```sql
72 •    SELECT *
73      FROM customers
74      WHERE country = 'USA';
75
76
```

customer_id	customer_name	email	country
1	Alice Smith	alice@example.com	USA
3	Carol Lee	carol@example.com	USA
NULL	NULL	NULL	NULL

```sql
76 •    SELECT product_name, price
77      FROM products
78      ORDER BY price DESC
79      LIMIT 5;
80
```

product_name	price
Laptop	999.99
Smartphone	699.99
Headphones	129.99
T-Shirt	19.99
Novel	14.99

```sql
81  SELECT c.category_name, SUM(oi.quantity * oi.unit_price) AS total_sales
82  FROM order_items oi
83  JOIN products p ON oi.product_id = p.product_id
84  JOIN categories c ON p.category_id = c.category_id
85  GROUP BY c.category_name
86  ORDER BY total_sales DESC;
87
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: A

category_name	total_sales
Electronics	1829.97
Clothing	39.98
Books	14.99

```sql
88  SELECT o.order_id, cu.customer_name, p.product_name, oi.quantity
89  FROM orders o
90  JOIN customers cu ON o.customer_id = cu.customer_id
91  JOIN order_items oi ON o.order_id = oi.order_id
92  JOIN products p ON oi.product_id = p.product_id
93  ORDER BY o.order_id;
94
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: A

order_id	customer_name	product_name	quantity
1001	Alice Smith	Smartphone	1
1001	Alice Smith	T-Shirt	2
1002	Bob Johnson	Novel	1
1003	Carol Lee	Laptop	1
1004	Alice Smith	Headphones	1

```sql
94
95 ●   SELECT customer_id, COUNT(*) AS num_orders
96     FROM orders
97     GROUP BY customer_id
98 ⊖  HAVING COUNT(*) > (
99         SELECT AVG(order_count)
100 ⊖       FROM (
101             SELECT COUNT(*) AS order_count
102             FROM orders
103             GROUP BY customer_id
104         ) AS avg_orders
105   );
106
```

| Result Grid | 🔢 | Filter Rows: | | Export: 💾 | Wrap Cell Content: 🔤 |

	customer_id	num_orders
▶	1	2

```
 93        ORDER BY o.order_id;
 94
 95  •    SELECT customer_id, COUNT(*) AS num_orders
 96        FROM orders
 97        GROUP BY customer_id
 98        HAVING COUNT(*) > (
 99            SELECT AVG(order_count)
100            FROM (
101                SELECT COUNT(*) AS order_count
102                FROM orders
103                GROUP BY customer_id
104            ) AS avg_orders
105        );
106
107
108  •    CREATE VIEW customer_order_totals AS
109        SELECT o.customer_id, cu.customer_name, SUM(oi.quantity * oi.unit_price) AS total_value
110        FROM orders o
111        JOIN customers cu ON o.customer_id = cu.customer_id
112        JOIN order_items oi ON o.order_id = oi.order_id
113        GROUP BY o.customer_id, cu.customer_name;
114
115
```

```
114
115  •    SELECT * FROM customer_order_totals ORDER BY total_value DESC;
116
```

customer_id	customer_name	total_value
3	Carol Lee	999.99
1	Alice Smith	869.96
2	Bob Johnson	14.99

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator:

SQL File 4* ×

SCHEMAS

Filter objects

▼ ecomm
  ▼ Tables
    ▶ categories
    ▶ customers
    ▶ order_items
    ▶ orders
    ▶ products
    Views
    Stored Procedures
    Functions
  ▶ sys
  ▶ xyz

Administration Schemas

Information

No object selected

```
 97        GROUP BY customer_id
 98        HAVING COUNT(*) > (
 99            SELECT AVG(order_count)
100            FROM (
101                SELECT COUNT(*) AS order_count
102                FROM orders
103                GROUP BY customer_id
104            ) AS avg_orders
105        );
106
107
108  •    CREATE VIEW customer_order_totals AS
109        SELECT o.customer_id, cu.customer_name, SUM(oi.quantity * oi.unit_price) AS total_value
110        FROM orders o
111        JOIN customers cu ON o.customer_id = cu.customer_id
112        JOIN order_items oi ON o.order_id = oi.order_id
113        GROUP BY o.customer_id, cu.customer_name;
114
115  •    SELECT * FROM customer_order_totals ORDER BY total_value DESC;
116
117  •    CREATE INDEX idx_orders_customer_id ON orders(customer_id);
118  •    CREATE INDEX idx_order_items_product_id ON order_items(product_id);
119
```