East Melb 1bed apt, East Melbourne, VIC, 3002, Melbourne"
}
{

        "listingDetails" : "**just renovated** Bright & Spacious StKildaEast**, St Kilda East, VIC, 3183, Port Phillip"
}
{

        "listingDetails" : "2 bedrooms-ideal for friends/family, Prahran, VIC, 3181, Stonnington"
}
{

        "listingDetails" : "3 Bedroom Apartment MT111, Southbank, VIC, 3006, Melbourne"
}
{

        "listingDetails" : "3 Level Loft with Views close to City, Richmond, VIC, 3121, Yarra"
}
{

        "listingDetails" : "5 mins Melbourne CBD in Richmond., Richmond, VIC, 3121, Yarra"
}
{

        "listingDetails" : "A POP-UP BEDROOM NEAR CITY AND AIRPORT, Reservoir, VIC, 3073, Darebin"
}
{ "listingDetails" : "A Room Near the Park, Melbourne, VIC, 3187, Bayside" }
{
        "listingDetails" : "AWESOME BEACH HOUSE BnB -, Mordialloc, VIC, 3195, Kingston"
}
{

        "listingDetails" : "Albert Park: Between St Kilda & CBD - Perfect spot, Albert Park, VIC, 3206, Port Phillip"
}
{

        "listingDetails" : "Angelus Retreat @ Mount Dandenong, Menzies Creek, VIC, 3159, Yarra Ranges"
}
{

        "listingDetails" : "Attractive room in leafy Deepdene, Balwyn, VIC, 3103, Boroondara"
}
{

        "listingDetails" : "Beautiful Northcote home with heart -close to city, Melbourne, VIC, 3070, Darebin"
}
{

        "listingDetails" : "Beautiful Room & House, Bulleen, VIC, 3105, Manningham"
}
```

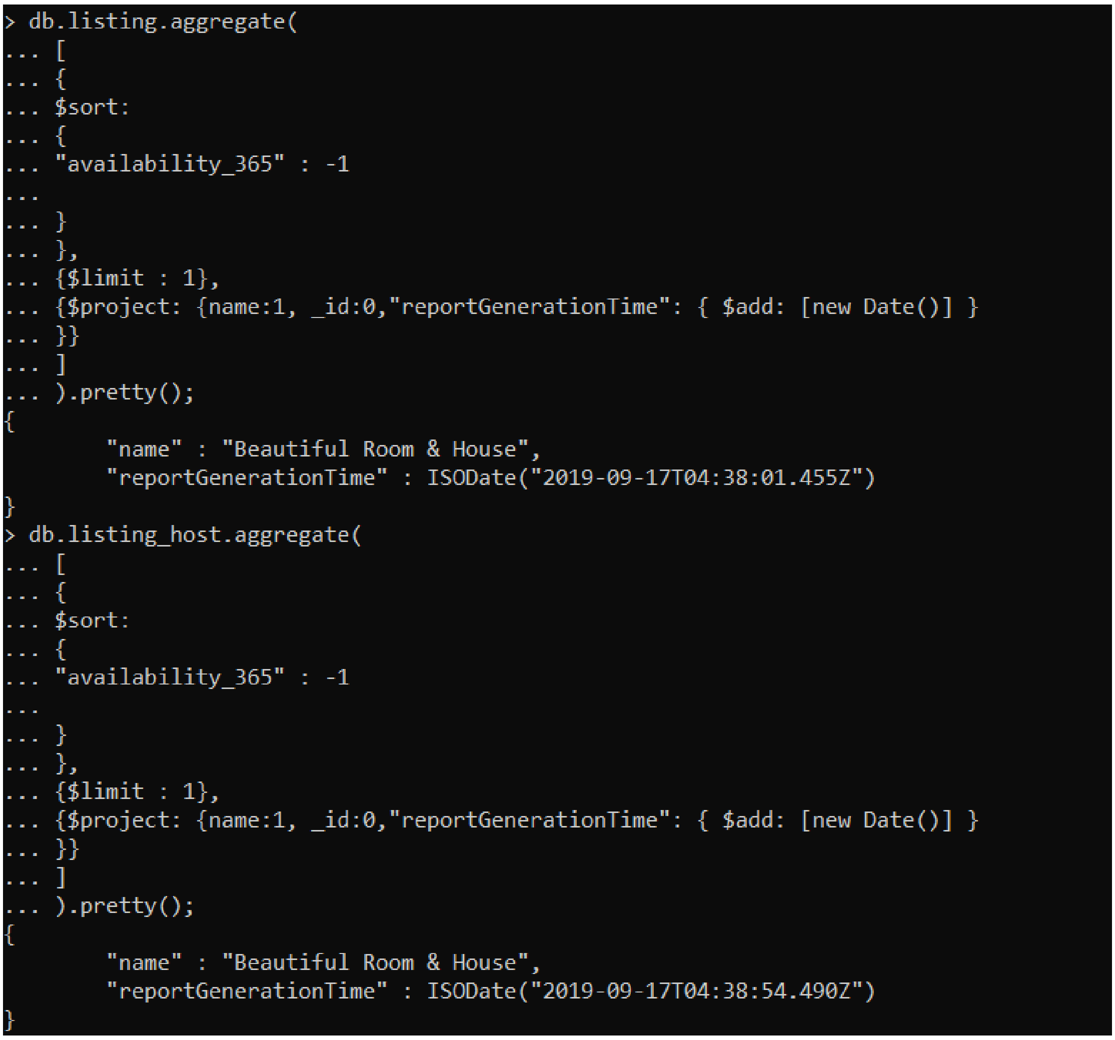
Q12

Reference

```
db.listing.aggregate(
        [
                {
                        $sort:
                        {
                                "availability_365" : -1


                        }
                },
                {$limit : 1},
                {$project: {name:1, _id:0,"reportGenerationTime": { $add: [new Date()] }
                }}
        ]
).pretty();
```

Embedded

```
db.listing_host.aggregate(
    [
        {
            $sort:
            {
                "availability_365" : -1

            }
        },
        {$limit : 1},
        {$project: {name:1, _id:0,"reportGenerationTime": { $add: [new Date()] }
        }}
    ]
).pretty();
```



```
> db.listing.aggregate(
... [
... {
... $sort:
... {
... "availability_365" : -1
...
... }
... },
... {$limit : 1},
... {$project: {name:1, _id:0,"reportGenerationTime": { $add: [new Date()] }
... }}
... ]
... ).pretty();
{
        "name" : "Beautiful Room & House",
        "reportGenerationTime" : ISODate("2019-09-17T04:38:01.455Z")
}
> db.listing_host.aggregate(
... [
... {
... $sort:
... {
... "availability_365" : -1
...
... }
... },
... {$limit : 1},
... {$project: {name:1, _id:0,"reportGenerationTime": { $add: [new Date()] }
... }}
... ]
... ).pretty();
{
        "name" : "Beautiful Room & House",
        "reportGenerationTime" : ISODate("2019-09-17T04:38:54.490Z")
}
```

Q13

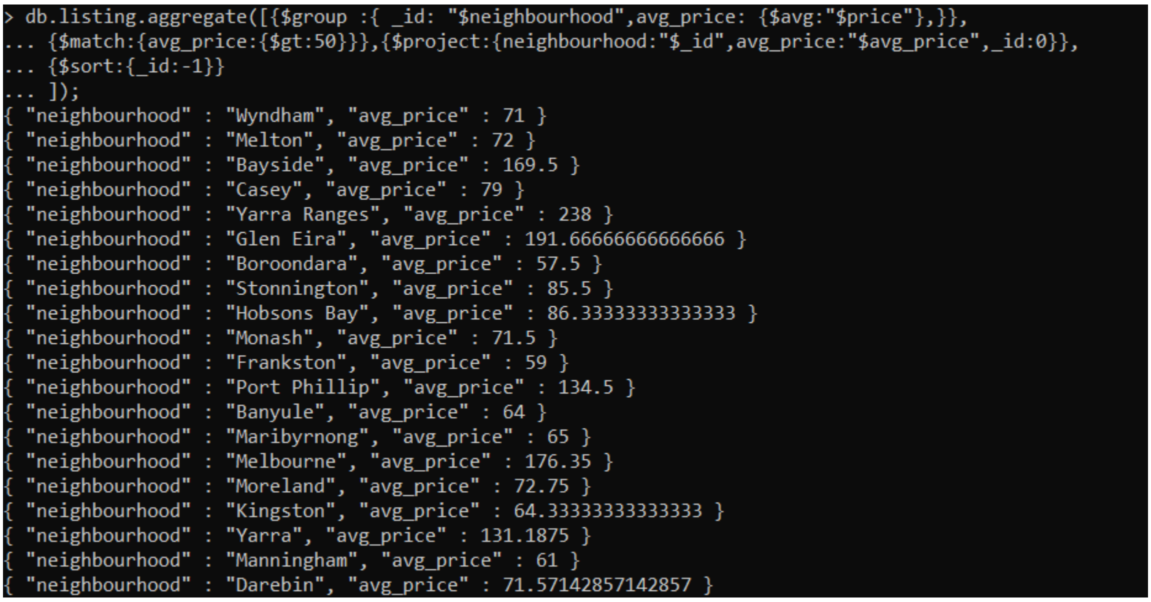
Reference

db.listing.aggregate([{$group :{ _id: "$neighbourhood",avg_price: {$avg:"$price"},}},

{$match:{avg_price:{$gt:50}}},{$project:{neighbourhood:"$_id",avg_price:"$avg_price",_id:0}},

{$sort:{_id:-1}}

]);



```
> db.listing.aggregate([{$group :{ _id: "$neighbourhood",avg_price: {$avg:"$price"},}},
... {$match:{avg_price:{$gt:50}}},{$project:{neighbourhood:"$_id",avg_price:"$avg_price",_id:0}},
... {$sort:{_id:-1}}
... ]);
{ "neighbourhood" : "Wyndham", "avg_price" : 71 }
{ "neighbourhood" : "Melton", "avg_price" : 72 }
{ "neighbourhood" : "Bayside", "avg_price" : 169.5 }
{ "neighbourhood" : "Casey", "avg_price" : 79 }
{ "neighbourhood" : "Yarra Ranges", "avg_price" : 238 }
{ "neighbourhood" : "Glen Eira", "avg_price" : 191.66666666666666 }
{ "neighbourhood" : "Boroondara", "avg_price" : 57.5 }
{ "neighbourhood" : "Stonnington", "avg_price" : 85.5 }
{ "neighbourhood" : "Hobsons Bay", "avg_price" : 86.33333333333333 }
{ "neighbourhood" : "Monash", "avg_price" : 71.5 }
{ "neighbourhood" : "Frankston", "avg_price" : 59 }
{ "neighbourhood" : "Port Phillip", "avg_price" : 134.5 }
{ "neighbourhood" : "Banyule", "avg_price" : 64 }
{ "neighbourhood" : "Maribyrnong", "avg_price" : 65 }
{ "neighbourhood" : "Melbourne", "avg_price" : 176.35 }
{ "neighbourhood" : "Moreland", "avg_price" : 72.75 }
{ "neighbourhood" : "Kingston", "avg_price" : 64.33333333333333 }
{ "neighbourhood" : "Yarra", "avg_price" : 131.1875 }
{ "neighbourhood" : "Manningham", "avg_price" : 61 }
{ "neighbourhood" : "Darebin", "avg_price" : 71.57142857142857 }
```

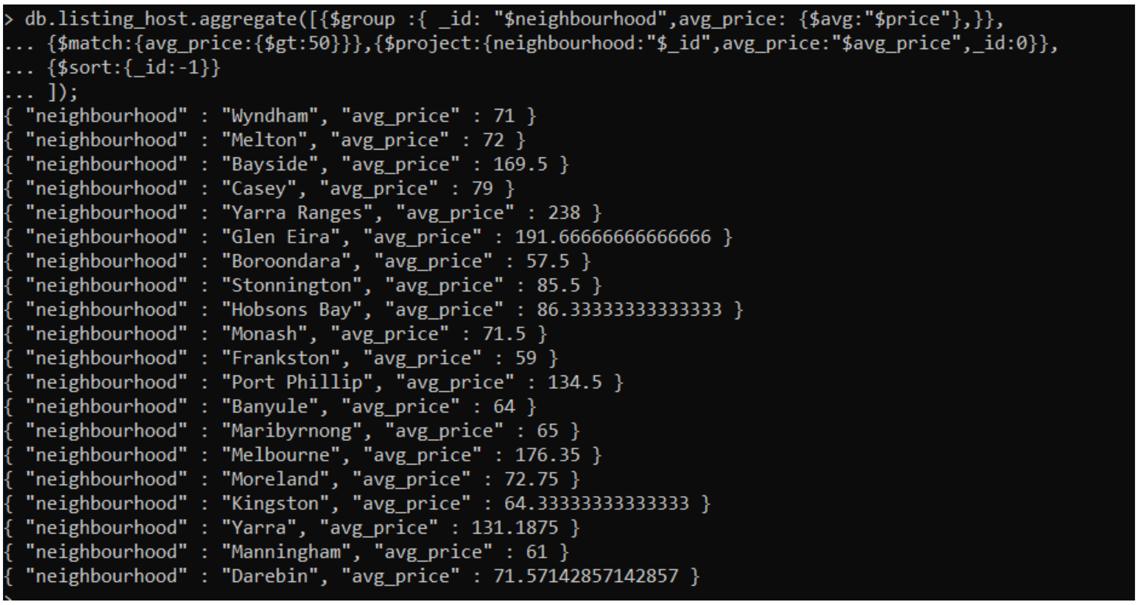
Embedded

db.listing_host.aggregate([{$group :{ _id: "$neighbourhood",avg_price: {$avg:"$price"},}},

{$match:{avg_price:{$gt:50}}},{$project:{neighbourhood:"$_id",avg_price:"$avg_price",_id:0}},

{$sort:{_id:-1}}

]);



```
> db.listing_host.aggregate([{$group :{ _id: "$neighbourhood",avg_price: {$avg:"$price"},}},
... {$match:{avg_price:{$gt:50}}},{$project:{neighbourhood:"$_id",avg_price:"$avg_price",_id:0}},
... {$sort:{_id:-1}}
... ]);
{ "neighbourhood" : "Wyndham", "avg_price" : 71 }
{ "neighbourhood" : "Melton", "avg_price" : 72 }
{ "neighbourhood" : "Bayside", "avg_price" : 169.5 }
{ "neighbourhood" : "Casey", "avg_price" : 79 }
{ "neighbourhood" : "Yarra Ranges", "avg_price" : 238 }
{ "neighbourhood" : "Glen Eira", "avg_price" : 191.66666666666666 }
{ "neighbourhood" : "Boroondara", "avg_price" : 57.5 }
{ "neighbourhood" : "Stonnington", "avg_price" : 85.5 }
{ "neighbourhood" : "Hobsons Bay", "avg_price" : 86.33333333333333 }
{ "neighbourhood" : "Monash", "avg_price" : 71.5 }
{ "neighbourhood" : "Frankston", "avg_price" : 59 }
{ "neighbourhood" : "Port Phillip", "avg_price" : 134.5 }
{ "neighbourhood" : "Banyule", "avg_price" : 64 }
{ "neighbourhood" : "Maribyrnong", "avg_price" : 65 }
{ "neighbourhood" : "Melbourne", "avg_price" : 176.35 }
{ "neighbourhood" : "Moreland", "avg_price" : 72.75 }
{ "neighbourhood" : "Kingston", "avg_price" : 64.33333333333333 }
{ "neighbourhood" : "Yarra", "avg_price" : 131.1875 }
{ "neighbourhood" : "Manningham", "avg_price" : 61 }
{ "neighbourhood" : "Darebin", "avg_price" : 71.57142857142857 }
```

Q14

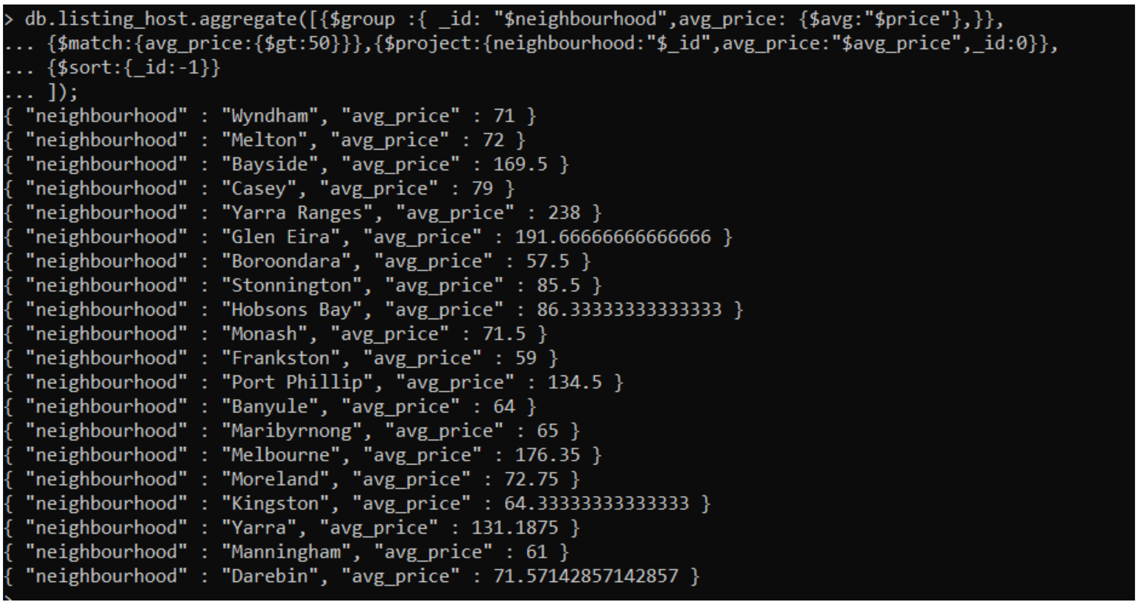
Reference

db.host.aggregate([

{$project :{_id : 0, host_id: 1, host_name: 1,"number of verification methods" : { $size: "$host_verifications" }}},

{$sort:{"number of verification methods":-1}}

]);



```
> db.listing_host.aggregate([{$group :{ _id: "$neighbourhood",avg_price: {$avg:"$price"},}},
... {$match:{avg_price:{$gt:50}}},{$project:{neighbourhood:"$_id",avg_price:"$avg_price",_id:0}},
... {$sort:{_id:-1}}
... ]);
{ "neighbourhood" : "Wyndham", "avg_price" : 71 }
{ "neighbourhood" : "Melton", "avg_price" : 72 }
{ "neighbourhood" : "Bayside", "avg_price" : 169.5 }
{ "neighbourhood" : "Casey", "avg_price" : 79 }
{ "neighbourhood" : "Yarra Ranges", "avg_price" : 238 }
{ "neighbourhood" : "Glen Eira", "avg_price" : 191.66666666666666 }
{ "neighbourhood" : "Boroondara", "avg_price" : 57.5 }
{ "neighbourhood" : "Stonnington", "avg_price" : 85.5 }
{ "neighbourhood" : "Hobsons Bay", "avg_price" : 86.33333333333333 }
{ "neighbourhood" : "Monash", "avg_price" : 71.5 }
{ "neighbourhood" : "Frankston", "avg_price" : 59 }
{ "neighbourhood" : "Port Phillip", "avg_price" : 134.5 }
{ "neighbourhood" : "Banyule", "avg_price" : 64 }
{ "neighbourhood" : "Maribyrnong", "avg_price" : 65 }
{ "neighbourhood" : "Melbourne", "avg_price" : 176.35 }
{ "neighbourhood" : "Moreland", "avg_price" : 72.75 }
{ "neighbourhood" : "Kingston", "avg_price" : 64.33333333333333 }
{ "neighbourhood" : "Yarra", "avg_price" : 131.1875 }
{ "neighbourhood" : "Manningham", "avg_price" : 61 }
{ "neighbourhood" : "Darebin", "avg_price" : 71.57142857142857 }
```

Embedded

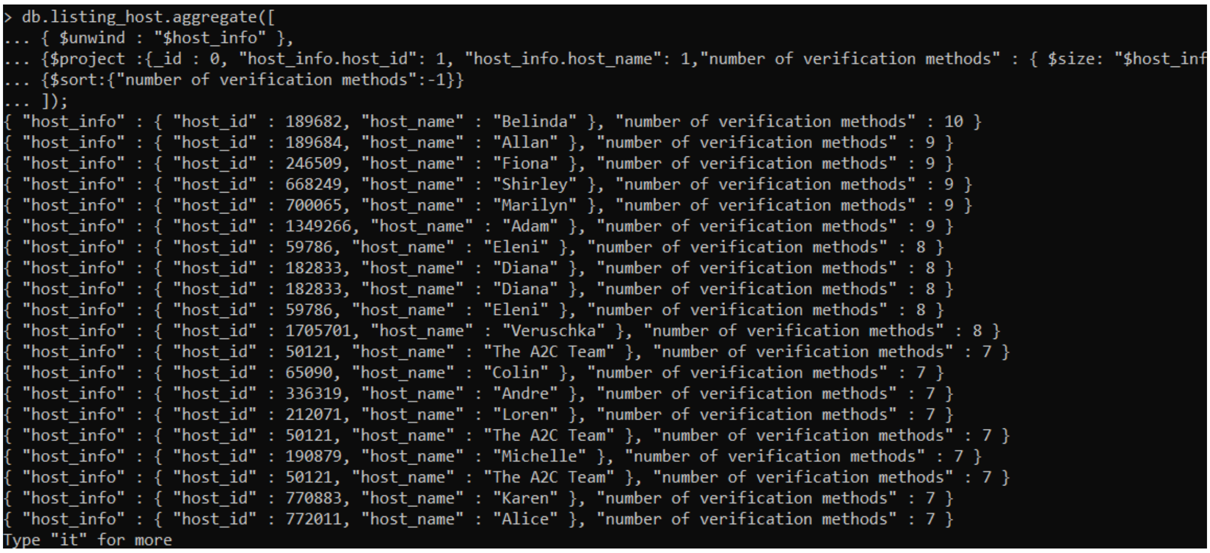
db.listing_host.aggregate([

{ $unwind : "$host_info" },

{$project :{_id : 0, "host_info.host_id": 1, "host_info.host_name": 1,"number of verification methods" : { $size: "$host_info.host_verifications" }}},

{$sort:{"number of verification methods":-1}}

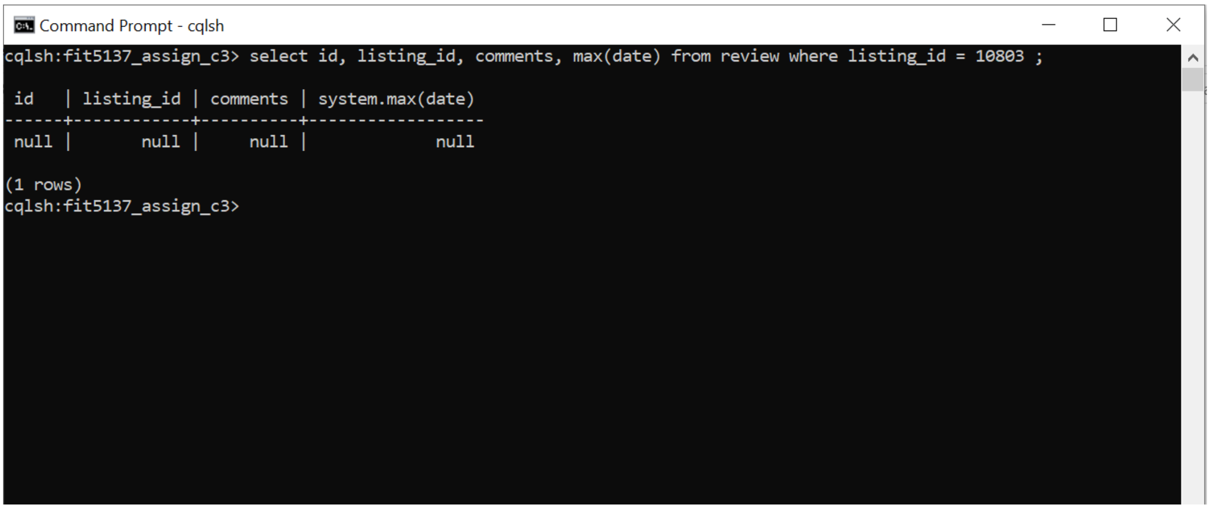
]);



```
> db.listing_host.aggregate([
... { $unwind : "$host_info" },
... {$project :{_id : 0, "host_info.host_id": 1, "host_info.host_name": 1,"number of verification methods" : { $size: "$host_inf
... {$sort:{"number of verification methods":-1}}
... ]);
{ "host_info" : { "host_id" : 189682, "host_name" : "Belinda" }, "number of verification methods" : 10 }
{ "host_info" : { "host_id" : 189684, "host_name" : "Allan" }, "number of verification methods" : 9 }
{ "host_info" : { "host_id" : 246509, "host_name" : "Fiona" }, "number of verification methods" : 9 }
{ "host_info" : { "host_id" : 668249, "host_name" : "Shirley" }, "number of verification methods" : 9 }
{ "host_info" : { "host_id" : 700065, "host_name" : "Marilyn" }, "number of verification methods" : 9 }
{ "host_info" : { "host_id" : 1349266, "host_name" : "Adam" }, "number of verification methods" : 9 }
{ "host_info" : { "host_id" : 59786, "host_name" : "Eleni" }, "number of verification methods" : 8 }
{ "host_info" : { "host_id" : 182833, "host_name" : "Diana" }, "number of verification methods" : 8 }
{ "host_info" : { "host_id" : 182833, "host_name" : "Diana" }, "number of verification methods" : 8 }
{ "host_info" : { "host_id" : 59786, "host_name" : "Eleni" }, "number of verification methods" : 8 }
{ "host_info" : { "host_id" : 1705701, "host_name" : "Veruschka" }, "number of verification methods" : 8 }
{ "host_info" : { "host_id" : 50121, "host_name" : "The A2C Team" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 65090, "host_name" : "Colin" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 336319, "host_name" : "Andre" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 212071, "host_name" : "Loren" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 50121, "host_name" : "The A2C Team" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 190879, "host_name" : "Michelle" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 50121, "host_name" : "The A2C Team" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 770883, "host_name" : "Karen" }, "number of verification methods" : 7 }
{ "host_info" : { "host_id" : 772011, "host_name" : "Alice" }, "number of verification methods" : 7 }
Type "it" for more
```

Q15

select id, listing_id, comments, max(date) from review where listing_id = 10803 ;

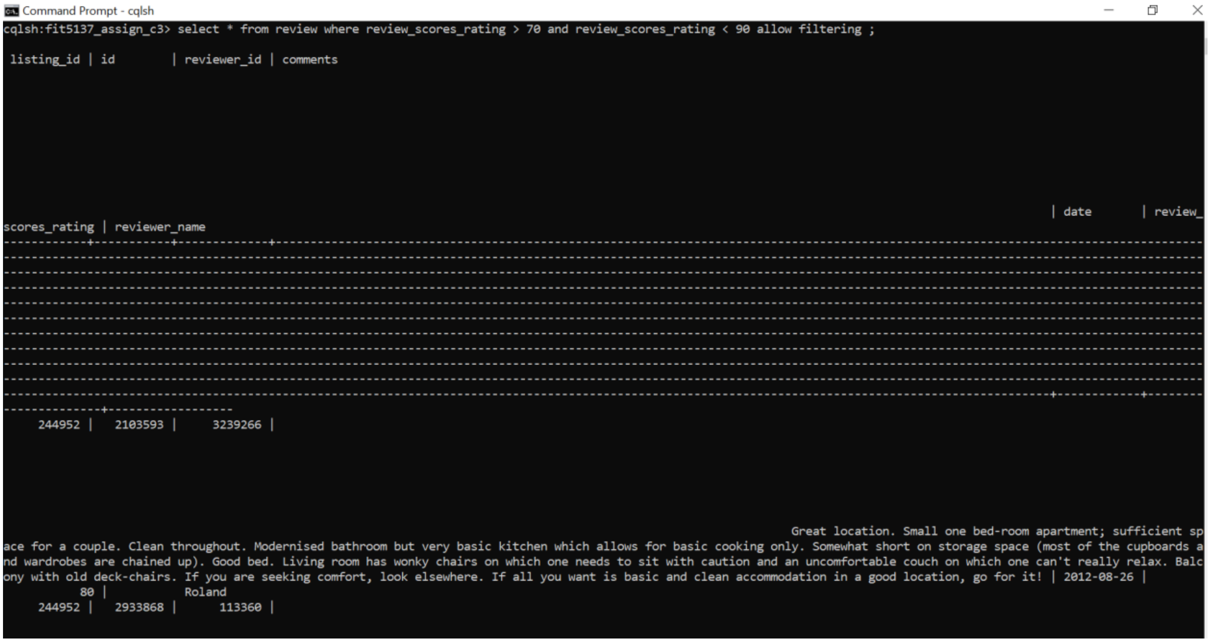


```
cqlsh:fit5137_assign_c3> select id, listing_id, comments, max(date) from review where listing_id = 10803 ;

 id   | listing_id | comments | system.max(date)
------+------------+----------+------------------
 null |       null |     null |             null

(1 rows)
cqlsh:fit5137_assign_c3>
```

//Windows OS

Q16

select * from review where review_scores_rating > 70 and review_scores_rating < 90 allow filtering ;

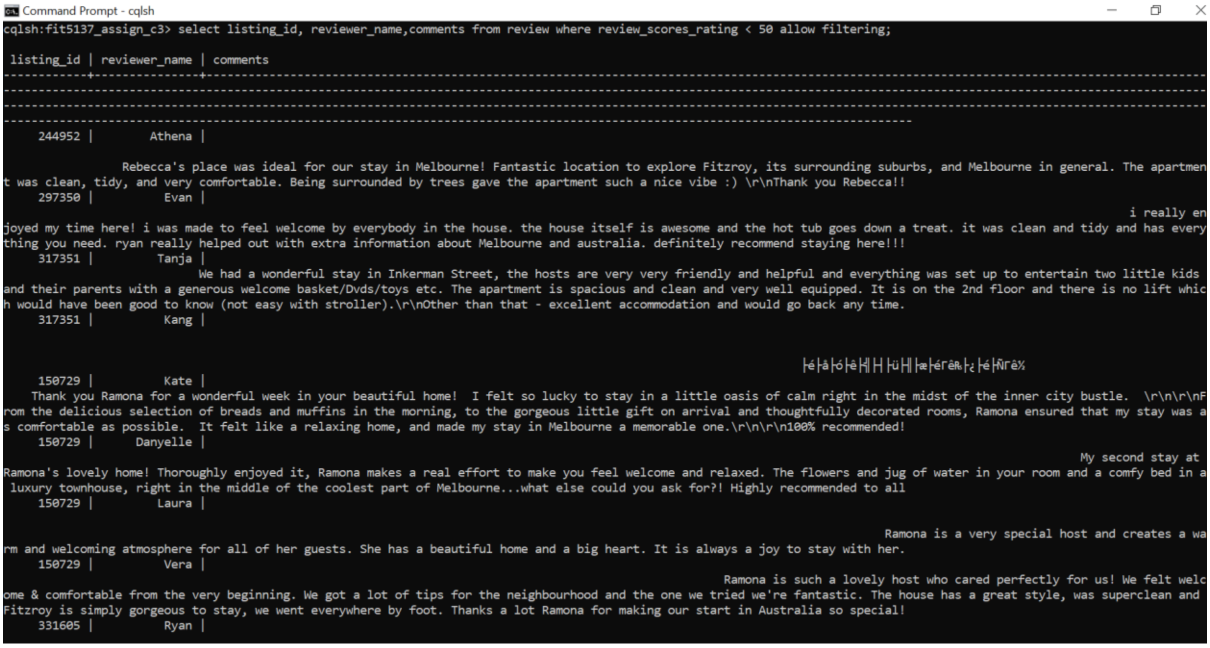


Command Prompt - cqlsh

```
cqlsh:fit5137_assign_c3> select * from review where review_scores_rating > 70 and review_scores_rating < 90 allow filtering ;

 listing_id | id       | reviewer_id | comments







                                                                                                                                    | date       | review_
scores_rating | reviewer_name
------------+----------+-------------+----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------+------------+------
--------------+-----------------
     244952 |   2103593 |     3239266 |






                                                                                          Great location. Small one bed-room apartment; sufficient sp
ace for a couple. Clean throughout. Modernised bathroom but very basic kitchen which allows for basic cooking only. Somewhat short on storage space (most of the cupboards a
nd wardrobes are chained up). Good bed. Living room has wonky chairs on which one needs to sit with caution and an uncomfortable couch on which one can't really relax. Balc
ony with old deck-chairs. If you are seeking comfort, look elsewhere. If all you want is basic and clean accommodation in a good location, go for it! | 2012-08-26 |
          80 |          Roland
     244952 |   2933868 |      113360 |
```

Q17

select listing_id, reviewer_name,comments from review where review_scores_rating < 50 allow filtering;

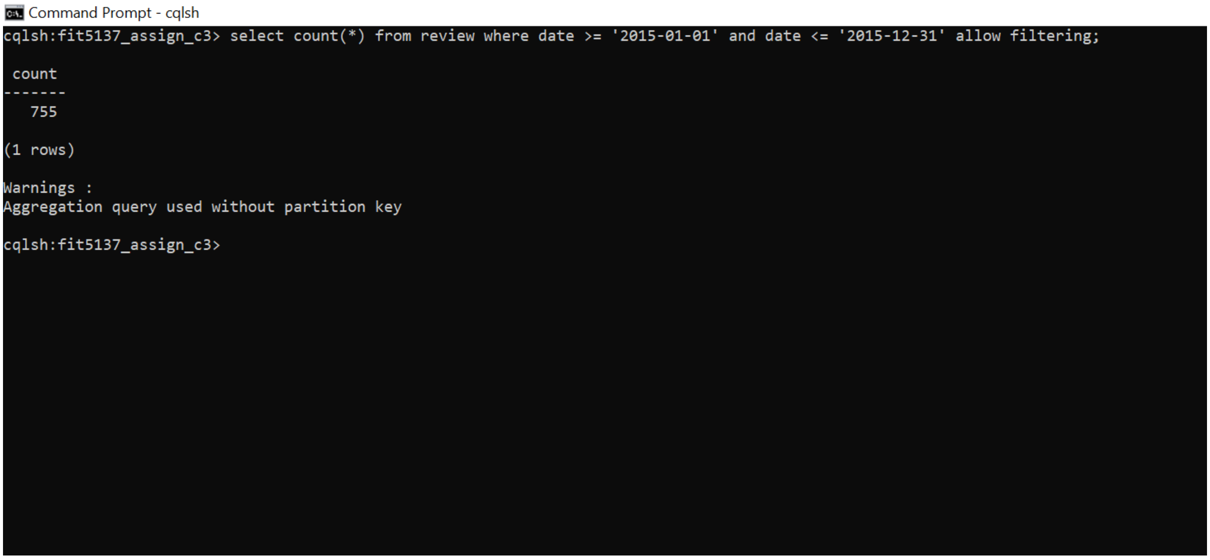


Command Prompt - cqlsh

```
cqlsh:fit5137_assign_c3> select listing_id, reviewer_name,comments from review where review_scores_rating < 50 allow filtering;

 listing_id | reviewer_name | comments
------------+---------------+--------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------------------------------
     244952 |        Athena |

               Rebecca's place was ideal for our stay in Melbourne! Fantastic location to explore Fitzroy, its surrounding suburbs, and Melbourne in general. The apartmen
t was clean, tidy, and very comfortable. Being surrounded by trees gave the apartment such a nice vibe :) \r\nThank you Rebecca!!
     297350 |          Evan |
                                                                                                                                                     i really en
joyed my time here! i was made to feel welcome by everybody in the house. the house itself is awesome and the hot tub goes down a treat. it was clean and tidy and has every
thing you need. ryan really helped out with extra information about Melbourne and australia. definitely recommend staying here!!!
     317351 |         Tanja |
                         We had a wonderful stay in Inkerman Street, the hosts are very very friendly and helpful and everything was set up to entertain two little kids
and their parents with a generous welcome basket/Dvds/toys etc. The apartment is spacious and clean and very well equipped. It is on the 2nd floor and there is no lift whic
h would have been good to know (not easy with stroller).\r\nOther than that - excellent accommodation and would go back any time.
     317351 |          Kang |


                                                                                                  ├é├á├ó├ê├┤├┤├ü├┤├æ├ér├ê₨├¿├é├Ñr├ê%
     150729 |          Kate |
    Thank you Ramona for a wonderful week in your beautiful home!  I felt so lucky to stay in a little oasis of calm right in the midst of the inner city bustle.  \r\n\r\nF
rom the delicious selection of breads and muffins in the morning, to the gorgeous little gift on arrival and thoughtfully decorated rooms, Ramona ensured that my stay was a
s comfortable as possible.  It felt like a relaxing home, and made my stay in Melbourne a memorable one.\r\n\r\n100% recommended!
     150729 |      Danyelle |
                                                                                                                                                   My second stay at
Ramona's lovely home! Thoroughly enjoyed it, Ramona makes a real effort to make you feel welcome and relaxed. The flowers and jug of water in your room and a comfy bed in a
 luxury townhouse, right in the middle of the coolest part of Melbourne...what else could you ask for?! Highly recommended to all
     150729 |         Laura |

                                                                                                            Ramona is a very special host and creates a wa
rm and welcoming atmosphere for all of her guests. She has a beautiful home and a big heart. It is always a joy to stay with her.
     150729 |          Vera |
                                                                                         Ramona is such a lovely host who cared perfectly for us! We felt welc
ome & comfortable from the very beginning. We got a lot of tips for the neighbourhood and the one we tried we're fantastic. The house has a great style, was superclean and
Fitzroy is simply gorgeous to stay, we went everywhere by foot. Thanks a lot Ramona for making our start in Australia so special!
     331605 |          Ryan |
```

Q18

select count(*) from review where date >= '2015-01-01' and date <= '2015-12-31' allow filtering;



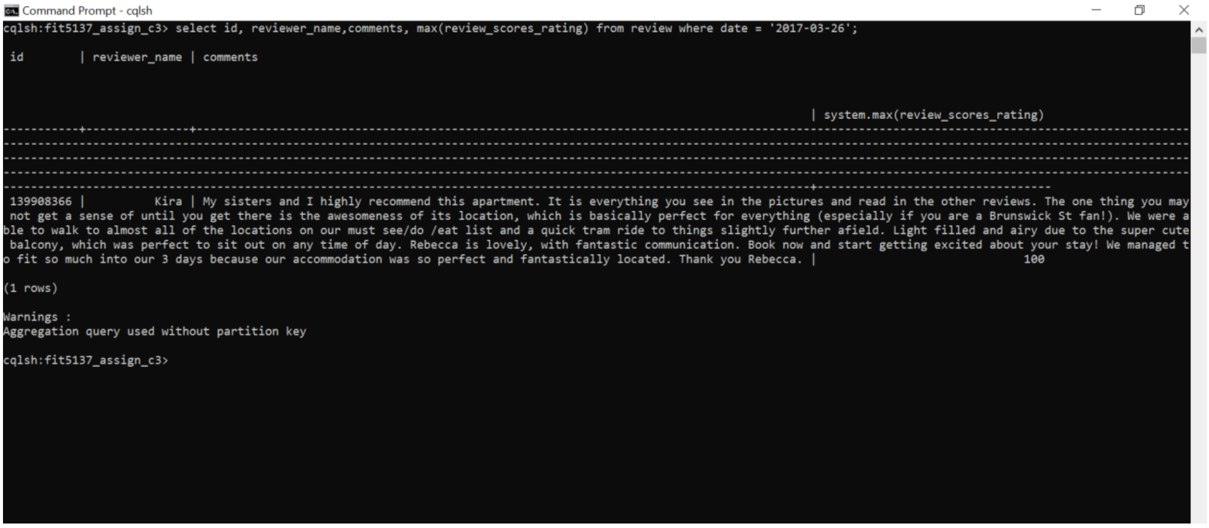
Command Prompt - cqlsh

```
cqlsh:fit5137_assign_c3> select count(*) from review where date >= '2015-01-01' and date <= '2015-12-31' allow filtering;

 count
-------
   755

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:fit5137_assign_c3>
```

Q19

create index on review(date);

select id, reviewer_name,comments, max(review_scores_rating) from review where date = '2017-03-26';



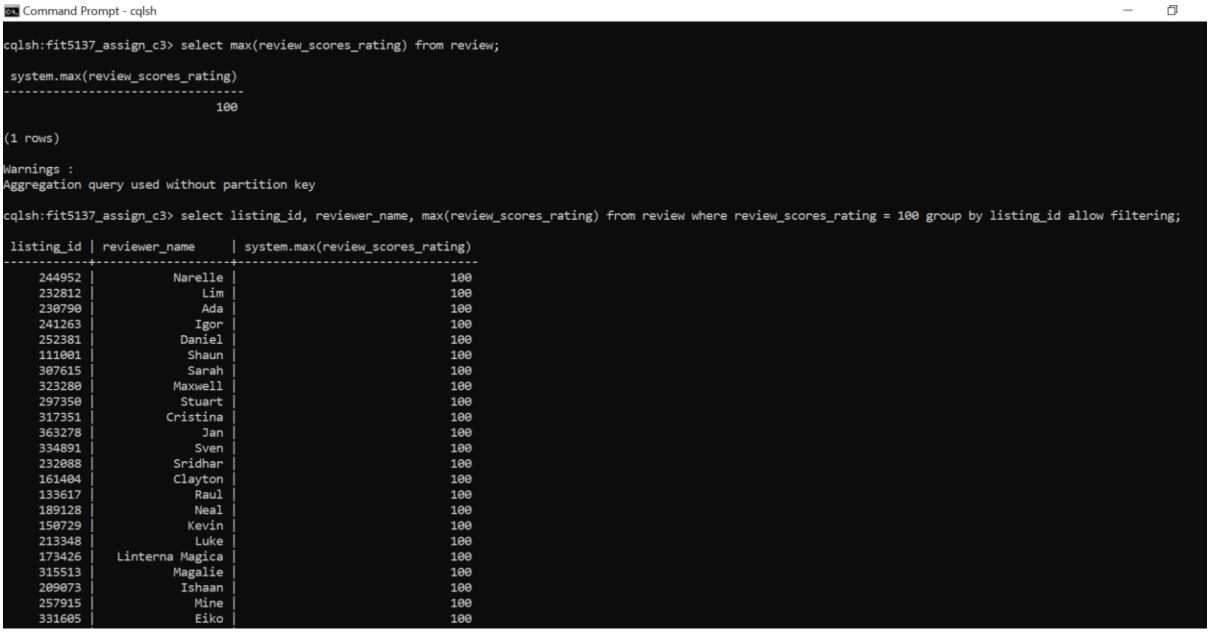
Command Prompt - cqlsh

```
cqlsh:fit5137_assign_c3> select id, reviewer_name,comments, max(review_scores_rating) from review where date = '2017-03-26';

 id        | reviewer_name | comments



                                                                                                                                    | system.max(review_scores_rating)
-----------+---------------+--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------+----------------------------------
 139908366 |          Kira | My sisters and I highly recommend this apartment. It is everything you see in the pictures and read in the other reviews. The one thing you may
 not get a sense of until you get there is the awesomeness of its location, which is basically perfect for everything (especially if you are a Brunswick St fan!). We were a
ble to walk to almost all of the locations on our must see/do /eat list and a quick tram ride to things slightly further afield. Light filled and airy due to the super cute
 balcony, which was perfect to sit out on any time of day. Rebecca is lovely, with fantastic communication. Book now and start getting excited about your stay! We managed t
o fit so much into our 3 days because our accommodation was so perfect and fantastically located. Thank you Rebecca. |                              100

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:fit5137_assign_c3>
```

Q20

select max(review_scores_rating) from review;

select listing_id, reviewer_name, max(review_scores_rating) from review where review_scores_rating = 100 group by listing_id allow filtering;

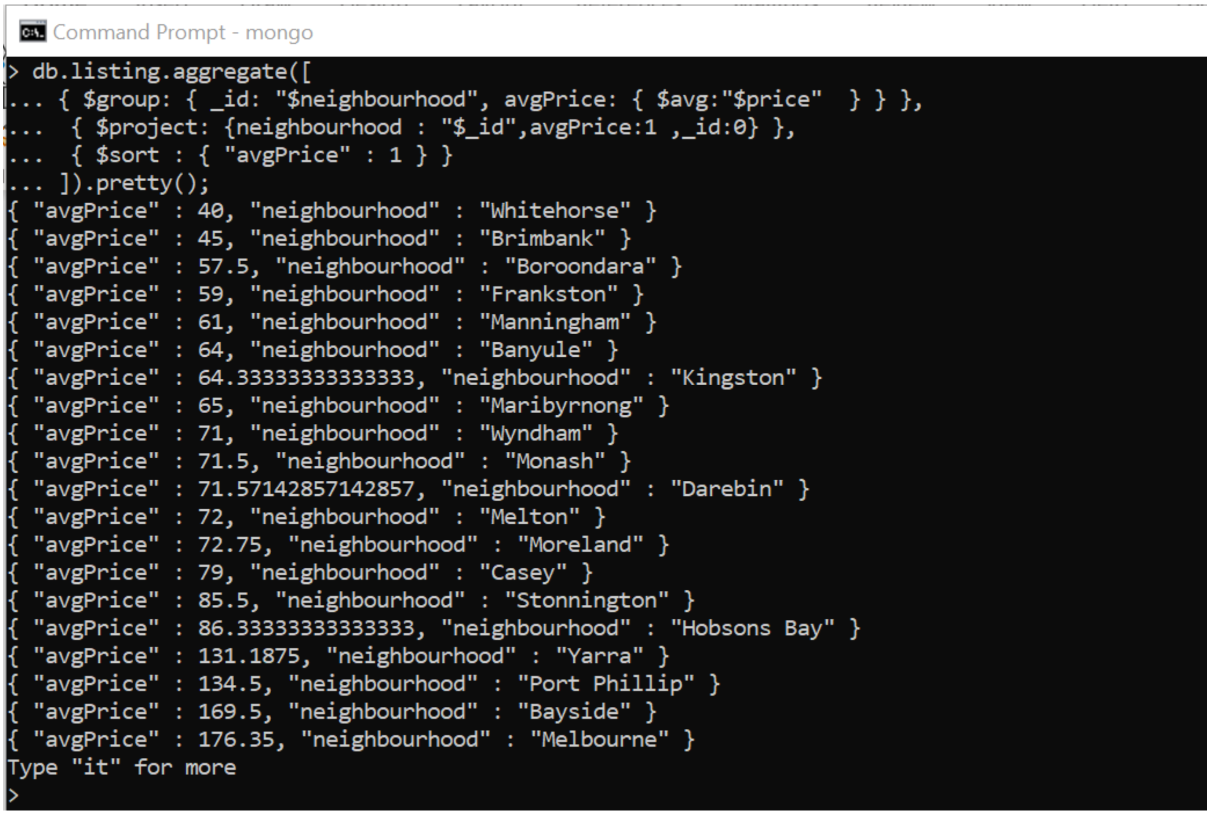


```
cqlsh:fit5137_assign_c3> select max(review_scores_rating) from review;

 system.max(review_scores_rating)
----------------------------------
                              100

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:fit5137_assign_c3> select listing_id, reviewer_name, max(review_scores_rating) from review where review_scores_rating = 100 group by listing_id allow filtering;

 listing_id | reviewer_name     | system.max(review_scores_rating)
------------+-------------------+----------------------------------
     244952 |           Narelle |                              100
     232812 |               Lim |                              100
     230790 |               Ada |                              100
     241263 |              Igor |                              100
     252381 |            Daniel |                              100
     111001 |             Shaun |                              100
     307615 |             Sarah |                              100
     323280 |           Maxwell |                              100
     297350 |            Stuart |                              100
     317351 |          Cristina |                              100
     363278 |               Jan |                              100
     334891 |              Sven |                              100
     232088 |           Sridhar |                              100
     161404 |           Clayton |                              100
     133617 |              Raul |                              100
     189128 |              Neal |                              100
     150729 |             Kevin |                              100
     213348 |              Luke |                              100
     173426 |   Linterna Magica |                              100
     315513 |           Magalie |                              100
     209073 |            Ishaan |                              100
     257915 |              Mine |                              100
     331605 |              Eiko |                              100
```

--EXTRA QUERIES--

1.Display the cheapest neighbourhoods based on average price.

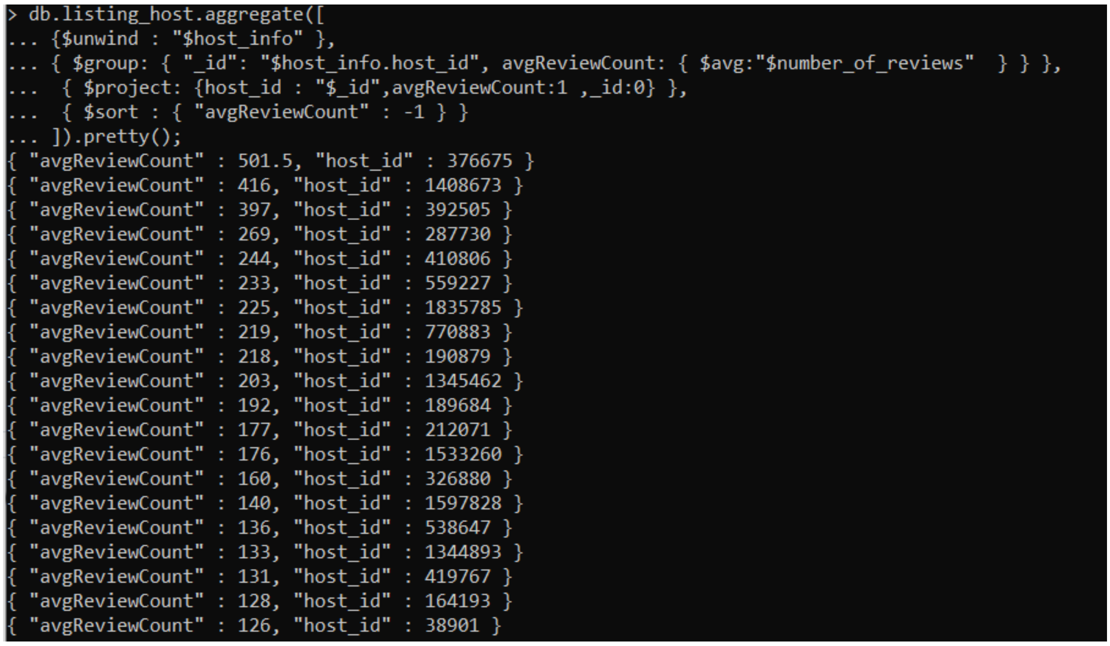
```
db.listing.aggregate([
{ $group: { _id: "$neighbourhood", avgPrice: { $avg:"$price"  } } },
 { $project: {neighbourhood : "$_id",avgPrice:1 ,_id:0} },
 { $sort : { "avgPrice" : 1 } }
]).pretty();
```



```
> db.listing.aggregate([
... { $group: { _id: "$neighbourhood", avgPrice: { $avg:"$price"  } } },
...  { $project: {neighbourhood : "$_id",avgPrice:1 ,_id:0} },
...  { $sort : { "avgPrice" : 1 } }
... ]).pretty();
{ "avgPrice" : 40, "neighbourhood" : "Whitehorse" }
{ "avgPrice" : 45, "neighbourhood" : "Brimbank" }
{ "avgPrice" : 57.5, "neighbourhood" : "Boroondara" }
{ "avgPrice" : 59, "neighbourhood" : "Frankston" }
{ "avgPrice" : 61, "neighbourhood" : "Manningham" }
{ "avgPrice" : 64, "neighbourhood" : "Banyule" }
{ "avgPrice" : 64.33333333333333, "neighbourhood" : "Kingston" }
{ "avgPrice" : 65, "neighbourhood" : "Maribyrnong" }
{ "avgPrice" : 71, "neighbourhood" : "Wyndham" }
{ "avgPrice" : 71.5, "neighbourhood" : "Monash" }
{ "avgPrice" : 71.57142857142857, "neighbourhood" : "Darebin" }
{ "avgPrice" : 72, "neighbourhood" : "Melton" }
{ "avgPrice" : 72.75, "neighbourhood" : "Moreland" }
{ "avgPrice" : 79, "neighbourhood" : "Casey" }
{ "avgPrice" : 85.5, "neighbourhood" : "Stonnington" }
{ "avgPrice" : 86.33333333333333, "neighbourhood" : "Hobsons Bay" }
{ "avgPrice" : 131.1875, "neighbourhood" : "Yarra" }
{ "avgPrice" : 134.5, "neighbourhood" : "Port Phillip" }
{ "avgPrice" : 169.5, "neighbourhood" : "Bayside" }
{ "avgPrice" : 176.35, "neighbourhood" : "Melbourne" }
Type "it" for more
>
```

2.Display the average number of reviews for a host.

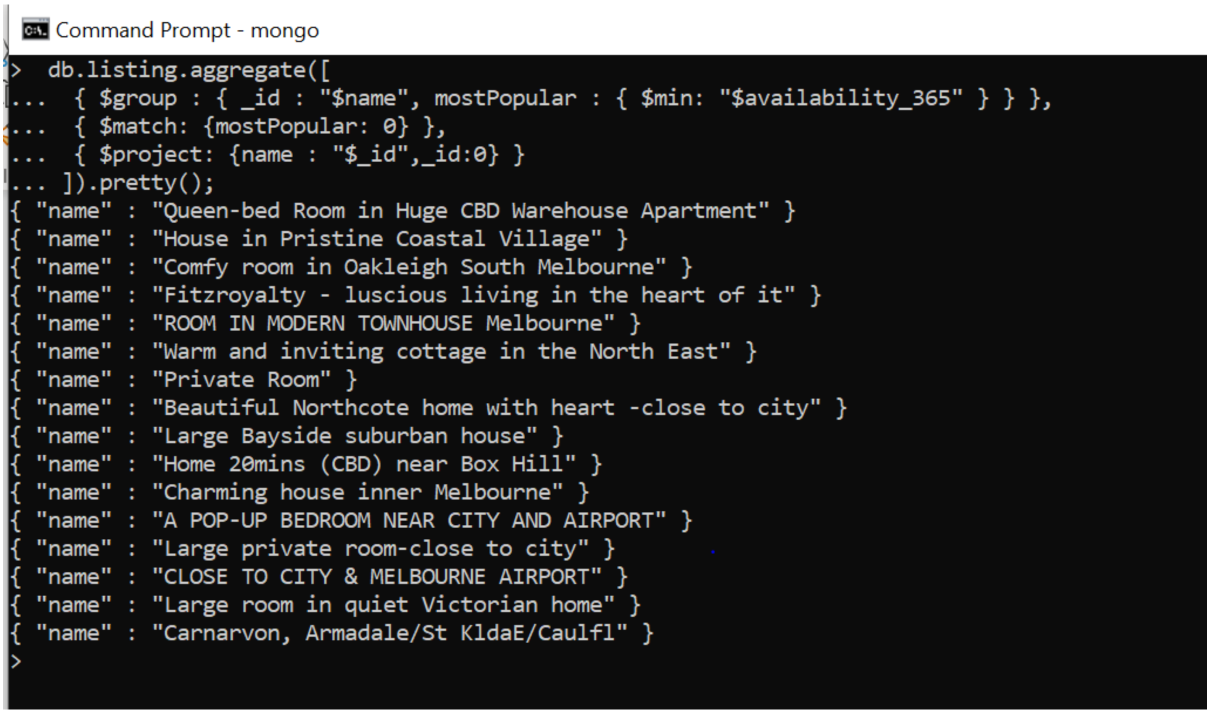
```
db.listing_host.aggregate([

{$unwind : "$host_info" },

{ $group: { "_id": "$host_info.host_id", avgReviewCount: { $avg:"$number_of_reviews"  } } },

 { $project: {host_id : "$_id",avgReviewCount:1 ,_id:0} },

 { $sort : { "avgReviewCount" : -1 } }

]).pretty();
```



```
> db.listing_host.aggregate([
... {$unwind : "$host_info" },
... { $group: { "_id": "$host_info.host_id", avgReviewCount: { $avg:"$number_of_reviews"  } } },
...  { $project: {host_id : "$_id",avgReviewCount:1 ,_id:0} },
...  { $sort : { "avgReviewCount" : -1 } }
... ]).pretty();
{ "avgReviewCount" : 501.5, "host_id" : 376675 }
{ "avgReviewCount" : 416, "host_id" : 1408673 }
{ "avgReviewCount" : 397, "host_id" : 392505 }
{ "avgReviewCount" : 269, "host_id" : 287730 }
{ "avgReviewCount" : 244, "host_id" : 410806 }
{ "avgReviewCount" : 233, "host_id" : 559227 }
{ "avgReviewCount" : 225, "host_id" : 1835785 }
{ "avgReviewCount" : 219, "host_id" : 770883 }
{ "avgReviewCount" : 218, "host_id" : 190879 }
{ "avgReviewCount" : 203, "host_id" : 1345462 }
{ "avgReviewCount" : 192, "host_id" : 189684 }
{ "avgReviewCount" : 177, "host_id" : 212071 }
{ "avgReviewCount" : 176, "host_id" : 1533260 }
{ "avgReviewCount" : 160, "host_id" : 326880 }
{ "avgReviewCount" : 140, "host_id" : 1597828 }
{ "avgReviewCount" : 136, "host_id" : 538647 }
{ "avgReviewCount" : 133, "host_id" : 1344893 }
{ "avgReviewCount" : 131, "host_id" : 419767 }
{ "avgReviewCount" : 128, "host_id" : 164193 }
{ "avgReviewCount" : 126, "host_id" : 38901 }
```

3.Display the most popular listings (based on availability_365).

```
 db.listing.aggregate([
 { $group : { _id : "$name", mostPopular : { $min: "$availability_365" } } },
 { $match: {mostPopular: 0} },
 { $project: {name : "$_id",_id:0} }
]).pretty();
```

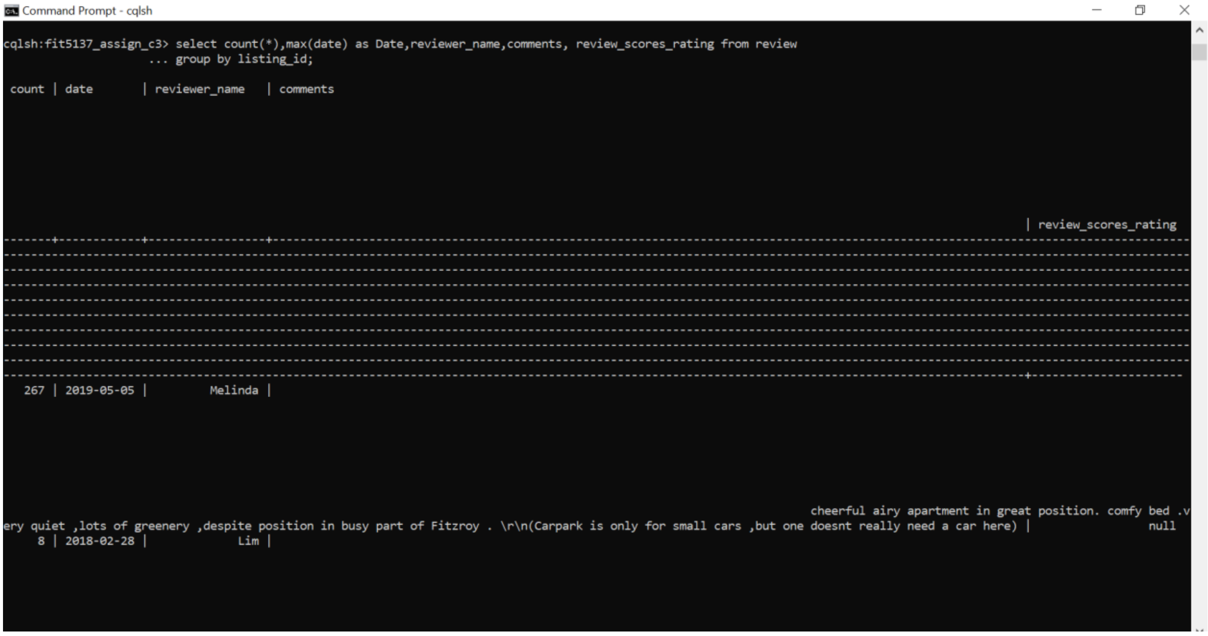


```
>  db.listing.aggregate([
...   { $group : { _id : "$name", mostPopular : { $min: "$availability_365" } } },
...   { $match: {mostPopular: 0} },
...   { $project: {name : "$_id",_id:0} }
...  ]).pretty();
{ "name" : "Queen-bed Room in Huge CBD Warehouse Apartment" }
{ "name" : "House in Pristine Coastal Village" }
{ "name" : "Comfy room in Oakleigh South Melbourne" }
{ "name" : "Fitzroyalty - luscious living in the heart of it" }
{ "name" : "ROOM IN MODERN TOWNHOUSE Melbourne" }
{ "name" : "Warm and inviting cottage in the North East" }
{ "name" : "Private Room" }
{ "name" : "Beautiful Northcote home with heart -close to city" }
{ "name" : "Large Bayside suburban house" }
{ "name" : "Home 20mins (CBD) near Box Hill" }
{ "name" : "Charming house inner Melbourne" }
{ "name" : "A POP-UP BEDROOM NEAR CITY AND AIRPORT" }
{ "name" : "Large private room-close to city" }
{ "name" : "CLOSE TO CITY & MELBOURNE AIRPORT" }
{ "name" : "Large room in quiet Victorian home" }
{ "name" : "Carnarvon, Armadale/St KldaE/Caulfl" }
>
```

4. Display each listing with the count of total reviews, the most recent date of the review, reviewer name and its rating.

select count(*),max(date) as Date,reviewer_name,comments, review_scores_rating from review

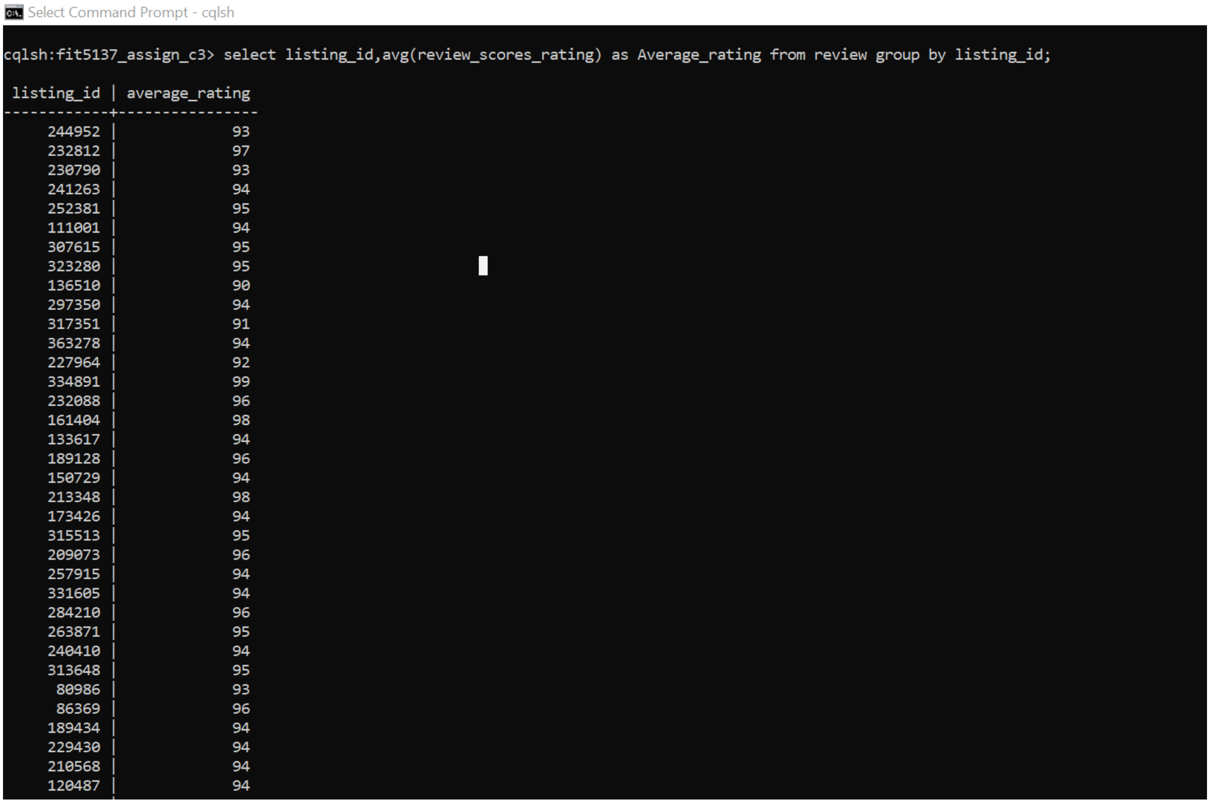
group by listing_id;



```
cqlsh:fit5137_assign_c3> select count(*),max(date) as Date,reviewer_name,comments, review_scores_rating from review
                     ... group by listing_id;

 count | date       | reviewer_name    | comments

                                                                                                                                                  | review_scores_rating
-------+------------+-----------------+------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------------------------+----------------------
   267 | 2019-05-05 |         Melinda |

                                                                                                                          cheerful airy apartment in great position. comfy bed .v
ery quiet ,lots of greenery ,despite position in busy part of Fitzroy . \r\n(Carpark is only for small cars ,but one doesnt really need a car here) |                 null
     8 | 2018-02-28 |             Lim |
```

5. Display the average rating of the each listing.

select listing_id,avg(review_scores_rating) as Average_rating from review group by listing_id;



```
cqlsh:fit5137_assign_c3> select listing_id,avg(review_scores_rating) as Average_rating from review group by listing_id;

 listing_id | average_rating
------------+----------------
     244952 |             93
     232812 |             97
     230790 |             93
     241263 |             94
     252381 |             95
     111001 |             94
     307615 |             95
     323280 |             95
     136510 |             90
     297350 |             94
     317351 |             91
     363278 |             94
     227964 |             92
     334891 |             99
     232088 |             96
     161404 |             98
     133617 |             94
     189128 |             96
     150729 |             94
     213348 |             98
     173426 |             94
     315513 |             95
     209073 |             96
     257915 |             94
     331605 |             94
     284210 |             96
     263871 |             95
     240410 |             94
     313648 |             95
      80986 |             93
      86369 |             96
     189434 |             94
     229430 |             94
     210568 |             94
     120487 |             94
```

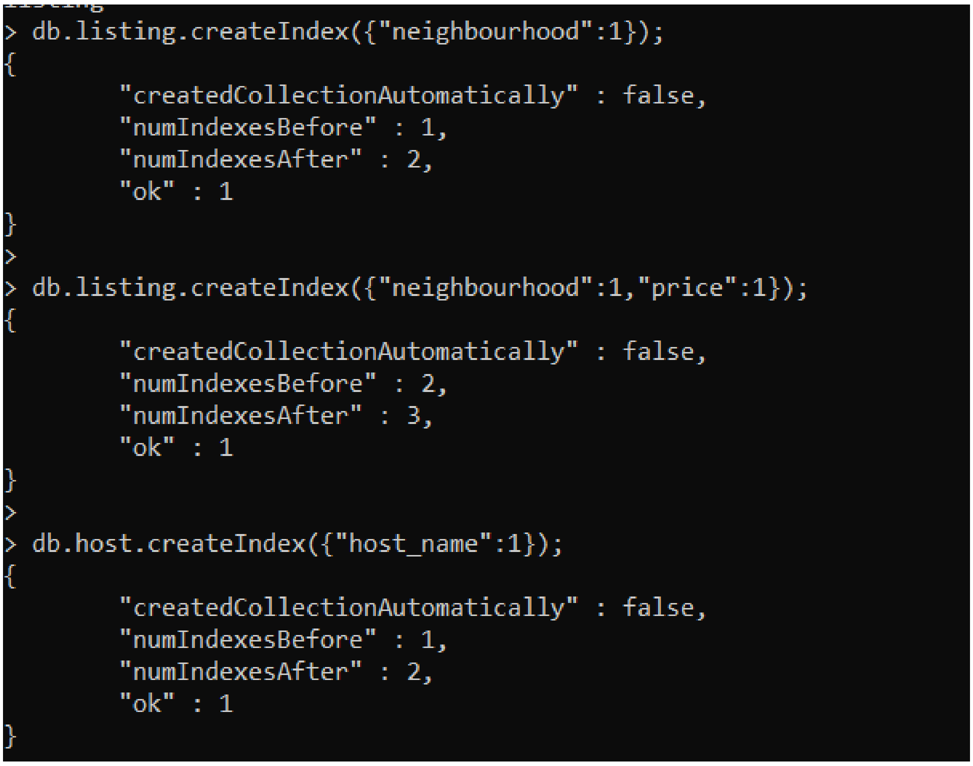
--Indexing--

Create index for referencing model

db.listing.createIndex({"neighbourhood":1});

db.listing.createIndex({"neighbourhood":1,"price":1});

db.host.createIndex({"host_name":1});



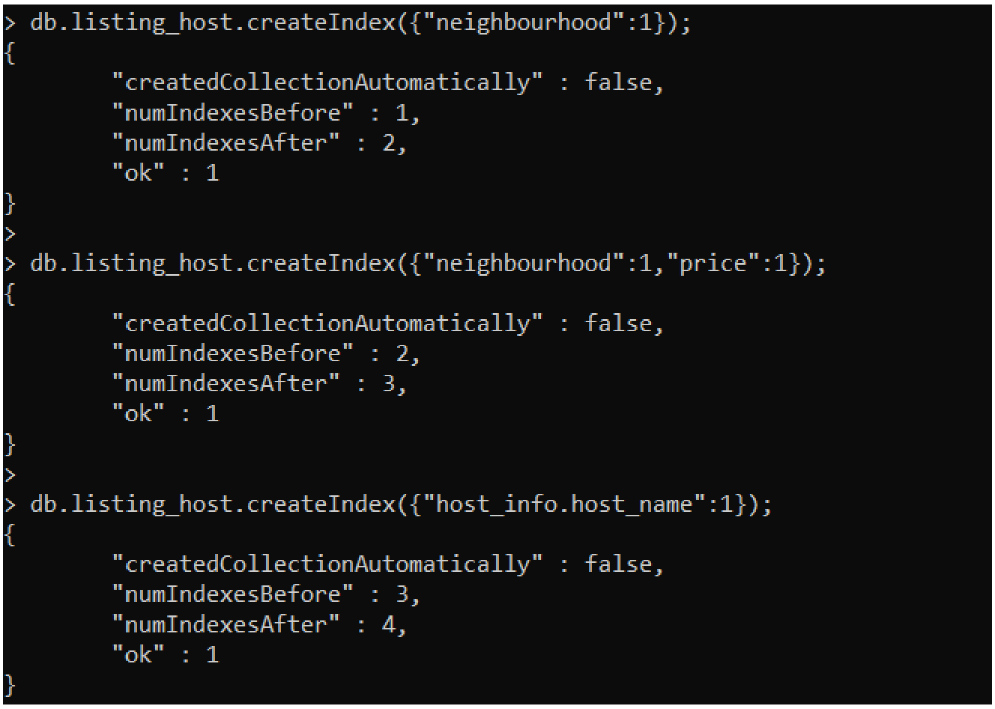
```
> db.listing.createIndex({"neighbourhood":1});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}
>
> db.listing.createIndex({"neighbourhood":1,"price":1});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 2,
        "numIndexesAfter" : 3,
        "ok" : 1
}
>
> db.host.createIndex({"host_name":1});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}
```

Create index for embedding model

db.listing_host.createIndex({"neighbourhood":1});

db.listing_host.createIndex({"neighbourhood":1,"price":1});

db.listing_host.createIndex({"host_info.host_name":1});



```
> db.listing_host.createIndex({"neighbourhood":1});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}
>
> db.listing_host.createIndex({"neighbourhood":1,"price":1});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 2,
        "numIndexesAfter" : 3,
        "ok" : 1
}
>
> db.listing_host.createIndex({"host_info.host_name":1});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 3,
        "numIndexesAfter" : 4,
        "ok" : 1
}
```

The columns considered for indexing are neighbourhood, price and host_name. We believe that according to the business case specified and the queries implemented, most frequently used columns will be regarding the neighbourhood or location, host of the accommodation or listings and the price specified for it. Since embedded model still has the same columns as the referencing model, the indexing keys still remain the same.

--C.4. Database Comparison. REPORT--

● Decide which database to use (either MongoDB or Cassandra).

We have decided to go with Cassandra. The reasons for the same are mentioned further in this section.

● List of the steps (including an explanation and a flowchart) on how the merging process could be implemented.

There are multiple ways online for implementing this task. Using the write and read branching techniques, using individual insert query for JSON objects (supported by Cassandra Version 3.0) or java looping on insert and many more techniques.

The easiest and most efficient solution discovered and implemented as of now is by exporting the collection of MongoDB to the csv file and importing in Cassandra (MongoDB Documentation, 2019). Cassandra provides only limited support for JSON. The steps for the same are mentioned in the form of a flowchart and explained as follows.

Step 1:

We already have JSON files for the collections *Listing* and *Host.* If we assume that the JSON files are not yet ready, we need to export the collections from MongoDB to JSON or direct CSV format using "mongoexport" command-line tool which is to be run directly from the system command line.

mongoexport --host --collection collection_name --db keyspace --out filename.csv [additional options]

use --type=csv to export the file in csv format.

Note: If Step 1 doesn't work, export the files into JSON format and then convert into CSV using JSON to CSV convertor.

Step 2:

Run the Cassandra Server by giving the command "Cassandra" in cmd.

Step 3:

Open a different cmd and drop to cql shell by giving the command "cqlsh".

Step 4:

Select the keyspace which already has the table(column family) "Review" in Cassandra by the command "use keyspace_name".

Step 5:

Create table *Listing* and *Host* with proper data types of the columns and primary keys.

Step 6:

Import both the CSV files using the command

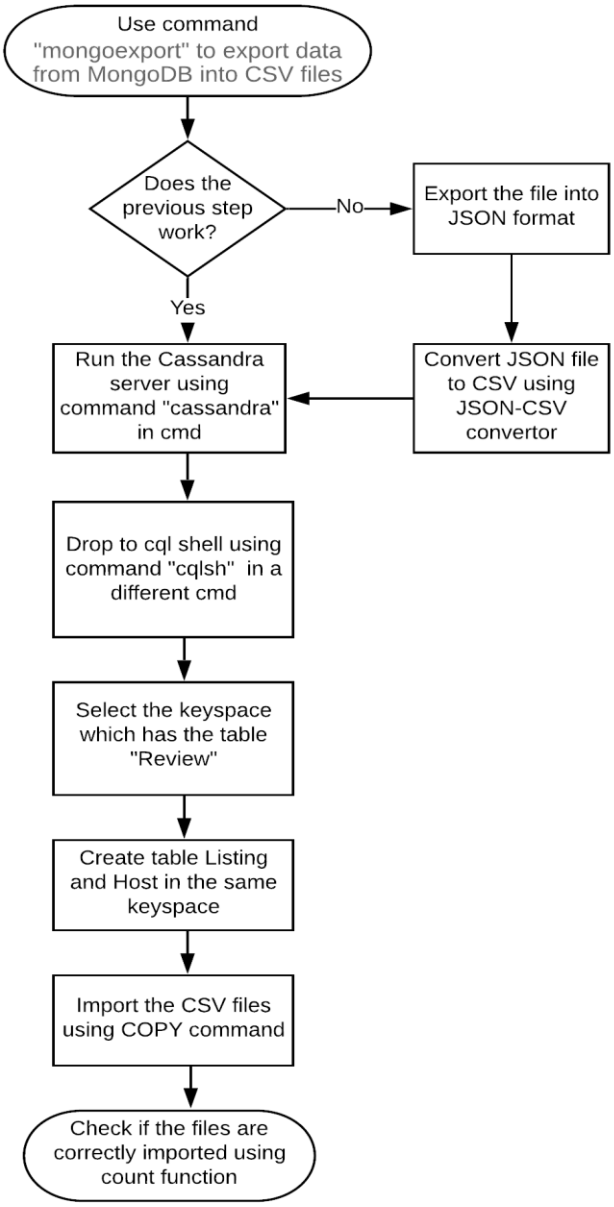
“COPY keypsace.tableName (col1,col2,col3.....) FROM 'file/file.csv' WITH DELIMITER=',' AND HEADER=TRUE;”

Step 7:

Check if the files are imported correctly by

“Select count(*) from keyspace.table;”

- **Flow Chart**


Use command "mongoexport" to export data from MongoDB into CSV files
Does the previous step work?
No
Export the file into JSON format
Yes
Run the Cassandra server using command "cassandra" in cmd
Convert JSON file to CSV using JSON-CSV convertor
Drop to cql shell using command "cqlsh" in a different cmd
Select the keyspace which has the table "Review"
Create table Listing and Host in the same keyspace
Import the CSV files using COPY command
Check if the files are correctly imported using count function

● Comparison in a tabular format with details on the main strengths and weaknesses of each database (Cassandra Vs MongoDB In 2018(Panoply blog, 2018)).

Category	NO	MongoDB	Cassandra
Syntax	1	MongoDB requires to learn a new language altogether.	CQL is easier than MongoDB if you already know SQL.
Replication	2	Replication is built-in, but requires additional setup	Replication is built-in and easy to use.
Data Storage	3	Requires JSON format to retrieve data and data is stored as documents in BSON	Flexible wide-column. i.e. Tables called as column families. Non-relational way
Analysis	4	Limits the number of users who can make changes to the database. This is because, the user access rights given by the administrator can either be full access or read-only.	The access rights for the users are defined per object by the database administrator. Hence, any user can get respective access from the DBA.
	5	MongoDB does support number of programming languages. eg. LISP, Matlab, Powershell etc.	Does not support much programming languages as compared to MongoDB
	6	Good for data without clear schema definition.	It can handle large amount of unstructured data.
	7	MongoDB can be a great choice if you need scalability and caching for real-time analytics	Can be speedily scaled with minimal increase of DBA work while having high reliability
	8	Good where you need high insert rate i.e. where write load is high.	Easy to setup and maintain if you expect rapid growth in database.
Use Case	9	Real-time analytics, mobile, IOT	Messaging system, Real-time analytics, Fraud detection, Storage

● Explanation on why we have selected Cassandra database over MongoDB.

The case study suggests few requirements which are efficiently satisfied by the specifications and properties of Cassandra. The requirements according to their respective justification is briefly explained as given (MongoDB vs Cassandra - The Next Generation NOSQL Enterprise Database Environments, 2016).

1. MonashBnB wishes to design a new database due to the increase in the volume of people occupying the accommodations listed. Cassandra supports large amount of unstructured data and can be rapidly scaled up in case of large volume efficiently.
2. The management team wishes to have a control of the environment without any manual hassles and maintenance. Also, they are migrating the data from manual entries to a database. Cassandra suffices the requirement as it can handle unstructured data and store it in a non-relational way. Also, it is easy to maintain without any much DBA work.

3. The CQL is easy to learn for a layman as compared to that of MongoDB. As the management team is not as experienced, Cassandra suffices the requirement.
4. The access rights for the users can be defined by the DBA. As any member from the team can easily access for the rights.

- References
-

MongoDB Documentation. (2019). Retrieved from https://docs.mongodb.com/manual/reference/program/mongoexport/#bin.mongoexport

MongoDB vs Cassandra - The Next Generation NOSQL Enterprise Database Environments. (2016). Plus Company Updates.

Panoply blog.(2018).Cassandra Vs MongoDB In 2018. Retrieved from https://blog.panoply.io/cassandra-vs-mongodb