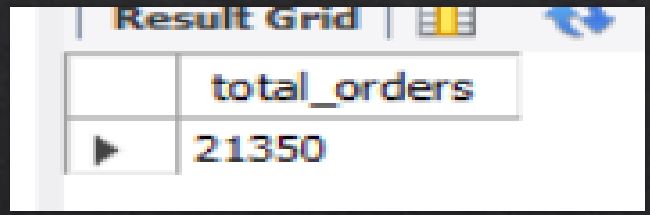
SQL PROJECT FOR PIZZA SALES ANALYSIS

The total number of orders placed.

CODE

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

OUTPUT

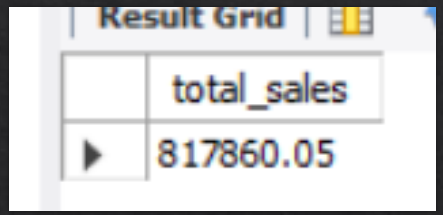


The total revenue generated from pizza sales.

CODE

```
select round(sum(orders_details.quantity*pizzas.price),2) as total_sales
from orders_details join pizzas ON pizzas.pizza_id = orders_details.pizza_id
```

OUTPUT

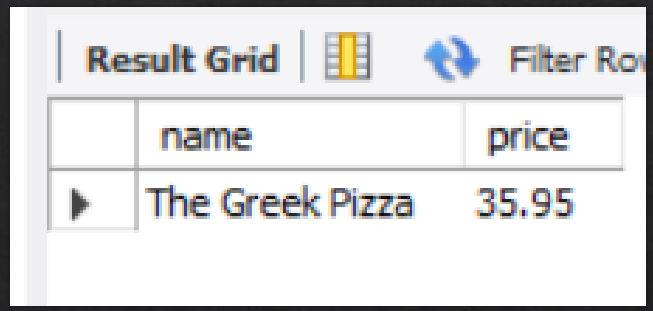


The highest-priced pizza.

CODE

```
select pizza_types.name,pizzas.price
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
order by pizzas.price desc limit 1;
```

OUTPUT

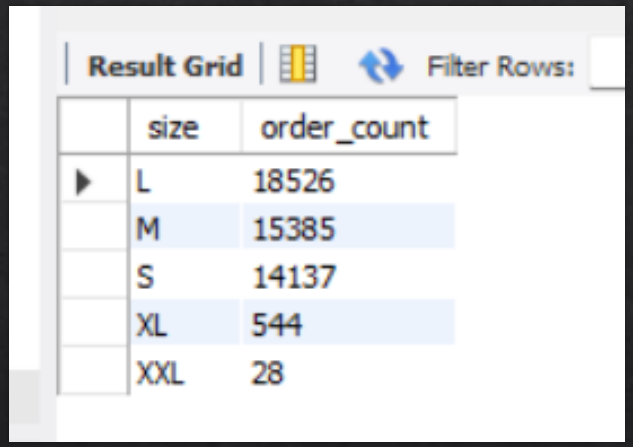


The most common pizza size ordered.

CODE

```
select pizzas.size, count(orders_details.order_details_id) as order_count
from pizzas join orders_details
on pizzas.pizza_id = orders_details.pizza_id
group by pizzas.size order by order_count desc ;
```

OUTPUT



size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

The top 5 most ordered pizza types along with their quantities.

CODE

```
select pizza_types.name,sum(orders_details.quantity) as quantity
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id = pizzas.pizza_id
group by pizza_types.name
order by quantity desc
limit 5;
```

OUTPUT

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

the total quantity of each pizza category ordered.

CODE

```sql
select pizza_types.category,sum(orders_details.quantity) as quantity
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id = pizzas.pizza_id
group by pizza_types.category
order by quantity desc;
```

OUTPUT

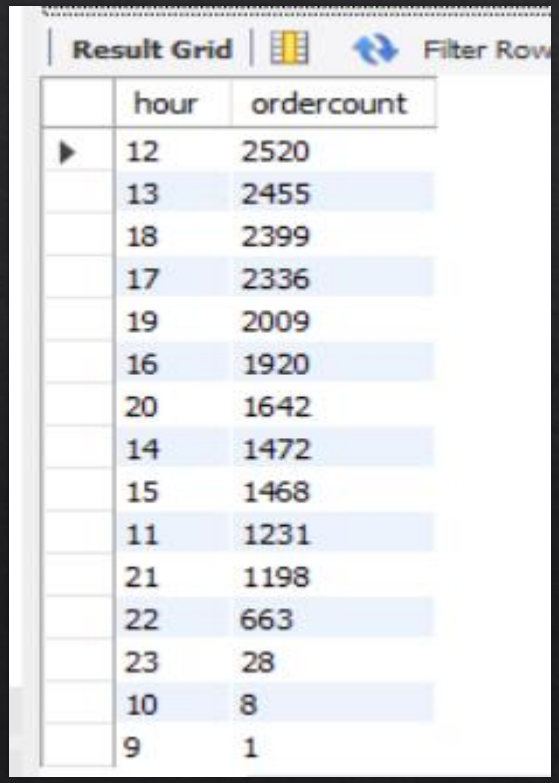
category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

The distribution of orders by hour of the day.

CODE

```
select hour(order_time) as hour , count(order_id) as ordercount
from orders
group by hour(order_time)
order by ordercount desc;
```

OUTPUT



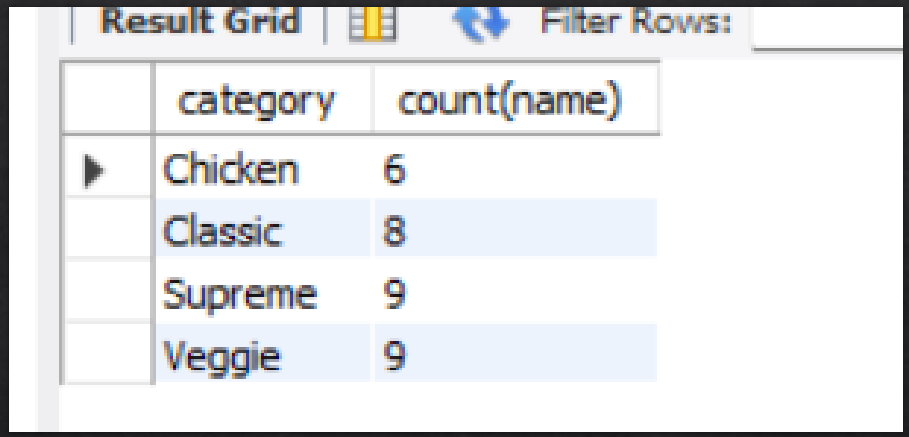
hour	ordercount
12	2520
13	2455
18	2399
17	2336
19	2009
16	1920
20	1642
14	1472
15	1468
11	1231
21	1198
22	663
23	28
10	8
9	1

The category-wise distribution of pizzas.

CODE

```
select category,count(name)
from pizza_types
group by category;
```

OUTPUT



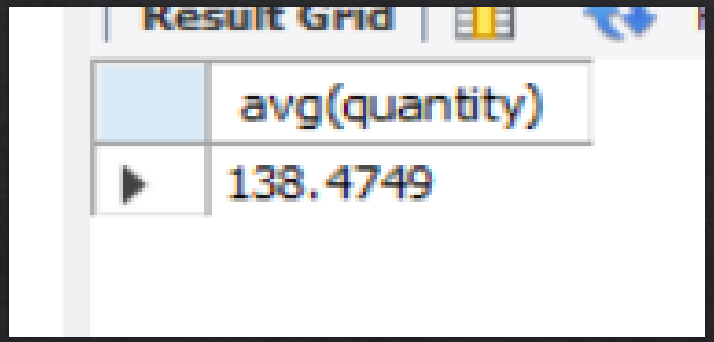
category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

CODE

```
select avg(quantity) from
(select orders.order_date,sum(orders_details.quantity) as quantity
from orders join orders_details
on orders.order_id = orders_details.order_id
group by orders.order_date) as orderquantity;
```

OUTPUT



The top 3 most ordered pizza types based on revenue.

CODE

```
select pizza_types.name, sum(orders_details.quantity * pizzas.price) as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on pizzas.pizza_id = orders_details.pizza_id
group by pizza_types.name order by revenue desc limit 3;
```

OUTPUT

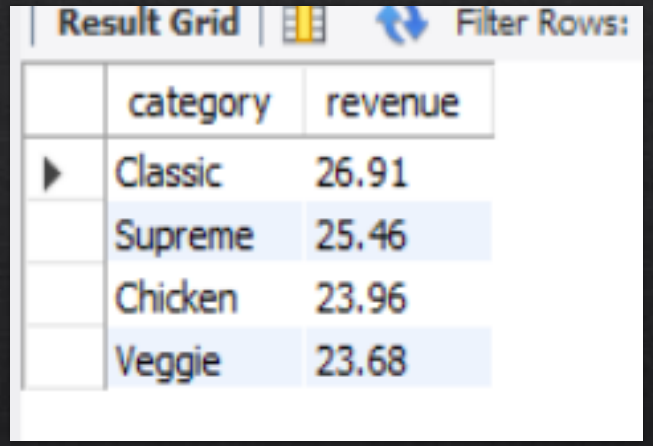
name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

The percentage contribution of each pizza type to total revenue.

CODE

```
select pizza_types.category,
round(sum(orders_details.quantity*pizzas.price)/(select sum(orders_details.quantity*pizzas.price) as total_sales
from orders_details join pizzas ON pizzas.pizza_id = orders_details.pizza_id) * 100,2)  as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id = pizzas.pizza_id
group by pizza_types.category order by revenue desc;
```

OUTPUT



category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

The cumulative revenue generated over time.

CODE

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

OUTPUT

order_date	cum_sales
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001

The top 3 most ordered pizza types based on revenue for each pizza category

CODE

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

OUTPUT

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
Supreme	The Sicilian Pizza	30940.5
Veggie	The Four Cheese Pizza	32265.70000000065
Veggie	The Mexicana Pizza	26780.75
Veggie	The Five Cheese Pizza	26066.5