Restaurant Operations Analysis

The Situation:

You've just been hired as a Data Analyst for the Taste of the World Café, a restaurant that has diverse menu offerings and serves generous portions.

The Assignment:

The Taste of the World Café debuted a new menu at the start of the year. You've been asked to dig into the customer data to see which menu items are doing well / not well and what the top customers seem to like best.

The Data Set:

A quarter's worth of orders from a fictitious restaurant serving international cuisine, including the date and time of each order, the items ordered, and additional details on the type, name and price of the items.

The Objectives:

- Explore the menu_items table to get an idea of what's on the new menu.
- Explore the order_details table to get an idea of the data that's been collected.
- Use both tables to understand how customers are reacting to the new menu.

Objective 1 - Explore the items table:

1. View the menu_items table and write a query to find the number of items on the menu
2. What are the least and most expensive items on the menu?
3. How many Italian dishes are on the menu? What are the least and most expensive Italian dishes on the menu?
4. How many dishes are in each category? What is the average dish price within each category?

Objective 2 - Explore the orders table:

5. View the order_details table. What is the date range of the table?
6. How many orders were made within this date range? How many items were ordered within this date range?
7. Which orders had the most number of items?
8. How many orders had more than 12 items?

Objective 3 - Analyze customer behavior:

9. Combine the menu_items and order_details tables into a single table.
10. What were the least and most ordered items? What categories were they in?
11. What were the top 5 orders that spent the most money?
12. View the details of the highest spend order. Which specific items were purchased?
13. View the details of the top 5 highest spend orders.

Objective 1:

1. View the menu_items table and write a query to find the number of items on the menu.

   Query:

```
USE restaurant_db;
SELECT
   COUNT(*) AS number_of_items
FROM
   menu_items;
```

Output:

number_of_items
32

2. What are the least and most expensive items on the menu?

Query:

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

Output (Only partial output shown):

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

3. How many Italian dishes are on the menu? What are the least and most expensive Italian dishes on the menu?

Query:

```
SELECT
   category, COUNT(item_name) AS count_of_items
FROM
   menu_items
WHERE
   category = 'Italian'
GROUP BY category;
```

Output:

category	count_of_items
Italian	9

4. How many dishes are in each category? What is the average dish price within each category?

Query:

```
SELECT
   category,
   COUNT(item_name) AS num_of_dishes,
   AVG(price) AS average_price
FROM
   menu_items
GROUP BY category;
```

Output:

category	num_of_dishes	average_price
American	6	10.066667
Asian	8	13.475000
Mexican	9	11.800000
Italian	9	16.750000

Objective 2:

5. View the order_details table. What is the date range of the table?

Query:

```
SELECT
   *
FROM
   order_details;


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

Output:

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

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

6. How many orders were made within this date range? How many items were ordered within this date range?

Query:

```
SELECT
   COUNT(DISTINCT order_id) AS num_of_orders,
   COUNT(*) AS num_of_items_orderd
FROM
   order_details;
```

Output:

num_of_orders	num_of_items_orderd
5370	12234

7. Which orders had the most number of items?

Query:

```
SELECT
   order_id,
   COUNT(item_id) AS num_of_items
FROM
   order_details
GROUP BY order_id
ORDER BY num_of_items DESC;
```

Output (Only partial output shown):

order_id	num_of_items
330	14
440	14
443	14
1957	14
2675	14
3473	14
4305	14
1274	13
1569	13
1685	13

8. How many orders had more than 12 items?

Query:

```
SELECT
    order_id, COUNT(item_id) AS num_of_items
FROM
    order_details
GROUP BY order_id
ORDER BY num_of_items DESC;
```

Output:

num_orders
20

Objective 3:

9. Combine the menu_items and order_details tables into a single table.

Query:

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

Output (Only partial output shown):

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

10. What were the least and most ordered items? What categories were they in?

Query:

```
SELECT
    item_name,
    category,
    COUNT(order_details_id) AS num_of_times_ordered
FROM
    (SELECT
        *
    FROM
        order_details
    LEFT JOIN menu_items ON order_details.item_id = menu_items.menu_item_id) AS
combined_table
GROUP BY item_name , category
ORDER BY num_of_times_ordered DESC
LIMIT 1;


SELECT
    item_name,
    category,
    COUNT(order_details_id) AS num_of_times_ordered
FROM
    (SELECT
        *
```

```
    FROM
        order_details
    LEFT JOIN menu_items ON order_details.item_id = menu_items.menu_item_id) AS
combined_table
GROUP BY item_name , category
ORDER BY num_of_times_ordered
LIMIT 1;
```

Output:

item_name	category	num_of_times_ordered
Hamburger	American	622

item_name	category	num_of_times_ordered
Chicken Tacos	Mexican	123

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

Query:

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

Output:

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

12. View the details of the highest spend order. Which specific items were purchased?

Query:

```
SELECT
    *
FROM
    (SELECT
        *
    FROM
        order_details
    LEFT JOIN menu_items ON order_details.item_id = menu_items.menu_item_id) AS
combined_table
WHERE order_id = 440;
```

Output (Only partial output shown):

order_details_id	order_id	order_date	order_time	item_id	menu_item_id	item_name	category	price
1003	440	2023-01-08	12:16:34	116	116	Steak Tacos	Mexican	13.95
1004	440	2023-01-08	12:16:34	103	103	Hot Dog	American	9.00
1005	440	2023-01-08	12:16:34	124	124	Spaghetti	Italian	14.50
1006	440	2023-01-08	12:16:34	125	125	Spaghetti & Meatballs	Italian	17.95
1007	440	2023-01-08	12:16:34	125	125	Spaghetti & Meatballs	Italian	17.95
1008	440	2023-01-08	12:16:34	126	126	Fettuccine Alfredo	Italian	14.50
1009	440	2023-01-08	12:16:34	126	126	Fettuccine Alfredo	Italian	14.50
1010	440	2023-01-08	12:16:34	109	109	Korean Beef Bowl	Asian	17.95

13. View the details of the top 5 highest spend orders.

Query:

```
SELECT
    *
FROM
    (SELECT
        *
    FROM
        order_details
    LEFT JOIN menu_items ON order_details.item_id = menu_items.menu_item_id) AS
combined_table
WHERE order_id IN (440, 2075, 1957, 330, 2675);
```

Output (Only partial output shown):

order_details_id	order_id	order_date	order_time	item_id	menu_item_id	item_name	category	price
750	330	2023-01-06	13:27:11	107	107	Orange Chicken	Asian	16.50
751	330	2023-01-06	13:27:11	103	103	Hot Dog	American	9.00
752	330	2023-01-06	13:27:11	108	108	Tofu Pad Thai	Asian	14.50
753	330	2023-01-06	13:27:11	108	108	Tofu Pad Thai	Asian	14.50
754	330	2023-01-06	13:27:11	124	124	Spaghetti	Italian	14.50
755	330	2023-01-06	13:27:11	125	125	Spaghetti & Meatballs	Italian	17.95
756	330	2023-01-06	13:27:11	109	109	Korean Beef Bowl	Asian	17.95
757	330	2023-01-06	13:27:11	112	112	Salmon Roll	Asian	14.95