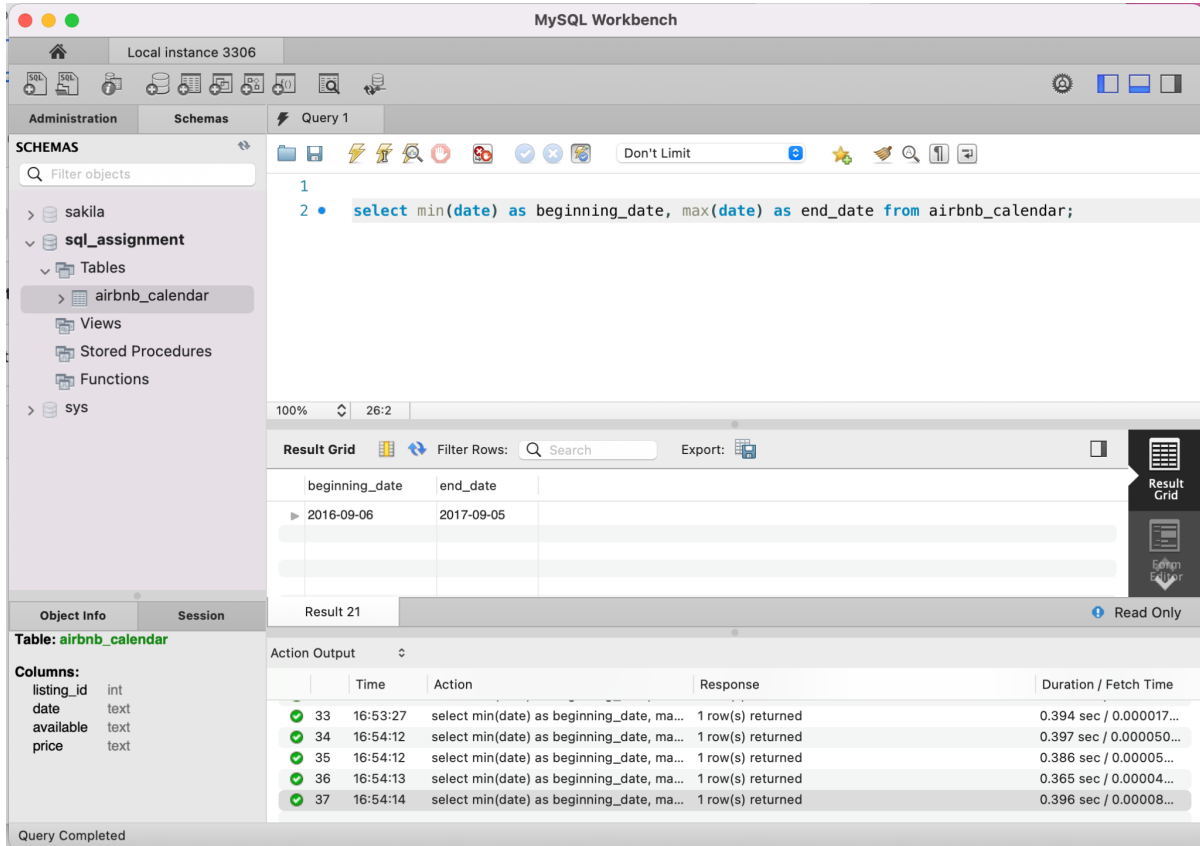


SQL Assignment

Name:- Rana Dilendra Singh

Q-1) What is the time period used?



The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'sakila' database selected, with 'sql_assignment' and 'airbnb_calendar' tables listed. The 'Query 1' pane shows the following SQL query:

```
1
2 • select min(date) as beginning_date, max(date) as end_date from airbnb_calendar;
```

The 'Result Grid' pane shows the results of the query:

beginning_date	end_date
2016-09-06	2017-09-05

The 'Object Info' pane shows the structure of the 'airbnb_calendar' table:

Columns:	
listing_id	int
date	text
available	text
price	text

The 'Action Output' pane shows the execution log:

	Time	Action	Response	Duration / Fetch Time
✓ 33	16:53:27	select min(date) as beginning_date, ma...	1 row(s) returned	0.394 sec / 0.000017...
✓ 34	16:54:12	select min(date) as beginning_date, ma...	1 row(s) returned	0.397 sec / 0.000050...
✓ 35	16:54:12	select min(date) as beginning_date, ma...	1 row(s) returned	0.386 sec / 0.00005...
✓ 36	16:54:13	select min(date) as beginning_date, ma...	1 row(s) returned	0.365 sec / 0.00004...
✓ 37	16:54:14	select min(date) as beginning_date, ma...	1 row(s) returned	0.396 sec / 0.00008...

Q-2) How many properties have duplicate entries? Remove duplicate rows (say a row appears 3 times, remove 2 and keep 1)

MySQL Workbench

Local Instance 3306 (sql_assignment) Local Instance 3306 (sql_assignment)

Administration Schemas Query 1

SCHEMAS

Filter objects

sakila
 sql_assignment
 Tables
 airbnb_calendar
 unique_airbnb_cal...
 Views
 Stored Procedures
 Functions
 sys

```

1 # properties having duplicate entries
2
3 select listing_id,date, count(*) from airbnb_calendar
4 group by listing_id,date having count(*)>1;
5
6 # remove duplicate rows by making new
7 create table unique_airbnb_calendar as
8 select distinct * from airbnb_calendar;
9
10

```

100% 24:3

Result Grid Filter Rows: Search Export: Result Grid

listing_id	date	count(*)
12898806	2017-06-13	2
12898806	2017-06-12	2
12898806	2017-06-11	2
12898806	2017-06-10	2
12898806	2017-06-09	2

Result 19 Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
52	22:25:27	DROP TABLE 'sql_assignment'. 'unique...	0 row(s) affected	0.043 sec
53	22:25:50	select listing_id,date, count(*) from air...	365 row(s) returned	3.130 sec / 0.00012 s...
54	22:25:53	create table unique_airbnb_calendar a...	1308525 row(s) affected Records: 1308525 Duplicat...	5.236 sec

Query Completed

Q-3) For each property, find out the number of days the property was available and not available (create a table with listing_id, available days, unavailable days and available days as a fraction of total days)

MySQL Workbench

Local Instance 3306 (sql_assignment) Local Instance 3306 (sql_assignment)

Administration Schemas Query 1

SCHEMAS

Filter objects

sakila
 sql_assignment
 Tables
 airbnb_calendar
 unique_airbnb_cal...
 Views
 Stored Procedures
 Functions
 sys

```

1
2 create table property_detail as (
3   select listing_id, sum(available='t') as is_available, sum(available='f') as not_available,
4   sum(available='t')/365 as available_fraction
5   from unique_airbnb_calendar group by (listing_id) );
6
7 select * from property_detail;

```

100% 31:7

Result Grid Filter Rows: Search Export: Fetch rows: Result Grid

listing_id	is_available	not_available	available_fra...
12147973	0	365	0.0000
3075044	359	6	0.9836
6976	319	46	0.8740
1436513	98	267	0.2685
7651065	334	31	0.9151
12386020	58	307	0.1589
5706995	344	21	0.9425
1247446	365	0	1.0000

property_detail 24 Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
71	22:55:54	DROP TABLE 'sql_assignment'. 'property_detail'	0 row(s) affected	0.0070 sec
72	22:55:59	create table property_detail as (select listing_id, sum(available='t') as is_av...	3585 row(s) affected, 1024 warning(s): 1265 Data tru...	0.701 sec
73	22:55:59	select * from property_detail	3585 row(s) returned	0.00049 sec / 0.0007...

Query Completed

Q-4) i) How many properties were available on more than 50% of the days?

The screenshot shows MySQL Workbench with a query executed. The query is:

```
1 select count(*) as availability_more_than_50_percent
2 from property_detail
3 where available_fraction>0.5
4
```

The result grid shows one row with the value 1732.

availability_more_than_50_percent
1732

The Action Output table shows the following data:

	Time	Action	Response	Duration / Fetch Time
✓ 84	23:11:42	select count(*) as availability_more_than_50_percent from property_detail...	2 row(s) returned	0.0052 sec / 0.00001...
✓ 85	23:12:20	select count(*) as availability_more_than_50_percent from property_detail...	1 row(s) returned	0.0031 sec / 0.00001...
✓ 86	23:12:32	select count(*) as availability_more_than_50_percent from property_detail...	1 row(s) returned	0.0031 sec / 0.00001...

Q-4) ii) How many properties were available on more than 75% of the days?

The screenshot shows MySQL Workbench with a query executed. The query is:

```
1 select count(*) as availability_more_than_75_percent
2 from property_detail
3 where available_fraction>0.75;
4
```

The result grid shows one row with the value 1429.

availability_more_than_75_perc...
1429

The Action Output table shows the following data:

	Time	Action	Response	Duration / Fetch Time
✓ 85	23:12:20	select count(*) as availability_more_than_50_percent from property_detail...	1 row(s) returned	0.0031 sec / 0.00001...
✓ 86	23:12:32	select count(*) as availability_more_than_50_percent from property_detail...	1 row(s) returned	0.0031 sec / 0.00001...
✓ 87	23:14:10	select count(*) as availability_more_than_75_percent from property_detail...	1 row(s) returned	0.0026 sec / 0.00001...

Q-5) Create a table with max, min and average price of each property.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' panel with a tree view showing the 'sakila' database and the 'sql_assignment' schema. The 'Query 1' tab is active, showing a SQL query to create a table 'property_prices_detail' and insert data from 'unique_airbnb_calendar'.

```

1
2 • create table property_prices_detail as
3 select listing_id, min(price) as min_price, max(price) as max_price,
4 avg(price) as average_price
5 from unique_airbnb_calendar
6 where available='t'
7 group by listing_id;
8
9 select * from property_prices_detail;

```

The 'Result Grid' shows the output of the query, displaying columns: listing_id, min_price, max_price, and average_price. The 'Action Output' panel shows the execution details of the queries.

listing_id	min_price	max_price	average_price
3075044	65	75	67.8134
6976	65	65	65.0000
1436513	75	75	75.0000
7651065	79	79	79.0000
12386020	75	75	75.0000
5706985	100	200	111.7558
2843445	75	75	75.0000
753446	58	69	59.3631

Time	Action	Response	Duration / Fetch Time
49 10:51:30	DROP TABLE `sql_assignment`.`property_...	0 row(s) affected	0.0065 sec
50 10:51:45	create table property_prices_detail as sel...	2898 row(s) affected, 1024 warning(s): 1265 Data tru...	0.452 sec
51 10:51:45	select * from property_prices_detail	2898 row(s) returned	0.00046 sec / 0.0007...

Q-6) Extract properties with an average price of more than \$500

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' panel with a tree view showing the 'sakila' database and the 'sql_assignment' schema. The 'Query 1' tab is active, showing a SQL query to select properties with an average price greater than \$500.

```

1
2 • select listing_id, average_price from property_prices_detail
3 where average_price > 500;

```

The 'Result Grid' shows the output of the query, displaying columns: listing_id and average_price. The 'Action Output' panel shows the execution details of the queries.

listing_id	average_price
115936	525.0000
50032	725.0000
10153739	553.7094
10153536	553.7094
475259	506.8966
10136854	553.7094
13445221	563.1714
14813006	584.0653

Time	Action	Response	Duration / Fetch Time
55 10:55:36	select listing_id, average_price from propert...	74 row(s) returned	0.0022 sec / 0.00002...
56 10:55:39	select listing_id, average_price from propert...	74 row(s) returned	0.0023 sec / 0.00002...
57 10:56:04	select listing_id, average_price from propert...	74 row(s) returned	0.0025 sec / 0.00002...