Ch7 Problems Part 3

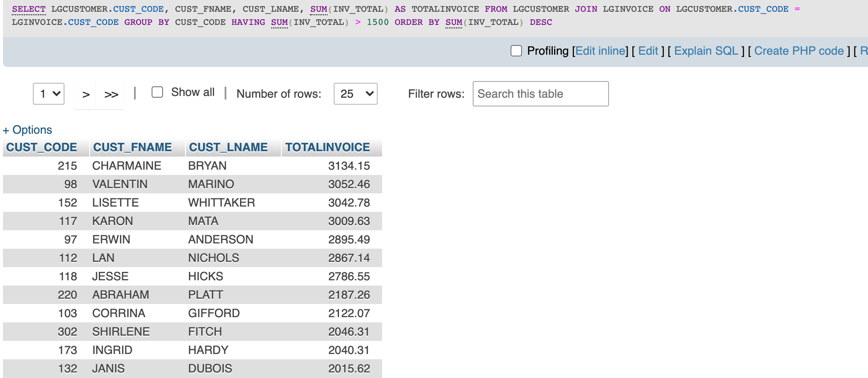
41. Write a query to display the customer code, first name, last name, and sum of all invoice totals for customers with cumulative invoice totals greater than $1,500. Sort the output by the sum of invoice totals in descending order (Figure P7.41)

SELECT LGCUSTOMER.CUST\_CODE, CUST\_FNAME, CUST\_LNAME, SUM(INV\_TOTAL) AS TOTALINVOICE

FROM LGCUSTOMER JOIN LGINVOICE ON LGCUSTOMER.CUST\_CODE = LGINVOICE.CUST\_CODE

GROUP BY CUST\_CODE

HAVING SUM(INV\_TOTAL) > 1500

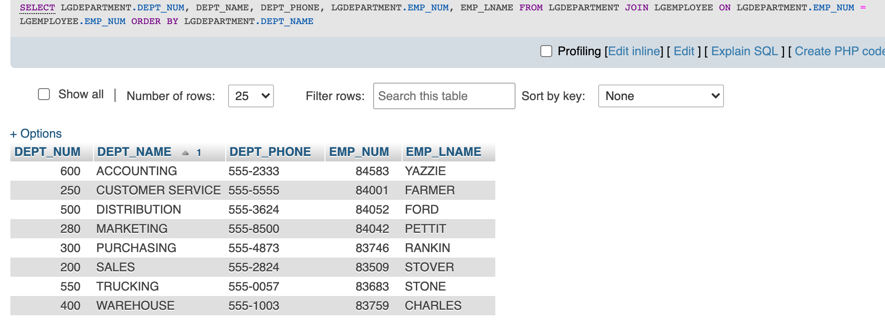
ORDER BY SUM(INV\_TOTAL) DESC;

1. Write a query to display the department number, department name, department phone number, employee number, and last name of each department manager. Sort the output by department name (Figure P7.42).

SELECT LGDEPARTMENT.DEPT\_NUM, DEPT\_NAME, DEPT\_PHONE, LGDEPARTMENT.EMP\_NUM, EMP\_LNAME

FROM LGDEPARTMENT JOIN LGEMPLOYEE ON LGDEPARTMENT.EMP\_NUM = LGEMPLOYEE.EMP\_NUM

ORDER BY LGDEPARTMENT.DEPT\_NAME;



1. Write a query to display the vendor ID, vendor name, brand name, and number of products of each brand supplied by each vendor. Sort the output by vendor name and then by brand name (Figure P7.43).

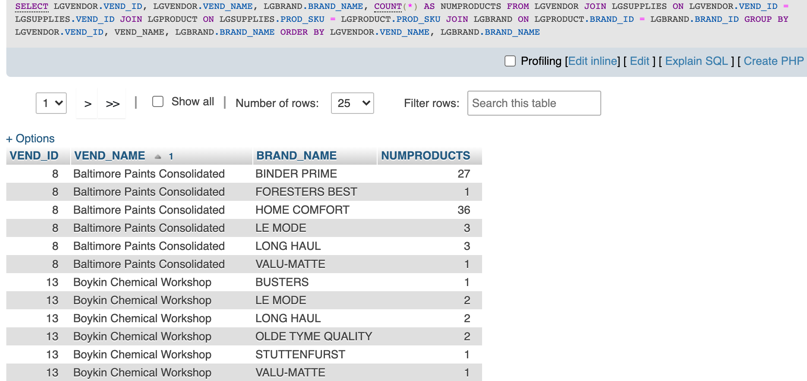
SELECT LGVENDOR.VEND\_ID, LGVENDOR.VEND\_NAME, LGBRAND.BRAND\_NAME, COUNT(\*) AS NUMPRODUCTS

FROM LGVENDOR JOIN LGSUPPLIES ON LGVENDOR.VEND\_ID = LGSUPPLIES.VEND\_ID

JOIN LGPRODUCT ON LGSUPPLIES.PROD\_SKU = LGPRODUCT.PROD\_SKU

JOIN LGBRAND ON LGPRODUCT.BRAND\_ID = LGBRAND.BRAND\_ID

GROUP BY LGVENDOR.VEND\_ID, VEND\_NAME, LGBRAND.BRAND\_NAME

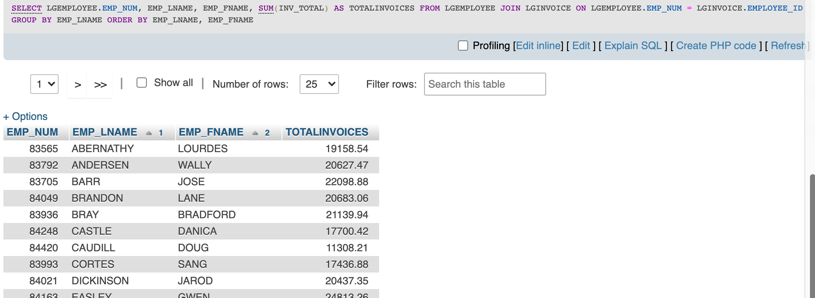
ORDER BY LGVENDOR.VEND\_NAME, LGBRAND.BRAND\_NAME;

1. Write a query to display the employee number, last name, first name, and sum of invoice totals for all employees who completed an invoice. Sort the output by employee last name and then by first name (Figure P7.44).

SELECT LGEMPLOYEE.EMP\_NUM, EMP\_LNAME, EMP\_FNAME, SUM(INV\_TOTAL) AS TOTALINVOICES

FROM LGEMPLOYEE JOIN LGINVOICE ON LGEMPLOYEE.EMP\_NUM = LGINVOICE.EMPLOYEE\_ID

GROUP BY EMP\_LNAME

ORDER BY EMP\_LNAME, EMP\_FNAME;

1. Write a query to display the largest average product price of any brand (FigureP7.45).

SELECT MAX(AVGPRICE) AS 'LARGEST AVERAGE'

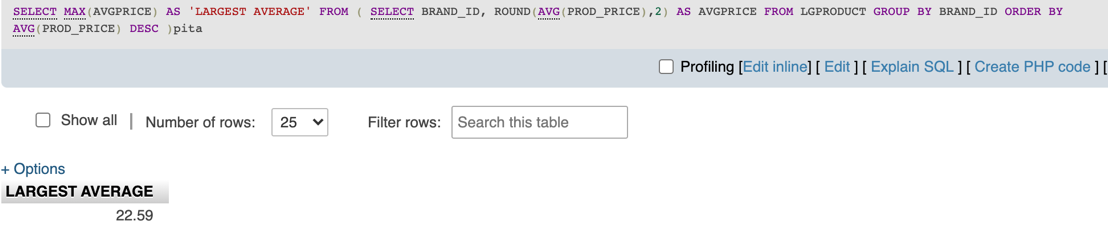
FROM (

SELECT BRAND\_ID, ROUND(AVG(PROD\_PRICE),2) AS AVGPRICE

FROM LGPRODUCT

GROUP BY BRAND\_ID

ORDER BY AVG(PROD\_PRICE) DESC

 )pita;

1. Write a query to display the brand ID, brand name, brand type, and average price of products for the brand that has the largest average product price (Figure P7.46).

SELECT BRAND\_ID, BRAND\_NAME, BRAND\_TYPE, MAX(AVGPRICE) AS 'AVGPRICE'

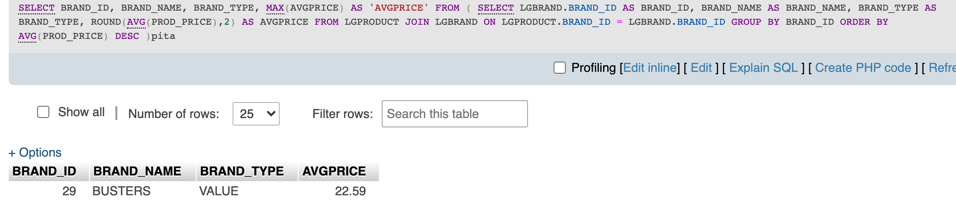
FROM (

SELECT LGBRAND.BRAND\_ID AS BRAND\_ID, BRAND\_NAME AS BRAND\_NAME, BRAND\_TYPE AS BRAND\_TYPE, ROUND(AVG(PROD\_PRICE),2) AS AVGPRICE

FROM LGPRODUCT JOIN LGBRAND ON LGPRODUCT.BRAND\_ID = LGBRAND.BRAND\_ID

GROUP BY BRAND\_ID

ORDER BY AVG(PROD\_PRICE) DESC

 )pita;

1. Write a query to display the manager name, department name, department phone number, employee name, customer name, invoice date, and invoice total for the department manager of the employee who made a sale to a customer whose last name is Hagan on May 18, 2015 (Figure P7.47).

SELECT

m.EMP\_FNAME AS 'Manager FNAME',

m.EMP\_LNAME 'Manager LName',

DEPT\_NAME, DEPT\_PHONE,

e.EMP\_FNAME AS 'Employee FName',

e.EMP\_LNAME AS 'Employee LName',

CUST\_FNAME, CUST\_LNAME,

INV\_DATE, INV\_TOTAL

FROM

LGEMPLOYEE m JOIN LGDEPARTMENT ON m.EMP\_NUM = LGDEPARTMENT.EMP\_NUM

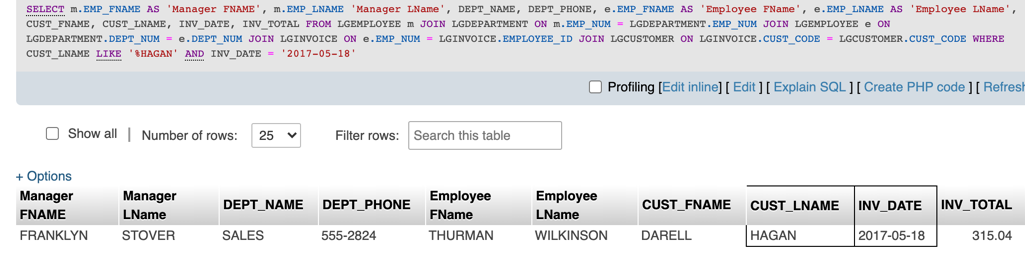
JOIN LGEMPLOYEE e ON LGDEPARTMENT.DEPT\_NUM = e.DEPT\_NUM

JOIN LGINVOICE ON e.EMP\_NUM = LGINVOICE.EMPLOYEE\_ID

JOIN LGCUSTOMER ON LGINVOICE.CUST\_CODE = LGCUSTOMER.CUST\_CODE

WHERE

CUST\_LNAME LIKE '%HAGAN' AND INV\_DATE = '2017-05-18';

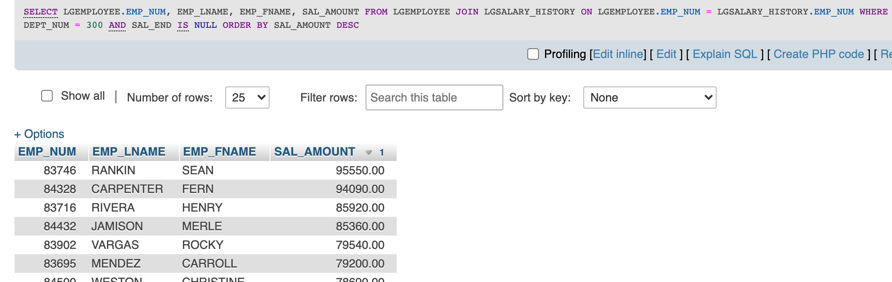


1. Write a query to display the current salary for each employee in department 300. Assume that only current employees are kept in the system, and therefore the most current salary for each employee is the entry in the salary history with a NULL end date. Sort the output in descending order by salary amount (Figure P7.48).

SELECT LGEMPLOYEE.EMP\_NUM, EMP\_LNAME, EMP\_FNAME, SAL\_AMOUNT

FROM LGEMPLOYEE JOIN LGSALARY\_HISTORY ON LGEMPLOYEE.EMP\_NUM = LGSALARY\_HISTORY.EMP\_NUM

WHERE DEPT\_NUM = 300 AND SAL\_END IS NULL

ORDER BY SAL\_AMOUNT DESC;

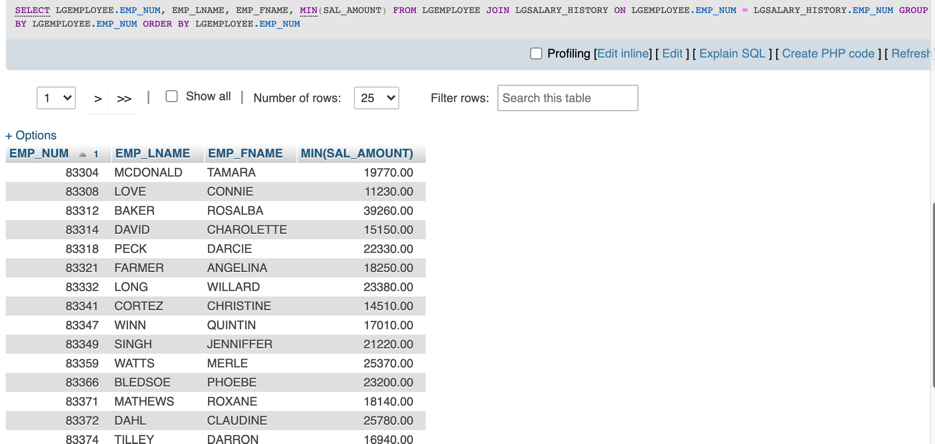
1. Write a query to display the starting salary for each employee. The starting salary would be the entry in the salary history with the oldest salary start date for each employee. Sort the output by employee number (Figure P7.49).

SELECT LGEMPLOYEE.EMP\_NUM, EMP\_LNAME, EMP\_FNAME, MIN(SAL\_AMOUNT)

FROM LGEMPLOYEE JOIN LGSALARY\_HISTORY ON LGEMPLOYEE.EMP\_NUM = LGSALARY\_HISTORY.EMP\_NUM

GROUP BY LGEMPLOYEE.EMP\_NUM

ORDER BY LGEMPLOYEE.EMP\_NUM;



1. Write a query to display the invoice number, line numbers, product SKUs, product descriptions, and brand ID for sales of sealer and top coat products of the same brand on the same invoice. Sort the results by invoice number in ascending order, first line number in ascending order, and then by second line number in descending order (Figure P7.50).

SELECT

c1.INV\_NUM,

c1.LINE\_NUM AS 'p1 Line Num',

c1.PROD\_SKU AS 'p1 Prod Sku',

c1.PROD\_DESCRIPT AS 'p1 Prod Descript',

c2.LINE\_NUM AS 'p2 Line Num',

c2.PROD\_SKU AS 'p2 Prod Sku',

c2.PROD\_DESCRIPT AS 'p2 Prod Descript',

c1.BRAND\_ID

FROM

(SELECT l.INV\_NUM, l.LINE\_NUM , p.PROD\_SKU , p.PROD\_DESCRIPT , p.BRAND\_ID, p.PROD\_CATEGORY

FROM

LGLINE l JOIN LGPRODUCT p ON l.PROD\_SKU = p.PROD\_SKU

WHERE p.PROD\_CATEGORY = 'Sealer') c1,

(SELECT l2.LINE\_NUM , p2.PROD\_SKU , p2.PROD\_DESCRIPT , p2.BRAND\_ID, l2.INV\_NUM, p2.PROD\_CATEGORY

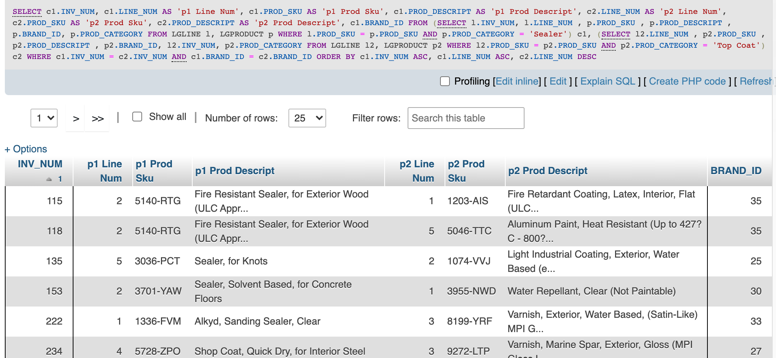
FROM

LGLINE l2 JOIN LGPRODUCT p2 ON l2.PROD\_SKU = p2.PROD\_SKU

WHERE p2.PROD\_CATEGORY = 'Top Coat') c2

WHERE c1.INV\_NUM = c2.INV\_NUM AND c1.BRAND\_ID = c2.BRAND\_ID

ORDER BY c1.INV\_NUM ASC, c1.LINE\_NUM ASC, c2.LINE\_NUM DESC;



1. The Binder Prime Company wants to recognize the employee who sold the most of its products during a specified period. Write a query to display the employee number, employee first name, employee last name, email address, and total units sold for the employee who sold the most Binder Prime brand products between November 1, 2015, and December 5, 2015. If there is a tie for most units sold, sort the output by employee last name (Figure P7.51).

SELECT EMP\_NUM, EMP\_FNAME, EMP\_LNAME, EMP\_EMAIL, SUM(LGLINE.LINE\_QTY) AS TOTAL

FROM

LGEMPLOYEE JOIN LGINVOICE ON LGEMPLOYEE.EMP\_NUM = LGINVOICE.EMPLOYEE\_ID

JOIN LGLINE ON LGINVOICE.INV\_NUM = LGLINE.INV\_NUM

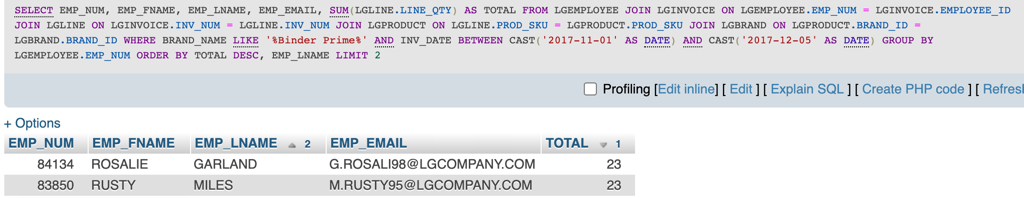
JOIN LGPRODUCT ON LGLINE.PROD\_SKU = LGPRODUCT.PROD\_SKU

JOIN LGBRAND ON LGPRODUCT.BRAND\_ID = LGBRAND.BRAND\_ID

WHERE BRAND\_NAME LIKE '%Binder Prime%' AND INV\_DATE BETWEEN CAST('2017-11-01' AS DATE) AND CAST('2017-12-05' AS DATE)

GROUP BY LGEMPLOYEE.EMP\_NUM

ORDER BY TOTAL DESC, EMP\_LNAME

LIMIT 2;

1. Write a query to display the customer code, first name, and last name of all customers who have had at least one invoice completed by employee 83649 and at least one invoice completed by employee 83677. Sort the output by customer last name and then first name (Figure P7.52).

SELECT c1.CUST\_CODE, c1.CUST\_FNAME, c1.CUST\_LNAME

FROM

(

SELECT LGCUSTOMER.CUST\_CODE, CUST\_FNAME, CUST\_LNAME

FROM LGCUSTOMER JOIN LGINVOICE ON LGCUSTOMER.CUST\_CODE = LGINVOICE.CUST\_CODE

WHERE EMPLOYEE\_ID = 83649

)c1,

(

SELECT LGCUSTOMER.CUST\_CODE, CUST\_FNAME, CUST\_LNAME

FROM LGCUSTOMER JOIN LGINVOICE ON LGCUSTOMER.CUST\_CODE = LGINVOICE.CUST\_CODE

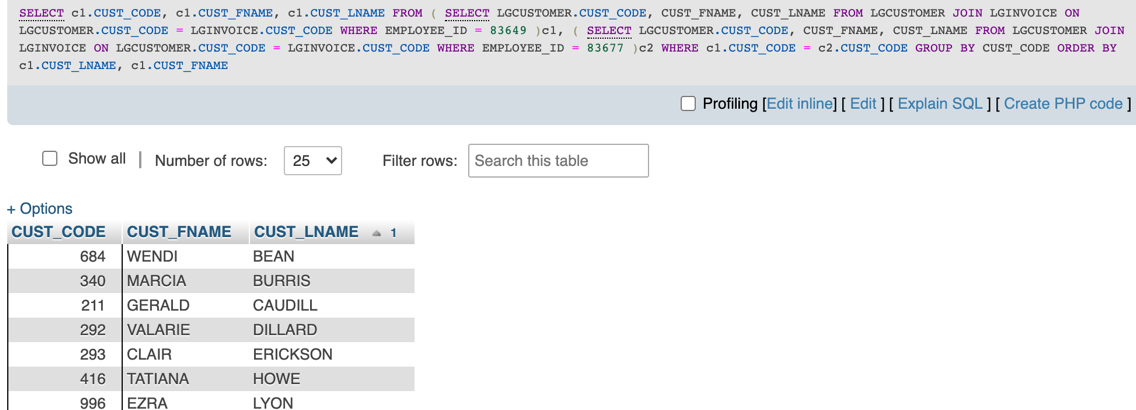
WHERE EMPLOYEE\_ID = 83677

)c2

WHERE c1.CUST\_CODE = c2.CUST\_CODE

GROUP BY CUST\_CODE

ORDER BY c1.CUST\_LNAME, c1.CUST\_FNAME;



1. LargeCo is planning a new promotion in Alabama (AL) and wants to know about the largest purchases made by customers in that state. Write a query to display the customer code, customer first name, last name, full address, invoice date, and invoice total of the largest purchase made by each customer in Alabama. Be certain to include any customers in Alabama who have never made a purchase; their invoice dates should be NULL and the invoice totals should display as 0. Sort the results by customer last name and then first name (Figure P7.53).

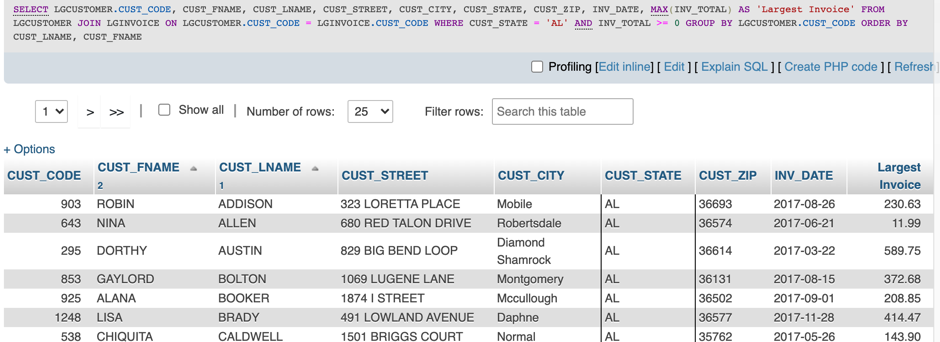
(Our LargeCo database did not contain any Inv\_Total = 0 instances)

SELECT LGCUSTOMER.CUST\_CODE, CUST\_FNAME, CUST\_LNAME, CUST\_STREET, CUST\_CITY, CUST\_STATE, CUST\_ZIP, INV\_DATE, MAX(INV\_TOTAL) AS 'Largest Invoice'

FROM LGCUSTOMER JOIN LGINVOICE ON LGCUSTOMER.CUST\_CODE = LGINVOICE.CUST\_CODE

WHERE CUST\_STATE = 'AL' AND INV\_TOTAL >= 0

GROUP BY LGCUSTOMER.CUST\_CODE

ORDER BY CUST\_LNAME, CUST\_FNAME;

1. One of the purchasing managers is interested in the impact of product prices on the sale of products of each brand. Write a query to display the brand name, brand type, average price of products of each brand, and total units sold of products of each brand. Even if a product has been sold more than once, its price should only be included once in the calculation of the average price. However, you must be careful because multiple products of the same brand can have the same price, and each of those products must be included in the calculation of the brand’s average price. Sort the result by brand name (Figure P7.54).

SELECT BRAND\_NAME, BRAND\_TYPE, c1.Average\_Price AS 'Average Price', c2.Units\_Sold AS 'Units Sold'

FROM(

SELECT BRAND\_NAME, BRAND\_TYPE, LGPRODUCT.BRAND\_ID, ROUND(SUM(PROD\_PRICE)/COUNT(DISTINCT(LGPRODUCT.PROD\_SKU)),2) AS Average\_Price

FROM LGPRODUCT JOIN LGBRAND ON LGPRODUCT.BRAND\_ID = LGBRAND.BRAND\_ID

GROUP BY LGBRAND.BRAND\_ID

ORDER BY BRAND\_NAME

)c1,

(

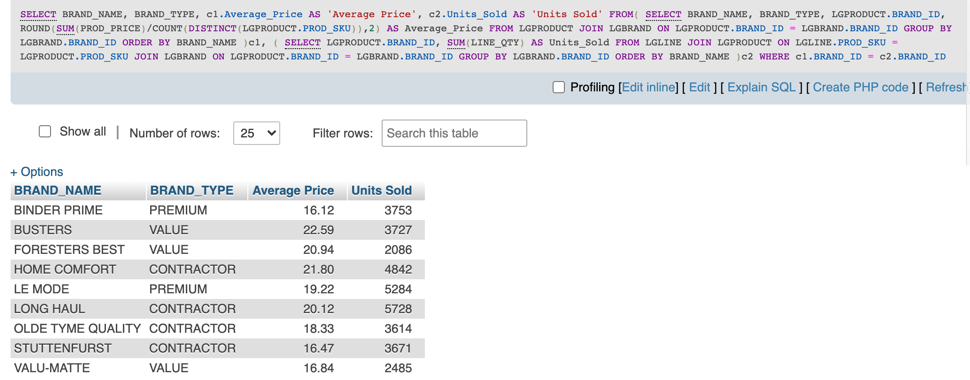
SELECT LGPRODUCT.BRAND\_ID, SUM(LINE\_QTY) AS Units\_Sold

FROM LGLINE JOIN LGPRODUCT ON LGLINE.PROD\_SKU = LGPRODUCT.PROD\_SKU JOIN LGBRAND ON LGPRODUCT.BRAND\_ID = LGBRAND.BRAND\_ID

GROUP BY LGBRAND.BRAND\_ID

ORDER BY BRAND\_NAME

)c2

WHERE c1.BRAND\_ID = c2.BRAND\_ID;

1. The purchasing manager is still concerned about the impact of price on sales. Write a query to display the brand name, brand type, product SKU, product description, and price of any products that are not a premium brand, but that cost more than the most expensive premium brand products (Figure P7.55).

SELECT c1.BRAND\_NAME, c1.BRAND\_TYPE, c1.PROD\_SKU, c1.PROD\_DESCRIPT, PROD\_PRICE

FROM(

SELECT BRAND\_NAME, BRAND\_TYPE, PROD\_SKU, PROD\_DESCRIPT, PROD\_PRICE

FROM LGPRODUCT JOIN LGBRAND ON LGPRODUCT.BRAND\_ID = LGBRAND.BRAND\_ID

WHERE BRAND\_TYPE != 'PREMIUM'

)c1,

(

SELECT BRAND\_NAME, BRAND\_TYPE, PROD\_SKU, PROD\_DESCRIPT, MAX(PROD\_PRICE) AS MAXPRICE

FROM LGPRODUCT JOIN LGBRAND ON LGPRODUCT.BRAND\_ID = LGBRAND.BRAND\_ID

WHERE BRAND\_TYPE = 'PREMIUM'

)c2

WHERE c1.PROD\_PRICE > c2.MAXPRICE

