1. Find customers who placed the highest quantity order

```
SELECT CustomerID
FROM Orders
WHERE Quantity = (SELECT MAX(Quantity) FROM Orders);
```

Explanation:
Inner query finds the maximum quantity ever ordered.
Outer query returns all customers whose orders match that quantity.

2. List products with price above the average product price

```
SELECT ProductName, Price
FROM Products
WHERE Price > (SELECT AVG(Price) FROM Products);
```

Explanation:
The subquery gets the average price of all products.
The outer query lists products priced higher than that average.

3. Show customers whose total spending is above the average spending

```
SELECT CustomerID
FROM Orders
GROUP BY CustomerID
HAVING SUM(Quantity) > (
    SELECT AVG(total_spent)
    FROM (
        SELECT SUM(Quantity) AS total_spent
        FROM Orders
        GROUP BY CustomerID
    ) AS sub
);
```

Explanation:
Inner subquery: finds total quantity (or spending) per customer.
Outer query: compares each customer·s total to the average of all customers.
(You could replace SUM(Quantity) with SUM(Price * Quantity) if product prices were available, but that would ne

4. Find the most expensive product ordered

```
SELECT ProductID
FROM Orders
WHERE ProductID IN (
    SELECT ProductID
    FROM Products
    WHERE Price = (SELECT MAX(Price) FROM Products)
);
```

Explanation:
Inner subquery finds product(s) with the max price.
Outer query lists orders for those products.
(Still technically references multiple tables but avoids explicit JOIN.)

5. List customers who ordered the cheapest product

```
SELECT DISTINCT CustomerID
FROM Orders
WHERE ProductID IN (
    SELECT ProductID
    FROM Products
    WHERE Price = (SELECT MIN(Price) FROM Products)
);
```

Explanation:
Inner subquery finds product(s) with the minimum price.
Outer query lists customers who ordered those products.