Chapter 04 In-class Lab Assignment

ISTA-420, T-SQL Fundamentals

In-class Lab — Subqueries

Use subqueries to execute the following queries. Note that it may be possible to use joins and sets to do the same thing.

Using the TSQLV4 database, part 1

1. What is our highest priced product? Report the product id, product name, and unit price. Use a self-contained, scalar valued subquery
2. What is our most popular product in terms of quantity sold? Report the product name and product id. Use a self-contained, scalar valued subquery.
3. Who is our top salesperson overall? Include the employee id, title, first name, and last name. Use a self-contained, scalar valued subquery.
4. I want to examine the individual orders. Specifically, I want to look at each order and compare the total of each order line with the average total of all order lines in the order. Report the order id, the total of each order line, and the average of all order lines for each order. Use a correlated, scalar valued subquery.
5. What is the largest quantity ordered by a customer for every order? Report the order id, the product id, and the quantity ordered. Use a correlated subquery in the WHERE clause.
6. I need to see all orders placed on the first day of the month. Report the order id, the customer id, the employee id, and the order date.Use a correlated, scalar valued subquery in the WHERE clause.
7. What orders requested beverages? Report the ordeer id and the product id. Use nested self-contained, scalar valued subqueries in the WHERE clause.
8. I need a list of all our foreign suppliers, i.e., non-American. Include the supplier id, supplier name, and country. Use a self-contained, list valued subquery.
9. Report the nae of all meat and poultry products. Use nested subqueries. The outer nested subquery is a self-contained, list valued subqueery. The inner subquery is a scalar valued, self-contained subquery.

Using the Northwind database

1. Create a report that shows all orders taken by Janet.
2. Create a report that shows all products by name that are in the Seafood category.
3. Create a report that shows all orders taken by any employee whose last name begins with “A.”
4. Create a report that shows the product name and supplier id for all products supplied by Exotic Liquids, Grandma Kelly’s Homestead, and Tokyo Traders.
5. Create a report that shows all products supplied from the Pacific Ocean region.
6. Create a report that shows all companies by name that sell products in CategoryID 8.

7. Create a report in two parts that shows the date of the last sale made by each employee, and the date of the first sale made by each employee.
8. What is the product number of our most expensive product? Create a report that shows the employee id and order id of every order for that product.
9. Create a report showing the date of the last sale for every product, ordered by product id.
10. Create a report that shows all companies by name that sell products in the Seafood category.
11. Create a report that lists the ten most expensive products.
12. Create a report that shows the date of the last order by all employees.
13. Create a line item report that contains a line for each product in the order with the following columns: the order id, the product id, the unit price, the quantity sold, the line item price, and the percent of that line item constitutes of the total amount of the order.

Using the TSQLV4 database, part 2

Use the book's database, TSQLV4, and do the exercises 1 through 10, beginning on page 150. The solutions are in the book beginning on page 154.

Solutions to the lab queries

Attempt to write the queries before you look at the solutions. Do not look at the solutions before you attempt to write the query.

TSQLV4 queries, part 1

```
1  -- 1
2  select p.productid, p.productname, p.unitprice
3    from production.products p
4    where p.unitprice = (
5      select max(pp.unitprice )
6      from production.products pp);
7
8  -- 2
9  select p.productid, p.productname
10   from production.products p
11   where p.productid = (
12     select top 1 od.productid
13         from sales.OrderDetails od
14     group by od.productid order by sum(od.qty) desc);
15
16 -- 3
17 select e.empid, e.title, e.firstname, e.lastname
18   from hr.Employees e
19   where e.empid = (
20     select top 1 o.empid
21     from Sales.Orders o
22     group by o.empid
23     order by sum(o.orderid) desc);
24
25 -- 4
26 select od.orderid, (od.unitprice * od.qty) as line_total,
27   (select avg(ood.unitprice * ood.qty)
28     from sales.orderdetails ood
29         where od.orderid = ood.orderid) as avg_orderline
30   from Sales.OrderDetails od order by od.orderid;
31
32 -- 5
33 select o.custid, o.orderid, (
34   select max(od.qty)
35     from sales.OrderDetails od
36     where o.orderid = od.orderid) as most_product
37  from sales.orders o;
38
39 -- 6
40 select o.orderid, o.custid, o.empid, o.orderdate
41   from sales.orders o
42   where o.orderdate = (
43     select oo.orderdate from sales.Orders oo
44           where datepart(day, oo.orderdate) = 1 and o.orderid = oo.orderid);
45
46 -- 7
47 select od.orderid, od.productid from sales.OrderDetails od
48   where od.productid in (
49   select p.productid
50     from Production.Products p
51     where p.categoryid = (
52       select c.categoryid from Production.Categories c
53             where c.categoryname like 'Bev%'));
54 -- 8
55 select s.supplierid, s.companyname, s.country
56   from Production.Suppliers s where s.supplierid not in (
57   select ss.supplierid from Production.Suppliers ss
58   where ss.country like 'USA');
```

```
59
60 -- 9
61 select productname from Production.Products
62   where productid in (
63     select p.productid
64       from Production.Products p where p.categoryid = (
65   select c.categoryid from Production.Categories c
66     where categoryname like '%meat%'));
```

Northwind queries

```
1  -- 1
2  select orderid, orderdate from orders where employeeid =
3    (select employeeid from employees where firstname = 'Janet');
4
5  -- 2
6  SELECT ProductName
7  FROM Products
8  WHERE CategoryID = (SELECT CategoryID
9     FROM Categories
10    WHERE CategoryName = 'Seafood');
11
12 -- 3
13 select orderid, orderdate from orders where employeeid =
14  (select employeeid from employees where firstname like 'A%');
15
16 -- 4
17 SELECT ProductName, SupplierID
18 FROM Products
19 WHERE SupplierID IN (SELECT SupplierID
20     FROM Suppliers
21     WHERE CompanyName IN
22          ('Exotic Liquids', 'Grandma Kelly''s Homestead', 'Tokyo Traders'));
23
24 -- 5
25 select productname from products where supplierid in
26 (select supplierid from suppliers where country in ('Japan','Singapore','Australia'));
27
28 -- 6
29 SELECT CompanyName
30 FROM Suppliers
31 WHERE SupplierID IN (SELECT SupplierID
32    FROM Products
33    WHERE CategoryID = 8);
34
35 -- 7
36 select o1.employeeid, o1.orderdate from orders o1 where o1.orderdate =
37   (select max(o2.orderdate) from orders o2 where o2.employeeid = o1.employeeid)
38   order by o1.employeeid;
39 select o1.employeeid, o1.orderdate from orders o1 where o1.orderdate =
40   (select min(o2.orderdate) from orders o2 where o2.employeeid = o1.employeeid)
41   order by o1.employeeid;
42
43 -- 8
44 select o.orderid, o.employeeid from orders o where exists
45   (select od.orderid from order_details od
46     where od.productid = 38 and o.orderid = od.orderid);
47
48 -- 9
49 select p.productid, p.productname, lastdate.lastsale from products p,
50    (select od.productid, od.orderid, max(o.orderdate) as lastsale
51       from order_details od join orders o on od.orderid = o.orderid
52           group by od.productid) lastdate
53 where p.productid = lastdate.productid order by p.productid;
54
55 -- 10
56 SELECT CompanyName
```

```
57 FROM Suppliers
58 WHERE SupplierID IN (SELECT SupplierID
59    FROM Products
60    WHERE CategoryID = (SELECT CategoryID
61       FROM Categories
62       WHERE CategoryName = 'Seafood'));
63
64 -- 11
65 select * from
66 (
67     select distinct ProductName as Ten_Most_Expensive_Products,
68            UnitPrice
69     from Products
70     order by UnitPrice desc
71 ) as a
72   limit 10;
73
74 -- 12
75 SELECT OrderID, CustomerID, EmployeeID, OrderDate, RequiredDate
76 FROM Orders AS O1
77 WHERE OrderDate =
78   (SELECT MAX(OrderDate)
79    FROM Orders AS O2
80    WHERE O2.EmployeeID = O1.EmployeeID);
81
82 -- 13
83 select od1.orderid, od1.productid, od1.unitprice, od1.quantity,
84   (od1.unitprice * od1.quantity) as LineTotal,
85   round((od1.unitprice * od1.quantity) /
86   (select sum(od2.unitprice * od2.quantity) from order_details od2
87       where od1.orderid = od2.orderid) * 100, 2) as OrderTotal
88 from order_details od1 order by od1.orderid limit 50;
```