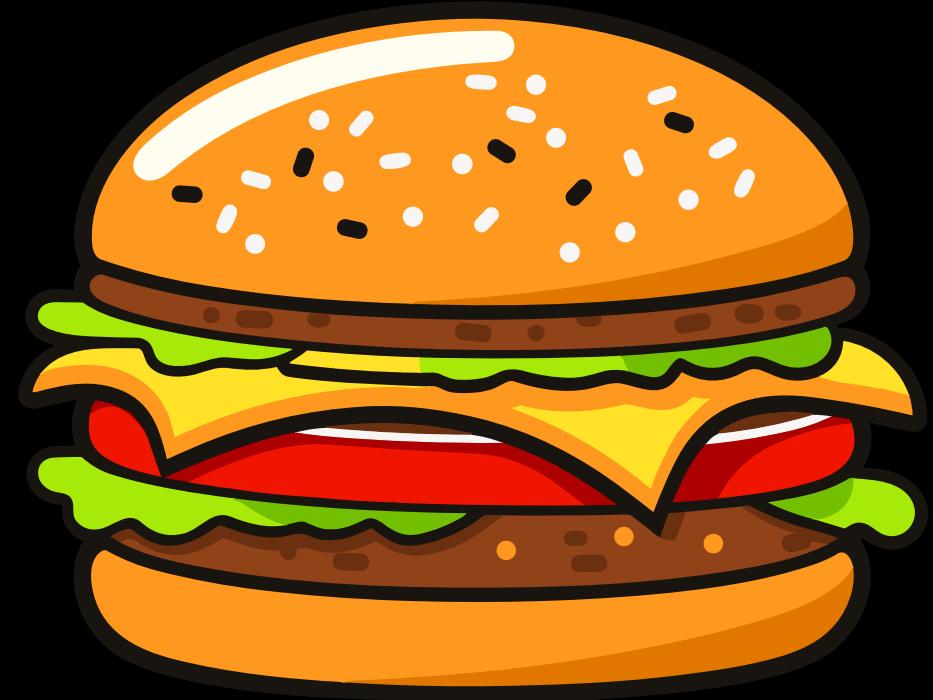
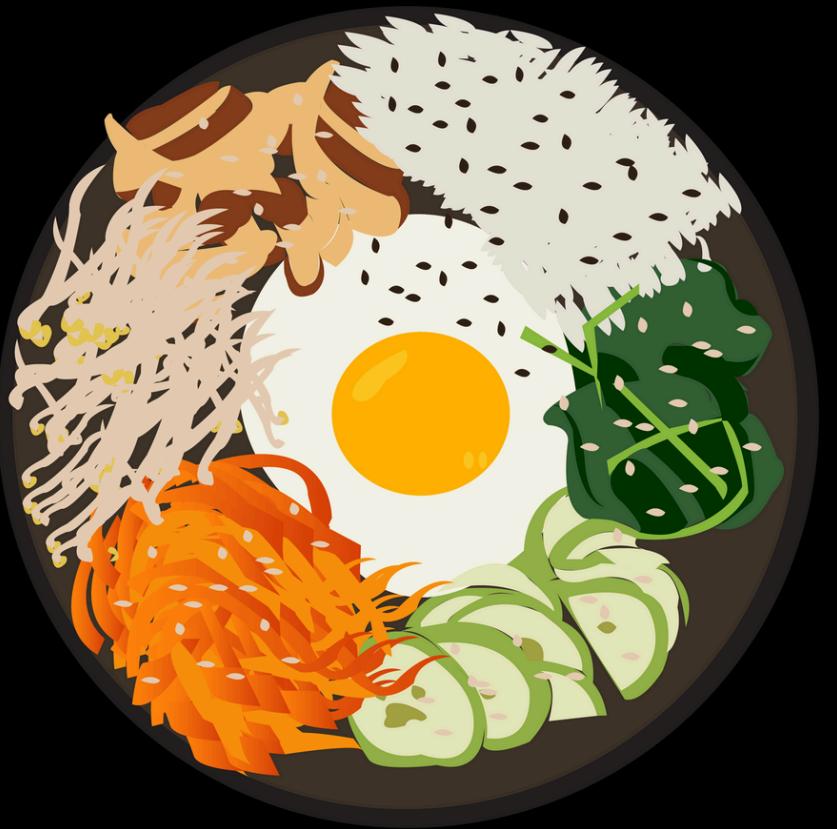


SQL PROJECT

Restaurant Order Analysis



Explore Menu_items Table

1. View the menu_items table

Query :

```
SELECT * FROM menu_items;
```

menu_item_id	item_name	category	price
101	Hamburger	American	12.95
102	Cheeseburger	American	13.95
103	Hot Dog	American	9.00
104	Veggie Burger	American	10.50
105	Mac & Cheese	American	7.00
106	French Fries	American	7.00
107	Orange Chicken	Asian	16.50
108	Tofu Pad Thai	Asian	14.50
109	Korean Beef Bowl	Asian	17.95
110	Pork Ramen	Asian	17.95
111	California Roll	Asian	11.95
112	Salmon Roll	Asian	14.95
113	Edamame	Asian	5.00
114	Potstickers	Asian	9.00
115	Chicken Tacos	Mexican	11.95
116	Steak Tacos	Mexican	13.95
117	Chicken Burrito	Mexican	12.95

118	Steak Burrito	Mexican	14.95
119	Chicken Torta	Mexican	11.95
120	Steak Torta	Mexican	13.95
121	Cheese Quesadillas	Mexican	10.50
122	Chips & Salsa	Mexican	7.00
123	Chips & Guacamole	Mexican	9.00
124	Spaghetti	Italian	14.50
125	Spaghetti & Mea...	Italian	17.95
126	Fettuccine Alfredo	Italian	14.50
127	Meat Lasagna	Italian	17.95
128	Cheese Lasagna	Italian	15.50
129	Mushroom Ravioli	Italian	15.50
130	Shrimp Scampi	Italian	19.95
131	Chicken Parmesan	Italian	17.95
132	Eggplant Parmesan	Italian	16.95

2. find the number of items on the menu

Query :

```
SELECT count(item_name) FROM  
menu_items;
```

Item_Count
32



3. How many Items are there in each category?

```
SELECT CATEGORY,  
COUNT(DISTINCT(item_name))  
AS ITEM_COUNT  
FROM MENU_ITEMS  
GROUP BY CATEGORY;
```

CATEGORY	ITEM_COUNT
American	6
Asian	8
Italian	9
Mexican	9



4. What are the most expensive items on the menu?

```
SELECT * FROM menu_items  
ORDER BY price DESC;
```

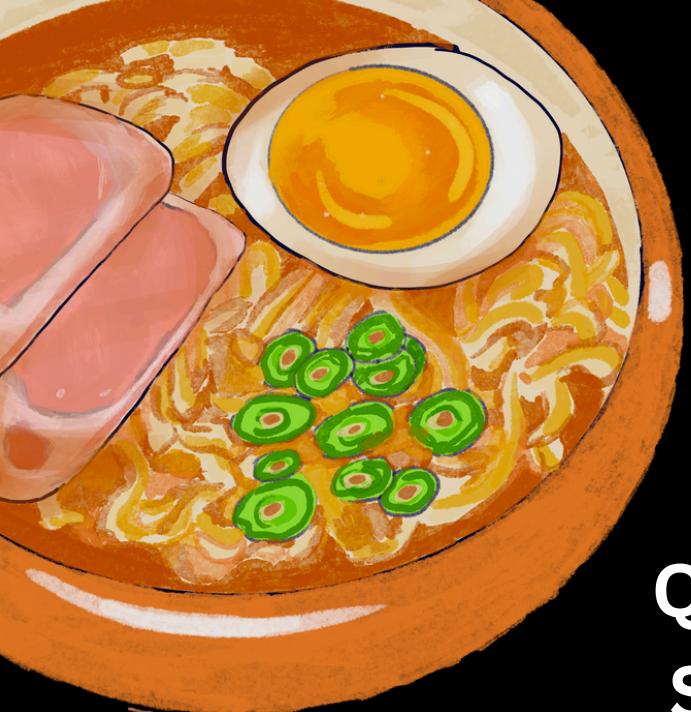
menu_item_id	item_name	category	price
130	Shrimp Scampi	Italian	19.95
109	Korean Beef Bowl	Asian	17.95
110	Pork Ramen	Asian	17.95
125	Spaghetti & Meatballs	Italian	17.95
127	Meat Lasagna	Italian	17.95
131	Chicken Parmesan	Italian	17.95
132	Eggplant Parmesan	Italian	16.95
107	Orange Chicken	Asian	16.50
128	Cheese Lasagna	Italian	15.50
129	Mushroom Ravioli	Italian	15.50
112	Salmon Roll	Asian	14.95
118	Steak Burrito	Mexican	14.95
108	Tofu Pad Thai	Asian	14.50
124	Spaghetti	Italian	14.50
126	Fettuccine Alfredo	Italian	14.50
102	Cheeseburger	American	13.95

5. What are the least expensive items on the menu?

```
SELECT * FROM menu_items  
ORDER BY price;
```

menu_item_id	item_name	category	price
113	Edamame	Asian	5.00
105	Mac & Cheese	American	7.00
106	French Fries	American	7.00
122	Chips & Salsa	Mexican	7.00
103	Hot Dog	American	9.00
114	Potstickers	Asian	9.00
123	Chips & Guacamole	Mexican	9.00
104	Veggie Burger	American	10.50
121	Cheese Quesadillas	Mexican	10.50
111	California Roll	Asian	11.95
115	Chicken Tacos	Mexican	11.95
119	Chicken Torta	Mexican	11.95
101	Hamburger	American	12.95
117	Chicken Burrito	Mexican	12.95
102	Cheeseburger	American	13.95
116	Steak Tacos	Mexican	13.95





Explore Order_details Table

1. Explore order_details table

Query :

```
SELECT * FROM order_details;
```

order_details_id	order_id	order_date	order_time	item_id
1	1	2023-01-01	11:38:36	109
2	2	2023-01-01	11:57:40	108
3	2	2023-01-01	11:57:40	124
4	2	2023-01-01	11:57:40	117
5	2	2023-01-01	11:57:40	129
6	2	2023-01-01	11:57:40	106
7	3	2023-01-01	12:12:28	117
8	3	2023-01-01	12:12:28	119
9	4	2023-01-01	12:16:31	117
10	5	2023-01-01	12:21:30	117
11	6	2023-01-01	12:29:36	101
12	6	2023-01-01	12:29:36	114
13	7	2023-01-01	12:50:37	123
14	8	2023-01-01	12:51:37	123
15	9	2023-01-01	12:52:01	108
16	9	2023-01-01	12:52:01	126
17	9	2023-01-01	12:52:01	110
18	9	2023-01-01	12:52:01	117
19	9	2023-01-01	12:52:01	117
20	9	2023-01-01	12:52:01	129
21	9	2023-01-01	12:52:01	122
22	9	2023-01-01	12:52:01	130

2. What is the date range of the table?

Query :

```
SELECT MIN(order_date), MAX(order_date) FROM  
order_details;
```

MIN(order_date)	MAX(order_date)
2023-01-01	2023-03-31

3. How many orders were made within this date range?

```
SELECT COUNT(DISTINCT order_id) FROM  
order_details;
```

COUNT(DISTINCT order_id)
5370



4. How Many items were ordered in the date range?

Query :
SELECT COUNT(*) FROM
order_details;

COUNT(*)
12234

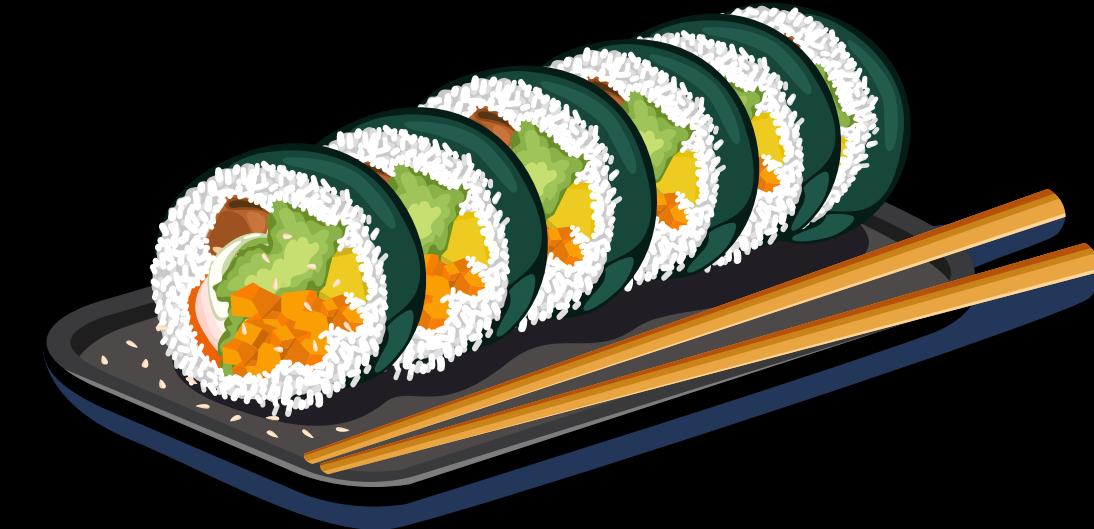


5. Which orders had the most number of items

Query :

```
SELECT order_id,  
count(item_id) AS num_items  
FROM order_details  
GROUP BY order_id  
ORDER BY num_items DESC;
```

order_id	num_items
4305	14
3473	14
1957	14
330	14
440	14
443	14
2675	14
5066	13



6. How many orders had more than 12 items?

```
SELECT COUNT(*) AS More_than_12_items  
FROM  
(SELECT order_id, count(item_id) AS  
num_items  
FROM order_details  
GROUP BY order_id  
HAVING num_items > 12) AS num_orders;
```

More_than_12_items
20

Analysing Customer behaviour

1. Join two tables

Query :

```
SELECT * FROM order_details od
LEFT JOIN menu_items mi
ON od.item_id = mi.menu_item_id;
```



order_details_id	order_id	order_date	order_time	item_id	menu_item_id	item_name	category	price
1	1	2023-01-01	11:38:36	109	109	Korean Beef Bowl	Asian	17.95
2	2	2023-01-01	11:57:40	108	108	Tofu Pad Thai	Asian	14.50
3	2	2023-01-01	11:57:40	124	124	Spaghetti	Italian	14.50
4	2	2023-01-01	11:57:40	117	117	Chicken Burrito	Mexican	12.95
5	2	2023-01-01	11:57:40	129	129	Mushroom Ravioli	Italian	15.50
6	2	2023-01-01	11:57:40	106	106	French Fries	American	7.00
7	3	2023-01-01	12:12:28	117	117	Chicken Burrito	Mexican	12.95
8	3	2023-01-01	12:12:28	119	119	Chicken Torta	Mexican	11.95
9	4	2023-01-01	12:16:31	117	117	Chicken Burrito	Mexican	12.95
10	5	2023-01-01	12:21:30	117	117	Chicken Burrito	Mexican	12.95
11	6	2023-01-01	12:29:36	101	101	Hamburger	American	12.95
12	6	2023-01-01	12:29:36	114	114	Potstickers	Asian	9.00
13	7	2023-01-01	12:50:37	123	123	Chips & Guacamole	Mexican	9.00
14	8	2023-01-01	12:51:37	123	123	Chips & Guacamole	Mexican	9.00
15	9	2023-01-01	12:52:01	108	108	Tofu Pad Thai	Asian	14.50
16	9	2023-01-01	12:52:01	126	126	Fettuccine Alfredo	Italian	14.50
17	9	2023-01-01	12:52:01	110	110	Pork Ramen	Asian	17.95

2. What were the most ordered items? What categories were they in?

```
SELECT category, item_name, COUNT(order_details_id)
AS num_purchases
FROM order_details od LEFT JOIN menu_items mi
ON od.item_id = mi.menu_item_id
GROUP BY item_name, category
ORDER BY num_purchases DESC;
```

category	item_name	num_purchases
American	Hamburger	622
Asian	Edamame	620
Asian	Korean Beef Bowl	588
American	Cheeseburger	583
American	French Fries	571
Asian	Tofu Pad Thai	562
Mexican	Steak Torta	489
Italian	Spaghetti & Meatballs	470
American	Mac & Cheese	463
Mexican	Chips & Salsa	461
Asian	Orange Chicken	456
Mexican	Chicken Burrito	455
Italian	Eggplant Parmesan	420
Mexican	Chicken Torta	379
Italian	Spaghetti	367
Italian	Chicken Parmesan	364
Asian	Pork Ramen	360
Italian	Mushroom Ravioli	359



3. What were the top 5 orders that spent the most money?

```
SELECT order_id, SUM(price) AS total_spend  
FROM order_details od LEFT JOIN  
menu_items mi  
ON od.item_id = mi.menu_item_id  
GROUP BY order_id  
ORDER BY total_spend DESC  
LIMIT 5;
```

order_id	total_spend
440	192.15
2075	191.05
1957	190.10
330	189.70
2675	185.10

4. Details of the highest spend order. Which specific items were purchased?

```
SELECT order_id, category, COUNT(item_id) AS num_items  
FROM order_details od LEFT JOIN menu_items mi  
ON od.item_id = mi.menu_item_id  
WHERE order_id = 440  
GROUP BY category;
```

order_id	category	num_items
440	Mexican	2
440	American	2
440	Italian	8
440	Asian	2



5. Details of the top 5 highest spend orders

```
SELECT order_id, category, COUNT(item_id) AS num_items  
FROM order_details od LEFT JOIN menu_items mi  
ON od.item_id = mi.menu_item_id  
WHERE order_id IN (440, 2075, 1957, 330, 2675)  
GROUP BY order_id, category;
```

order_id	category	num_items
330	Asian	6
330	American	1
330	Italian	3
330	Mexican	4
440	Mexican	2
440	American	2
440	Italian	8
440	Asian	2
1957	Asian	3
1957	American	3
1957	Italian	5
1957	Mexican	3
2075	Asian	3
2075	Mexican	3
2075	American	1
2075	Italian	6
2675	American	3
2675	Asian	3
2675	Italian	4
2675	Mexican	4



6. How much was the most expensive order in the dataset?

```
SELECT od.order_id, SUM(mi.price) AS Total_Order_Amount  
FROM order_details od JOIN menu_items mi  
ON od.item_id = mi.menu_item_id  
GROUP BY od.order_id  
ORDER BY Total_Order_Amount DESC  
LIMIT 1;
```

order_id	Total_Order_Amount
440	192.15





**This badge is earned by successfully completing the
Restaurant Order Analysis guided project at Maven Analytics.**