RESEARCH PROJECT

RESTAURANT ORDER ANALYSIS USING MYSQL

BY MANYA GUPTA

RESTAURANT ORDER DATASET

- THERE ARE TWO TABLES IN THE DATASET
- 1.MENU_ITEMS

  CONTAINS

FIELD	DESCRIPTION
MENU_ITEM_ID	UNIQUE ID OF A MENU ITEM
ITEM_NAME	NAME OF A MENU ITEM
CATEGORY	CATEGORY OR TYPE OF CUISINE OF THE MENU ITEM
PRICE	PRICE OF THE MENU ITEM IN US $

2.ORDER_DETAILS
 CONTAINS

FIELD	DESCRIPTION
ORDER_DETAILS_ID	UNIQUE ID OF AN ITEM IN AN ORDER
ORDER_ID	ID OF AN ORDER
ORDER_DATE	DATE OF AN ORDER
ORDER_TIME	TIME OF AN ORDER
ITEM_ID	MATCHES THE MENU_ITEM_ID IN THE MENU_ITEMS _TABLE

CREATE DATABASE IN MY SQL

```
create database restaurant_orders_db ;
```

▶ 🗄 **restaurant_orders_db**

USE THE DATABASE IN MY SQL

```
use restaurant_orders_db;
```

```sql
create table menu_items (
    menu_item_id int,
    item_name text,
    category text,
    price double,
    PRIMARY KEY (menu_item_id)
);
```

```sql
create table order_details (
    order_details_id int,
    order_id int,
    order_date date,
    order_time time,
    item_id int,
    PRIMARY KEY (order_details_id),
    FOREIGN KEY(item_id) REFERENCES menu_items(menu_items_id)
);
```
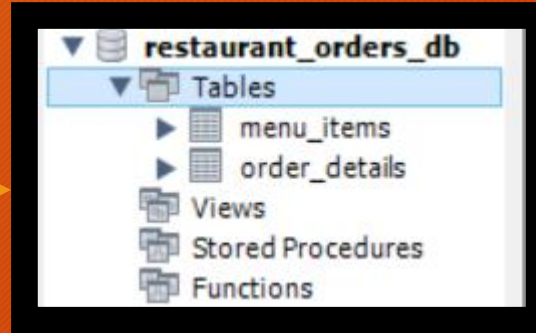
CREATE TABLES AND THEN IMPORT THE DATA BY USING TABLE DATA IMPORT WIZARD
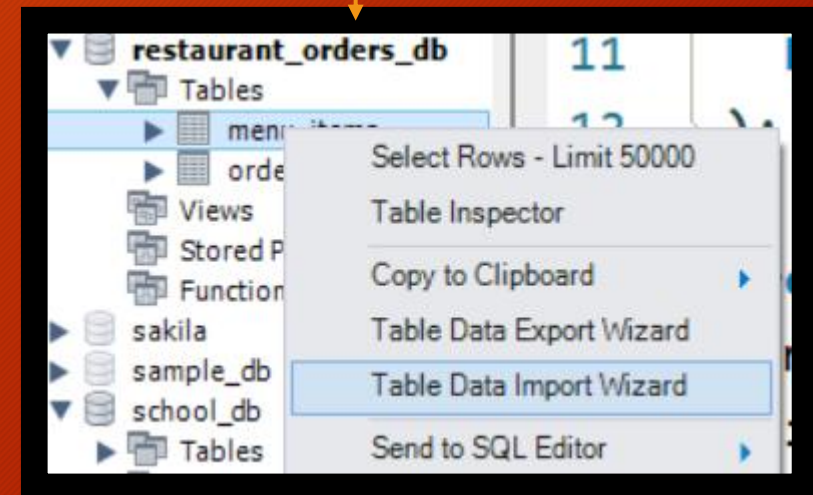
SNAPSHOT OF MENU_ITEMS_TABLE AND ORDER_DETAILS TABLE

```
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
104	Veggie Burger	American	10.5
105	Mac & Cheese	American	7
106	French Fries	American	American
107	Orange Chicken	Asian	16.5
108	Tofu Pad Thai	Asian	14.5
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

```
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

TOTAL NO. OF MENU_ITEMS

```
select count(*)as Total_items   from menu_items ;
```

Total_items
32

TOTAL NO.OF ORDERS

```
select count(*)as Total_orders from order_details ;
```

Total_orders
12234

OBJECTIVE

- TO ANALYZE CUSTOMER ORDRERS AND THEIR PREFERENCES BASED ON MENU_ITEMS,CATEGORY,AND PRICING TO OPTIMIZE MENU OFFERING AND IMPROVE SALES.

RESEARCH QUESTIONS

1.WHAT ARE THE MOST POPULAR MENU ITEMS ORDERER BY THE CUSTOMERS ?

2.WHAT ARE THE LEAST POPULAR MENU ITEMS ORDERED BY THE CUSTOMERS?

3.WHICH MENU CATEGORY GENERATE THE HIGHEST NUMBER OF ORDERS?

4.HOW DO PRICE OF MENU ITEMS INFLUENCE THE NO.OF ORDERS?

5.WHICH DAYS OF THE WEEK ARE THE BUSIEST IN TERMS OF ORDER?

6.WHAT IS THE ORDER DISTRIBUTION ACROSS DIFFERENT TIMES OF THE DAY(EXAMPLE-'MORNING','AFTERNOON')?

7.WHICH ARE THE MOST PROFITABLE ITEMS BASED ON THE TOTAL REVENUE GENERATED FROM ORDERS?

8.WHICH ARE THE LEAST PROFITABLE ITEMS BASED ON THE TOTAL REVENUE GENERATED FROM ORDERS?

9.WHICH ITEMS ARE ORDERED TOGETHER THE MOST ?

10.HOW DOES THE FREQUENCY OF ORDERS RELATE TO THE NUMBER OF ITEMS ORDERED PER TRANSACTION (SINGLE VS MULTIPLE ITEMS)?

1. What are the most popular menu items ordered by customers?

```sql
select m.item_name ,count(o.order_id) as orders_count
from order_details as o
left join menu_items as m
on o.item_id = m.menu_item_id
group by m.item_name
order by orders_count desc ;
```

item_name	orders_count
Hamburger	622
Edamame	620
Korean Beef Bowl	588
Cheeseburger	583
French Fries	571
Tofu Pad Thai	562
Chicken Parmesan	501
Steak Torta	489
Spaghetti & Meatballs	470
Mac & Cheese	463
Chips & Salsa	461
Orange Chicken	456
Chicken Burrito	455
Eggplant Parmesan	420

2. What are the least popular menu items ordered by customers?

```sql
select m.item_name ,count(o.order_id) as orders_count
from order_details as o
left join menu_items as m
on o.item_id = m.menu_item_id
group by m.item_name
order by orders_count asc ;
```

item_name	orders_count
Chicken Tacos	123
Potstickers	205
Cheese Lasagna	207
Steak Tacos	214
Cheese Quesadillas	233
Chips & Guacamole	237
Veggie Burger	238
Shrimp Scampi	239
Fettuccine Alfredo	249
Hot Dog	257
Meat Lasagna	273
Salmon Roll	324
Steak Burrito	354
California Roll	355

3. Which menu category generates the highest number of orders?

```sql
select m.category ,count(o.order_id) as orders_count
from order_details as o
left join menu_items as m
on o.item_id = m.menu_item_id
group by m.category
order by orders_count desc ;
```

category	orders_count
Asian	3470
Italian	3085
Mexican	2945
American	2734

4. How do prices of menu items influence the number of orders?

```sql
select m.item_name,round(avg(m.price),2)as avg_price,count(o.order_id)as orders_count
from order_details as o
left join menu_items as m
on o.item_id = m.menu_item_id
group by m.item_name
order by orders_count desc;
```

item_name	avg_price	orders_count
Hamburger	12.95	622
Edamame	5	620
Korean Beef Bowl	17.95	588
Cheeseburger	13.95	583
French Fries	7	571
Tofu Pad Thai	14.5	562
Chicken Parmesan	17.95	501 501
Steak Torta	13.95	489
Spaghetti & Meatballs	17.95	470
Mac & Cheese	7	463
Chips & Salsa	7	461
Orange Chicken	16.5	456
Chicken Burrito	12.95	455
Eggplant Parmesan	16.95	420
Chicken Torta	11.95	379
Spaghetti	14.5	367
Pork Ramen	17.95	360
Mushroom Ravioli	15.5	359
California Roll	11.95	355
Steak Burrito	14.95	354
Salmon Roll	14.95	324
Meat Lasagna	17.95	273
Hot Dog	9	257
Fettuccine Alfredo	14.5	249
Shrimp Scampi	19.95	239
Veggie Burger	10.5	238
Chips & Guacamole	9	237
Cheese Quesadillas	10.5	233
Steak Tacos	13.95	214
Cheese Lasagna	15.5	207
Potstickers	9	205
Chicken Tacos	11.95	123

5. Which days of the week are the busiest in terms of orders?

```sql
select dayname(order_date)as day_of_week, count(order_id)as orders_count
from order_details
group by day_of_week
order by orders_count  desc;
```

day_of_week	orders_count
Monday	2010
Friday	1822
Tuesday	1788
Sunday	1776
Thursday	1689
Saturday	1618
Wednesday	1531

6. What is the order distribution across different times of the day (e.g.'Morning','Afternoon')?

```sql
select case
        when extract(hour from o.order_time) between 6 and 11 then'Morning'
        WHEN Extract(hour from o.order_time) between 12 and 17 then 'Afternoon'
    else 'Evening'
    end as time_of_day, count(order_id)as order_count
from order_details o
join menu_items m on o.item_id = m.menu_item_id
group by  time_of_day
order by order_count desc;
```

time_of_day	order_count
Afternoon	7390
Evening	4209
Morning	635

7. which are the most profitable items based on the total revenue generated from orders?

```sql
with count_of_order as
( select m.menu_item_id, m.item_name,count(o.item_id)as order_count
from order_details as o
left join menu_items as m
on o.item_id = m. menu_item_id
group by m.menu_item_id , m.item_name)

select oc.item_name,oc.order_count, round((oc.order_count * m.price),2)as total_revenue
from count_of_order as oc
join menu_items as m
on oc.menu_item_id = m.menu_item_id
order by total_revenue desc;
```

item_name	order_count	total_revenue
Korean Beef Bowl	588	10554.6
Chicken Parmesan	501	8992.95
Spaghetti & Meatballs	470	8436.5
Tofu Pad Thai	562	8149
Cheeseburger	583	8132.85
Hamburger	622	8054.9

8. which are the least profitable items based on the total revenue generated from orders?

```sql
with count_of_order as
( select m.menu_item_id, m.item_name,count(o.item_id)as order_count
from order_details as o
left join menu_items as m
on o.item_id = m. menu_item_id
group by m.menu_item_id , m.item_name)

select oc.item_name,oc.order_count, round((oc.order_count * m.price),2)as total_revenue
from count_of_order as oc
join menu_items as m
on oc.menu_item_id = m.menu_item_id
order by total_revenue asc;
```

item_name	order_count	total_revenue
Chicken Tacos	123	1469,85
Potstickers	205	1845

9. Which Items are Ordered Together the Most?

```sql
select o1.item_id as Item1, o2.item_id as Item2 , count(*)as order_combination
from order_details as o1
join order_details as o2
on o1.order_id = o2.order_id
where o1.item_id > o2.item_id
group by o1.item_id , o2.item_id
order by order_combination desc ;
```

Item1	Item2	order_combination
113	101	90
113	102	88

10. How does the frequency of orders relate to the number of items ordered per transaction (single vs multiple items)?

```sql
select order_id, COUNT(item_id) AS item_count
from order_details
group by order_id
having item_count > 1
order by  item_count desc;
```

order_id	item_count
330	14
440	14
443	14
1957	14
2675	14
3473	14
4305	14
4482	14

11. What is the average number of items ordered in a single transaction?

```sql
select distinct
    avg(count(item_id)) over () as avg_items_per_order
from order_details
group by  order_id;
```

avg_items_per_order
2.2782

12. what is the total revenue generated throughout the year?

```sql
with ordercount as
(select year(order_date) as order_year ,count(order_id) as orders_count ,item_id
from order_details
group by order_year, item_id)

select oc.order_year ,round(sum(oc.orders_count*m.price),2)as revenue_for_year
from ordercount as oc
join menu_items as m
on oc.item_id = m. menu_item_id
group by oc.order_year;
```

order_year	revenue_for_year
2023	161677.05

FINDINGS

- The most popular menu items ordered by customers are **Hamburger(622)**, **Edamame(620)**, **Korean Beef Bowl(588)** ,**Cheeseburger(583)**, and **French Fries(571)**, with **Hamburger** being the top choice, followed closely by Edamame, indicating a strong preference for both classic comfort foods and lighter, healthier options.

- **Chicken Tacos** have been **ordered 123 times**, suggesting that they are one of the least popular menu items.

- The menu category that generates the highest number of orders is **Asian**, with a total of **3470** **orders.**

- From the data, we can observe that items with **lower prices** **(e.g., Edamame at $5 or French Fries at $7) tend to have higher order counts (620 and 571 orders**, respectively). In contrast, more **expensive items like Shrimp Scampi at $19.95 or Meat Lasagna at $17.95 generally have lower order counts (239 and 273 orders**, respectively). **Some high-priced items** still have a **decent number of orders. For instance, Korean Beef Bowl at $17.95 has 588 orders, showing that some more expensive items can still perform well.** This suggests that factors other than just price may also play a role, such as item appeal, popularity, or perceived value.**Generally, there is a negative relationship between the price of a menu item and its order count, with cheaper items seeing more orders and more expensive items seeing fewer. However, exceptions exist, and factors like popularity and customer preferences could also influence the number of orders.**

FINDINGS

- **The busiest days of the week in terms of orders are: Monday (2010 orders),Friday (1822 orders),Tuesday (1788 orders), Sunday (1776 orders).**

- The order distribution across different times of the day is as follows: Afternoon has 7,390 orders, Evening has 4,209 orders, and Morning has 635 orders.

- The most profitable item based on total revenue is the Korean Beef Bowl, which has generated $10,554.60 from 588 orders.

- The least profitable item based on total revenue is the Chicken Tacos , which has generated $1469.85 from 123 orders.

- Edamame and hamburger are the items that are ordered together the most.

- The no. of items ordered per transaction is 1-14 and the total revenue generated through out the year is **$161677.05**

RECOMMENDATIONS

- Feature top-sellers (Hamburger, Edamame, French Fries) more prominently.

- Create combo deals (e.g., Burger + Fries, Edamame + Burger).

- Low-Performing Items like chicken tacos need new versions or toppings.

- Offer combos like "Burger and Edamame Combo" to increase average order value.

- Target high-traffic days (Monday, Friday, Tuesday, Sunday) with promotions.

- Offer breakfast deals to boost morning orders and promote healthier options in the evening.

- Highlight high-performing items at the top of the menu.

- Promote less popular items with special offers or chef's specials

- If customers consistently place larger orders with a similar number of items, indicating a prefence of buying in larger quantities then give discounts on that items.

- Increase order size of customer purchasing 1 or 2 items with promotions like "Buy One, Get One Half Off." and many more special deals that encourage customer to buy multiple items in a single transaction.

- Offer personalized menu suggestions based on past orders.

- Monitor item profitability regularly and adjust pricing and update the menu accordingly.