1	--How many customers do we have in the data? SELECT COUNT (c.customer_name) FROM customers AS c;
2,3	--What was the city with the most profit for the company in 2015? --In 2015, what was the most profitable city's profit? SELECT DISTINCT o.shipping_city, SUM(od.order_profits) AS profits FROM orders AS o INNER JOIN order_details AS od ON od.order_id = o.order_id WHERE o.order_date BETWEEN '2015-01-01 00:00:00' AND '2015-12-31 00:00:00' GROUP BY o.shipping_city ORDER BY profits DESC LIMIT 1;
4	--How many different cities do we have in the data? SELECT COUNT(DISTINCT o.shipping_city) AS city_count FROM orders AS o;
5	--Show the total spent by customers from low to high. SELECT DISTINCT o.customer_id, c.customer_name, SUM(od.order_sales) AS total_amount_per_customer FROM orders AS o INNER JOIN customers AS c ON o.customer_id = c.customer_id INNER JOIN order_details AS od ON o.order_id = od.order_id GROUP BY 1,2 ORDER BY 3 ASC;
6	--What is the most profitable city in the State of Tennessee? SELECT DISTINCT o.shipping_city, SUM(od.order_profits) AS sum_profits FROM orders AS o RIGHT JOIN order_details AS od ON od.order_id = o.order_id WHERE o.shipping_state ='Tennessee' GROUP BY o.shipping_city ORDER BY 2 DESC;
7	--What's the average annual profit for that city across all years?

	``sql SELECT DISTINCT o.shipping_city, CAST(AVG(od.order_profits) AS DECIMAL(10,2)) FROM orders AS o RIGHT JOIN order_details AS od ON od.order_id = o.order_id WHERE o.shipping_state ='Tennessee' AND o.shipping_city = 'Lebanon' GROUP BY o.shipping_city ORDER BY 2 DESC; ```
8	```sql --What is the distribution of customer types in the data? SELECT CASE WHEN c.customer_segment='Home Office' THEN 'Home Office' WHEN c.customer_segment='Consumer' THEN 'Consumer' ELSE 'Corporate' END AS customer_type, COUNT (*) AS distribution FROM customers AS c GROUP BY 1 ORDER BY 2; ```
9	```sql --the most profitable product category on average in Iowa across all years? SELECT p.product_category, CAST(AVG (od.order_profits) AS DECIMAL(10,2)) FROM orders AS o INNER JOIN order_details AS od ON o.order_id = od.order_id INNER JOIN product AS p ON od.product_id = p.product_id WHERE o.shipping_state = 'Iowa' GROUP BY p.product_category ORDER BY 2 DESC; ```
10	```sql --the most popular product in that category across all states in 2016? SELECT DISTINCT p.product_name, SUM(od.quantity) as quantity_product_sold FROM orders AS o INNER JOIN order_details AS od ON o.order_id = od.order_id INNER JOIN product AS p ON od.product_id = p.product_id WHERE o.order_date BETWEEN '2016-01-01' AND '2016-31-12' AND p.product_category ='Furniture' GROUP BY p.product_name ORDER BY 2 DESC LIMIT 1; ```
11	```sql --Which customer got the most discount in the data? (in total amount) ```

	``sql SELECT c.customer_name, c.customer_id, CAST(SUM(od.order_discount*od.order_sales) AS DECIMAL (10,2)) FROM orders o INNER JOIN order_details od ON o.order_id = od.order_id INNER JOIN customers c ON o.customer_id = c.customer_id GROUP BY 1,2 ORDER BY 3 DESC; ```
12	```sql --How widely did monthly profits varied in 2018? WITH t1 AS (SELECT to_char(o.order_date, 'MM-YYYY') AS month, SUM(od.order_profits) AS monthly_profit FROM orders o INNER JOIN order_details od USING (order_id) WHERE DATE_PART('year', o.order_date) = 2018 GROUP BY 1) SELECT month, monthly_profit, COALESCE(monthly_profit - LAG(monthly_profit,1) OVER (ORDER BY month), monthly_profit) AS monthly_difference FROM t1; ```
13	```sql --Which order was the highest in 2015? SELECT o.order_id, SUM(od.order_sales) as sum_of_sales_per_order FROM orders AS o INNER JOIN order_details AS od ON o.order_id = od.order_id WHERE o.order_date BETWEEN 'Jan 01 2015' AND 'DEC 31 2015' GROUP BY 1 ORDER BY 2 DESC LIMIT 1; ```
14	```sql --What was the rank of each city in the East region in 2015? ```

	``sql SELECT RANK() OVER (order by SUM(od.quantity) DESC) rank_of_city, o.shipping_city FROM orders AS o INNER JOIN order_details AS od ON o.order_id = od.order_id WHERE o.order_date BETWEEN 'Jan 01 2015' AND 'DEC 31 2015' AND o.shipping_region ='East'-- GROUP BY 2 ```
15	```sql --Display customer names for customers who are in the segment 'Consumer' --or 'Corporate.' How many customers are there in total? SELECT DISTINCT c.customer_name, c.customer_segment FROM customers as c WHERE c.customer_segment ='Corporate' OR c.customer_segment ='Consumer' ORDER BY 2,1 ASC; ```
16	```sql --Calculate the difference between the largest and --smallest order quantities for product id '100.' SELECT MAX(od.quantity) - MIN(od.quantity) AS difference_between_max_min FROM order_details AS od INNER JOIN product AS p ON p.product_id = od.product_id WHERE p.product_id = 100 ```
17	```sql --Calculate the percent of products that are within the category 'Furniture.' SELECT p. product_category, ROUND(COUNT(*) * 100.0 / (SELECT COUNT(*) FROM product), 2) AS perccentage FROM product AS p WHERE p.product_category = 'Furniture' GROUP BY 1; ```
18	```sql --number of duplicate products based on their product manufacturer. SELECT p.product_manufacturer, COUNT(*) FROM product AS p GROUP BY 1 HAVING COUNT(*) > 1 ```
19	```sql --Show the order from most to least products and then by product_ ```

	``sql --subcategory name ascending. SELECT p.product_subcategory, COUNT(*) AS TotalProducts FROM product AS p GROUP BY 1 ORDER BY 2 DESC, 1 ASC; ```
20	```sql --Show the product_id(s), the sum of quantities, where the total sum of its --product quantities is greater than or equal to 100. Select od.product_id, SUM(od.quantity) AS quantity_product_min_100 FROM order_details AS od GROUP by 1 HAVING SUM(quantity) >= 100 ORDER by 2 DESC, 1 ASC; ```
Bonus	```sql --Join all database tables into one dataset that includes all unique --columns and download it as a .csv file SELECT * FROM customers JOIN orders USING (customer_id) JOIN order_details USING (order_id) JOIN product USING (product_id) ```