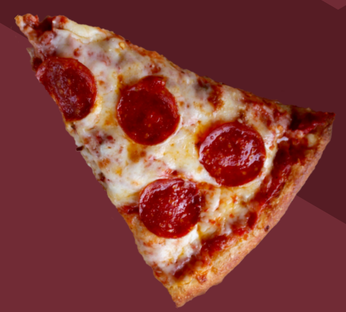
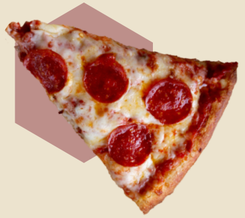



PIZZA SALES REPORT

SQL PROJECT



RETRIVE TOTAL NUMBER OF ORDER PLACED



```
SELECT
 COUNT(*) AS TOTAL_ORDERS_COUNT
FROM
 PUBLIC."orders";
```

	total_orders_count bigint
1	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT
 SUM(OD.QUANTITY * PZS.PRICE) AS
TOTAL_SALES
FROM
 PUBLIC."order_details" AS OD
 INNER JOIN PUBLIC."pizzas" AS PZS ON
OD.PIZZA_ID = PZS.PI
```

total_sales numeric
817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA.

```
SELECT PT.NAME, PZS.PRICE
FROM PUBLIC."pizzas" PZS
JOIN PUBLIC."pizza_types" PT ON
PZS.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID
ORDER BY PZS.PRICE DESC
LIMIT 1;
```

name character varying (200)	price numeric
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```
SELECT
 COUNT(OD.PIZZA_ID),
 PZS.SIZE
FROM
 PUBLIC."order_details" AS OD
 INNER JOIN PIZZAS AS PZS ON
OD.PIZZA_ID = PZS.PIZZA_ID
GROUP BY
 PZS.SIZE
ORDER BY
 COUNT(OD.PIZZA_ID) DESC
LIMIT
 1
```

count bigint	size character varying (200)
18526	L

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

```
SELECT
 PT.NAME,
 SUM(OD.QUANTITY)
FROM
 PUBLIC."order_details" AS OD
 INNER JOIN PUBLIC."pizzas" AS PZS ON OD.PIZZA_ID
= PZS.PIZZA_ID
 INNER JOIN PIZZA_TYPES AS PT ON
PZS.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID
GROUP BY
 PT.NAME
ORDER BY SUM(OD.QUANTITY) DESC LIMIT
 5;
```

name character varying (200)	count bigint
The Classic Deluxe Pizza	2416
The Barbecue Chicken Pizza	2372
The Hawaiian Pizza	2370
The Pepperoni Pizza	2369
The Thai Chicken Pizza	2315

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

```
SELECT
 PT.CATEGORY,
 SUM(OD.QUANTITY)
 FROM
 PUBLIC."order_details" AS OD
 INNER JOIN PUBLIC."pizzas" AS PZS ON
OD.PIZZA_ID = PZS.PIZZA_ID
 INNER JOIN PUBLIC."pizza_types" AS PT ON
PZS.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID
GROUP BY
 PT.CATEGORY
ORDER BY
 SUM(OD.QUANTITY) DESC;
```

category character varying (200)	sum numeric
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
SELECT
  EXTRACT(HOUR FROM time) AS order_hour,
  COUNT(order_id) AS total_orders
FROM
  public."orders"
GROUP BY
  order_hour
ORDER BY
  total_orders DESC;
```

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

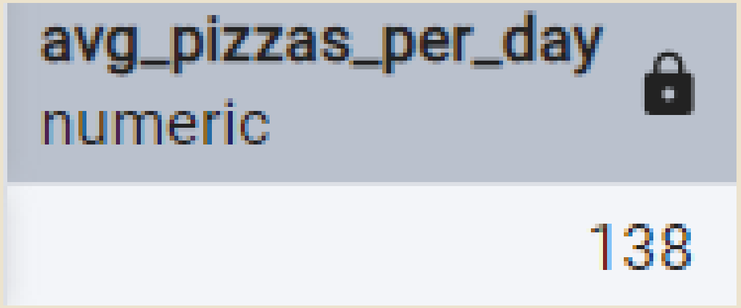
JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```
SELECT
 CATEGORY,
 COUNT(NAME) AS COUNT_OF_PIZZA
FROM
 PUBLIC."pizza_types"
GROUP BY
 CATEGORY
ORDER BY
 COUNT(NAME) DESC;
```

category character varying (200)	count_of_pizza bigint
Veggie	9
Supreme	9
Classic	8
Chicken	6

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

```
SELECT
 ROUND(AVG(Daily_Total), 0) AS Avg_Pizzas_Per_Day
FROM (
 SELECT
  O.DATE,
  SUM(OD.QUANTITY) AS Daily_Total
 FROM PUBLIC."orders" O
 JOIN PUBLIC."order_details" OD ON O.ORDER_ID =
OD.ORDER_ID
 GROUP BY O.DATE
) AS DailyOrders;
```



DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

```
SELECT
 DATA_TABLE.NAME,
 SUM(DATA_TABLE.QUANTITY * DATA_TABLE.PRICE) AS
REVENUE
FROM
 (SELECT * FROM
   PUBLIC."order_details" AS OD
   INNER JOIN PUBLIC."pizzas" AS PZS ON OD.PIZZA_ID =
PZS.PIZZA_ID
   INNER JOIN PIZZA_TYPES AS PT ON PZS.PIZZA_TYPE_ID
= PT.PIZZA_TYPE_ID
 ) AS DATA_TABLE
GROUP BY
 DATA_TABLE.NAME
ORDER BY
 SUM(DATA_TABLE.QUANTITY * DATA_TABLE.PRICE) DESC
LIMIT 3;
```

name character varying (200)	revenue numeric
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768.00
The California Chicken Pizza	41409.50

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
WITH REVENUE_DETAILS AS (
    SELECT
        os.date,
        pt.category,
        SUM(od.quantity * pzs.price) AS daily_revenue
    FROM
        public."order_details" AS od
    INNER JOIN
        public."pizzas" AS pzs ON od.pizza_id = pzs.pizza_id
    INNER JOIN
        public."pizza_types" AS pt ON pzs.pizza_type_id = pt.pizza_type_id
    INNER JOIN
        public."orders" AS os ON od.order_id = os.order_id
    GROUP BY
        os.date, pt.category
)
SELECT
    date,
    category,
    daily_revenue,
    SUM(daily_revenue) OVER (
        PARTITION BY category
        ORDER BY date
    ) AS cumulative_revenue
FROM
    REVENUE_DETAILS
ORDER BY
    category, date;
```

category character varying (200)	category_revenue numeric
Classic	26.9
Supreme	25.5
Chicken	24.0
Veggie	23.7

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

```
WITH TEMP AS (
  SELECT
   PT.NAME,
   PT.CATEGORY,
   SUM(OD.QUANTITY * PZ.PRICE) AS REVENUE
  FROM
   PUBLIC."order_details" AS OD
   INNER JOIN PUBLIC."pizzas" AS PZ ON OD.PIZZA_ID = PZ.PIZZA_ID
   INNER JOIN PUBLIC."pizza_types" AS PT ON PZ.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID
  GROUP BY
   PT.NAME,
   PT.CATEGORY
 )
SELECT * FROM
 (
  SELECT
   TEMP.CATEGORY,
   TEMP.NAME,
   TEMP.REVENUE,
   DENSE_RANK() OVER (
    PARTITION BY
     TEMP.CATEGORY
    ORDER BY
     TEMP.REVENUE DESC
   ) AS RANK FROM TEMP
 ) WHERE RANK <= 3
```

date date	category character varying (200)	daily_revenue numeric	cumulative_revenue numeric
2015-01-01	Chicken	667.00	667.00
2015-01-02	Chicken	552.00	1219.00
2015-01-03	Chicken	763.50	1982.50
2015-01-04	Chicken	505.00	2487.50
2015-01-05	Chicken	551.25	3038.75
2015-01-06	Chicken	567.25	3606.00
2015-01-07	Chicken	477.00	4083.00
2015-01-08	Chicken	563.25	4646.25

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
WITH REVENUE AS (
  SELECT
   SUM(OD.QUANTITY * PZS.PRICE) AS TOTAL_REVENUE
  FROM
   PUBLIC."order_details" AS OD
   INNER JOIN PUBLIC."pizzas" AS PZS ON OD.PIZZA_ID = PZS.PIZZA_ID
   INNER JOIN PIZZA_TYPES AS PT ON PZS.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID
 )
SELECT PT.CATEGORY,
 ROUND((
   SUM(OD.QUANTITY * PZS.PRICE) / (
    SELECT TOTAL_REVENUE FROM REVENUE)
  ) * 100, 1) AS CATEGORY_REVENUE
FROM
 PUBLIC."order_details" AS OD
 INNER JOIN PUBLIC."pizzas" AS PZS ON OD.PIZZA_ID = PZS.PIZZA_ID
 INNER JOIN PIZZA_TYPES AS PT ON PZS.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID
GROUP BY
 PT.CATEGORY
ORDER BY
 ROUND((
   SUM(OD.QUANTITY * PZS.PRICE) / (SELECT TOTAL_REVENUE
    FROM REVENUE)) * 100,1) DESC;
```

category character varying (200)	name character varying (200)	revenue numeric	rank bigint
Chicken	The Thai Chicken Pizza	43434.25	1
Chicken	The Barbecue Chicken Pizza	42768.00	2
Chicken	The California Chicken Pizza	41409.50	3
Classic	The Classic Deluxe Pizza	38180.5	1
Classic	The Hawaiian Pizza	32273.25	2
Classic	The Pepperoni Pizza	30161.75	3
Supreme	The Spicy Italian Pizza	34831.25	1
Supreme	The Italian Supreme Pizza	33476.75	2
Supreme	The Sicilian Pizza	30940.50	3
Veggie	The Four Cheese Pizza	32265.70	1
Veggie	The Mexicana Pizza	26780.75	2
Veggie	The Five Cheese Pizza	26066.5	3

Pizza Sales Analytics: Insights & Action Plan

Key Findings:

Revenue & Volume

- Total revenue: $817,860 from 21,350 orders
- Avg. 138 pizzas sold daily

Product Performance

- Top pizza: Thai Chicken ($43.4K revenue)
- Most ordered: Classic Deluxe (2,416 orders)
- Highest-priced: Greek Pizza ($35.95)

Customer Behavior

- Peak hours: 12 PM (2,520 orders), 6 PM (2,399 orders)
- Size preference: Large (L) pizzas = 87% of orders

Category Trends

- Classic pizzas drive 27% of revenue (highest share)
- Veggie/Supreme categories have most varieties (9 types each) but lower sales

Area	Action
Peak Hour Optimization	Bundle "Lunch Rush Combos" + staff surge scheduling
Menu Engineering	Highlight Thai Chicken in ads/menu
Upselling	Train staff to recommend Large (L) size upgrades
Off-Peak Boost	Launch "Early Bird" breakfast pizzas (9–11 AM)
Inventory Focus	Stock 2X dough/toppings before 12 PM & 5 PM rushes


THANK YOU