



Jenson USA: America's Premier Online Cycling Store

Discover the legacy of Jenson USA, a pioneer in online cycling retail, dedicated to providing exceptional service and an expansive selection of products to the cycling community across America.

1. Find the total number of products sold by each store along with the store name

The screenshot shows a MySQL Workbench interface with a query editor and a results grid.

Query Editor:

```
1 SELECT
2     s.store_name,
3     SUM(oi.quantity) AS total_products_sold
4 FROM orders o
5 JOIN order_items oi ON o.order_id = oi.order_id
6 JOIN stores s ON o.store_id = s.store_id
7 GROUP BY s.store_name
8 ORDER BY total_products_sold DESC;
```

Results Grid:

store_name	total_products_s...
Baldwin Bikes	4779
Santa Cruz Bikes	1516
Rowlett Bikes	783

Action Output:

	Time	Action	Response	Duration / Fetch Time
1	10:40:32	SELECT s.store_name, SUM(oi.quantity) AS total_pr...	3 row(s) returned	0.014 sec / 0.000012...

2. Calculate the cumulative sum of quantities sold for each product over time

The screenshot shows a MySQL Workbench interface. The SQL editor window contains the following query:

```
1 • SELECT
2     p.product_name,
3     o.order_date,
4     SUM(oi.quantity) OVER (PARTITION BY p.product_id ORDER BY o.order_date) AS cumulative_quantity
5   FROM order_items oi
6   JOIN orders o ON oi.order_id = o.order_id
7   JOIN products p ON oi.product_id = p.product_id
8   ORDER BY p.product_name, o.order_date;
9
```

The results grid displays the following data:

product_name	order_date	cumulative_quantity
Electra Amsterdam Fashion 3i Ladies' - 2017/20...	2018-01-01	1
Electra Amsterdam Fashion 3i Ladies' - 2017/20...	2018-01-21	3
Electra Amsterdam Fashion 3i Ladies' - 2017/20...	2018-04-30	5
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-01-29	2
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-02-28	3
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-03-03	4
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-03-09	6
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-04-06	7
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-04-15	9
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-04-16	10
Electra Amsterdam Fashion 7i Ladies' - 2017	2017-06-27	11

The status bar at the bottom shows the query was executed at 10:42:23 and returned 4722 rows in 0.043 seconds.

3. Find the product with the highest total sales (quantity * price) for each category

```
1 • - WITH prod_sales AS (
2     SELECT
3         p.product_id,
4         p.product_name,
5         p.category_id,
6         SUM(oi.quantity * oi.list_price) AS total_sales
7     FROM order_items oi
8     JOIN products p ON oi.product_id = p.product_id
9     GROUP BY p.product_id, p.product_name, p.category_id
10    ),
11    ranked AS (
12        SELECT
13            ps.*,
14            ROW_NUMBER() OVER (PARTITION BY ps.category_id ORDER BY ps.total_sales DESC) AS rn
15        FROM prod_sales ps
16    )
17    SELECT
18        c.category_name,
19        r.product_name,
20        r.total_sales
21    FROM ranked r
22    JOIN categories c ON r.category_id = c.category_id
23    WHERE r.rn = 1
24    ORDER BY c.category_name;
25
```

100% 1:25

Result Grid Filter Rows: Search

Export:

category_name product_name

total_sales

Result Grid

4. Find the customer who spent the most money on orders

The screenshot shows a MySQL Workbench interface with the following details:

- Toolbar:** Includes icons for file operations (New, Save, Print, Paste), search, and various preferences.
- Query Editor:** Displays the following SQL query:

```
1 •  SELECT
2      CONCAT(c.first_name, ' ', c.last_name) AS customer_name,
3      ROUND(SUM(oi.quantity * oi.list_price * (1 - oi.discount)), 2) AS total_spent
4  FROM customers c
5  JOIN orders o ON c.customer_id = o.customer_id
6  JOIN order_items oi ON o.order_id = oi.order_id
7  GROUP BY c.customer_id
8  ORDER BY total_spent DESC
9  LIMIT 1;
10
```
- Status Bar:** Shows "100%" zoom and "1:10" ratio.
- Result Grid:** Shows the results of the query:

customer_name	total_spent
Kellie Franco	0.00
- Right Panel:** A sidebar titled "Result Grid" with a grid icon.

5. Find the highest-priced product for each category name

The screenshot shows a MySQL Workbench interface. The SQL editor window contains the following query:

```
1 SELECT
2     c.category_name,
3     p.product_name,
4     p.list_price
5 FROM products p
6 JOIN categories c ON p.category_id = c.category_id
7 WHERE p.list_price = (
8     SELECT MAX(p2.list_price)
9     FROM products p2
10    WHERE p2.category_id = p.category_id
11 );
```

The result grid displays the following data:

category_name	product_name	list_price
Children Bicycles	Electra Straight 8 3i (20-inch) - Boy's - 2017	48999.00
Children Bicycles	Electra Townie 3i EQ (20-inch) - Boys' - 2017	48999.00
Children Bicycles	Trek Superfly 24 - 2017/2018	48999.00
Comfort Bicycles	Electra Townie Go! 8i - 2017/2018	259999.00
Cruisers Bicycles	Electra Townie Commute Go! - 2018	299999.00
Cruisers Bicycles	Electra Townie Commute Go! Ladies' - 2018	299999.00
Cyclocross Bicycles	Trek Boone 7 Disc - 2018	399999.00
Electric Bikes	Trek Powerfly 8 FS Plus - 2017	499999.00
Electric Bikes	Trek Powerfly 7 FS - 2018	499999.00
Electric Bikes	Trek Super Commuter+ 8S - 2018	499999.00
Mountain Bikes	Trek Fuel EX 98 275 Plus - 2017	529999.00
Mountain Bikes	Trek Remedy 98 - 2017	529999.00
Road Bikes	Trek Domane SLR 9 Disc - 2018	1199999....

The sidebar on the right shows icons for Result Grid, Form Editor, and Field Types.

6. Find the total number of orders placed by each customer per store

The screenshot shows a MySQL Workbench interface. The query editor window contains the following SQL code:

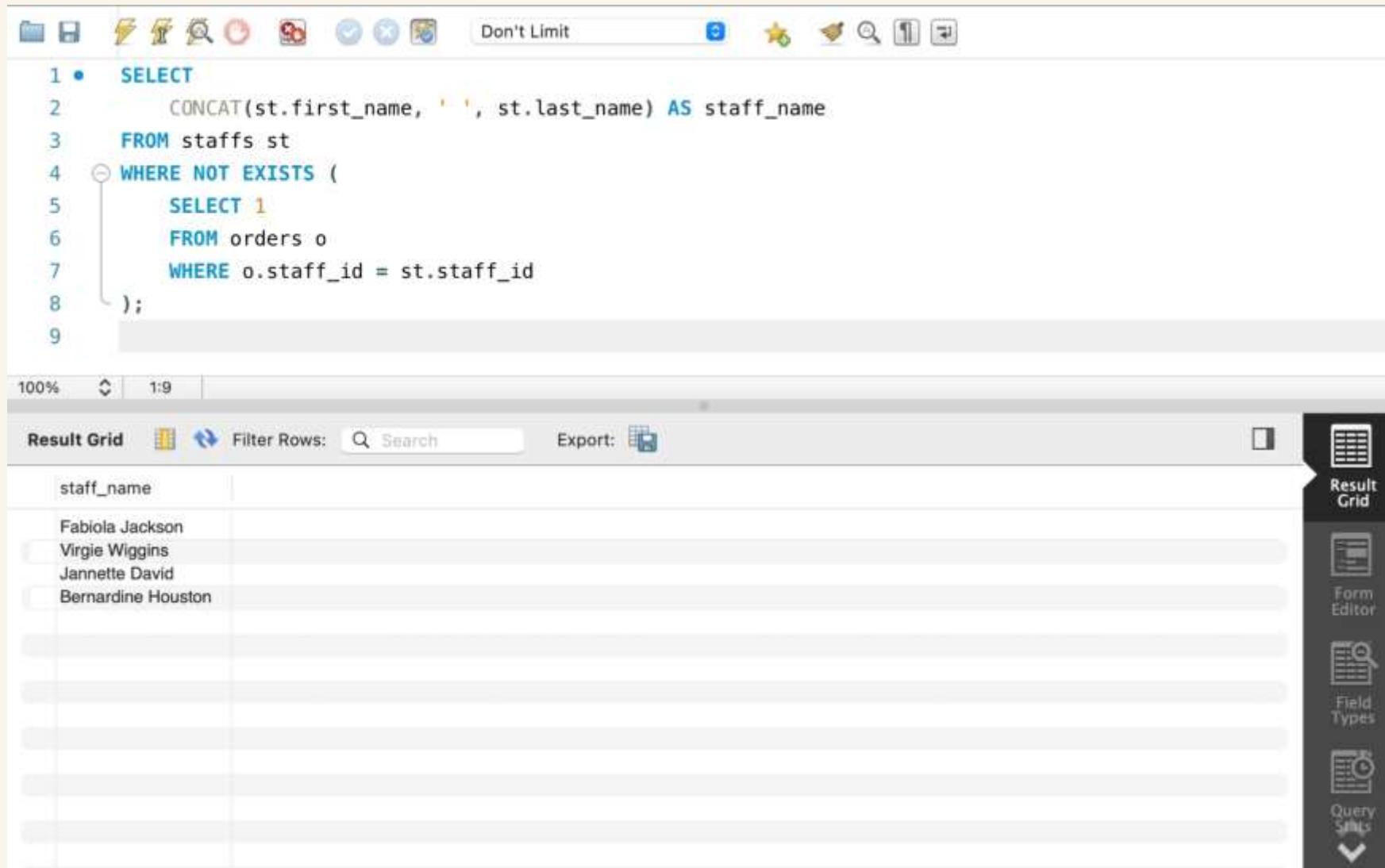
```
1 •  SELECT
2      s.store_name,
3      CONCAT(c.first_name, ' ', c.last_name) AS customer_name,
4      COUNT(o.order_id) AS total_orders
5  FROM orders o
6  JOIN customers c ON o.customer_id = c.customer_id
7  JOIN stores s ON o.store_id = s.store_id
8  GROUP BY s.store_name, c.customer_id
9  ORDER BY s.store_name, total_orders DESC;
10
```

The result grid displays the following data:

store_name	customer_name	total_orders
Baldwin Bikes	Genoveva Baldwin	3
Baldwin Bikes	Lyndsey Bean	3
Baldwin Bikes	Latasha Hays	3
Baldwin Bikes	Robby Sykes	3
Baldwin Bikes	Caren Stephens	3
Baldwin Bikes	Linnie Branch	3
Baldwin Bikes	Pamelia Newman	3
Baldwin Bikes	Emmitt Sanchez	3
Baldwin Bikes	Jacqueline Duncan	3
Baldwin Bikes	Debra Burks	3
Baldwin Bikes	Daryl Spence	3
Baldwin Bikes	Theo Reese	2
Baldwin Bikes	Tenisha Lyons	2
Baldwin Bikes	Santos Valencia	2
Baldwin Bikes	Damien Dorsey	2

The right sidebar of the interface includes icons for Result Grid, Form Editor, Field Types, and Query Stats.

7.Find the names of staff members who have not made any sales



The screenshot shows a MySQL Workbench interface. The query editor window contains the following SQL code:

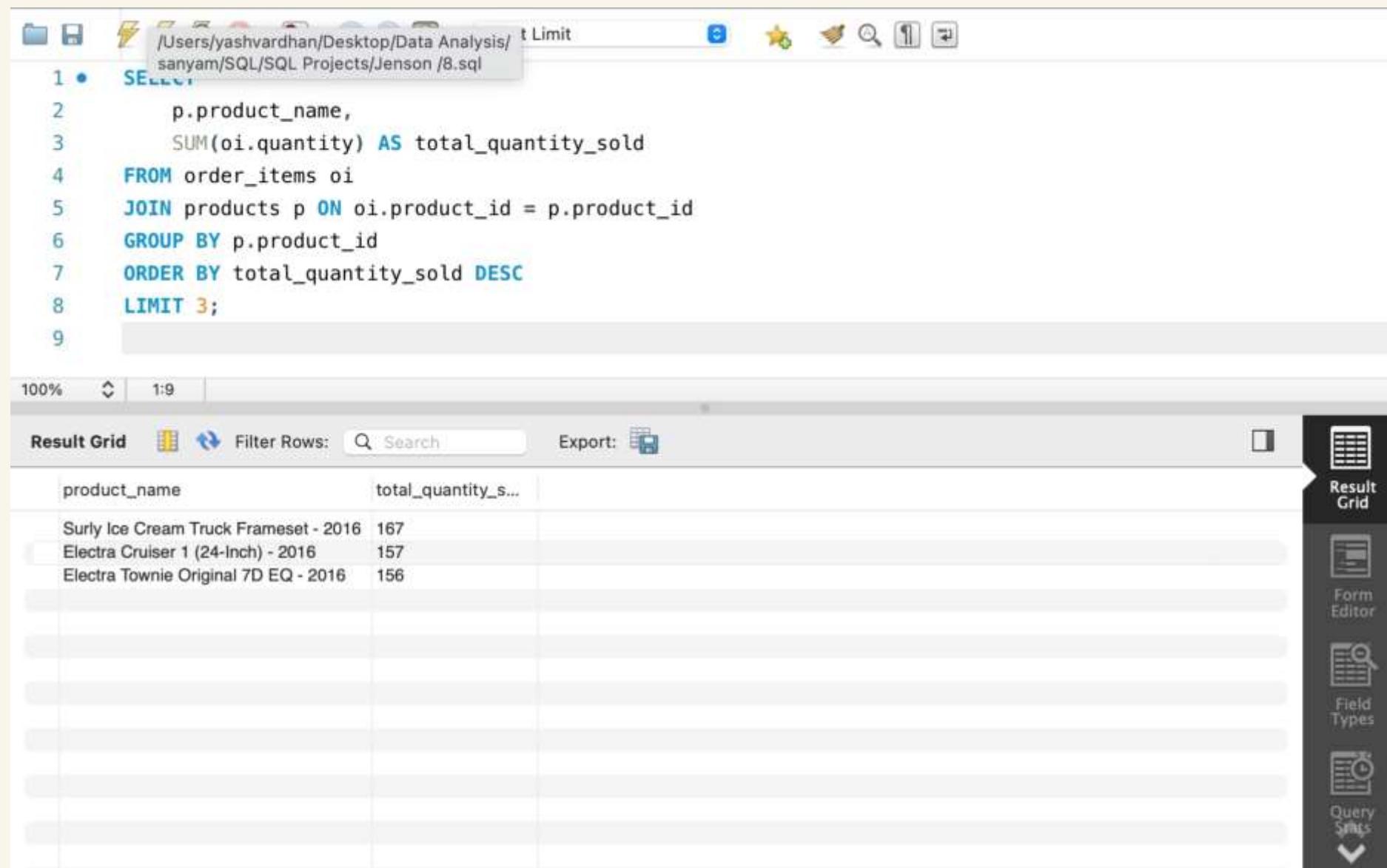
```
1 •  SELECT
2      CONCAT(st.first_name, ' ', st.last_name) AS staff_name
3  FROM staffs st
4 WHERE NOT EXISTS (
5     SELECT 1
6     FROM orders o
7     WHERE o.staff_id = st.staff_id
8 );
9
```

The result grid displays the following data:

staff_name
Fabiola Jackson
Virgie Wiggins
Jannette David
Bernardine Houston

The right sidebar shows the following tabs: Result Grid, Form Editor, Field Types, and Query Skills.

8.Find the top 3 most sold products in terms of quantity



The screenshot shows the MySQL Workbench interface. The query editor window displays the following SQL code:

```
1 •  SELECT
2      p.product_name,
3          SUM(oi.quantity) AS total_quantity_sold
4  FROM order_items oi
5  JOIN products p ON oi.product_id = p.product_id
6  GROUP BY p.product_id
7  ORDER BY total_quantity_sold DESC
8  LIMIT 3;
9
```

The result grid shows the following data:

product_name	total_quantity_s...
Surly Ice Cream Truck Frameset - 2016	167
Electra Cruiser 1 (24-Inch) - 2016	157
Electra Townie Original 7D EQ - 2016	156

The right sidebar contains icons for Result Grid, Form Editor, Field Types, and Query Schemas.

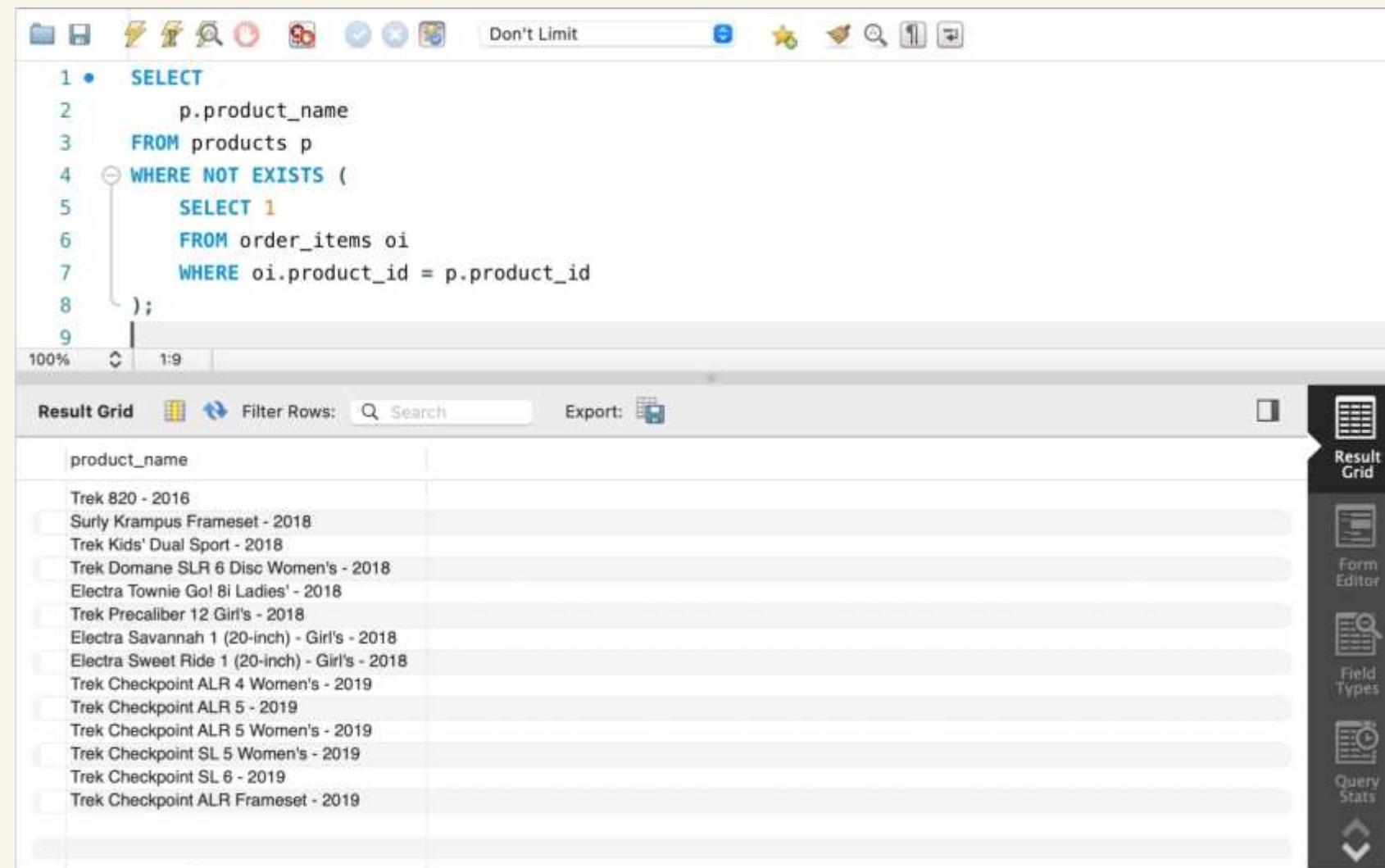
9. Find the median value of the price list

The screenshot shows a database query editor interface with the following details:

- Toolbar:** Includes icons for file operations, search, and various database functions.
- Text Editor:** Displays the SQL code for calculating the median price. The code uses a common table expression (CTE) named `temp` to rank items by price and then determine the median based on the count of items (`cn`).

```
1 • WITH temp AS (
2     SELECT
3         list_price,
4         ROW_NUMBER() OVER (ORDER BY list_price) AS rn,
5         COUNT(*) OVER () AS cn
6     FROM order_items
7 )
8 SELECT CASE
9     WHEN cn % 2 = 0 THEN (
10         SELECT AVG(list_price)
11         FROM temp
12         WHERE rn IN (cn/2, cn/2 + 1)
13     )
14     ELSE (
15         SELECT list_price
16         FROM temp
17         WHERE rn = (cn + 1)/2
18     )
19     END AS median
20 FROM temp
21 LIMIT 1;
```
- Status Bar:** Shows the current time as 21:19.
- Result Grid:** A table showing the result of the query. The single row contains the column `median` with the value `59999...`.
- Buttons:** Includes "Result Grid" and "Export" buttons.

10. List all products that have never been ordered (use EXISTS)



The screenshot shows a MySQL Workbench interface with a query editor and a results grid.

Query Editor:

```
1 • SELECT
2     p.product_name
3     FROM products p
4     WHERE NOT EXISTS (
5         SELECT 1
6         FROM order_items oi
7         WHERE oi.product_id = p.product_id
8     );
9
```

Results Grid:

product_name
Trek 820 - 2016
Surly Krampus Frameset - 2018
Trek Kids' Dual Sport - 2018
Trek Domane SLR 6 Disc Women's - 2018
Electra Townie Go! 8i Ladies' - 2018
Trek Precaliber 12 Girl's - 2018
Electra Savannah 1 (20-inch) - Girl's - 2018
Electra Sweet Ride 1 (20-inch) - Girl's - 2018
Trek Checkpoint ALR 4 Women's - 2019
Trek Checkpoint ALR 5 - 2019
Trek Checkpoint ALR 5 Women's - 2019
Trek Checkpoint SL 5 Women's - 2019
Trek Checkpoint SL 6 - 2019
Trek Checkpoint ALR Frameset - 2019

Right Panel:

- Result Grid
- Form Editor
- Field Types
- Query Stats

11.List the names of staff members who have made more sales than the average number of sales by all staff members

```
1 •  SELECT
2      CONCAT(st.first_name, ' ', st.last_name) AS staff_name,
3      COUNT(o.order_id) AS total_sales
4  FROM staffs st
5  JOIN orders o ON st.staff_id = o.staff_id
6  GROUP BY st.staff_id
7  HAVING COUNT(o.order_id) > (
8      SELECT AVG(staff_sales)
9      FROM (
10         SELECT COUNT(o2.order_id) AS staff_sales
11         FROM orders o2
12         GROUP BY o2.staff_id
13     ) AS avg_sales
14 );
15 |
```

100% ▾ 1:15

Result Grid



Filter Rows:

Search

Export:



Result
Grid

staff_name	total_sales
------------	-------------

Marcelene Boyer	553
-----------------	-----

12. Identify the customers who have ordered all types of products (from every category)

The screenshot shows a MySQL Workbench interface. The query editor window contains the following SQL code:

```
1 • SELECT
2     CONCAT(c.first_name, ' ', c.last_name) AS customer_name
3     FROM customers c
4     JOIN orders o ON c.customer_id = o.customer_id
5     JOIN order_items oi ON o.order_id = oi.order_id
6     JOIN products p ON oi.product_id = p.product_id
7     GROUP BY c.customer_id
8     HAVING COUNT(DISTINCT p.category_id) = (SELECT COUNT(*) FROM categories);
9
```

The results grid shows one row with the customer name "Genoveva Baldwin".

customer_name
Genoveva Baldwin

At the bottom right, there are tabs for "Result Grid" and "Form Editor".

Thank You!

Sharma

By Sanyam