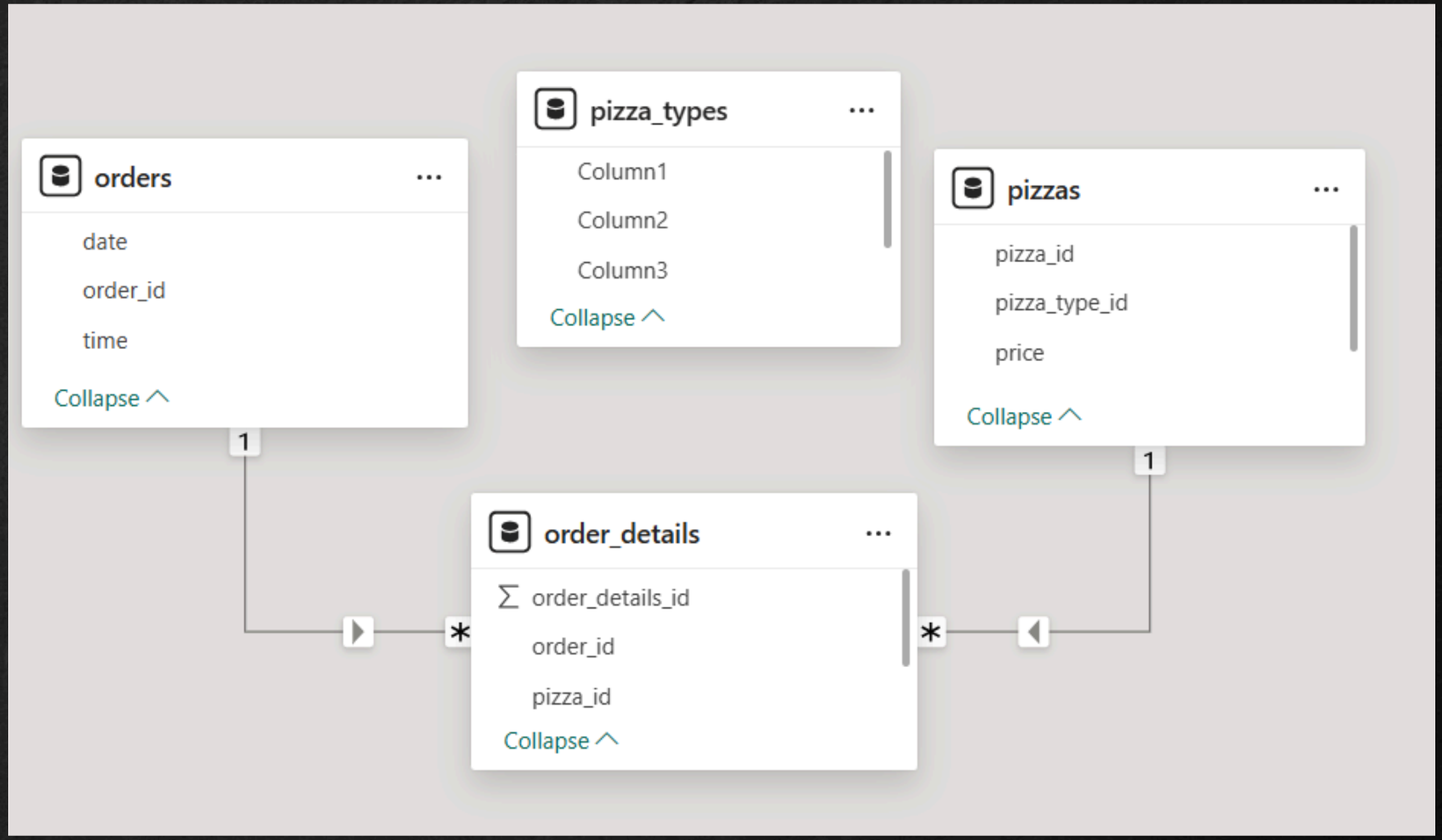
SQL PROJECT ON PIZZA SALES

HELLO !!

My name is Loveleen Kaur. In this project, I have utilised SQL queries to solve questions that were related to pizza sales.

DATA MODEL OVERVIEW



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```
SELECT COUNT(A.order_id) AS TOTAL_ORDERS
FROM ORDERS AS A
```

Results | Messages

	TOTAL_ORDERS
1	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

```sql
SELECT ROUND(SUM(A.quantity * B.price),2) AS TOTAL_SALES FROM order_details AS A
INNER JOIN pizzas AS B
ON A.pizza_id = B.pizza_id
```

	TOTAL_SALES
1	817860.05

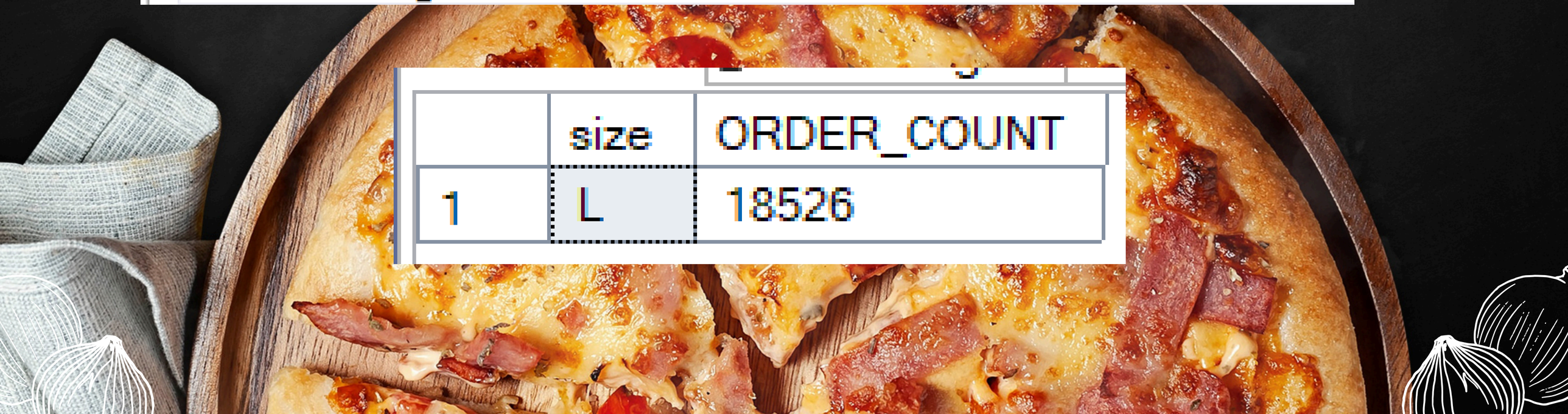
IDENTIFY THE HIGHEST PRICED PIZZA

```sql
SELECT TOP 1 A.name,ROUND(B.price,2) FROM pizza_types AS A
INNER JOIN pizzas AS B
ON A.pizza_type_id = B.pizza_type_id
ORDER BY B.price DESC
```

	name	(No column name)
1	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```sql
SELECT TOP 1 A.size,COUNT(B.quantity) AS ORDER_COUNT FROM pizzas AS A
INNER JOIN order_details AS B
ON A.pizza_id = B.pizza_id
GROUP BY A.size
ORDER BY ORDER_COUNT DESC
```

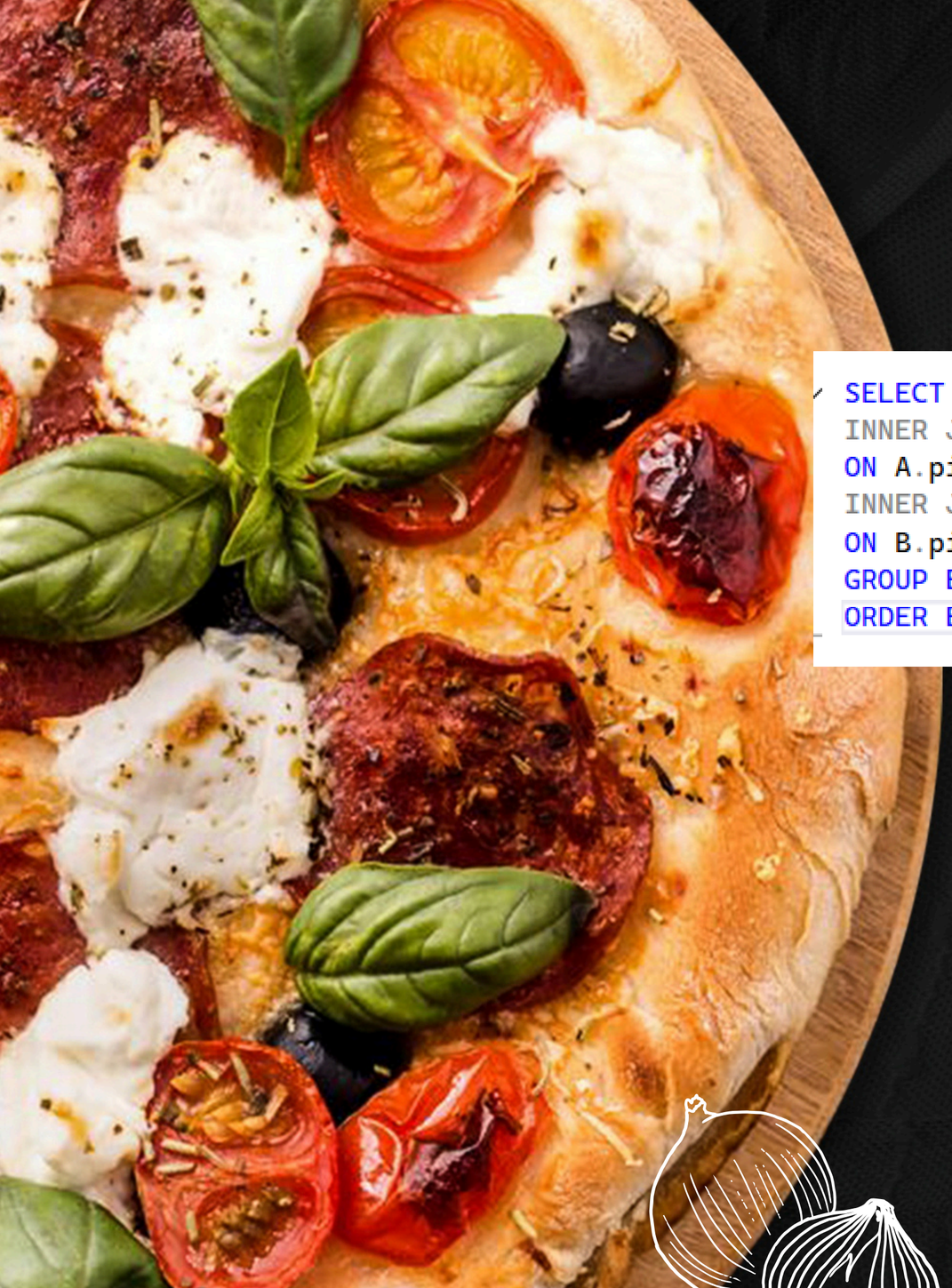


	size	ORDER_COUNT
1	L	18526

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES

```
SELECT TOP 5 A.name,SUM(CAST(C.quantity AS INT)) AS QUANTITY FROM pizza_types AS A
INNER JOIN pizzas AS B
ON A.pizza_type_id = B.pizza_type_id
INNER JOIN order_details AS C
ON B.pizza_id = C.pizza_id
GROUP BY A.name
ORDER BY QUANTITY DESC
```

	name	QUANTITY
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371



JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY

```sql
SELECT A.category,SUM(CAST(C.quantity AS INT)) AS QUANTITY FROM pizza_types AS A
INNER JOIN pizzas AS B
ON A.pizza_type_id = B.pizza_type_id
INNER JOIN order_details AS C
ON B.pizza_id = C.pizza_id
GROUP BY A.category
ORDER BY QUANTITY DESC
```

	category	QUANTITY
1	Classic	14888
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
SELECT DATEPART(HOUR,A.time) AS HOUR,COUNT(A.order_id) AS ORDER_COUNT FROM orders AS A
GROUP BY DATEPART(HOUR,A.time)
```

	HOUR	ORDER_COUNT
1	23	28
2	15	1468
3	9	1
4	12	2520
5	21	1198
6	18	2399
7	10	8
8	19	2009
9	13	2455
10	22	663
11	16	1920
12	17	2336
13	11	1231
14	20	1642
15	14	1472

JOIN RELEVANT TABLES TO FIND THE CATEGORY WISE DISTRIBUTION OF PIZZAS

```
SELECT CATEGORY,COUNT(A.name) AS TOTAL_COUNT FROM pizza_types AS A
GROUP BY CATEGORY
```

	CATEGORY	TOTAL_COUNT
1	Chicken	6
2	Classic	8
3	Supreme	9
4	Veggie	9

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY

```
SELECT AVG(QUANTITY) AS AVERAGE FROM
(SELECT A.date,SUM(CAST(B.quantity AS INT)) AS QUANTITY FROM orders AS A
INNER JOIN order_details AS B
ON A.order_id = B.order_id
GROUP BY A.date) AS ORDER_QUANTITY
```

	AVERAGE
1	138

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```sql
SELECT TOP 3 A.name,SUM(C.quantity * B.price) AS REVENUE FROM pizza_types AS A
INNER JOIN pizzas AS B
ON A.pizza_type_id = B.pizza_type_id
INNER JOIN order_details AS C
ON B.pizza_id = C.pizza_id
GROUP BY  A.name
ORDER BY REVENUE DESC
```

	name	REVENUE
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

```
SELECT A.category AS CATEGORY,
ROUND(SUM(C.quantity * B.price) / (SELECT ROUND(SUM(D.quantity*E.price),2) AS TOTAL_SALES FROM order_details AS D
                                   INNER JOIN pizzas AS E
                                   ON D.pizza_id = E.pizza_id) * 100,2) AS REVENUE
FROM pizza_types AS A
INNER JOIN pizzas AS B
ON A.pizza_type_id = B.pizza_type_id
INNER JOIN order_details AS C
ON B.pizza_id = C.pizza_id
GROUP BY CATEGORY
ORDER BY REVENUE DESC
```

	CATEGORY	REVENUE
1	Classic	26.91
2	Supreme	25.46
3	Chicken	23.96
4	Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```sql
SELECT D.date,
SUM(D.REVENUE)  OVER(ORDER BY D.date) AS CUM_REVENUE
FROM
(SELECT C.date,SUM(A.quantity * B.price) AS REVENUE
FROM order_details AS A
INNER JOIN pizzas AS B
ON A.pizza_id = B.pizza_id
INNER JOIN orders AS C
ON C.order_id = A.order_id
GROUP BY C.date) AS D
```

	date	CUM_REVENUE
1	2015-01-01	2713.85000228882
2	2015-01-02	5445.7500038147
3	2015-01-03	8108.15000724792
4	2015-01-04	9863.60000801086
5	2015-01-05	11929.5500087738
6	2015-01-06	14358.5000114441
7	2015-01-07	16560.700012207
8	2015-01-08	19399.0500183105
9	2015-01-09	21526.4000225067
10	2015-01-10	23990.350025177
11	2015-01-11	25862.6500263214
12	2015-01-12	27781.7000274658
13	2015-01-13	29831.3000278473
14	2015-01-14	32358.7000293732
15	2015-01-15	34343.5000324249
16	2015-01-16	36937.6500339508
17	2015-01-17	39001.7500343323

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```sql
SELECT E.name,E.REVENUE FROM
(SELECT D.category,D.name,D.REVENUE,
RANK()  OVER(PARTITION BY D.category ORDER BY D.REVENUE DESC) AS RN
FROM
(SELECT A.category,A.name,
SUM((C.quantity) * B.price) AS REVENUE
FROM pizza_types AS A
INNER JOIN pizzas AS B
ON A.pizza_type_id = B.pizza_type_id
INNER JOIN order_details AS C
ON C.pizza_id = B.pizza_id
GROUP BY A.category,A.name) AS D) AS E
WHERE RN<=3
```

	name	REVENUE
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5
4	The Classic Deluxe Pizza	38180.5
5	The Hawaiian Pizza	32273.25
6	The Pepperoni Pizza	30161.75
7	The Spicy Italian Pizza	34831.25
8	The Italian Supreme Pizza	33476.75
9	The Sicilian Pizza	30940.5
10	The Four Cheese Pizza	32265.7010040283
11	The Mexicana Pizza	26780.75
12	The Five Cheese Pizza	26066.5

THANK YOU!