Situation

Work as a Data Analyst for the Taste of the World Cafe, a restaurant that has diverse menu offerings and serves generous portions .

Assignment

Dig into the customer data to see which meni items are doing well / not well and what the top customers seem to like best .

Objective

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

Table details

Table | Field | Description
menu_items | menu_item_id | Unique ID of a menu item
menu_items | item_name | Name of a menu item
menu_items | category | Category or type of cuisine of the menu item
menu_items | price | Price of the menu item (US Dollars $)
order_details | order_details_id | Unique ID of an item in an order
order_details | order_id | ID of an order
order_details | order_date | Date an order was put in (MM / DD / YY)
order_details | order_time | Time an order was put in (HH:MM :SS AM / PM)
order_details | item_id | Matches the menu_item_id in the menu_items table

Objective 1

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

```sql
SELECT *

FROM menu_items ;
```

```sql
SELECT COUNT( menu_item_id ) AS number_of_menu_items

FROM menu_items ;
```

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

```sql
SELECT

   item_name,
   price
FROM menu_items
WHERE price = ( SELECT MIN(price) FROM menu_items )
   OR price = ( SELECT MAX(price) FROM menu_items )
ORDER BY price DESC;
```

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

```sql
SELECT COUNT( menu_item_id ) AS number_of_italian_dishes

FROM menu_items
WHERE category = 'Italian' ;
```

```sql
SELECT

   item_name,
   price
FROM menu_items
WHERE category = 'Italian'
   AND price = (SELECT MIN(price) FROM menu_items WHERE category = 'Italian');
```

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

```sql
SELECT

    COUNT(DISTINCT menu_item_id) AS number_of_dishes,
    category,
    AVG(price) AS average_price
FROM menu_items
GROUP BY category ;
```

Objective 2

1. View the *order_details* table. What is the date range of the table?

```sql
SELECT *

FROM order_details ;
```

```sql
SELECT MIN(order_date) AS earliest_order,

       MAX(order_date) AS latest_order
FROM order_details ;
```

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

```sql
SELECT

    COUNT( DISTINCT order_id) AS total_orders,
    COUNT(item_id) AS total_items_ordered
FROM order_details
```

1. Which orders had the most number of items?

```sql
SELECT

    order_id,
    COUNT( item_id )
FROM order_details
GROUP BY order_id
ORDER BY COUNT( item_id ) DESC
```

```sql
-- alternative approach to get only the orders with the most items
```

```
SELECT
    order_id,
    COUNT( item_id )
FROM order_details
GROUP BY order_id
HAVING COUNT( item_id ) =

    ( SELECT COUNT( item_id )
      FROM order_details
      GROUP BY order_id
      ORDER BY COUNT( item_id ) DESC
      LIMIT 1
    )
ORDER BY COUNT( item_id ) DESC
```

1. How many orders had more than 12 items?

```
SELECT

    order_id,
    COUNT( item_id )
FROM order_details
GROUP BY order_id
HAVING COUNT( item_id ) > 12
ORDER BY COUNT( item_id ) DESC ;
```

```
-- using subquery to show only number of count of orders with more than 12 items

SELECT COUNT(order_id)
FROM
    (
        SELECT
            order_id,
            COUNT( item_id )
        FROM order_details
        GROUP BY order_id

        HAVING COUNT( item_id ) > 12
        ORDER BY COUNT( item_id ) DESC
    ) AS number_of_orders  ;
```

Objective 3

1. Combine the *menu_items* and *order details* table into a single table

```
SELECT *

FROM order_details
LEFT JOIN menu_items

ON order_details.item_id = menu_items.menu_item_id ;
```

```
-- using Views to avoid repeating the need to do JOIN
```

```sql
CREATE OR REPLACE VIEW joined AS (
    SELECT *
    FROM order_details
    LEFT JOIN menu_items
        ON order_details.item_id = menu_items.menu_item_id
) ;
```

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

```sql
-- use CTE to categorize the table, then subquery to get the min and max

WITH order_counts AS (
    SELECT
        item_name,
        category,
        COUNT(order_details_id) AS number_of_orders
    FROM joined
    GROUP BY item_name, category
)

( SELECT *
FROM order_counts
WHERE number_of_orders = ( SELECT MAX(number_of_orders) FROM order_counts )
)

UNION


( SELECT *
FROM order_counts
WHERE number_of_orders = ( SELECT MIN(number_of_orders) FROM order_counts )
)
```

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

```sql
SELECT

    order_id,
    SUM(price) AS total_spent
FROM joined
GROUP BY order_id
ORDER BY total_spent DESC

LIMIT 5 ;
```

1. View the details of the highest spent order. What insights can you gather from the results?

```sql
SELECT *

FROM joined
WHERE order_id = 440 ;
```

```sql
-- highest spent order = order_id 440

SELECT COUNT(order_details_id) as count_items_ordered
FROM joined
WHERE order_id = 440 ;
```

```sql
SELECT

    category,
    COUNT(order_details_id) as count_items_ordered,
    SUM(price) as total_spent
FROM joined
WHERE order_id = 440
GROUP BY category
ORDER BY
    count_items_ordered DESC,
    total_spent DESC;
```

Insights – Highest Spend Order (Order ID: 440)

1. The customer ordered a total of 14 items.
2. Italian cuisine was the most frequently ordered category by quantity. (8)
3. The highest spending within the order also went to Italian food. ($132.25)
4. View the details of the top 5 highest spend orders. What insights can you gather from the results?

```sql
-- top 5 highest spend orders are order_id 440, 2075, 1957, 330, 2675

SELECT
    order_id,
    COUNT(order_details_id) as count_items_ordered
FROM joined
WHERE order_id IN (440, 2075, 1957, 330, 2675)
GROUP BY order_id
ORDER BY count_items_ordered DESC;
```

```sql
-- to know the average items ordered in the top 5 highest spend orders

SELECT AVG(items_per_order) AS avg_items_per_order
FROM (
    SELECT
        order_id,
        COUNT(order_details_id) AS items_per_order
    FROM joined
    WHERE order_id IN (440, 2075, 1957, 330, 2675)
    GROUP BY order_id
) AS sub;
```

```sql
-- to know the breakdown of items ordered and total spent by category in the top 5
highest spend orders

SELECT
    category,
    COUNT(order_details_id) as count_items_ordered,
```

```
    SUM(price) as total_spent

FROM joined

WHERE order_id IN (440, 2075, 1957, 330, 2675)
GROUP BY category
ORDER BY
    count_items_ordered DESC,
    total_spent DESC;
```

```
-- to know the breakdown of items ordered and total spent by category in the each of
the top 5 highest spend orders

SELECT
    order_id,
    category,
    COUNT(order_details_id) as count_items_ordered,

    SUM(price) as total_spent
FROM joined
WHERE order_id IN (440, 2075, 1957, 330, 2675)
GROUP BY order_id, category

ORDER BY
    order_id,
    count_items_ordered DESC,
    total_spent DESC;
```

Insights – Top 5 Highest Spend Order (Order ID: 440, 2075, 1957, 330, 2675)

1. Across the five orders, customers purchased a total of 69 items, averaging 13.8 items per order.
2. Italian cuisine was the most frequently ordered category with 26 items, while American cuisine ranked lowest with only 10 items.
3. In terms of spending, Italian cuisine also generated the highest revenue at $430.65, whereas American cuisine recorded the lowest at $88.35.
4. For 4 out of the 5 orders, Italian cuisine dominated as the top choice. Conversely, American cuisine consistently appeared as either the joint-lowest or the least ordered category within those orders.