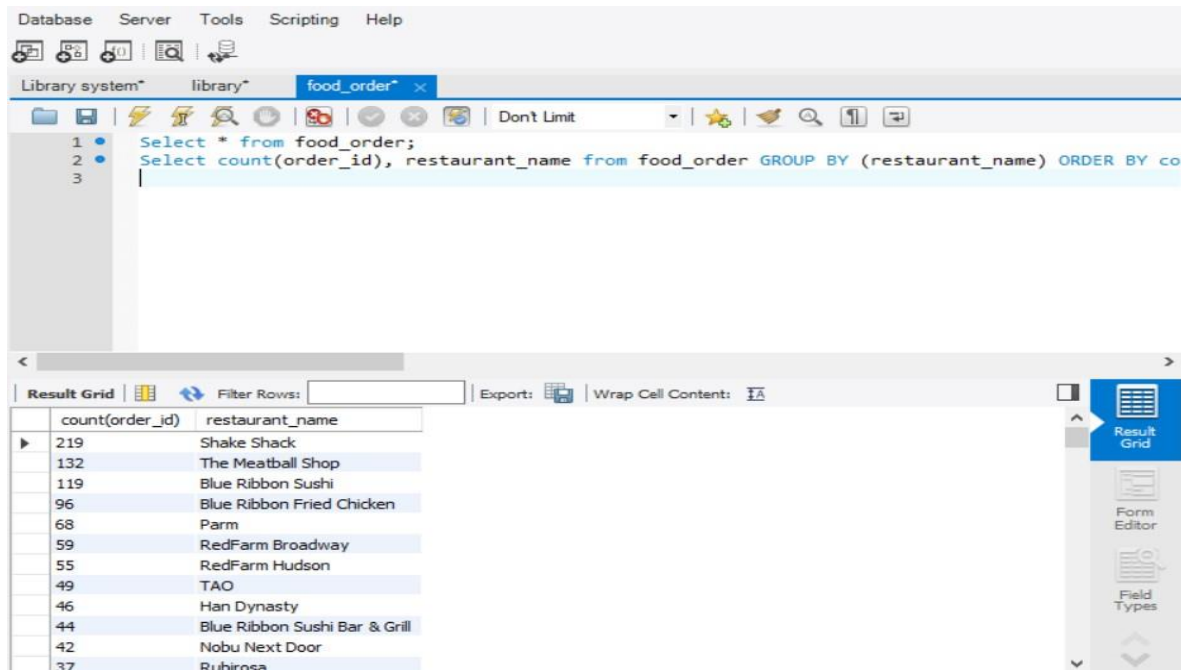


# FOOD ORDERS EDA PROJECT

Select \* from food\_order;

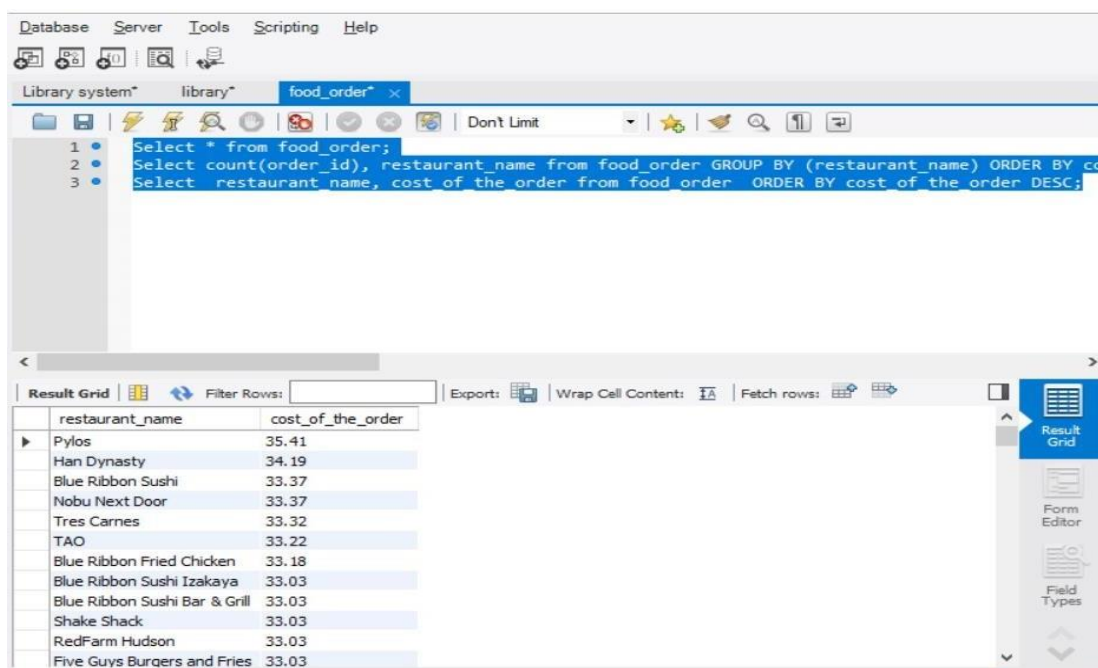
Select count(order\_id), restaurant\_name from food\_order GROUP BY (restaurant\_name) ORDER BY count(order\_id) DESC;



The screenshot shows a database IDE with a menu bar (Database, Server, Tools, Scripting, Help) and a toolbar. The 'Scripting' tab is active, showing a SQL query in a text editor. The query is: `Select * from food_order;` followed by `Select count(order_id), restaurant_name from food_order GROUP BY (restaurant_name) ORDER BY count(order_id) DESC;`. Below the editor, the 'Result Grid' is displayed, showing the results of the second query. The grid has two columns: 'count(order\_id)' and 'restaurant\_name'. The results are sorted in descending order of count.

count(order_id)	restaurant_name
219	Shake Shack
132	The Meatball Shop
119	Blue Ribbon Sushi
96	Blue Ribbon Fried Chicken
68	Parm
59	RedFarm Broadway
55	RedFarm Hudson
49	TAO
46	Han Dynasty
44	Blue Ribbon Sushi Bar & Grill
42	Nobu Next Door
37	Rubirosa

Select restaurant\_name, cost\_of\_the\_order from food\_order ORDER BY cost\_of\_the\_order DESC;



The screenshot shows the same database IDE with a new SQL query in the 'Scripting' tab. The query is: `Select restaurant_name, cost of the order from food order ORDER BY cost of the order DESC;`. The 'Result Grid' displays the results of this query, showing two columns: 'restaurant\_name' and 'cost\_of\_the\_order'. The results are sorted in descending order of cost.

restaurant_name	cost_of_the_order
Pylos	35.41
Han Dynasty	34.19
Blue Ribbon Sushi	33.37
Nobu Next Door	33.37
Tres Carnes	33.32
TAO	33.22
Blue Ribbon Fried Chicken	33.18
Blue Ribbon Sushi Izakaya	33.03
Blue Ribbon Sushi Bar & Grill	33.03
Shake Shack	33.03
RedFarm Hudson	33.03
Five Guys Burgers and Fries	33.03

Select restaurant\_name, delivery\_time from food\_order WHERE delivery\_time > 30 ORDER BY delivery\_time DESC ;

The screenshot shows a database query editor with a SQL query in the top pane and its results in the bottom pane. The query is:

```

1  ;
2  restaurant_name from food_order GROUP BY (restaurant_name) ORDER BY count(order_id) DESC;
3  , cost of the order from food_order ORDER BY cost of the order DESC;
4  delivery time from food_order WHERE delivery_time > 30 ORDER BY delivery_time DESC ;
5

```

The results are displayed in a grid with two columns: restaurant\_name and delivery\_time. The data is sorted by delivery\_time in descending order.

restaurant_name	delivery_time
RedFarm Hudson	33
Blue Ribbon Fried Chicken	33
Blue Ribbon Fried Chicken	33
Sushi of Gari 46	33
Parm	33
Shake Shack	33
Tarallucci e Vino Restaurant	33
The Meatball Shop	33
Chipotle Mexican Grill \$1.99 Delivery	33
TAO	33
Melt Shop	33
...	...

Select count(order\_id), cuisine\_type from food\_order group by(cuisine\_type) ORDER BY cuisine\_type DESC;

The screenshot shows a database query editor with a SQL query in the top pane and its results in the bottom pane. The query is:

```

1  order GROUP BY (restaurant_name) ORDER BY count(order_id) DESC;
2  od_order ORDER BY cost_of_the_order DESC;
3  rder WHERE delivery_time > 30 ORDER BY delivery_time DESC ;
4  er) ,avg(cost_of_the_order) FROM food_order;
5  er group by(cuisine_type) ORDER BY count(order_id) DESC;
6

```

The results are displayed in a grid with two columns: count(order\_id) and cuisine\_type. The data is sorted by cuisine\_type in descending order.

count(order_id)	cuisine_type
584	American
470	Japanese
298	Italian
215	Chinese
77	Mexican
73	Indian
49	Middle Eastern
46	Mediterranean
19	Thai
18	French
17	Southern
...	...



SELECT min(cost\_of\_the\_order), max(cost\_of\_the\_order) ,avg(cost\_of\_the\_order) FROM food\_order;

```

1 •  food_order;
2 •  (order_id), restaurant_name from food_order GROUP BY (restaurant_name) ORDER BY count(order_id)
3 •  urant_name, cost_of_the_order from food_order ORDER BY cost_of_the_order DESC;
4 •  urant name, delivery_time from food_order WHERE delivery_time > 30 ORDER BY delivery_time D
5 •  ost of the order), max(cost of the order) ,avg(cost of the order) FROM food_order;
6 •  (order_id), cuisine_type from food_order group by(cuisine_type) ORDER BY count(order_id) DESC;
  
```

Result Grid

	min(cost_of_the_order)	max(cost_of_the_order)	avg(cost_of_the_order)
▶	4.47	35.41	16.498851422550082

Result Grid

- **Insights** - Kambi Ramen House has higher average costs per order, should carry out analyses to match their pricing models with consumer demands 2) With the highest average rating of 4.83, Spanish food comes in first.
- This knowledge can help direct marketing campaigns and shape promotional plans in order to take advantage of the favorable reaction to Spanish food offerings.