PIZZAHUT

SALES

SQL PROJECT

Schema

```
create table orders(
order_id int not null,
order_date datetime not null,
order_time time not null,
primary key(order_id)
);

create table order_details(
order_details_id int not null,
order_id int not null,
pizza_id text not null,
quantity int not null,
primary key(order_details_id)
);
```

Schema

```
create table pizzas(
pizza_id text not null,
price int not null,
pizza_type_id text not null,
size text not null
);

create table pizza_types(
ingredients text not null,
name text not null,
category text not null,
pizza_type_id text not null
);
```

Retrieve the total number of orders placed.

```
select count(order_id) as total_orders
from orders;
```

Result Grid

total_orders
21350

Calculate the total revenue generated from pizza sales.

```
SELECT
   ROUND(SUM(order_details.quantity * pizzas.price),
         2) AS Total_revenue
FROM
   order_details
      JOIN
   pizzas ON order_details.pizza_id = pizzas.pizza_id;
```

Result Grid

	Total_revenue
▶	817860.05

Identify the highest-priced pizza.

```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id =
pizza_types.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

Identify the most common pizza size ordered.

```
SELECT
   pizzas.size, SUM(order_details.quantity) AS Quantity
FROM
   pizzas
      JOIN
   order_details ON pizzas.pizza_id =
order_details.pizza_id
GROUP BY pizzas.size
ORDER BY Quantity DESC
LIMIT 1;
```

Result Grid

	size	Quantity
▶	L	18956

List the top 5 most ordered pizza types along with their quantities.

```
SELECT
    pizza_types.name, SUM(order_details.quantity) AS
quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
        JOIN
    order_details ON pizzas.pizza_id =
order_details.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

Result Grid | Filter Rows:

name	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Join the necessary tables to find the total quantity of each pizza category ordered.

```
SELECT
    pizza_types.category AS Category,
    SUM(order_details.quantity) AS Quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id =
pizzas.pizza_id
GROUP BY Category
ORDER BY Quantity DESC;
```

Result Grid | Filter Rows:

Category	Quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day.

```
SELECT
   HOUR(order_time) AS Hour_time,
   COUNT(order_id) AS Number_of_orders
FROM
   orders
GROUP BY Hour_time;
```

Hour_time	Number_of_orders
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009

Join relevant tables to find the category-wise distribution of pizzas.

```
SELECT
    category, COUNT(name) AS count_pizza
FROM
    pizza_types
GROUP BY category;
```

category	count_pizza
Chicken	6
Classic	8
Supreme	9
Veggie	9

Group the orders by date and calculate the average number of pizzas ordered per day.

```
SELECT
  ROUND(AVG(quantity), 2)
FROM
  (SELECT
    orders.order_date AS Day,
      SUM(order_details.quantity) AS quantity
  FROM
    orders
  JOIN order_details ON orders.order_id =
order_details.order_id
  GROUP BY Day) AS order_quantity;
```

	round(avg(quantity), 2)
▶	138.47

Determine the top 3 most ordered pizza types based on revenue.

```
SELECT
    pizza_types.name AS Pizza_type,
    SUM(order_details.quantity * pizzas.price) AS Revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id =
pizzas.pizza_id
GROUP BY Pizza_type
ORDER BY Revenue DESC limit 3;
```

	Pizza_type	Revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5

Calculate the percentage contribution of each pizza type to total revenue.

```
SELECT
    pizza_types.category AS Pizza_type,
    SUM(order_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(order_details.quantity * pizzas.price),
                2) AS Total_revenue
      FROM
        order_details
          JOIN
        pizzas ON order_details.pizza_id = pizzas.pizza_id) *
100 AS Revenue
FROM
    pizza_types
      JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
      JOIN
    order_details ON order_details.pizza_id =
pizzas.pizza_id
GROUP BY Pizza_type
ORDER BY Revenue DESC;
```

Pizza_type	Revenue
Classic	26.90596025566967
Supreme	25.45631126009862
Chicken	23.955137556847287
Veggie	23.682590927384577

Calculate the percentage contribution of each pizza type to total revenue.

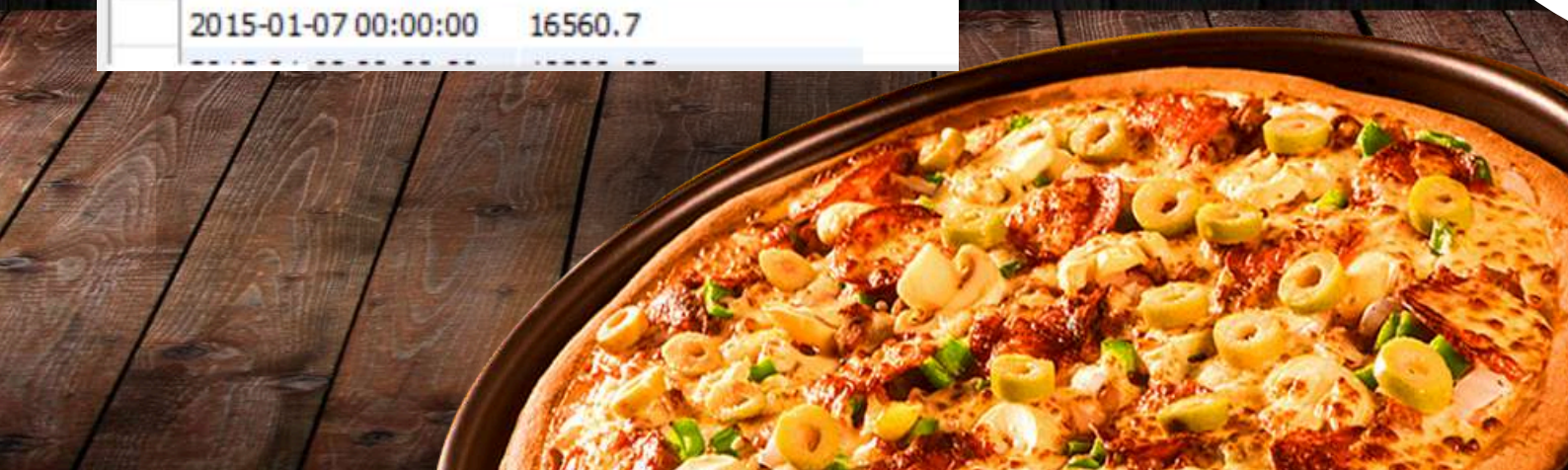
```
SELECT
    pizza_types.category AS Pizza_type,
    SUM(order_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(order_details.quantity * pizzas.price),
                2) AS Total_revenue
      FROM
        order_details
          JOIN
        pizzas ON order_details.pizza_id = pizzas.pizza_id) *
100 AS Revenue
FROM
    pizza_types
      JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
      JOIN
    order_details ON order_details.pizza_id =
pizzas.pizza_id
GROUP BY Pizza_type
ORDER BY Revenue DESC;
```

Pizza_type	Revenue
Classic	26.90596025566967
Supreme	25.45631126009862
Chicken	23.955137556847287
Veggie	23.682590927384577

Analyze the cumulative revenue generated over time.

```
select order_date, sum(Revenue)
over(order by order_date) as
cum_revenue from
(select orders.order_date,
sum(order_details.quantity *
pizzas.price) as Revenue
from order_details join orders on
order_details.order_id = orders.order_id
join pizzas on pizzas.pizza_id =
order_details.pizza_id
group by orders.order_date) as sales;
```

order_date	cum_revenue
2015-01-01 00:00:00	2713.8500000000004
2015-01-02 00:00:00	5445.75
2015-01-03 00:00:00	8108.15
2015-01-04 00:00:00	9863.6
2015-01-05 00:00:00	11929.55
2015-01-06 00:00:00	14358.5
2015-01-07 00:00:00	16560.7



Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```
select category, name, Revenue
from
(select name, category, Revenue, rank() over(partition by
category order by Revenue desc) as rn
from (select pizza_types.category,pizza_types.name,
sum(order_details.quantity * pizzas.price) as Revenue
from pizza_types join pizzas on pizzas.pizza_type_id =
pizza_types.pizza_type_id
   join order_details on order_details.pizza_id =
pizzas.pizza_id
   group by pizza_types.category, pizza_types.name)
as a) b where rn <= 3;
```

category	name	Revenue
Chicken	The Thai Chicken Pizza	43434.25
Chicken	The Barbecue Chicken Pizza	42768
Chicken	The California Chicken Pizza	41409.5
Classic	The Classic Deluxe Pizza	38180.5
Classic	The Hawaiian Pizza	32273.25
Classic	The Pepperoni Pizza	30161.75
Supreme	The Spicy Italian Pizza	34831.25
Supreme	The Italian Supreme Pizza	33476.75