Prince Angulo
Matthew Zaldana
Hunter Lewis
Brayan Fuentes Lopez

SQL Practice

44. What product that makes us the most money (qty*price) across all orders for that product? Returns 1.

```
SELECT  p.PRODUCTCODE, p.PRODUCTNAME, p.PRODUCTLINE,
        (o.QUANTITYORDERED * o.PRICEEACH) AS MostMoney
FROM    ORDERDETAILS o
INNER JOIN
        PRODUCTS p
ON      o.PRODUCTCODE = p.PRODUCTCODE
WHERE   (o.QUANTITYORDERED * o.PRICEEACH) =
        (SELECT MAX(QUANTITYORDERED * PRICEEACH)
        FROM    ORDERDETAILS o);
```

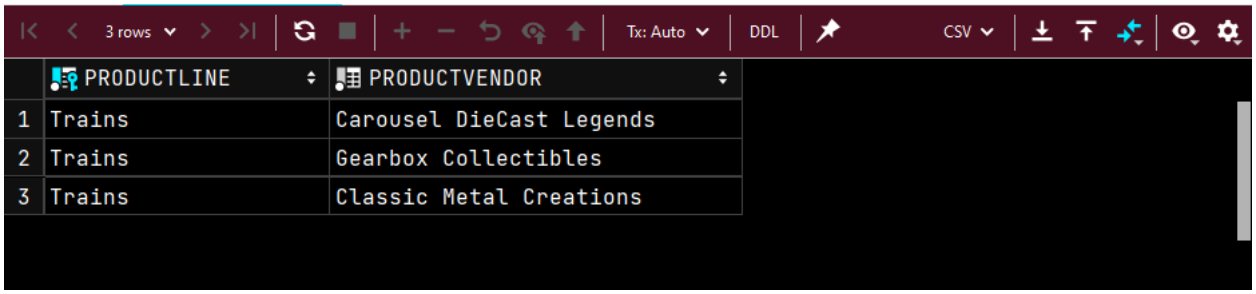
1 row | CSV

	PRODUCTCODE	PRODUCTNAME	PRODUCTLINE	MOSTMONEY
1	S10_4698	2003 Harley-Davidson Eagle Drag Bi…	Motorcycles	11503.14

45.

List the product lines and vendors for product lines which are supported by < 5 vendors. That is, there are < 5 vendors making products within that product line. Returns 3.

```
SELECT PRODUCTLINE, PRODUCTVENDOR FROM PRODUCTS
WHERE PRODUCTLINE =
(SELECT PRODUCTLINE FROM PRODUCTS
GROUP BY PRODUCTLINE
HAVING COUNT(*) < 5);
```



3 rows | Tx: Auto | DDL | CSV

	PRODUCTLINE	PRODUCTVENDOR
1	Trains	Carousel DieCast Legends
2	Trains	Gearbox Collectibles
3	Trains	Classic Metal Creations

46.List the products in the product line with the most number of products. Returns 38.

```
SELECT PRODUCTNAME, PRODUCTLINE
FROM PRODUCTS
WHERE PRODUCTLINE =
(
    SELECT PRODUCTLINE AS PL
    FROM PRODUCTS
    GROUP BY PRODUCTLINE
    HAVING COUNT(PRODUCTNAME) =
    (
        SELECT MAX(PL) AS MP
        FROM
        (
            SELECT PRODUCTLINE, COUNT(PRODUCTNAME) AS NOP
            FROM PRODUCTS
            GROUP BY PRODUCTLINE
        ) AS MNOP
    )
);
```

Output | APP.PRODUCTS

38 rows | Tx: Auto | DDL | CSV

	PRODUCTNAME	PRODUCTLINE
1	1952 Alpine Renault 1300	Classic Cars
2	1972 Alfa Romeo GTA	Classic Cars
3	1962 LanciaA Delta 16V	Classic Cars
4	1968 Ford Mustang	Classic Cars
5	2001 Ferrari Enzo	Classic Cars
6	1969 Corvair Monza	Classic Cars
7	1968 Dodge Charger	Classic Cars
8	1969 Ford Falcon	Classic Cars
9	1970 Plymouth Hemi Cuda	Classic Cars
10	1969 Dodge Charger	Classic Cars
11	1993 Mazda RX-7	Classic Cars
12	1965 Aston Martin DB5	Classic Cars
13	1948 Porsche 356-A Roadster	Classic Cars
14	1995 Honda Civic	Classic Cars
15	1998 Chrysler Plymouth Prowler	Classic Cars
16	1999 Indy 500 Monte Carlo SS	Classic Cars
17	1992 Ferrari 360 Spider red	Classic Cars
18	1985 Toyota Supra	Classic Cars
19	1969 Dodge Super Bee	Classic Cars
20	1976 Ford Gran Torino	Classic Cars
21	1948 Porsche Type 356 Roadster	Classic Cars
22	1970 Triumph Spitfire	Classic Cars
23	1957 Corvette Convertible	Classic Cars
24	1957 Ford Thunderbird	Classic Cars
25	1970 Chevy Chevelle SS 454	Classic Cars
26	1970 Dodge Coronet	Classic Cars
27	1966 Shelby Cobra 427 S/C	Classic Cars

47. Find the first name and last name of all customer contacts whose customer is located in the same state as the San Francisco office. Returns 11.

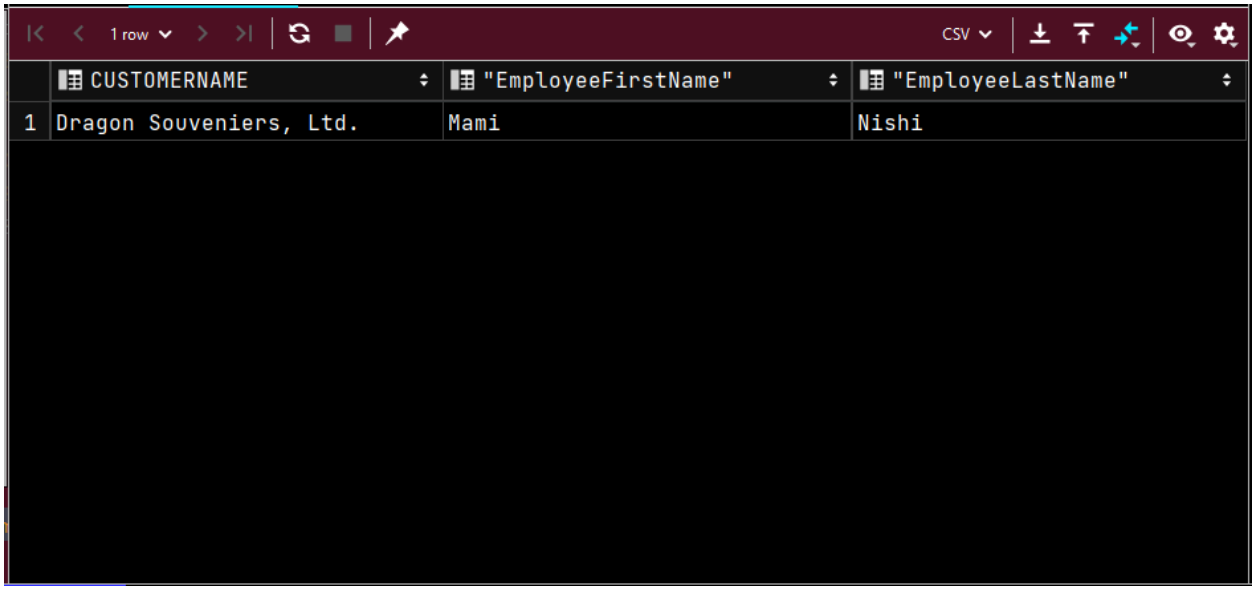
```
SELECT  CONTACTFIRSTNAME, CONTACTLASTNAME
FROM    CUSTOMERS
WHERE STATE = (SELECT STATE
               FROM OFFICES
               WHERE CITY = 'San Francisco');
```

11 rows | Tx: Auto | DDL | CSV

	CONTACTFIRSTNAME	CONTACTLASTNAME
1	Susan	Nelson
2	Julie	Murphy
3	Juri	Hashimoto
4	Julie	Young
5	Mary	Young
6	Valarie	Thompson
7	Julie	Brown
8	Brian	Chandler
9	Sue	Frick
10	Steve	Thompson
11	Sue	Taylor

48. What is the customer and salesperson of the highest priced order? The price of the order is the sum of the quantity ordered * the price each for all the items within that order. Returns 1.

```
SELECT c.CUSTOMERNAME, e.FIRSTNAME AS "EmployeeFirstName", e.LASTNAME AS
"EmployeeLastName"
FROM CUSTOMERS c INNER JOIN EMPLOYEES e ON c.SALESREPEMPLOYEENUMBER =
e.EMPLOYEENUMBER
INNER JOIN ORDERS o ON c.CUSTOMERNUMBER = o.CUSTOMERNUMBER
INNER JOIN ORDERDETAILS od ON o.ORDERNUMBER = od.ORDERNUMBER
GROUP BY od.ORDERNUMBER, c.CUSTOMERNAME, e.FIRSTNAME, e.LASTNAME
HAVING SUM(QUANTITYORDERED * PRICEEACH) =
   (SELECT MAX(TOTALPRICE) FROM
   (SELECT SUM(QUANTITYORDERED * PRICEEACH) AS TOTALPRICE FROM ORDERDETAILS
   GROUP BY ORDERNUMBER) piceTable);
```

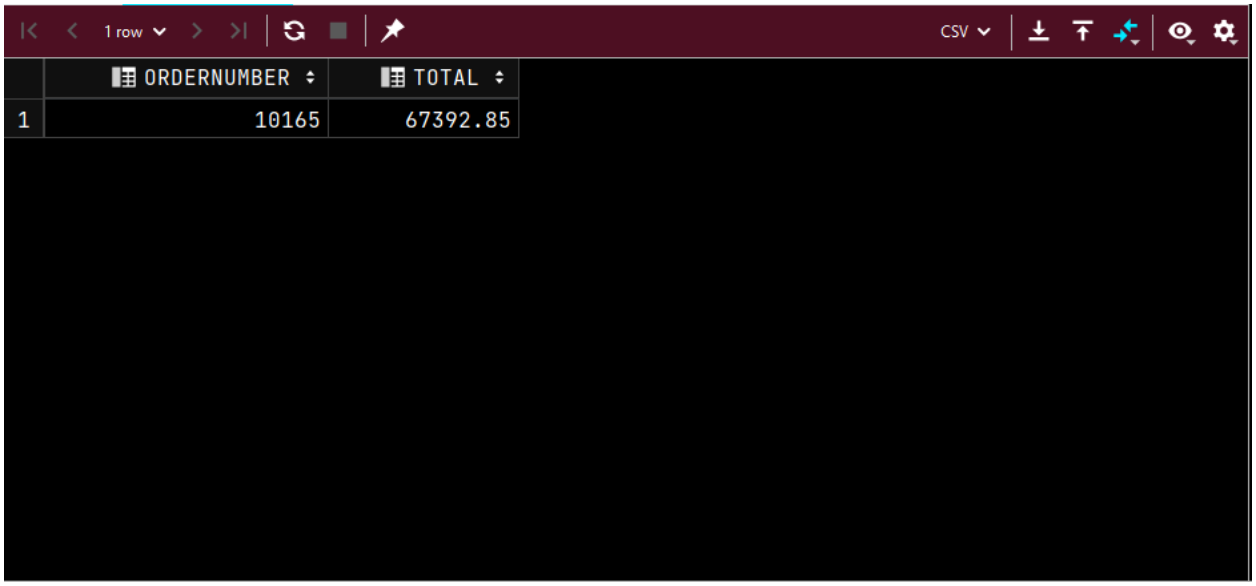


	CUSTOMERNAME	"EmployeeFirstName"	"EmployeeLastName"
1	Dragon Souveniers, Ltd.	Mami	Nishi

49.

What is the order number and the cost of the order for the most expensive orders? Note that there could be more than one order which all happen to add up to the same cost, and that same cost could be the highest cost among all orders. The cost of an order is the sum of the quantity ordered * the price each for all the items within that order. Returns 1.

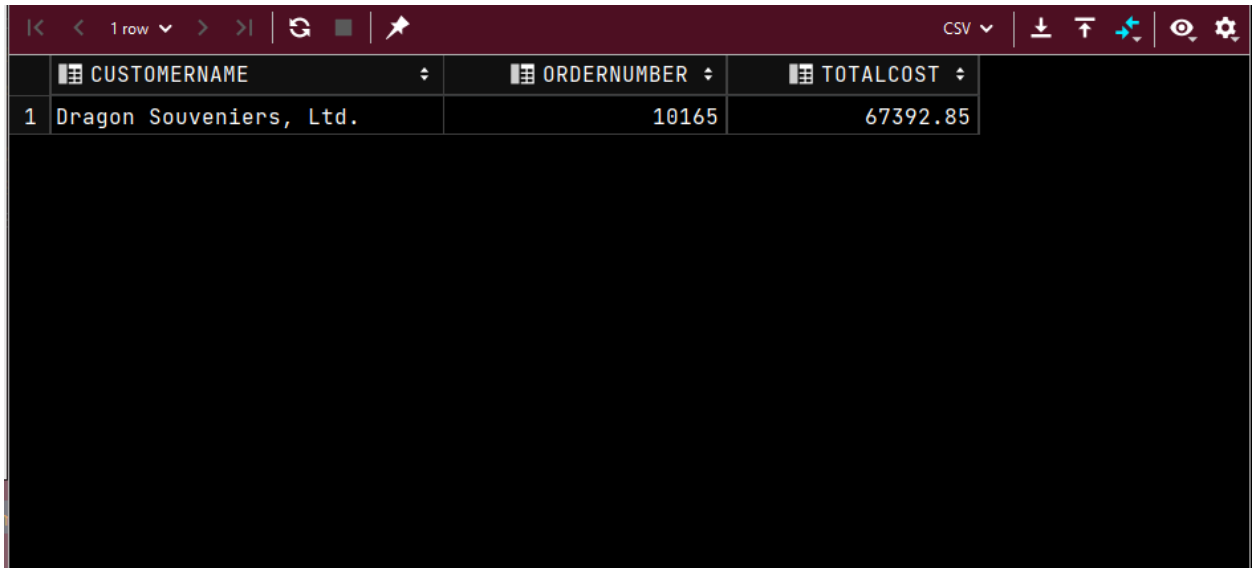
```
SELECT ORDERNUMBER, SUM(priceEach*quantityOrdered) AS TOTAL
FROM ORDERDETAILS
GROUP BY ORDERNUMBER
HAVING SUM(priceEach*quantityOrdered) =
(
    SELECT MAX(ORDERTOTALS.orderTotal)
    FROM
    (
        SELECT SUM(priceEach*QUANTITYORDERED) AS ORDERTOTAL
        FROM ORDERDETAILS
        GROUP BY ORDERNUMBER
    ) AS ORDERTOTALS
);
```



	ORDERNUMBER	TOTAL
1	10165	67392.85

50.What is the name of the customer, the order number, and the total cost of the most expensive orders? Returns 1.

```
SELECT c.CUSTOMERNAME, od.ORDERNUMBER, SUM(QUANTITYORDERED * PRICEEACH) AS
TotalCost
FROM CUSTOMERS c INNER JOIN ORDERS o ON c.CUSTOMERNUMBER = o.CUSTOMERNUMBER
INNER JOIN ORDERDETAILS od ON o.ORDERNUMBER = od.ORDERNUMBER
GROUP BY od.ORDERNUMBER, c.CUSTOMERNAME
HAVING SUM(QUANTITYORDERED * PRICEEACH) =
   (SELECT MAX(TotalPrice) FROM
   (SELECT SUM(QUANTITYORDERED * PRICEEACH) AS TotalPrice FROM ORDERDETAILS
   GROUP BY orderNumber) piceTable);
```

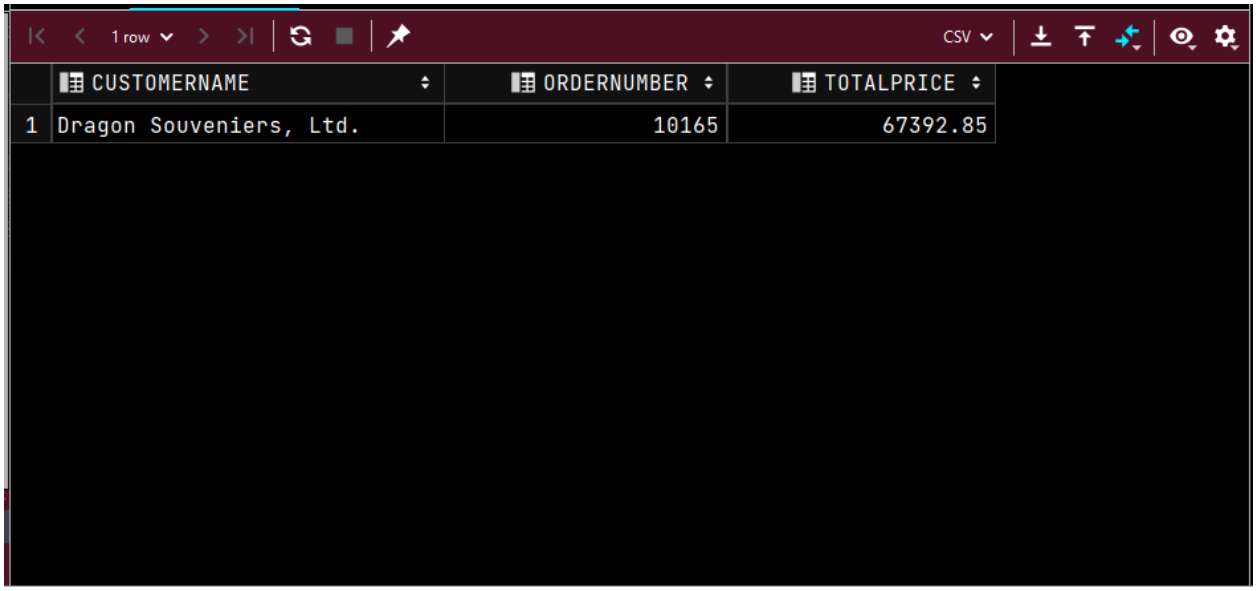


	CUSTOMERNAME	ORDERNUMBER	TOTALCOST
1	Dragon Souveniers, Ltd.	10165	67392.85

51.Take some portion of the above query and put that into a view. Then rewrite the above query to use the view that you just created and consider how incorporating the view made the query easier to understand. If you do not know how many rows this returns, please come see me immediately.

```
CREATE VIEW TotalCosts AS
SELECT SUM(QUANTITYORDERED * PRICEEACH) AS TotalPrice, orderNumber FROM
ORDERDETAILS
GROUP BY orderNumber;

SELECT c.CUSTOMERNAME, o.ORDERNUMBER, tc.TOTALPRICE
FROM CUSTOMERS c INNER JOIN Orders o ON c.CUSTOMERNUMBER = o.CUSTOMERNUMBER
INNER JOIN TotalCosts tc ON o.ORDERNUMBER = tc.ORDERNUMBER
GROUP BY o.ORDERNUMBER, c.CUSTOMERNAME, tc.TOTALPRICE
HAVING tc.TOTALPRICE =
    (SELECT MAX(TotalPrice) FROM TotalCosts);
```

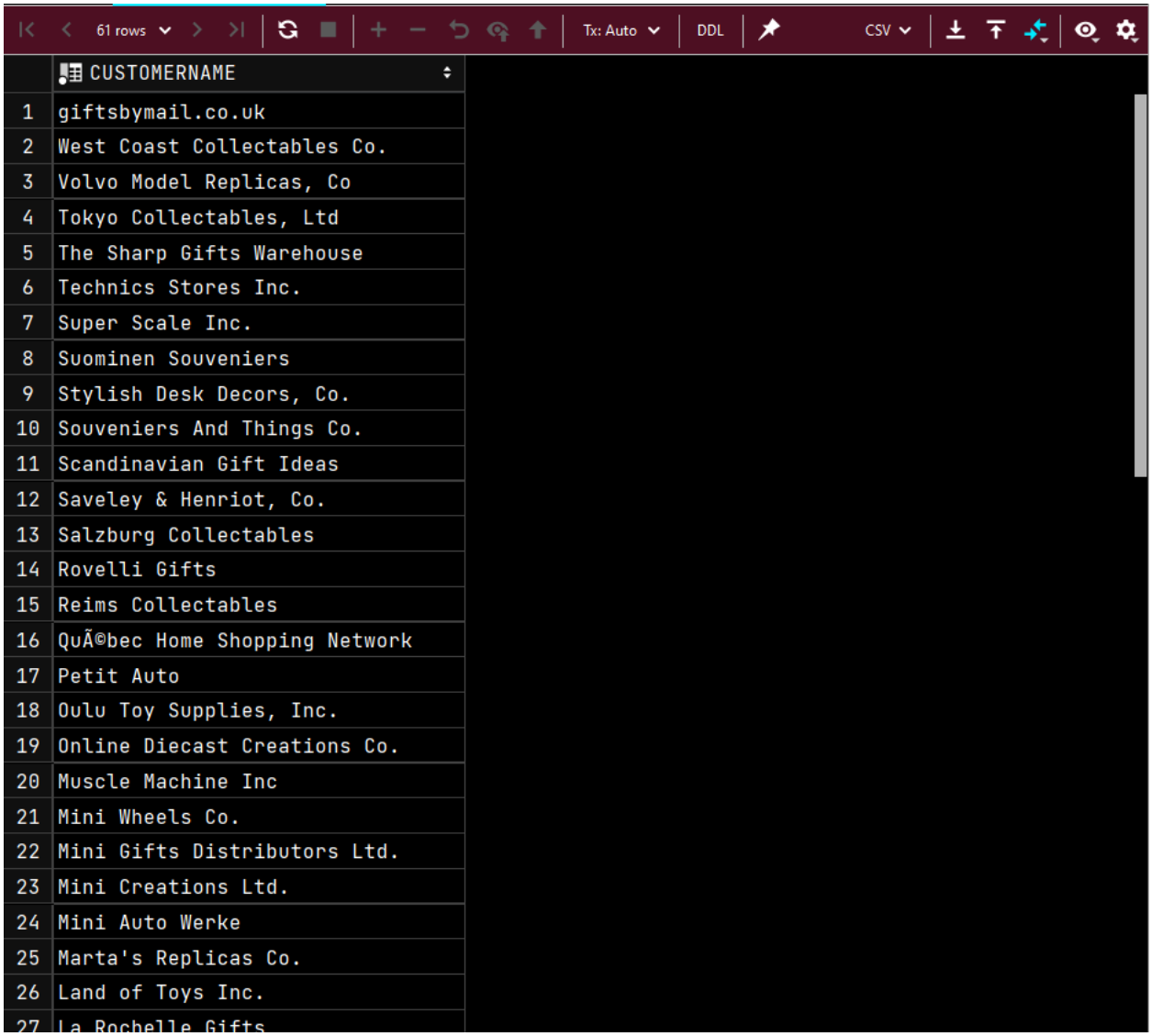


1 row CSV

	CUSTOMERNAME	ORDERNUMBER	TOTALPRICE
1	Dragon Souveniers, Ltd.	10165	67392.85

52.Show all of the customers who have ordered at least one product with the name “Ford” in it, that “Dragon Souveniers, Ltd.” has also ordered. List them in reverse alphabetical order, and do not consider the case of the letters in the customer name in the ordering. Show each customer no more than once. Returns 61.

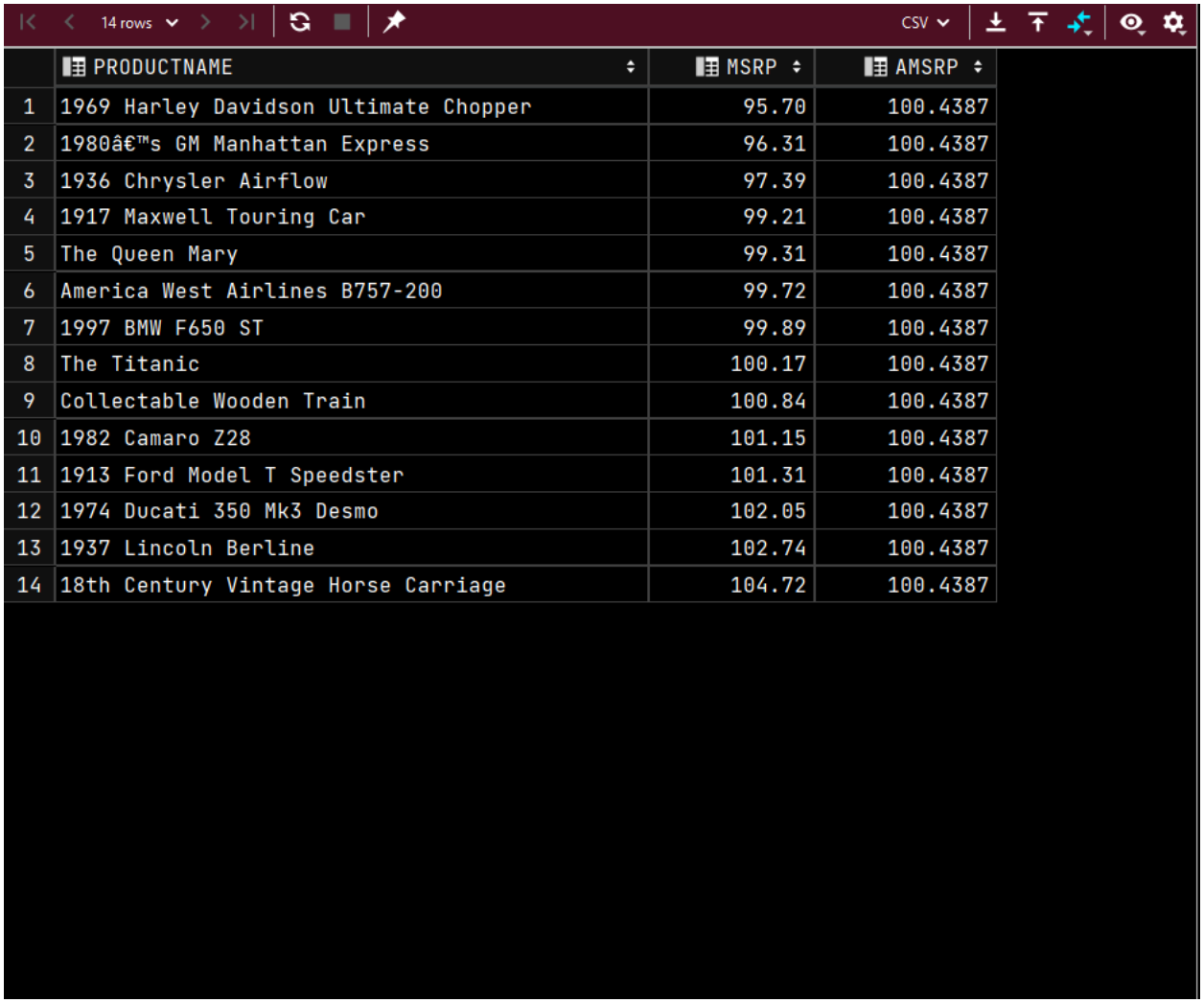
```
SELECT customerName FROM CUSTOMERS
NATURAL JOIN ORDERS NATURAL JOIN ORDERDETAILS NATURAL JOIN PRODUCTS
WHERE productName IN
(SELECT productName FROM CUSTOMERS
NATURAL JOIN ORDERS NATURAL JOIN ORDERDETAILS NATURAL JOIN PRODUCTS
WHERE customerName = 'Dragon Souveniers, Ltd.'
AND productname LIKE '%Ford%')
GROUP BY customerName
HAVING COUNT(productName) >= 1
ORDER BY customerName DESC;
```



	CUSTOMERNAME
1	giftsbymail.co.uk
2	West Coast Collectables Co.
3	Volvo Model Replicas, Co
4	Tokyo Collectables, Ltd
5	The Sharp Gifts Warehouse
6	Technics Stores Inc.
7	Super Scale Inc.
8	Suominen Souveniers
9	Stylish Desk Decors, Co.
10	Souveniers And Things Co.
11	Scandinavian Gift Ideas
12	Saveley & Henriot, Co.
13	Salzburg Collectables
14	Rovelli Gifts
15	Reims Collectables
16	QuÃ©bec Home Shopping Network
17	Petit Auto
18	Oulu Toy Supplies, Inc.
19	Online Diecast Creations Co.
20	Muscle Machine Inc
21	Mini Wheels Co.
22	Mini Gifts Distributors Ltd.
23	Mini Creations Ltd.
24	Mini Auto Werke
25	Marta's Replicas Co.
26	Land of Toys Inc.
27	La Rochelle Gifts

53.Which products have an MSRP within 5% of the average MSRP across all products? List the Product Name, the MSRP, and the average MSRP ordered by the product MSRP. If we denote the average MSRP as aMSRP, then the % difference between a particular MSRP and aMSRP is 100 * (MSRP - aMSRP)/aMSRP. Returns 14.

```
SELECT  productName, MSRP,
        (SELECT AVG(MSRP)
        FROM PRODUCTS) as aMSRP
FROM PRODUCTS
WHERE (100 * ((MSRP - (SELECT AVG(MSRP)
        FROM PRODUCTS)) / (SELECT AVG(MSRP)
        FROM PRODUCTS))) > -5
AND
(100 * ((MSRP - (SELECT AVG(MSRP)
        FROM PRODUCTS)) /(SELECT AVG(MSRP)
        FROM PRODUCTS))) < 5
ORDER BY MSRP;
```



	PRODUCTNAME	MSRP	AMSRP
1	1969 Harley Davidson Ultimate Chopper	95.70	100.4387
2	1980â€™s GM Manhattan Express	96.31	100.4387
3	1936 Chrysler Airflow	97.39	100.4387
4	1917 Maxwell Touring Car	99.21	100.4387
5	The Queen Mary	99.31	100.4387
6	America West Airlines B757-200	99.72	100.4387
7	1997 BMW F650 ST	99.89	100.4387
8	The Titanic	100.17	100.4387
9	Collectable Wooden Train	100.84	100.4387
10	1982 Camaro Z28	101.15	100.4387
11	1913 Ford Model T Speedster	101.31	100.4387
12	1974 Ducati 350 Mk3 Desmo	102.05	100.4387
13	1937 Lincoln Berline	102.74	100.4387
14	18th Century Vintage Horse Carriage	104.72	100.4387