

HOTEL RESERVATION ANALYSIS WITH SQL


By Suprakrit Debnath
Batch: MIP-DA-10

INTRODUCTION

Objective: Welcome to this presentation on analyzing a hotel reservation dataset to uncover insights into guest preferences, booking trends, and operational factors crucial to enhancing guest experiences and optimizing hotel operations.

Three parallel white diagonal lines are positioned in the bottom right corner of the slide, extending from the middle of the right edge towards the bottom left.

DATASET OVERVIEW: THE DATASET WE ARE EXPLORING CONTAINS COMPREHENSIVE INFORMATION ABOUT HOTEL RESERVATIONS, INCLUDING BOOKING DETAILS, GUEST DEMOGRAPHICS, STAY DURATION, AND BOOKING STATUS. BY LEVERAGING SQL QUERIES, WE AIM TO EXTRACT MEANINGFUL INSIGHTS THAT WILL INFORM STRATEGIC DECISIONS AND IMPROVE OVERALL SERVICE DELIVERY.

Three white diagonal lines of varying lengths and positions are located on the right side of the slide, extending from the middle towards the bottom right corner.

COLUMNS

Administration Schemas

Information

Table: hotel reservation dataset

Columns:

Booking_ID	text
no_of_adults	int
no_of_children	int
no_of_weekend_nights	int
no_of_week_nights	int
type_of_meal_plan	text
room_type_reserved	text
lead_time	int
arrival_date	text
market_segment_type	text
avg_price_per_room	double
booking_status	text

Object Info

Session

The screenshot displays the MySQL Workbench interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The left sidebar shows the 'SCHEMAS' panel with a tree view of 'new_schema' containing 'hotel reservation dataset', 'Views', 'Stored Procedures', and 'Functions'. The main workspace shows a query editor with the following SQL query:

```
1 select * from `hotel reservation dataset`
2
3
```

Below the query editor, the 'Result Grid' is displayed, showing the results of the query. The grid has columns: Booking_ID, no_of_adults, no_of_children, no_of_weekend_nights, no_of_week_nights, type_of_meal_plan, room_type_reserved, lead_time, and arrival_date. The results are as follows:

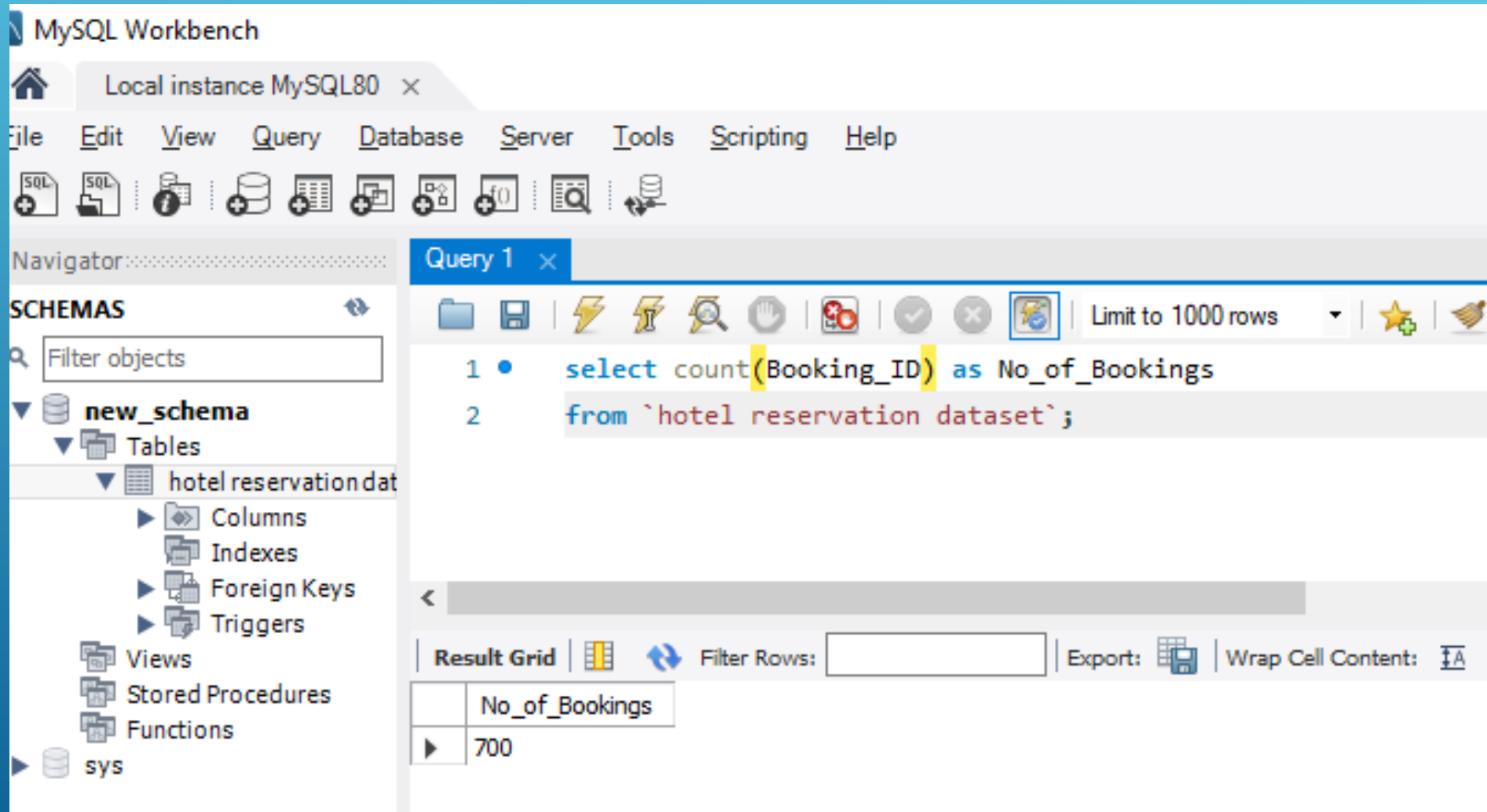
Booking_ID	no_of_adults	no_of_children	no_of_weekend_nights	no_of_week_nights	type_of_meal_plan	room_type_reserved	lead_time	arrival_date
INN00001	2	0	1	2	Meal Plan 1	Room_Type 1	224	02-10-2017
INN00002	2	0	2	3	Not Selected	Room_Type 1	5	06-11-2018
INN00003	1	0	2	1	Meal Plan 1	Room_Type 1	1	28-02-2018
INN00004	2	0	0	2	Meal Plan 1	Room_Type 1	211	20-05-2018
INN00005	2	0	1	1	Not Selected	Room_Type 1	48	11-04-2018
INN00006	2	0	0	2	Meal Plan 2	Room_Type 1	346	13-09-2018
INN00007	2	0	1	3	Meal Plan 1	Room_Type 1	34	15-10-2017
INN00008	2	0	1	3	Meal Plan 1	Room_Type 4	83	26-12-2018
INN00009	3	0	0	4	Meal Plan 1	Room_Type 1	121	06-07-2018

Below the result grid, the 'Output' panel shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	00:56:49	select * from `hotel reservation dataset` LIMIT 0, 1000	700 row(s) returned	0.000 sec / 0.000 sec

The right sidebar contains the 'SQLAdditions' panel with a 'Jump to' button and a message: 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.'

1. TOTAL NUMBER OF RESERVATIONS



The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' pane shows a database named 'new_schema' containing a table 'hotel reservation dat'. The main editor displays a SQL query in 'Query 1':

```
1 • select count(Booking_ID) as No_of_Bookings
2   from `hotel reservation dataset`;
```

The query is executed, and the 'Result Grid' at the bottom shows the result:

No_of_Bookings
700

2. MOST POPULAR MEAL PLAN

MySQL Workbench

Local instance MySQL80 x unconnected x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

new_schema

- Tables
 - hotel reservation dat
- Views
- Stored Procedures
- Functions

sys

Query 1 x

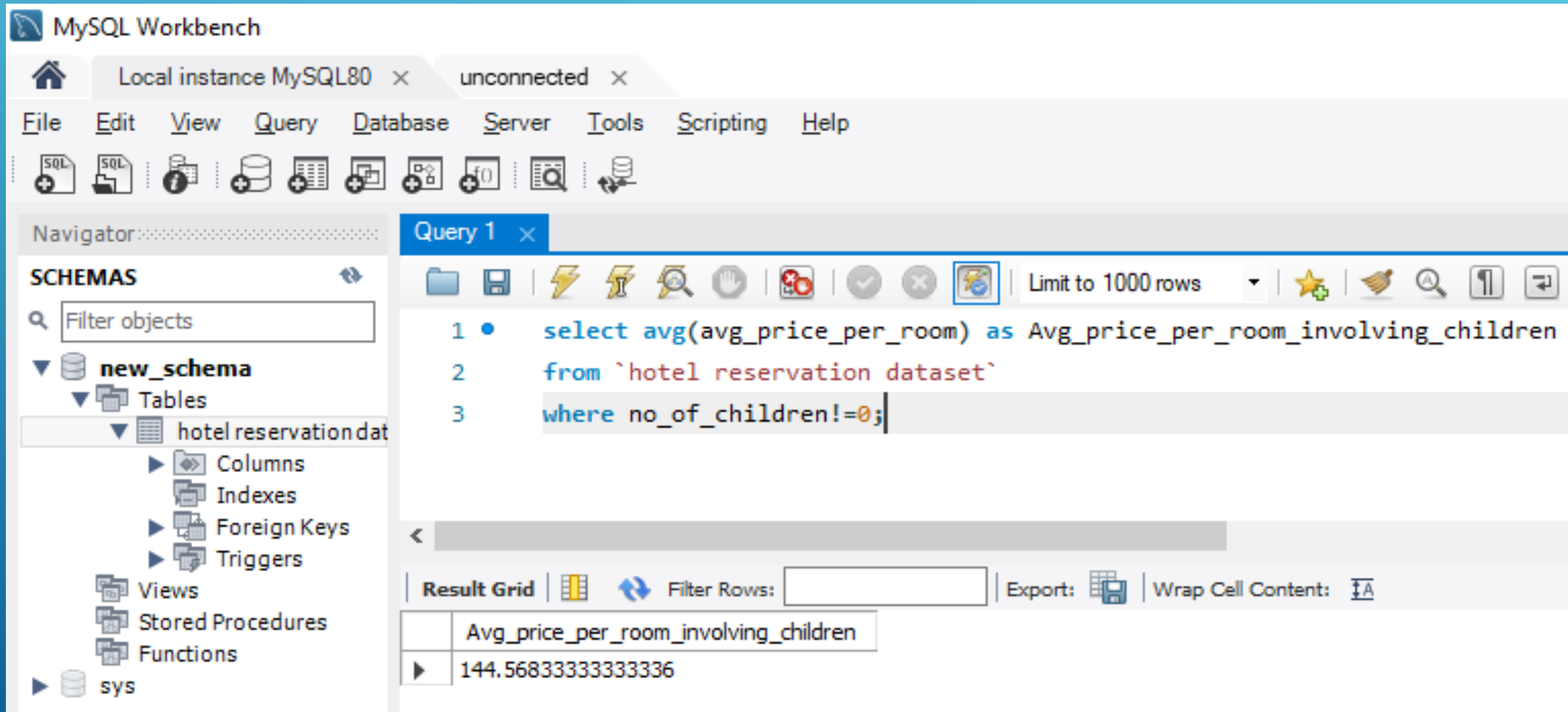
```
1 • select type_of_meal_plan, COUNT(type_of_meal_plan) as no_of_times_ordered
2   from `hotel reservation dataset`
3   group by type_of_meal_plan
4   order by no_of_times_ordered desc;
```

Limit to 1000 rows

Result Grid

	type_of_meal_plan	no_of_times_ordered
▶	Meal Plan 1	527
	Not Selected	109
	Meal Plan 2	64

3. AVERAGE PRICE PER ROOM FOR RESERVATIONS INVOLVING CHILDREN



The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'new_schema' expanded, showing 'hotel reservation dat' under 'Tables'. The main editor window, titled 'Query 1', contains the following SQL query:

```
1 • select avg(avg_price_per_room) as Avg_price_per_room_involving_children
2   from `hotel reservation dataset`
3   where no_of_children!=0;
```

Below the query editor, the 'Result Grid' is visible, showing the result of the query:

Avg_price_per_room_involving_children
144.56833333333336

4. MOST COMMONLY BOOKED ROOM TYPE

MySQL Workbench

Local instance MySQL80 x unconnected x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

new_schema

Tables

hotel reservation dat

Views

Stored Procedures

Functions

sys

Query 1 x

Limit to 1000 rows

```
1 select room_type_reserved, COUNT(room_type_reserved) as max_no_of_times_booked_room
2 from `hotel reservation dataset`
3 group by room_type_reserved
4 order by max_no_of_times_booked_room desc;
```

Result Grid

Filter Rows:

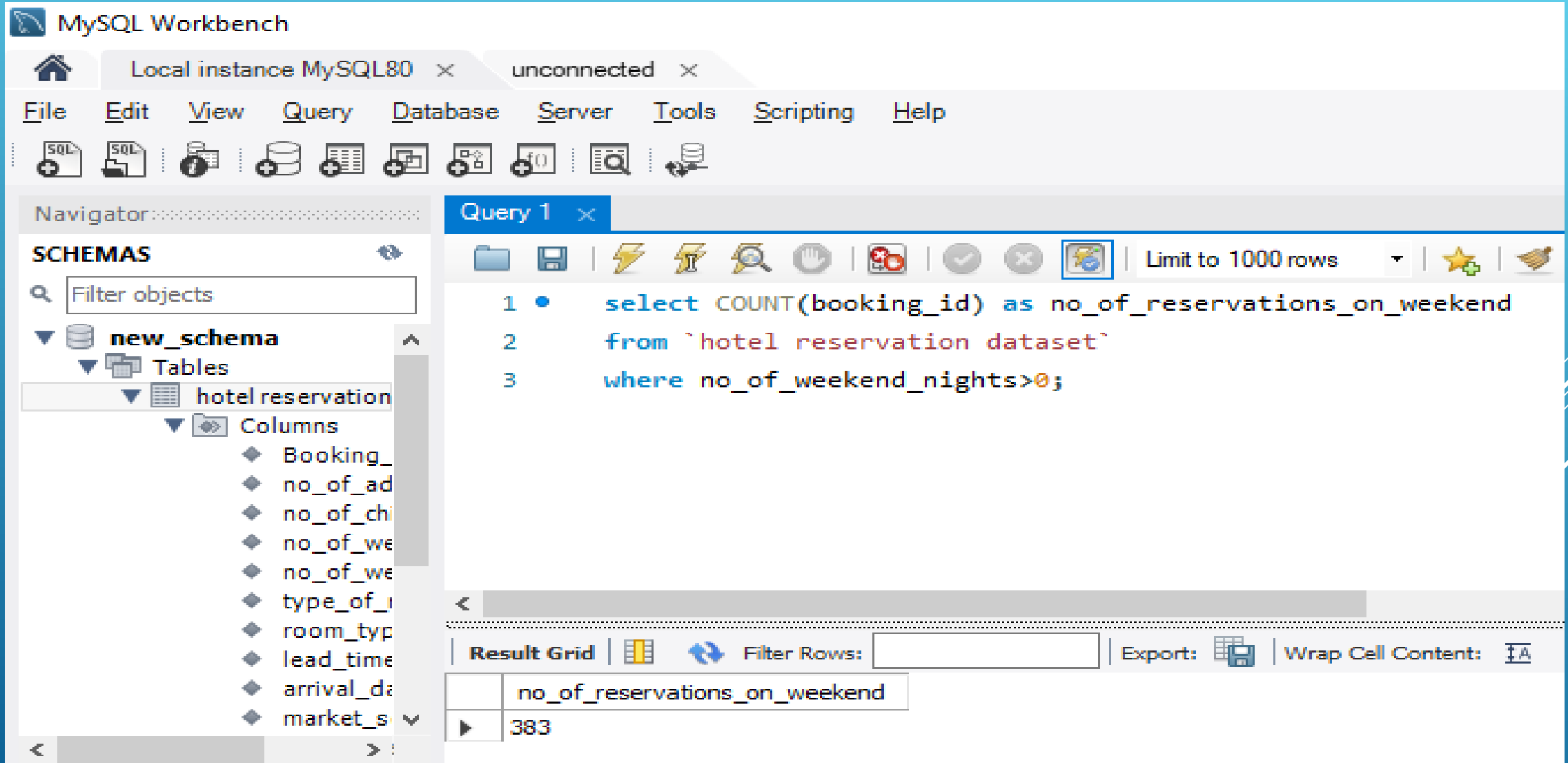
Export:

Wrap Cell Content:

room_type_reserved	max_no_of_times_booked_room
Room_Type 1	534
Room_Type 4	130
Room_Type 6	18
Room_Type 2	8
Room_Type 7	6
Room_Type 5	4

Administration Schemas

5. RESERVATIONS ON WEEKENDS



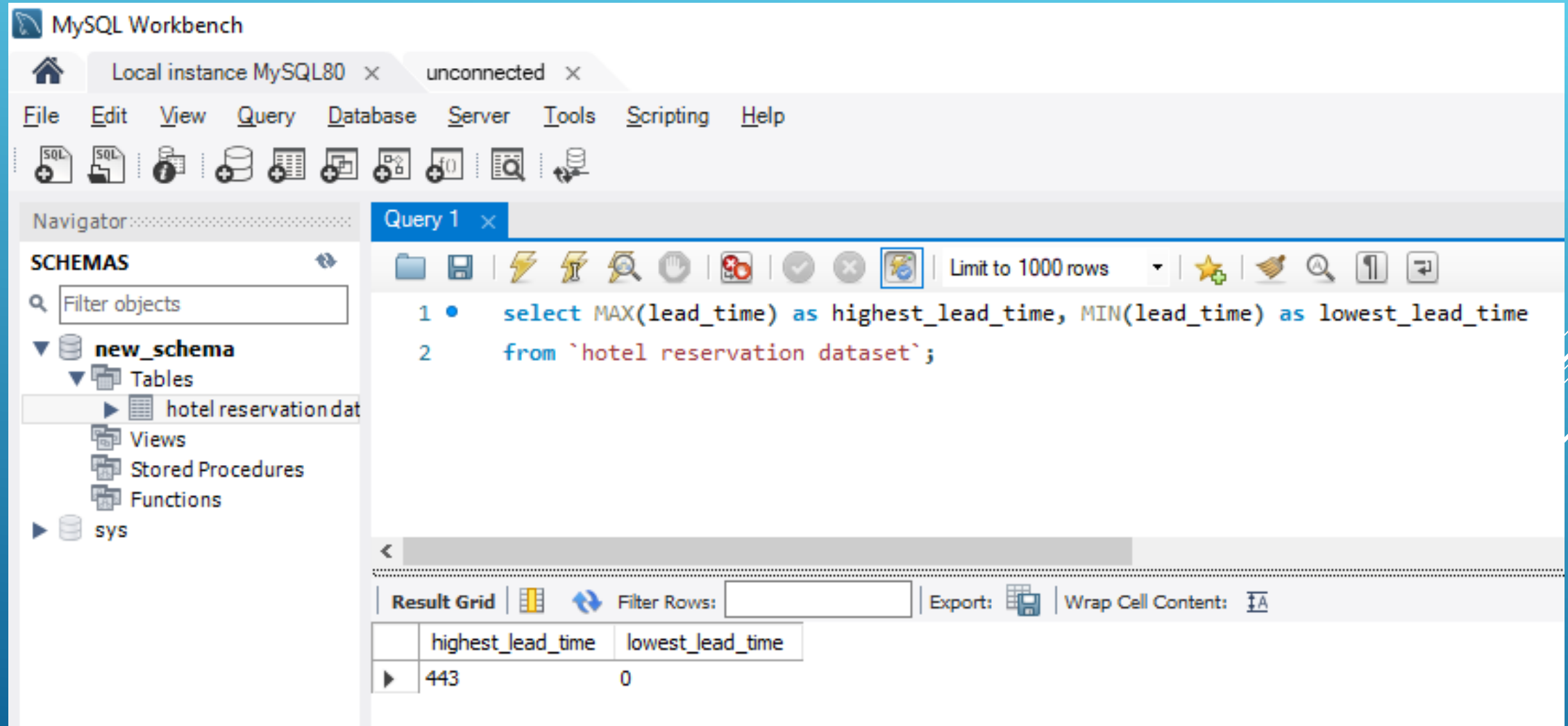
The screenshot shows the MySQL Workbench interface. On the left, the Navigator pane displays the 'new_schema' database with a table named 'hotel reservation'. The table's columns are listed: Booking_id, no_of_adults, no_of_children, no_of_weekend_nights, no_of_weeknights, type_of_booking, room_type, lead_time, arrival_date, and market_segment. The main query editor displays the following SQL query:

```
1 • select COUNT(booking_id) as no_of_reservations_on_weekend
2   from `hotel reservation dataset`
3   where no_of_weekend_nights>0;
```

Below the query editor, the 'Result Grid' tab is active, showing the query's output:

	no_of_reservations_on_weekend
▶	383

6. HIGHEST AND LOWEST LEAD TIME



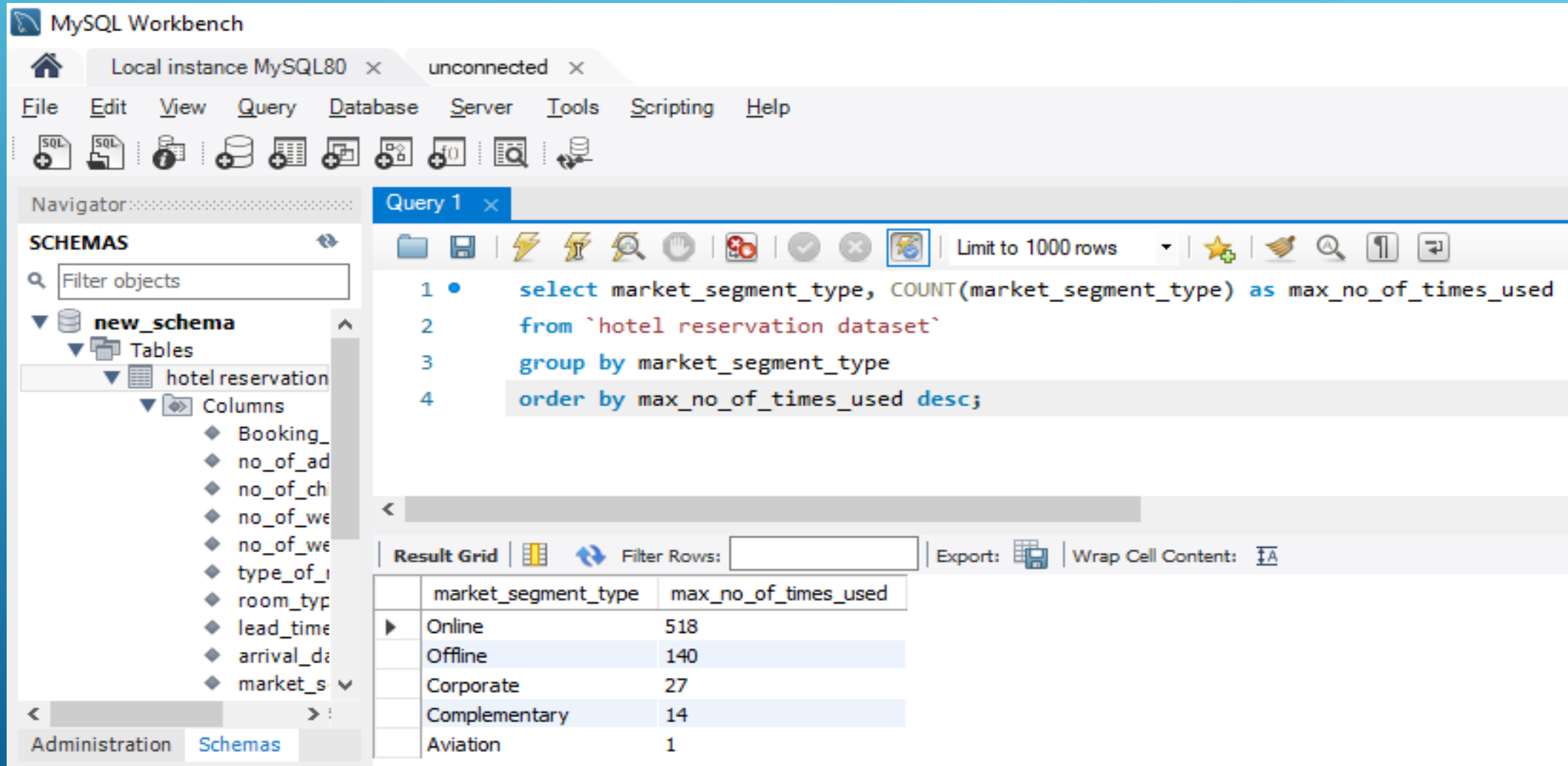
The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'new_schema' expanded, showing 'hotel reservation dat' under 'Tables'. The main editor window, titled 'Query 1', contains the following SQL query:

```
1 • select MAX(lead_time) as highest_lead_time, MIN(lead_time) as lowest_lead_time
2   from `hotel reservation dataset`;
```

Below the query editor, the 'Result Grid' is visible, showing the results of the query:

	highest_lead_time	lowest_lead_time
▶	443	0

7. MOST COMMON MARKET SEGMENT TYPE



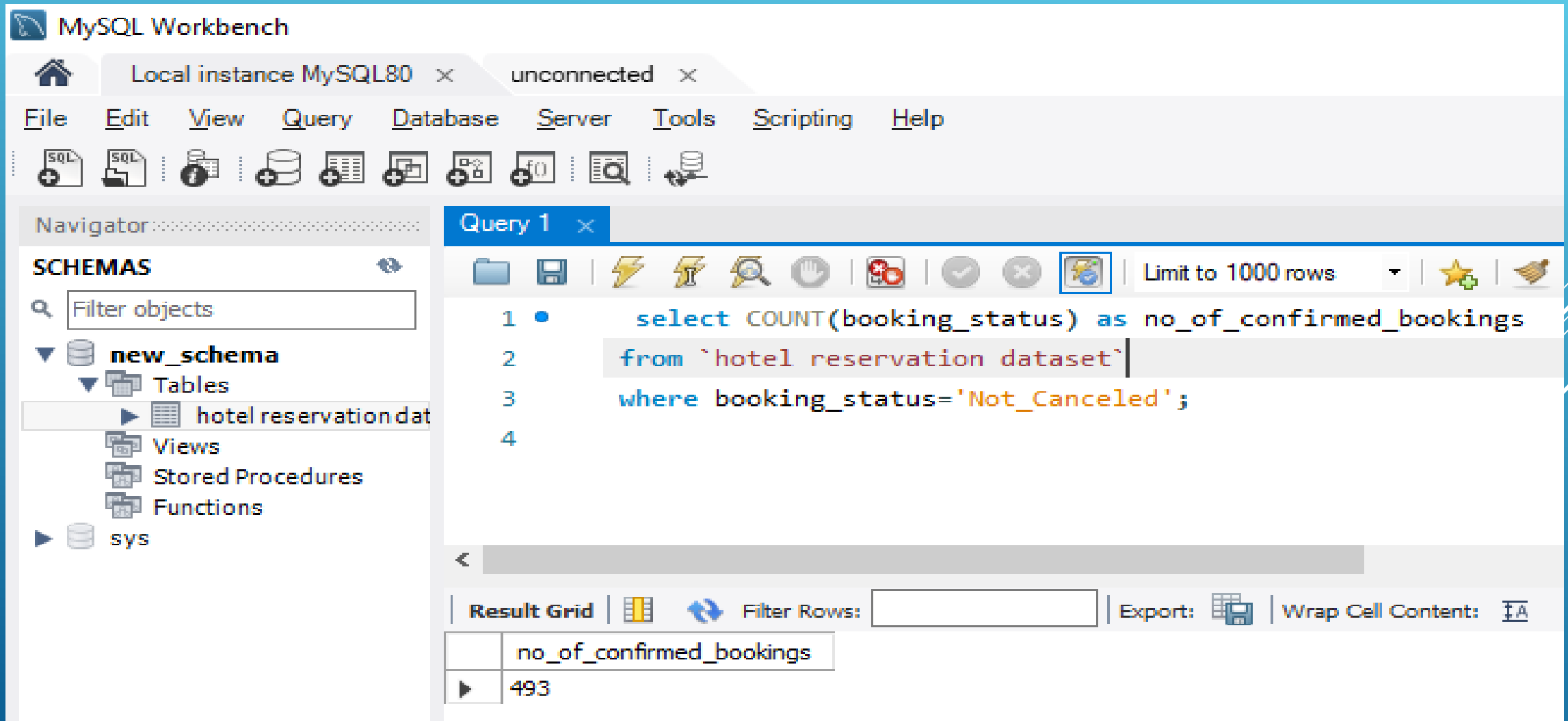
The screenshot displays the MySQL Workbench interface. On the left, the 'Navigator' pane shows the 'Schemas' section with a search filter 'Filter objects'. Under 'new_schema', the 'hotel reservation' table is expanded, showing columns like 'Booking_', 'no_of_ad', 'no_of_chi', 'no_of_we', 'no_of_we', 'type_of_i', 'room_typ', 'lead_time', 'arrival_da', and 'market_s'. The main area shows 'Query 1' with the following SQL code:

```
1 • select market_segment_type, COUNT(market_segment_type) as max_no_of_times_used
2   from `hotel reservation dataset`
3   group by market_segment_type
4   order by max_no_of_times_used desc;
```

Below the query editor, the 'Result Grid' is visible, showing the results of the query. The grid has two columns: 'market_segment_type' and 'max_no_of_times_used'. The results are as follows:

market_segment_type	max_no_of_times_used
Online	518
Offline	140
Corporate	27
Complementary	14
Aviation	1

8. RESERVATIONS WITH CONFIRMED BOOKING STATUS



The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'new_schema' expanded, showing 'hotel reservation dat' under 'Tables'. The main editor window, titled 'Query 1', contains the following SQL query:

```
1 select COUNT(booking_status) as no_of_confirmed_bookings
2 from `hotel reservation dataset`
3 where booking_status='Not_Canceled';
4
```

Below the query editor, the 'Result Grid' is visible, showing the output of the query:

	no_of_confirmed_bookings
▶	493

9. TOTAL NUMBER OF ADULTS AND CHILDREN

MySQL Workbench

Local instance MySQL80 x unconnected x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

new_schema

Tables

hotel reservation

Columns

- Booking_
- no_of_ad
- no_of_chi
- no_of_we
- no_of_we
- type_of_i
- room_typ
- lead_time
- arrival_da
- market s

Query 1 x

Limit to 1000 rows

```
1 • select sum(no_of_adults) as Total_no_of_adults,
2       sum(no_of_children) as Total_no_of_children,
3       sum(no_of_adults) + sum(no_of_children)
4 from `hotel reservation dataset`;
5
6
```

Result Grid

	Total_no_of_adults	Total_no_of_children	sum(no_of_adults) + sum(no_of_children)
▶	1316	69	1385

10. AVERAGE WEEKEND NIGHTS FOR RESERVATIONS INVOLVING CHILDREN

MySQL Workbench

Local instance MySQL80 x unconnected x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

new_schema

Tables

hotel reservation dat

Views

Stored Procedures

Functions

sys

Query 1 x

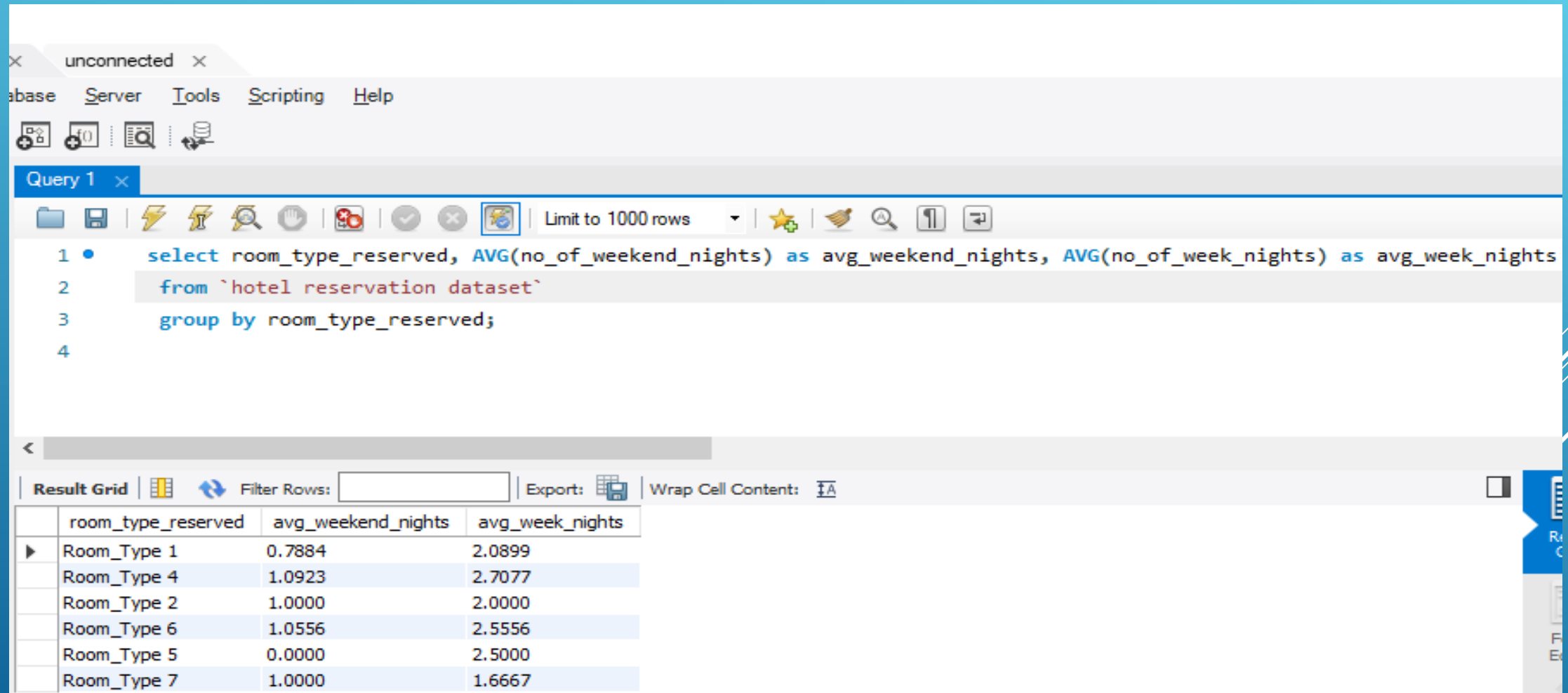
Limit to 1000 rows

```
1 • select AVG(no_of_weekend_nights) as avg_of_weekend_nights
2   from `hotel reservation dataset`
3   where no_of_children!=0;
4
5
6
```

Result Grid

	avg_of_weekend_nights
▶	1.0000

11. AVERAGE NIGHTS SPENT BY GUESTS FOR EACH ROOM TYPE



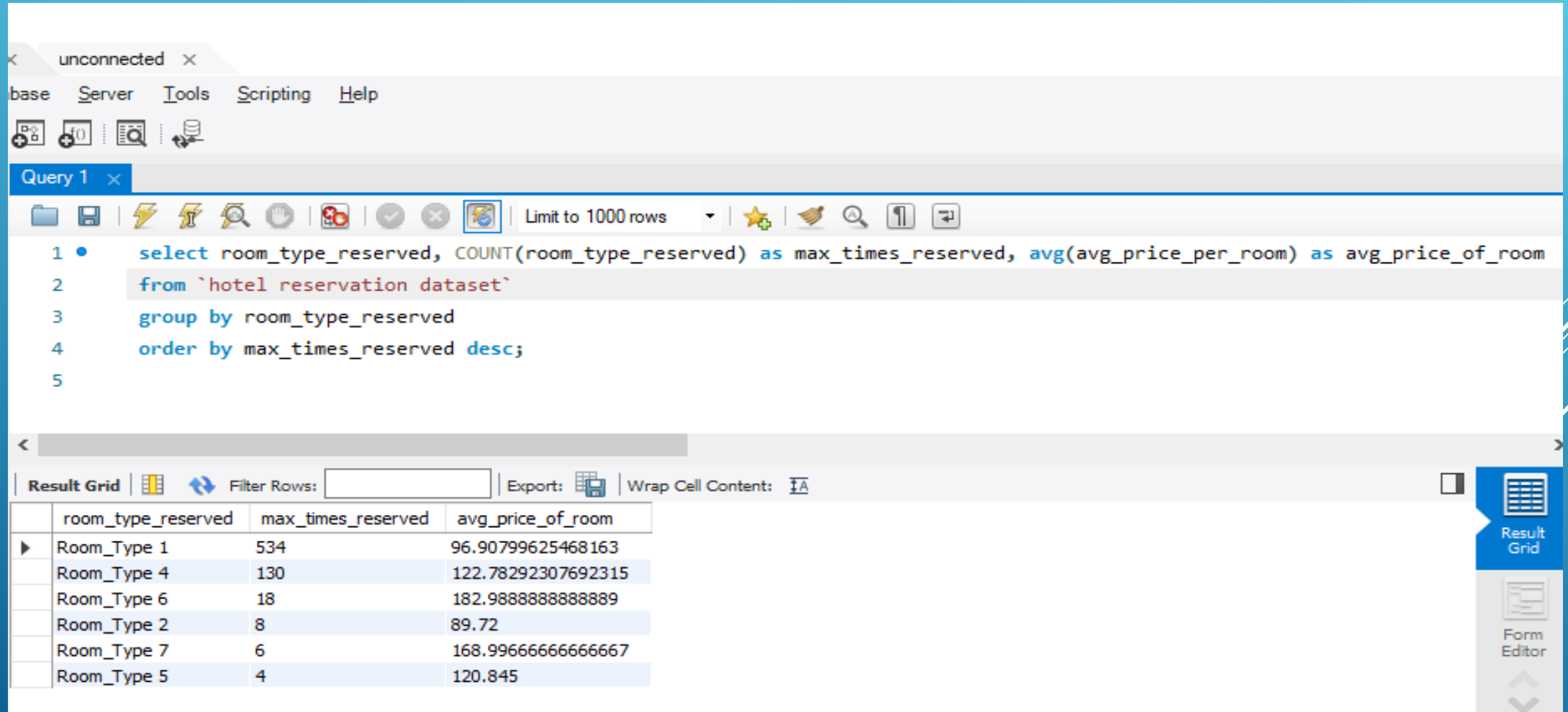
The screenshot shows a database query tool interface. At the top, there's a menu bar with 'Database', 'Server', 'Tools', 'Scripting', and 'Help'. Below the menu is a toolbar with various icons. The main area displays a SQL query in a text editor, labeled 'Query 1'. The query is as follows:

```
1 • select room_type_reserved, AVG(no_of_weekend_nights) as avg_weekend_nights, AVG(no_of_week_nights) as avg_week_nights
2   from `hotel reservation dataset`
3   group by room_type_reserved;
4
```

Below the query editor, there's a 'Result Grid' section. It includes a 'Filter Rows' input field, an 'Export' button, and a 'Wrap Cell Content' checkbox. The results are displayed in a table with the following data:

	room_type_reserved	avg_weekend_nights	avg_week_nights
▶	Room_Type 1	0.7884	2.0899
	Room_Type 4	1.0923	2.7077
	Room_Type 2	1.0000	2.0000
	Room_Type 6	1.0556	2.5556
	Room_Type 5	0.0000	2.5000
	Room_Type 7	1.0000	1.6667

12. MOST COMMON ROOM TYPE WITH CHILDREN AND AVERAGE PRICE



The screenshot shows a database query tool interface. At the top, there's a menu bar with 'base', 'Server', 'Tools', 'Scripting', and 'Help'. Below it is a toolbar with various icons. The main area displays a SQL query in a text editor, labeled 'Query 1'. The query is as follows:

```
1 • select room_type_reserved, COUNT(room_type_reserved) as max_times_reserved, avg(avg_price_per_room) as avg_price_of_room
2 from `hotel reservation dataset`
3 group by room_type_reserved
4 order by max_times_reserved desc;
5
```

Below the query editor, there's a 'Result Grid' section. It includes a 'Filter Rows' input field, an 'Export' button, and a 'Wrap Cell Content' checkbox. The results are displayed in a table with three columns: 'room_type_reserved', 'max_times_reserved', and 'avg_price_of_room'. The table shows the following data:

room_type_reserved	max_times_reserved	avg_price_of_room
Room_Type 1	534	96.90799625468163
Room_Type 4	130	122.78292307692315
Room_Type 6	18	182.98888888888889
Room_Type 2	8	89.72
Room_Type 7	6	168.99666666666667
Room_Type 5	4	120.845

On the right side of the interface, there are two buttons: 'Result Grid' and 'Form Editor'.

13. MARKET SEGMENT TYPE WITH HIGHEST AVERAGE PRICE

MySQL Workbench

Local instance MySQL80 x unconnected x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

new_schema

Tables

hotel reservation dat

Views

Stored Procedures

Functions

sys

Query 1 x

Limit to 1000 rows

```
1 select market_segment_type, max(avg_price_per_room) as highest_of_avg_price_of_room
2 from `hotel reservation dataset`
3 group by market_segment_type
4 order by highest_of_avg_price_of_room desc;
5
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

market_segment_type	highest_of_avg_price_of_room
Online	258
Offline	192.03
Corporate	160
Aviation	110
Complementary	29

Administration Schemas

14. RESERVATIONS MADE IN EACH MONTH OF THE YEAR

```
118 • SELECT
119     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 1 THEN 1 ELSE 0 END) AS January,
120     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 2 THEN 1 ELSE 0 END) AS February,
121     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 3 THEN 1 ELSE 0 END) AS March,
122     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 4 THEN 1 ELSE 0 END) AS April,
123     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 5 THEN 1 ELSE 0 END) AS May,
124     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 6 THEN 1 ELSE 0 END) AS June,
125     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 7 THEN 1 ELSE 0 END) AS July,
126     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 8 THEN 1 ELSE 0 END) AS August,
127     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 9 THEN 1 ELSE 0 END) AS September,
128     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 10 THEN 1 ELSE 0 END) AS October,
129     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 11 THEN 1 ELSE 0 END) AS November,
130     SUM(CASE WHEN MONTH(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 12 THEN 1 ELSE 0 END) AS December
131 FROM
132     hotel_reservation;
```

	January	February	March	April	May	June	July	August	September	October	November	December
►	11	28	52	67	55	84	44	70	80	103	54	52

15. RESERVATIONS MADE FOR A SPECIFIC YEAR

```
SELECT  
    COUNT(*) AS total_reservations_2017  
FROM  
    hotel_reservation  
WHERE  
    YEAR(STR_TO_DATE(arrival_date, '%d-%m-%Y')) = 2017;
```

Result Grid |   Filter Rows:

	total_reservations_2017
▶	123

CONCLUSION

The hotel reservation dataset provides valuable insights into guest preferences and booking trends. Key findings include Meal Plan 1 as the favoured choice among guests, with Room type 1 being the most frequently booked. Analysis of lead times reveals varied booking behaviours, while the majority of reservations are confirmed, indicating robust booking management. Market segmentation highlights online, influencing pricing strategies. Reservations involving children command 125+ , reflecting family-friendly offerings. These insights equip hotels to refine marketing approaches and optimize guest experiences, ensuring competitive edge and guest satisfaction.