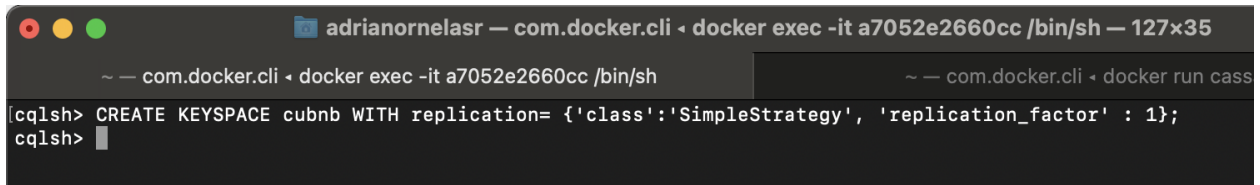


SECTION 1:

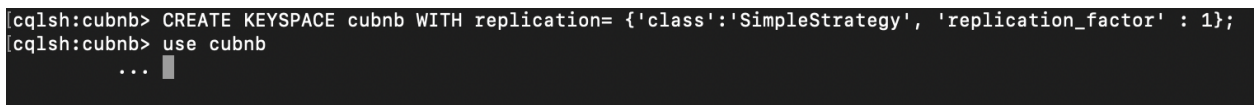
1)

```
CREATE KEYSPACE cubnb WITH replication= {'class':'SimpleStrategy', 'replication_factor' : 1};
```

A screenshot of a terminal window. The title bar shows 'adrianornelasr — com.docker.cli • docker exec -it a7052e2660cc /bin/sh — 127x35'. The terminal shows the prompt 'cqlsh>' followed by the command 'CREATE KEYSPACE cubnb WITH replication= {'class':'SimpleStrategy', 'replication_factor' : 1};'. The prompt returns to 'cqlsh>'.

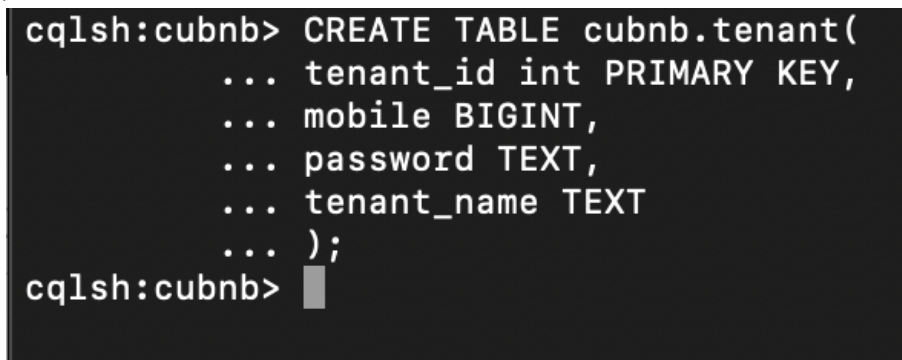
2)

```
USE cubnb;
```

A screenshot of a terminal window. The title bar shows 'adrianornelasr — com.docker.cli • docker exec -it a7052e2660cc /bin/sh — 127x35'. The terminal shows the prompt 'cqlsh:cubnb>' followed by the command 'CREATE KEYSPACE cubnb WITH replication= {'class':'SimpleStrategy', 'replication_factor' : 1};'. The prompt returns to 'cqlsh:cubnb>'.

3)

```
CREATE TABLE cubnb.tenant(  
    tenant_id int PRIMARY KEY,  
    mobile BIGINT,  
    password TEXT,  
    tenant_name TEXT  
);
```

A screenshot of a terminal window. The title bar shows 'adrianornelasr — com.docker.cli • docker exec -it a7052e2660cc /bin/sh — 127x35'. The terminal shows the prompt 'cqlsh:cubnb>' followed by the command 'CREATE TABLE cubnb.tenant(... tenant_id int PRIMARY KEY, ... mobile BIGINT, ... password TEXT, ... tenant_name TEXT ...);'. The prompt returns to 'cqlsh:cubnb>'.

4)

```
CREATE TABLE cubnb.property(  
    property_id int,  
    property_name TEXT,  
    host_id int,  
    location TEXT,  
    type TEXT,  
    price INT,  
    number_of_rooms INT,  
    max_occupancy INT,  
    number_of_reviews INT,  
    amenities set<TEXT>,  
    status TEXT,  
    primary key(property_id, host_id)
```

);

```
cqlsh:cubnb> CREATE TABLE cubnb.property(  
... property_id int,  
... property_name TEXT,  
... host_id int,  
... location TEXT,  
... type TEXT,  
... price INT,  
... number_of_rooms INT,  
... max_occupancy INT,  
... number_of_reviews INT,  
... amenities varchar,  
... status TEXT,  
... primary key(property_id, host_id)  
... );  
cqlsh:cubnb>
```

5)

```
CREATE TABLE cubnb.booking(  
    booking_id int,  
    property_id int,  
    total_occupancy int,  
    tenant_id int,  
    status TEXT,  
    no_of_days int,  
    start_date DATE,  
    end_date DATE,  
    primary key(booking_id, property_id, tenant_id)
```

);

```
cqlsh:cubnb> CREATE TABLE cubnb.booking(  
... booking_id int,  
... property_id int,  
... total_occupancy int,  
... tenant_id int,  
... status TEXT,  
... no_of_days int,  
... start_date DATE,  
... end_date DATE,  
... primary key(booking_id, property_id, tenant_id)  
... );  
cqlsh:cubnb>
```

Put cubnb-dataset.text (this is just to show the data and class's works)

```
cqlsh:cubnb>
cqlsh:cubnb> select count(*) from cubnb.property;

count
-----
200

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:cubnb> select count(*) from cubnb.booking;

count
-----
1500

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:cubnb> select count(*) from cubnb.tenant;

count
-----
100

(1 rows)

Warnings :
Aggregation query used without partition key

cqlsh:cubnb>
```

SECTION 2

1. Get the booking_id, property_id, tenant_id, and number of days a property is booked in the month of Feb '2022 for tenant-id 80855.

```
SELECT booking_id, property_id, tenant_id, no_of_days
FROM cubnb.booking
WHERE tenant_id=80855 AND
      start_date >= '2022-02-01' AND start_date <= '2022-02-28'
ALLOW FILTERING;
```

```
cqlsh:cubnb> SELECT booking_id, property_id, tenant_id, no_of_days
... FROM cubnb.booking
... WHERE tenant_id=80855 AND
...   start_date >= '2022-02-01' AND start_date <= '2022-02-28'
... ALLOW FILTERING;
```

booking_id	property_id	tenant_id	no_of_days
2471	72	80855	1
1332	133	80855	15
1753	154	80855	30
2678	79	80855	2
2252	53	80855	14
1441	42	80855	30

(6 rows)
cqlsh:cubnb> █

2. Get the count of properties whose reviews are more than 200.

```
SELECT COUNT(*) FROM cubnb.property WHERE number_of_reviews > 200 ALLOW FILTERING;
```

```
[cqlsh:cubnb> SELECT COUNT(*) FROM cubnb.property WHERE number_of_reviews > 200 ALLOW FILTERING;
```

count
45

(1 rows)

Warnings :
Aggregation query used without partition key

3. List all properties from “Brooklyn” location with type as apartment and have status as available.

```
SELECT property_id, property_name
... FROM cubnb.property
... WHERE location = 'Brooklyn' AND status = 'available' AND type = 'Apartment'
... ALLOW FILTERING;
```

```

[cqlsh:cubnb> SELECT property_id, property_name
... FROM cubnb.property
... WHERE location = 'Brooklyn' AND status = 'available' AND type = 'Apartment'
... ALLOW FILTERING;

```

property_id	property_name
144	Fully Furnished Basement Apartment
140	Huge Private Floor at The Waverly
94	2-bed brownstone duplex + garden
152	Lg Rm in Historic Prospect Heights
54	Greenpoint Place...Has It All!
75	BLUE TRIM GUEST HOUSE
179	Greenwich Village Stylish Apartment
196	Prime Location in Manhattan
25	MAISON DES SIRENES1,bohemian apartment

(9 rows)

cqlsh:cubnb>

4.

- a) Get all the properties which are handled by host_id 1029.

```

SELECT property_id, property_name
... FROM cubnb.property
... WHERE host_id = 1029
... ALLOW FILTERING;

```

```

[cqlsh:cubnb> SELECT property_id, property_name
... FROM cubnb.property
... WHERE host_id = 1029
... ALLOW FILTERING;

```

property_id	property_name
46	Entire apartment in central Brooklyn neighborhood.
8	Large Furnished Room Near Bway
122	3 Story Town House in Park Slope
7	BlissArtsSpace!
97	Light-filled classic Central Park
165	Charming and Cozy midtown loft any WEEK ENDS !!!

(6 rows)

cqlsh:cubnb>

- b) Add new amenity 'Heater' to the above results(in a single query).

```

UPDATE cubnb.property SET amenities = amenities + {'Heater'} WHERE property_id in (46, 8, 122, 7, 97, 165) AND host_id = 1029;

```

- c) Show the updated results from part b query.

```

cqlsh:cubnb> SELECT property_id, property_name, amenities FROM cubnb.property WHERE host_id = 1029 ALLOW FILTERING;

```

property_id	property_name	amenities
46	Entire apartment in central Brooklyn neighborhood	{'Air Conditioning', 'BeachView', 'Free Parking', 'Heater', 'Microwave Oven', 'Smoke Alarm', 'Washer', 'WiFi'}
8	Large Furnished Room Near Bway	{'Free Parking', 'GardenView', 'Heater', 'Microwave Oven', 'Pool', 'Smoke Alarm', 'Washer', 'WiFi'}
322	3 Story Town House in Park Slope	{'Air Conditioning', 'BeachView', 'Free Parking', 'GardenView', 'Heater', 'Microwave Oven', 'Pool', 'Security Camera', 'Smoke Alarm', 'Washer', 'WiFi'}
7	BlissArtsSpace!	{'Free Parking', 'GardenView', 'Heater', 'Microwave Oven', 'Pool', 'Smoke Alarm', 'Washer', 'WiFi'}
92	Light-filled classic Central Park	{'Air Conditioning', 'Free Parking', 'Heater', 'Microwave Oven', 'Security Camera', 'Smoke Alarm', 'Washer', 'WiFi'}
165	Charming and Cozy midtown loft any WEEK ENDS !!!	{'Air Conditioning', 'BeachView', 'Free Parking', 'Heater', 'Microwave Oven', 'Smoke Alarm', 'Washer', 'WiFi'}

```

(6 rows)
cqlsh:cubnb>

```

5.

- a) Add your details in the tenant table with tenant_id as 3287

```

INSERT INTO cubnb.tenant(tenant_id, mobile, password, tenant_name)
VALUES (
    3287,
    3034228869,
    'passwordPlaceholder999',
    'Adrian&John'
);

```

```

cqlsh:cubnb> INSERT INTO cubnb.tenant(tenant_id, mobile, password, tenant_name)
... VALUES (
... 3287,
... 3034228869,
... 'passwordPlaceholder999',
... 'Adrian&John'
... );
cqlsh:cubnb> select * from cubnb.tenant WHERE tenant_id = 3287;

```

tenant_id	mobile	password	tenant_name
3287	3034228869	passwordPlaceholder999	Adrian&John

```

(1 rows)

```

- b) Change the password of the tenant_id 3287 from the existing to 'password@123' and display the change

```

UPDATE cubnb.tenant
SET password = 'password@123'
WHERE tenant_id = 3287;

```



```
[cqlsh:cubnb> select * from cubnb.tenant WHERE tenant_id = 3287;
```

tenant_id	mobile	password	tenant_name
3287	3034228869	passwordPlaceholder999	Adrian&John

```
(1 rows)
[cqlsh:cubnb> UPDATE cubnb.tenant SET password = 'password@123' WHERE tenant_id = 3287;
[cqlsh:cubnb> select * from cubnb.tenant WHERE tenant_id = 3287;
```

tenant_id	mobile	password	tenant_name
3287	3034228869	password@123	Adrian&John

```
(1 rows)
cqlsh:cubnb>
```

6. Remove an amenity 'Garden View' for host_id 1012 and property_id 110 and display the results.

```
DELETE "amenities"['GardenView'] FROM cubnb.property WHERE host_id = 1012 AND property_id = 110;
cqlsh:cubnb> SELECT property_id, property_name, amenities FROM cubnb.property WHERE host_id = 1012 AND property_id = 110 ALLOW FILTERING;
```

```
[cqlsh:cubnb> DELETE "amenities"['GardenView'] FROM cubnb.property WHERE host_id = 1012 AND property_id = 110;
[cqlsh:cubnb> SELECT property_id, property_name, amenities FROM cubnb.property WHERE host_id = 1012 AND property_id = 110 ALLOW FILTERING;
```

property_id	property_name	amenities
110	Sunny 2-story Brooklyn townhouse w deck and garden	('Air Conditioning', 'BeachView', 'Free Parking', 'Microwave Oven', 'Pool', 'Security Camera', 'Smoke Alarm', 'Washer', 'WiFi')

```
(1 rows)
cqlsh:cubnb>
```

=

7. List the properties which have a 'BeachView' in it's amenities and whose rent price is more than 1000.

```
SELECT property_id, property_name, location, type, price
FROM cubnb.property
WHERE amenities CONTAINS 'BeachView'
AND price >= 1000
ALLOW FILTERING;
```

```
cqlsh:cubnb> SELECT property_id, property_name, location, type, price
... FROM cubnb.property
... WHERE amenities CONTAINS 'BeachView'
... AND price >= 1000
... ALLOW FILTERING;
```

property_id	property_name	location	type	price
110	Sunny 2-story Brooklyn townhouse w deck and garden	Queens	Apartment	1512
135	Most Central Location!	Brooklyn	Private Room	1042
85	Private room Great Deal at Lower East Side	Manhattan	Private Room	1512
81	Cozy 2 BR in Williamsburg	Queens	Private Room	1512
93	Large Parlor Room, Landmark Home 1 block to PRATT	Queens	Private Room	1042
163	Private, Large and Sunny 1BR w/WandD	Queens	Private Room	1512
165	Charming and Cozy midtown loft any WEEK ENDS !!!	Manhattan	Private Room	1042

```
(7 rows)
```

8. Get the number of highest occupancy of properties for 8 rooms.

```
SELECT MAX(max_occupancy) FROM cubnb.property WHERE number_of_rooms = 8 ALLOW FILTERING;
```

```
... /
cqlsh:cubnb> SELECT MAX(max_occupancy) FROM cubnb.property WHERE number_of_rooms = 8 ALLOW FILTERING;
system.max(max_occupancy)
-----
16
```