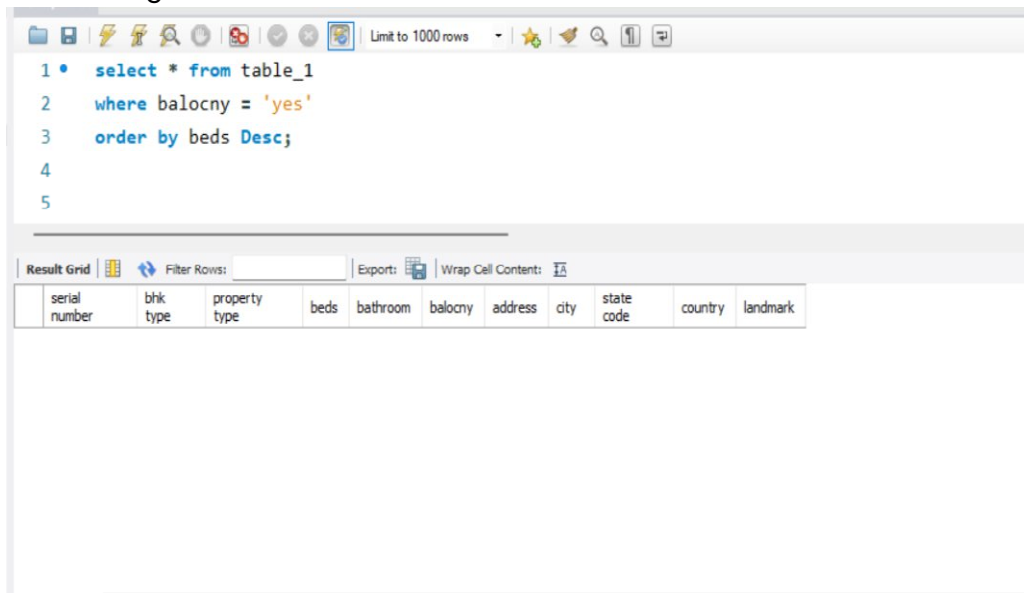
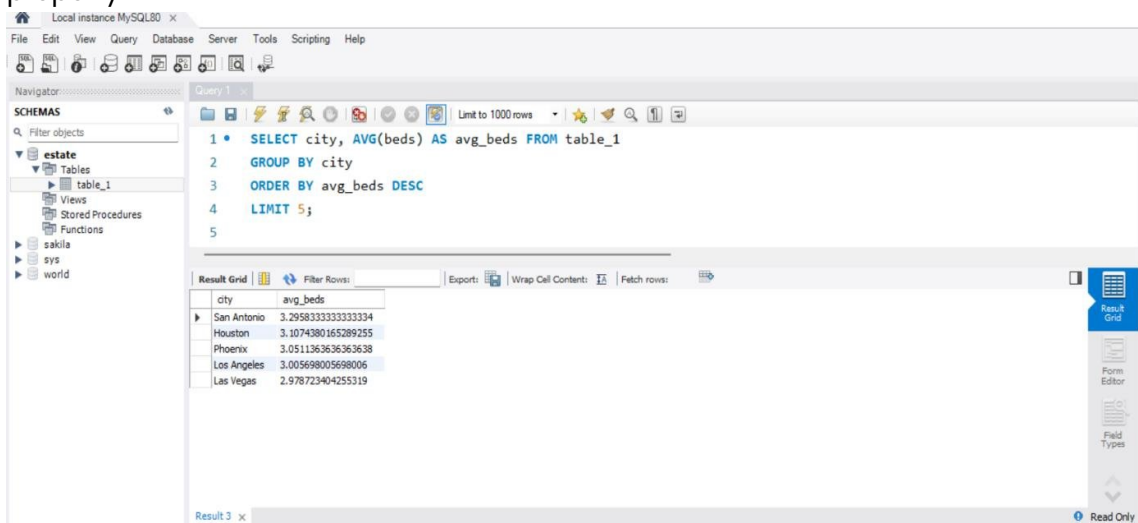


# SQL queries For Table 1

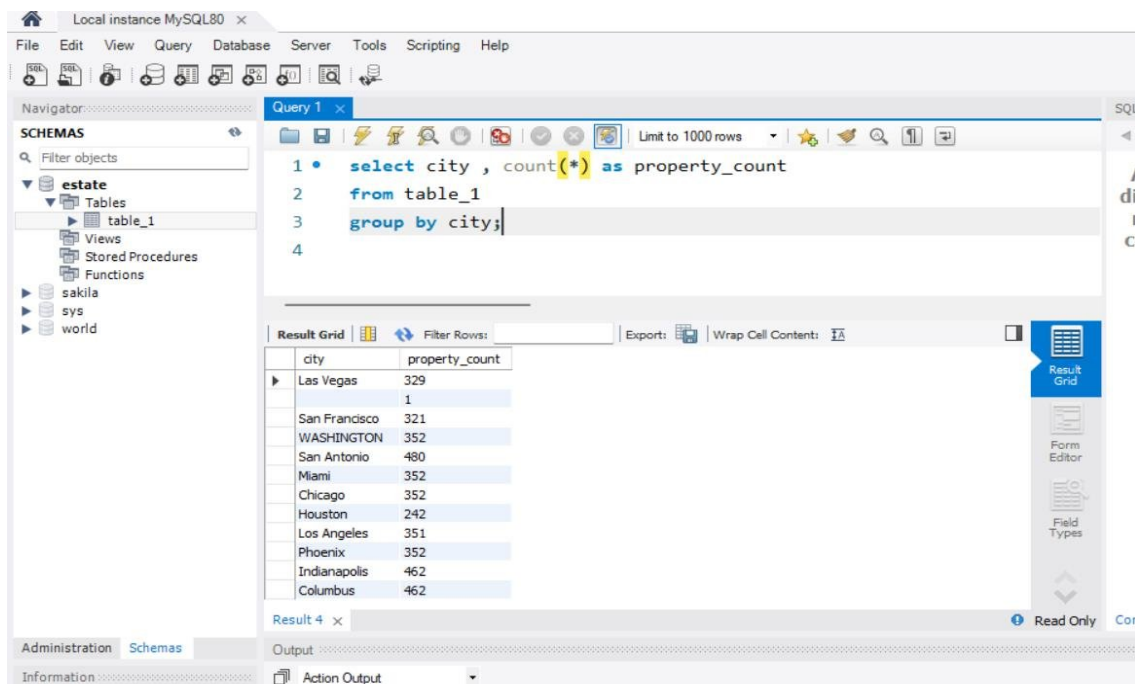
1. Retrieve properties with balconies, sorted by the number of bedrooms in descending order



2. Find the top 5 cities with the highest average number of bedrooms per property



### 3. Count the number of properties in each city.



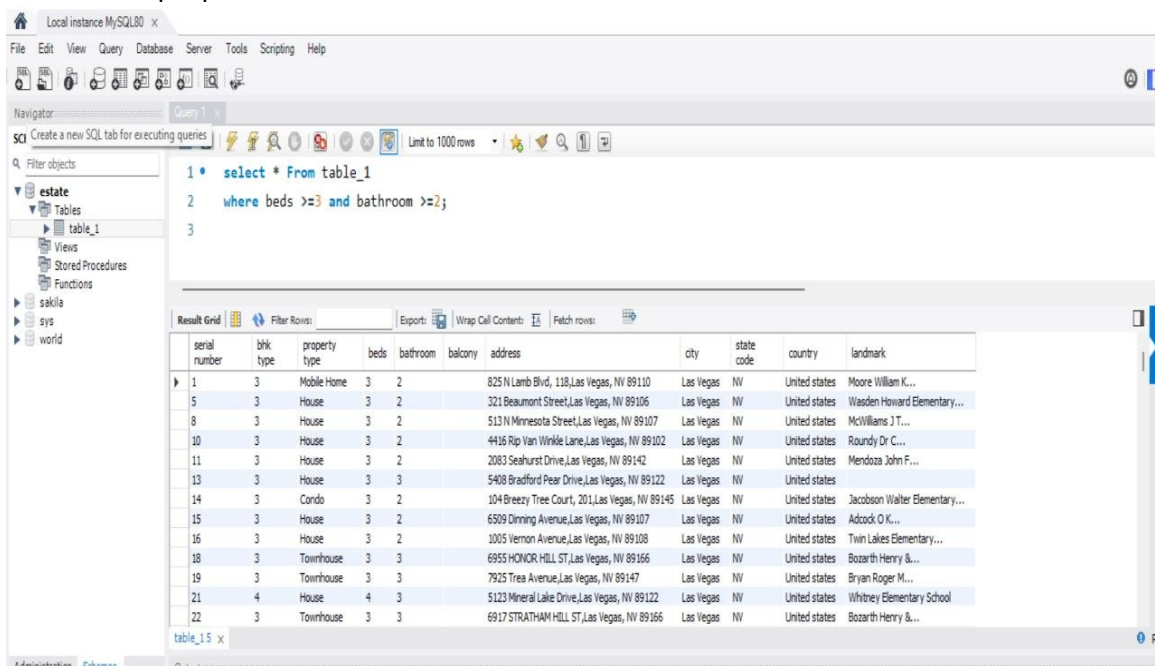
The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying the following SQL query:

```
1 • select city , count(*) as property_count
2   from table_1
3  group by city;
```

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

| city          | property_count |
|---------------|----------------|
| Las Vegas     | 329            |
| San Francisco | 321            |
| WASHINGTON    | 352            |
| San Antonio   | 480            |
| Miami         | 352            |
| Chicago       | 352            |
| Houston       | 242            |
| Los Angeles   | 351            |
| Phoenix       | 352            |
| Indianapolis  | 462            |
| Columbus      | 462            |

### 4. Retrieve all properties with at least 3 bedrooms and 2 bathrooms.



The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying the following SQL query:

```
1 • select * from table_1
2   where beds >=3 and bathroom >=2;
```

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

| serial number | bhk type | property type | beds | bathroom | balcony | address  | city      | state code | country       | landmark                      |
|---------------|----------|---------------|------|----------|---------|--|-----------|------------|---------------|-------------------------------|
| 1             | 3        | Mobile Home   | 3    | 2        |         | 825 N Lamb Blvd, 118,Las Vegas, NV 89110       | Las Vegas | NV         | United states | Moore William K...            |
| 5             | 3        | House         | 3    | 2        |         | 321 Beaumont Street,Las Vegas, NV 89106        | Las Vegas | NV         | United states | Warden Howard Elementary...   |
| 8             | 3        | House         | 3    | 2        |         | 513 N Minnesota Street,Las Vegas, NV 89107     | Las Vegas | NV         | United states | McWilliams J T...             |
| 10            | 3        | House         | 3    | 2        |         | 4416 Rip Van Winkle Lane,Las Vegas, NV 89102   | Las Vegas | NV         | United states | Roundy Dr C...                |
| 11            | 3        | House         | 3    | 2        |         | 2083 Seahurst Drive,Las Vegas, NV 89142        | Las Vegas | NV         | United states | Mendoza John F...             |
| 13            | 3        | House         | 3    | 3        |         | 5408 Bradford Pear Drive,Las Vegas, NV 89122   | Las Vegas | NV         | United states |                               |
| 14            | 3        | Condo         | 3    | 2        |         | 104 Breezy Tree Court, 201,Las Vegas, NV 89145 | Las Vegas | NV         | United states | Jacobson Walter Elementary... |
| 15            | 3        | House         | 3    | 2        |         | 6509 Dinning Avenue,Las Vegas, NV 89107        | Las Vegas | NV         | United states | Adcock O K...                 |
| 16            | 3        | House         | 3    | 2        |         | 1005 Vernon Avenue,Las Vegas, NV 89108         | Las Vegas | NV         | United states | Twin Lakes Elementary...      |
| 18            | 3        | Townhouse     | 3    | 3        |         | 6955 HONOR HILL ST,Las Vegas, NV 89166         | Las Vegas | NV         | United states | Bozarth Henry &...            |
| 19            | 3        | Townhouse     | 3    | 3        |         | 7925 Trea Avenue,Las Vegas, NV 89147           | Las Vegas | NV         | United states | Bryan Roger M...              |
| 21            | 4        | House         | 4    | 3        |         | 5123 Mineral Lake Drive,Las Vegas, NV 89122    | Las Vegas | NV         | United states | Whitney Elementary School     |
| 22            | 3        | Townhouse     | 3    | 3        |         | 6917 STRATHAM HILL ST,Las Vegas, NV 89166      | Las Vegas | NV         | United states | Bozarth Henry &...            |

- Find properties in a specific state with a certain landmark. (take state and landmark on your own ).

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'estate' expanded, showing 'table\_1' and its columns: serial number, bhk type, property type, beds, bathroom, balcony, address, city, state\_code, country, and landmark. The main query editor contains the following SQL:

```
1 • select * from table_1
2 where state_code = "NV" and landmark like 'Whitney Elementary School';
```

The 'Result Grid' shows the following data:

| serial number | bhk type | property type | beds | bathroom | balcony | address   | city      | state_code | country       | landmark                  |
|---------------|----------|---------------|------|----------|---------|---|-----------|------------|---------------|---------------------------|
| 21            | 4        | House         | 4    | 3        |         | 5123 Mineral Lake Drive, Las Vegas, NV 89122    | Las Vegas | NV         | United states | Whitney Elementary School |
| 94            | 2        | Condo         | 2    | 1        |         | 5350 Silvermist Court, 202, Las Vegas, NV 89122 | Las Vegas | NV         | United states | Whitney Elementary School |
| 119           | 3        | House         | 3    | 2        |         | 4940 Midnight Oil Drive, Las Vegas, NV 89122    | Las Vegas | NV         | United states | Whitney Elementary School |
| 134           | 4        | House         | 4    | 3        |         | 6327 Felicitas Avenue, Las Vegas, NV 89122      | Las Vegas | NV         | United states | Whitney Elementary School |
| 275           | 3        | House         | 3    | 3        |         | 6080 Forest Archer Place, Las Vegas, NV 89122   | Las Vegas | NV         | United states | Whitney Elementary School |
| 290           | 3        | House         | 3    | 3        |         | 6629 Crystal Run Lane, Las Vegas, NV 89122      | Las Vegas | NV         | United states | Whitney Elementary School |

## SQL Query for table 2

- Calculate the average price per square foot for properties built before 2010.

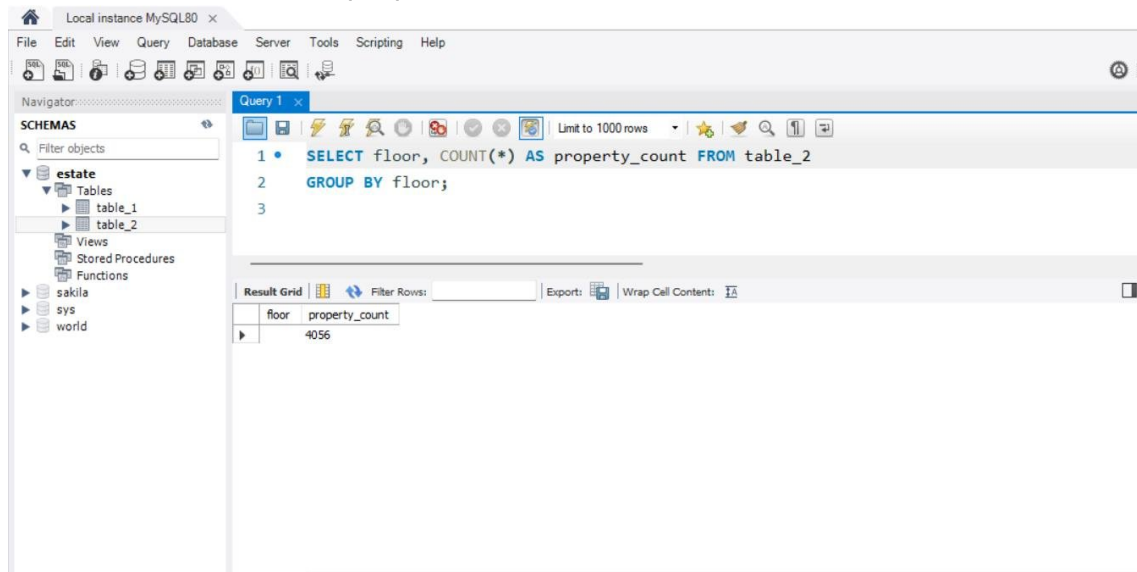
The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'estate' expanded, showing 'table\_2' and its columns: avg\_price\_per\_sqft, year built, and other columns. The main query editor contains the following SQL:

```
1 • SELECT AVG(COALESCE(`price per sqft`, 0)) AS avg_price_per_sqft
2 FROM table_2
3 WHERE `year built` < 2010;
```

The 'Result Grid' shows the following data:

| avg_price_per_sqft |
|--------------------|
| 26806.95621019108  |

- Find the total number of properties on each floor.



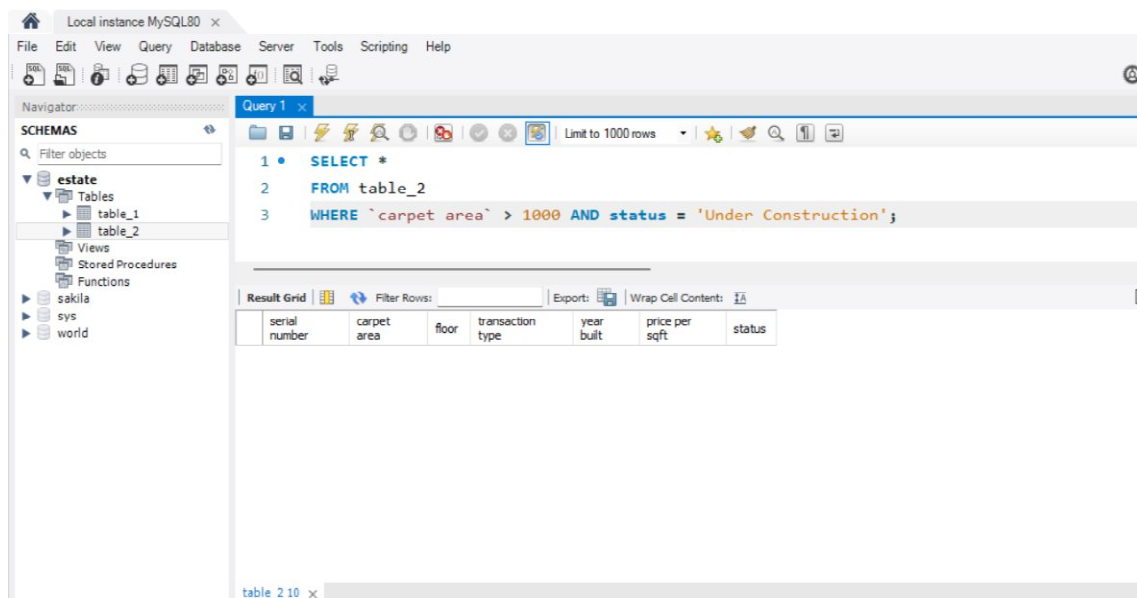
The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying a SQL query in the 'Query 1' editor. The query is:

```
1 • SELECT floor, COUNT(*) AS property_count FROM table_2
2   GROUP BY floor;
3
```

The 'Result Grid' shows the following data:

| floor | property_count |
|-------|----------------|
| 4056  |                |

- Retrieve properties with a carpet area greater than 1000 square feet and a status of 'Under Construction'.



The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying a SQL query in the 'Query 1' editor. The query is:

```
1 • SELECT *
2   FROM table_2
3  WHERE `carpet area` > 1000 AND status = 'Under Construction';
```

The 'Result Grid' shows the following data:

| serial number | carpet area | floor | transaction type | year built | price per sqft | status |
|---------------|-------------|-------|------------------|------------|----------------|--------|
|---------------|-------------|-------|------------------|------------|----------------|--------|

4. Calculate the average price per square foot for each transaction type.

The screenshot shows the MySQL Workbench interface. The 'Query 1' tab is active, displaying the following SQL query:

```
1 • SELECT `transaction type`, AVG(`price per sqft`) AS avg_price_per_sqft
2 FROM table_2
3 GROUP BY `transaction type`;
4
```

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

| transaction type | avg_price_per_sqft |
|------------------|--------------------|
|                  | 28133.476824457593 |

5. Find the properties with the highest price per square foot, sorted in descending order.

The screenshot shows the MySQL Workbench interface. The 'Query 1' tab is active, displaying the following SQL query:

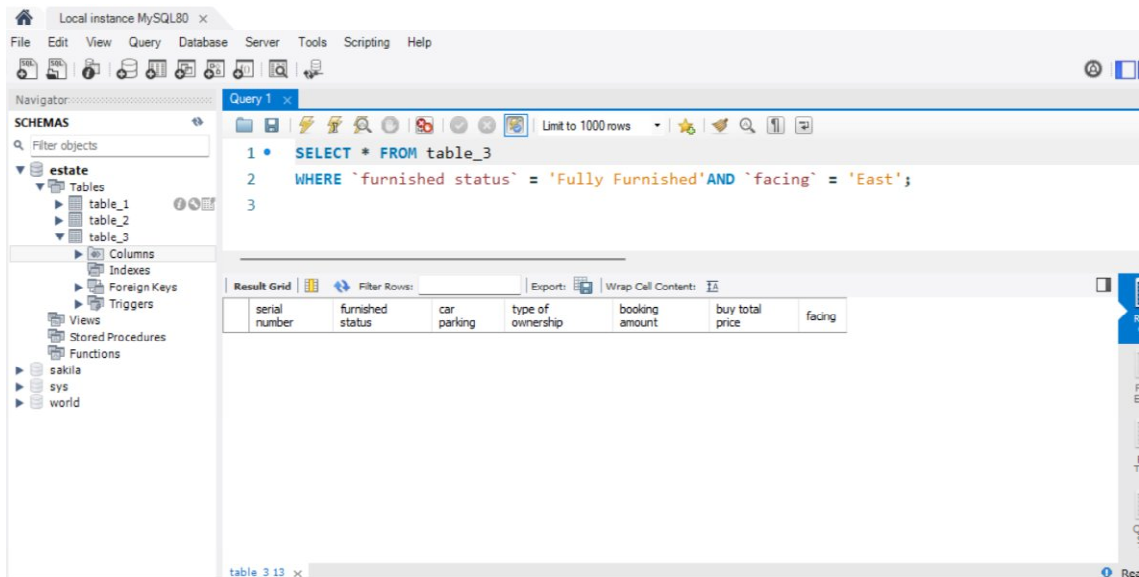
```
1 • SELECT * FROM table_2
2 ORDER BY `price per sqft` DESC;
```

The 'Result Grid' shows the results of the query, sorted by price per square foot in descending order:

| serial number | carpet area | floor | transaction type | year built | price per sqft | status |
|---------------|-------------|-------|------------------|------------|----------------|--------|
| 3207          | 1172        |       |                  | 1947       | 9986.0         | NOVA   |
| 3976          | 2515        |       |                  | 1992       | 9978.0         | NOVA   |
| 2332          | 2348        |       |                  | 2000       | 99765.0        | NOVA   |
| 3906          | 4526        |       |                  | 2000       | 9970.0         | NOVA   |
| 1245          | 1999        |       |                  | 2000       | 9961.0         | NOVA   |
| 3161          | 6091        |       |                  | 2000       | 9958.0         | NOVA   |
| 3005          | 882         |       |                  | 1978       | 9950.0         | NOVA   |
| 3789          | 1850        |       |                  | 1925       | 9944.0         | NOVA   |
| 2172          | 2777        |       |                  | 2000       | 9942.0         | NOVA   |
| 2126          | 1456        |       |                  | 2000       | 9940.0         | NOVA   |
| 893           | 2100        |       |                  | 2000       | 99396.0        | NOVA   |
| 3267          | 926         |       |                  | 1928       | 9938.0         | NOVA   |
| 3331          | 2522        |       |                  | 1958       | 9935.0         | NOVA   |

## SQL query for table 3

1. Retrieve all properties with a furnished status of 'Fully Furnished' and a facing direction of 'East'.

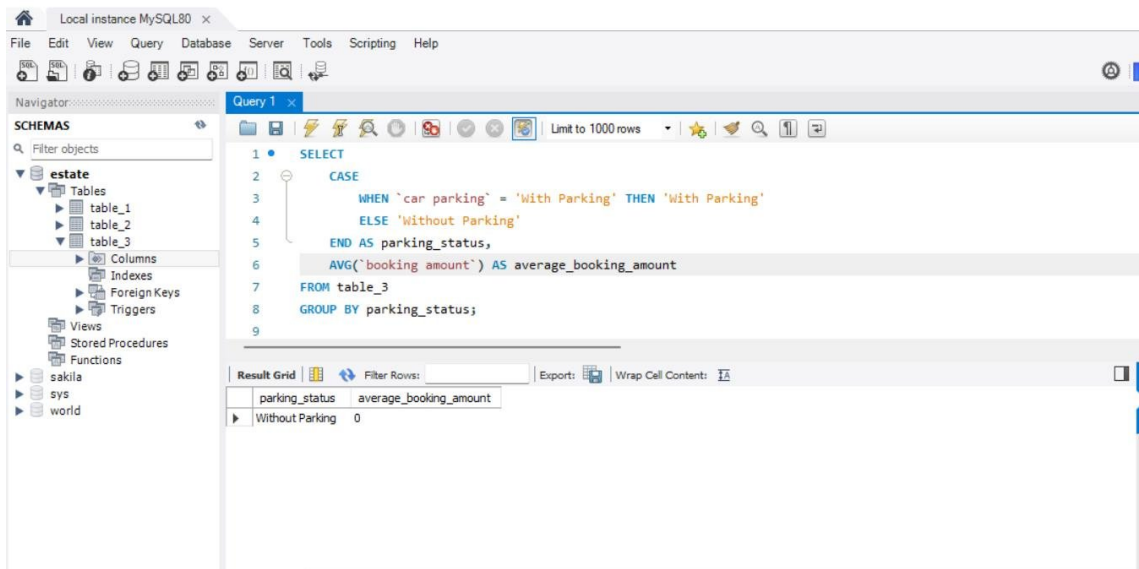


The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'estate' selected, containing 'table\_1', 'table\_2', and 'table\_3'. The main query editor shows the following SQL query:

```
1 • SELECT * FROM table_3
2 WHERE `furnished status` = 'Fully Furnished' AND `facing` = 'East';
3
```

The 'Result Grid' at the bottom shows the columns: serial number, furnished status, car parking, type of ownership, booking amount, buy total price, and facing. The query is limited to 1000 rows.

2. Calculate the average booking amount for properties with and without car parking:



The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'estate' selected, containing 'table\_1', 'table\_2', and 'table\_3'. The main query editor shows the following SQL query:

```
1 • SELECT
2 CASE
3 WHEN `car parking` = 'With Parking' THEN 'With Parking'
4 ELSE 'Without Parking'
5 END AS parking_status,
6 AVG(`booking amount`) AS average_booking_amount
7 FROM table_3
8 GROUP BY parking_status;
9
```

The 'Result Grid' at the bottom shows the columns: parking\_status and average\_booking\_amount. The query is limited to 1000 rows. The result shows one row for 'Without Parking' with an average booking amount of 0.

3. Find the total price of properties with different types of ownership

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'estate' expanded, showing 'table\_1', 'table\_2', and 'table\_3'. The 'Columns' for 'table\_3' are listed: serial number, furnished status, car parking, type of ownership, booking amount, buy total price, and facing. The main query editor contains the following SQL:

```
1 SELECT
2   `type of ownership`,
3   SUM(`buy total price`) AS total_price
4 FROM table_3
5 GROUP BY `type of ownership`;
```

The 'Result Grid' at the bottom shows the following data:

| type of ownership | total_price |
|-------------------|-------------|
| 235561340074      |             |

4. Retrieve properties with a booking amount greater than 50000 and a furnished status of 'Semi Furnished'.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'estate' expanded, showing 'table\_1', 'table\_2', and 'table\_3'. The 'Columns' for 'table\_3' are listed: serial number, furnished status, car parking, type of ownership, booking amount, buy total price, and facing. The main query editor contains the following SQL:

```
1 SELECT * FROM table_3
2 WHERE `booking amount` > 50000
3 AND `furnished status` = 'Semi Furnished';
```

The 'Result Grid' at the bottom shows the following data:

| serial number | furnished status | car parking | type of ownership | booking amount | buy total price | facing |
|---------------|------------------|-------------|-------------------|----------------|-----------------|--------|
|---------------|------------------|-------------|-------------------|----------------|-----------------|--------|

5. Find the property with the highest booking amount.

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' pane displays a tree view of the database structure. The 'estate' database is selected, showing tables 'table\_1', 'table\_2', and 'table\_3'. The 'Columns' pane for 'table\_3' lists: serial number, furnished status, car parking, type of ownership, booking amount, buy total price, and facing. The 'Query' pane shows the following SQL query:

```
1 • SELECT * FROM table_3
2 ORDER BY `booking amount` DESC
3 LIMIT 1;
4
5
6
7
```

The 'Result Grid' at the bottom displays the query results in a table format:

|     | serial number | furnished status | car parking | type of ownership | booking amount | buy total price | facing |
|-----|---------------|------------------|-------------|-------------------|----------------|-----------------|--------|
| ▶ 1 |               |                  |             |                   | 7772014        |                 | NULL   |