# SQL Assignment Solution

```
-- 1. Pull total number of orders that were completed on 18th March 2023
SELECT COUNT(1) AS total_orders
FROM SALES
WHERE DATE = '2023-03-18';

-- 2. Pull total number of orders that were completed on 18th March 2023 with the first name 'John' and last name 'Doe'
SELECT COUNT(s.Order_id) AS total_orders
FROM SALES AS s
INNER JOIN CUSTOMERS AS c ON s.Customer_id = c.Customer_id
WHERE s.DATE = '2023-03-18'
  AND c.first_name = 'John'
  AND c.last_name = 'Doe';

-- 3. Pull total number of customers that purchased in January 2023 and the average amount spent per customer
SELECT COUNT(DISTINCT s.Customer_id) AS total_customers,
       ROUND(AVG(customer_spent.total_spent), 2) AS average_spent
FROM (
    SELECT Customer_id, SUM(Revenue) AS total_spent
    FROM SALES
    WHERE DATE BETWEEN '2023-01-01' AND '2023-01-31'
    GROUP BY Customer_id
) AS customer_spent
```

```
JOIN SALES AS s ON customer_spent.Customer_id = s.Customer_id;


-- 4. Pull the departments that generated less than $600 in 2022
SELECT i.department, SUM(s.Revenue) AS total_revenue
FROM ITEMS AS i
INNER JOIN SALES AS s ON i.Item_id = s.Item_id
WHERE s.DATE BETWEEN '2022-01-01' AND '2022-12-31'
GROUP BY i.department
HAVING total_revenue < 600;


-- 5. What is the most and least revenue we have generated by an order
SELECT MAX(s.Revenue) AS most_revenue, MIN(s.Revenue) AS least_revenue
FROM SALES AS s;


-- 6. What were the orders that were purchased in our most lucrative order
WITH RevenueRanked AS (
    SELECT Order_id, Revenue, RANK() OVER (ORDER BY Revenue DESC) AS rank_order
    FROM SALES
)
SELECT s.*
FROM SALES AS s
JOIN RevenueRanked AS r ON s.Order_id = r.Order_id
WHERE r.rank_order = 1;
```