



SQL PROJECT ON PIZZA SALES





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(ORDERS_DETAILS.QUANTITY * PIZZAS.PRICE),2) AS 'TOTAL SALES'  
FROM ORDERS_DETAILS JOIN PIZZAS  
ON PIZZAS.PIZZA_ID = ORDERS_DETAILS.PIZZA_ID;
```

Result Grid	
	total sales
▶	817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA

```
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;
```

Result Grid			Filter Rows:
	name	price	
▶	The Greek Pizza	35.95	




IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```
SELECT PIZZAS.SIZE, COUNT(ORDERS_DETAILS.QUANTITY) 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;
```

Result Grid			Filter Rows:
	size	order_count	
▶	L	18526	
	M	15385	
	S	14137	
	XL	544	
	XXL	28	

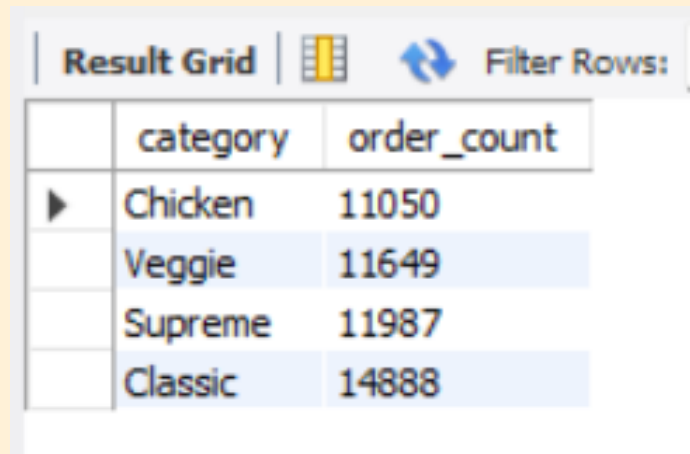
LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES

```
SELECT PIZZAS.PIZZA_TYPE_ID,PIZZA_TYPES.NAME,  
SUM(ORDERS_DETAILS.QUANTITY) AS ORDER_COUNT  
FROM PIZZAS JOIN  
ORDERS_DETAILS ON PIZZAS.PIZZA_ID = ORDERS_DETAILS.PIZZA_ID  
JOIN  
PIZZA_TYPES ON PIZZAS.PIZZA_TYPE_ID = PIZZA_TYPES.PIZZA_TYPE_ID  
GROUP BY PIZZAS.PIZZA_TYPE_ID , PIZZA_TYPES.NAME  
ORDER BY ORDER_COUNT DESC  
LIMIT 5;
```

Result Grid   Filter Rows: <input type="text"/> Export: 			
	pizza_type_id	name	order_count
▶	classic_dlx	The Classic Deluxe Pizza	2453
	bbq_ckn	The Barbecue Chicken Pizza	2432
	hawaiian	The Hawaiian Pizza	2422
	pepperoni	The Pepperoni Pizza	2418
	thai_ckn	The Thai Chicken Pizza	2371

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

```
SELECT PIZZA_TYPES.CATEGORY, SUM(ORDERS_DETAILS.QUANTITY) AS ORDER_COUNT  
FROM PIZZAS JOIN ORDERS_DETAILS  
ON PIZZAS.PIZZA_ID = ORDERS_DETAILS.PIZZA_ID  
JOIN PIZZA_TYPES  
ON PIZZAS.PIZZA_TYPE_ID = PIZZA_TYPES.PIZZA_TYPE_ID  
GROUP BY PIZZA_TYPES.CATEGORY  
ORDER BY ORDER_COUNT ;
```

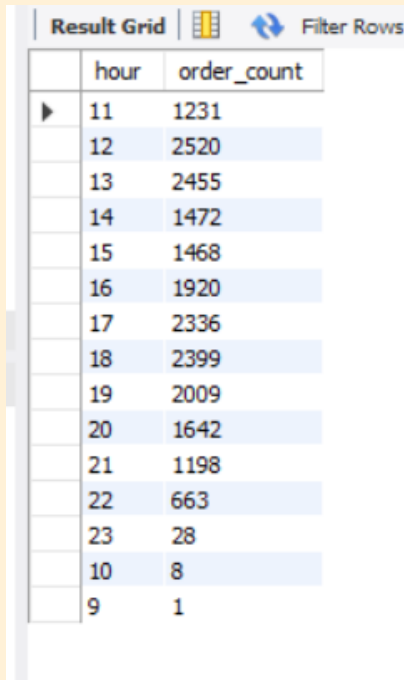


The screenshot shows a database interface with a 'Result Grid' tab. The grid displays the results of the SQL query, showing the total quantity ordered for each pizza category. The columns are 'category' and 'order_count'. The rows are ordered by 'order_count' in descending order.

	category	order_count
▶	Chicken	11050
	Veggie	11649
	Supreme	11987
	Classic	14888

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
SELECT HOUR(ORDER_TIME) AS HOUR, COUNT(ORDER_ID) AS ORDER_COUNT  
FROM ORDERS  
GROUP BY HOUR(ORDER_TIME);
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid displays the results of a SQL query, showing the hour of the day and the corresponding order count. The data is as follows:

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

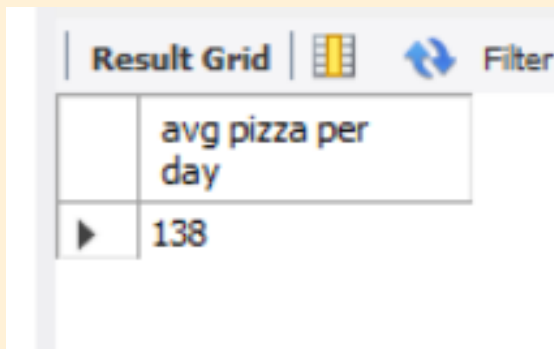
JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS

```
SELECT CATEGORY, COUNT(NAME)  
FROM PIZZA_TYPES  
GROUP BY CATEGORY;
```

Result Grid			Filter Rows:
	category	count(name)	
▶	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),0) AS 'AVG PIZZA PER DAY' FROM  
(SELECT O.ORDER_DATE, SUM(OD.QUANTITY) AS QUANTITY  
FROM ORDERS O JOIN ORDERS_DETAILS OD  
ON O.ORDER_ID = OD.ORDER_ID  
GROUP BY O.ORDER_DATE) AS ORDER_QUANTITY;
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid contains one row with the column header 'avg pizza per day' and a single data value '138'. There are icons for a grid, a refresh button, and a filter button at the top of the grid.

	avg pizza per day
▶	138

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
SELECT PT.NAME,  
SUM(OD.QUANTITY * P.PRICE) AS REVENUE  
FROM PIZZA_TYPES PT JOIN PIZZAS P  
ON PT.PIZZA_TYPE_ID = P.PIZZA_TYPE_ID  
JOIN ORDERS_DETAILS OD  
ON OD.PIZZA_ID = P.PIZZA_ID  
GROUP BY PT.NAME ORDER BY REVENUE DESC LIMIT 3;
```

Result Grid			Filter Rows:
	name	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 ,  
ROUND(SUM(ORDERS_DETAILS.QUANTITY * PIZZAS.PRICE) / (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)*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;
```

Result Grid			Filter
	category	revenue	
▶	Classic	26.91	
	Supreme	25.46	
	Chicken	23.96	
	Veggie	23.68	

-- 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(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;
```

Result Grid	Filter Rows:
order_date	cum_revenue
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	34343.500000000001
2015-01-16	36937.650000000001
2015-01-17	39001.750000000001
2015-01-18	40978.600000000006

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

```
SELECT NAME, REVENUE FROM  
(SELECT CATEGORY, NAME, REVENUE,  
RANK() OVER(PARTITION BY CATEGORY ORDER BY REVENUE DESC ) AS RN  
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 RN <= 3;
```

Result Grid			Filter Rows:	Exp
	name	revenue		
▶	The Thai Chicken Pizza	43434.25		
	The Barbecue Chicken Pizza	42768		
	The California Chicken Pizza	41409.5		
	The Classic Deluxe Pizza	38180.5		
	The Hawaiian Pizza	32273.25		
	The Pepperoni Pizza	30161.75		
	The Spicy Italian Pizza	34831.25		
	The Italian Supreme Pizza	33476.75		
	The Sicilian Pizza	30940.5		
	The Four Cheese Pizza	32265.70000000065		
	The Mexicana Pizza	26780.75		
	The Five Cheese Pizza	26066.5		