

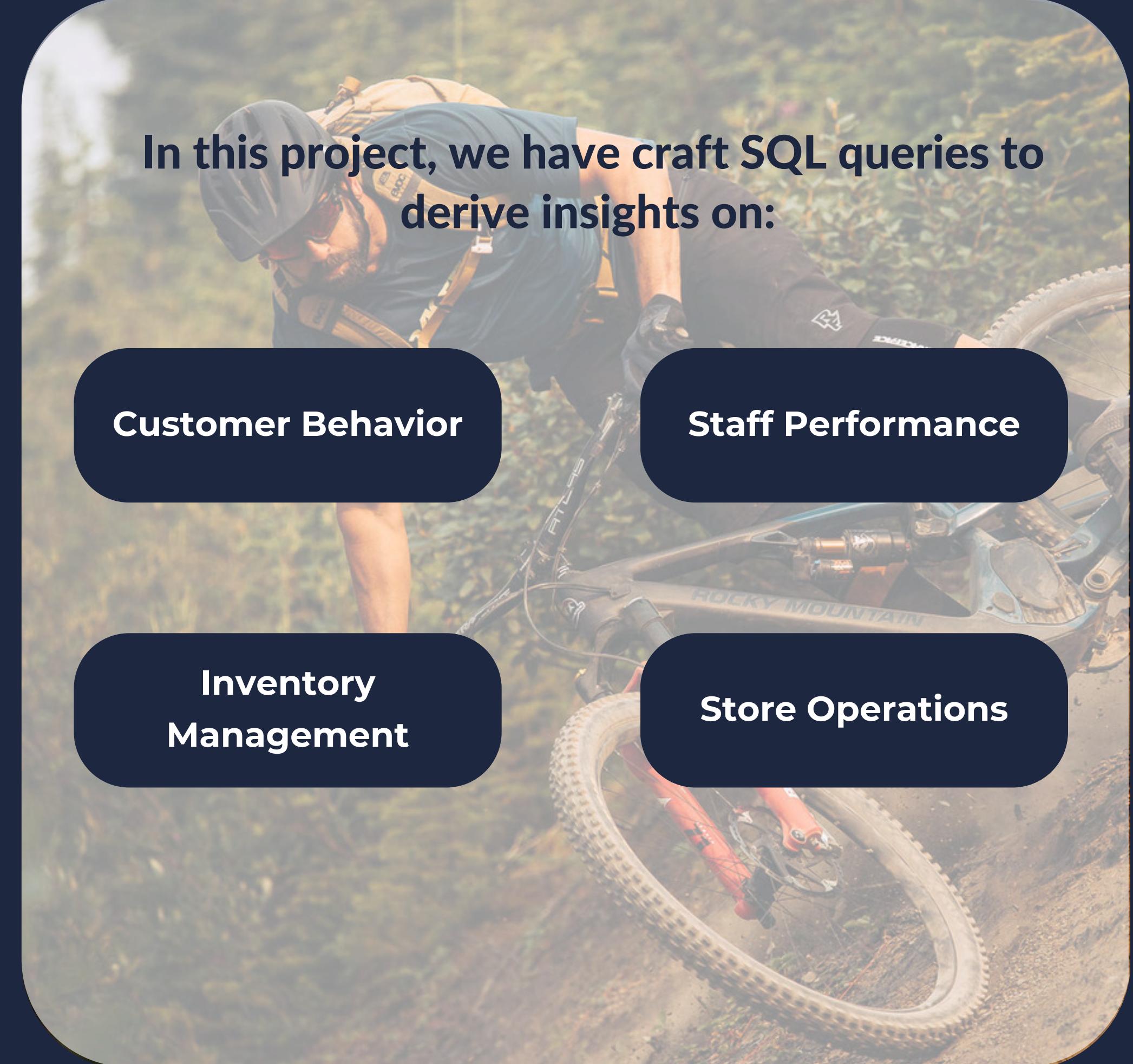


Driven Insights using Advanced
SQL Techniques

Presented by: Manik Gupta



Project: Jenson USA



In this project, we have craft SQL queries to derive insights on:

Customer Behavior

Staff Performance

Inventory
Management

Store Operations



Customer Behavior

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

```
select customers.customer_id,
concat(customers.first_name, " ", customers.last_name) customer_name
from customers join orders
on customers.customer_id = orders.customer_id
join order_items
on orders.order_id = order_items.order_id
join products
on order_items.product_id = products.product_id
group by customers.customer_id, customer_name
having count(distinct products.category_id) = (select count(category_id) from categories);
```

Result

customer_id	customer_name
9	Genoveva Baldwin

Customer Behavior

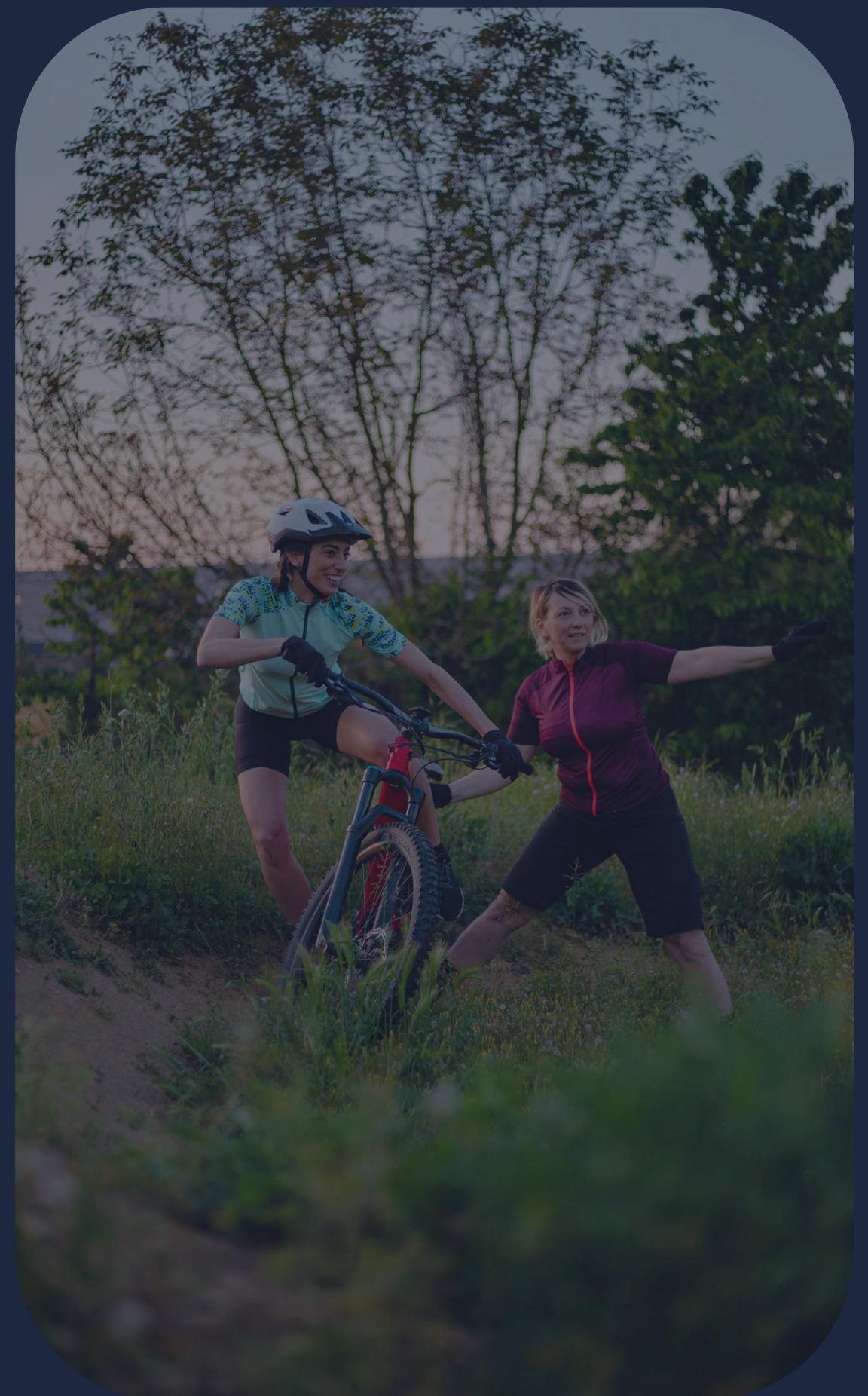
Find the customer who spent the most money on orders

```
with a as (select customers.customer_id,
concat(customers.first_name, " ",customers.last_name) full_name,
sum((order_items.list_price - order_items.discount) * order_items.quantity) money_spent
from customers join orders
on customers.customer_id = orders.customer_id
join order_items
on order_items.order_id = orders.order_id
group by customers.customer_id, full_name)

select * from
(select *, rank() over(order by money_spent desc) rnk from a) b
where rnk = 1;
```

Result

customer_id	full_name	money_spent	rnk
10	Pamelia Newman	3780140.00	1





Staff Performance

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

```
select staffs.staff_id,  
concat(staffs.first_name, " ",staffs.last_name) staff_full_name  
from staffs where not exists  
(select staff_id from orders  
where orders.staff_id = staffs.staff_id);
```

Result

staff_id	staff_full_name
1	Fabiola Jackson
4	Virgie Wiggins
5	Jannette David
10	Bernardine Houston

Staff Performance

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

```
select staffs.staff_id, concat(staffs.first_name, " ", staffs.last_name) staff_full_name,  
coalesce(sum((order_items.list_price-order_items.discount)*order_items.quantity),0) sales  
from order_items join orders  
on order_items.order_id = orders.order_id  
right join staffs  
on orders.staff_id = staffs.staff_id  
group by staffs.staff_id, staff_full_name  
having sales >  
  
    ⊕ (select avg(sales) from  
    ⊕ (select staffs.staff_id, concat(staffs.first_name, " ", staffs.last_name) staff_full_name,  
        coalesce(sum((order_items.list_price-order_items.discount)*order_items.quantity),0) sales  
        from order_items join orders  
        on order_items.order_id = orders.order_id  
        right join staffs  
        on orders.staff_id = staffs.staff_id  
        group by staffs.staff_id, staff_full_name) as a);
```

Result

staff_id	staff_full_name	sales
3	Genna Serrano	95269072.00
6	Marcelene Boyer	293879916.00
7	Venita Daniel	288726544.00





Store Operations

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

```
SELECT  
    stores.store_name,  
    SUM(order_items.quantity) AS total_quantity  
FROM  
    orders  
    JOIN  
    order_items ON order_items.order_id = orders.order_id  
    JOIN  
    stores ON stores.store_id = orders.store_id  
GROUP BY stores.store_name;
```

Result

store_name	total_quantity
Santa Cruz Bikes	1516
Rowlett Bikes	783
Baldwin Bikes	4779

Store Operations

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

```
SELECT  
    store_id, customer_id, COUNT(order_id) number_of_orders  
FROM  
    orders  
GROUP BY store_id , customer_id;
```

Result

store_id	customer_id	number_of_orders
1	259	1
2	1212	1
2	523	1
1	175	2
2	1324	1
2	94	2
2	324	1
2	1204	1
1	60	2
2	442	1
2	1326	1
1	91	2





Inventory Management

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

```
with a as (select orders.order_date, order_items.product_id,
sum(order_items.quantity) total_quantity
from orders join order_items
on orders.order_id = order_items.order_id
group by orders.order_date, order_items.product_id)

select product_id, order_date, total_quantity, sum(total_quantity)
over(partition by product_id order by order_date) cummulative_value from a;
```

Result

product_id	order_date	total_quantity	cummulative_value
2	2016-01-03	2	2
2	2016-01-14	2	4
2	2016-01-18	1	5
2	2016-02-05	1	6
2	2016-02-09	1	7
2	2016-02-26	1	8
2	2016-02-28	2	10
2	2016-03-08	1	11
2	2016-03-14	2	13

Inventory Management

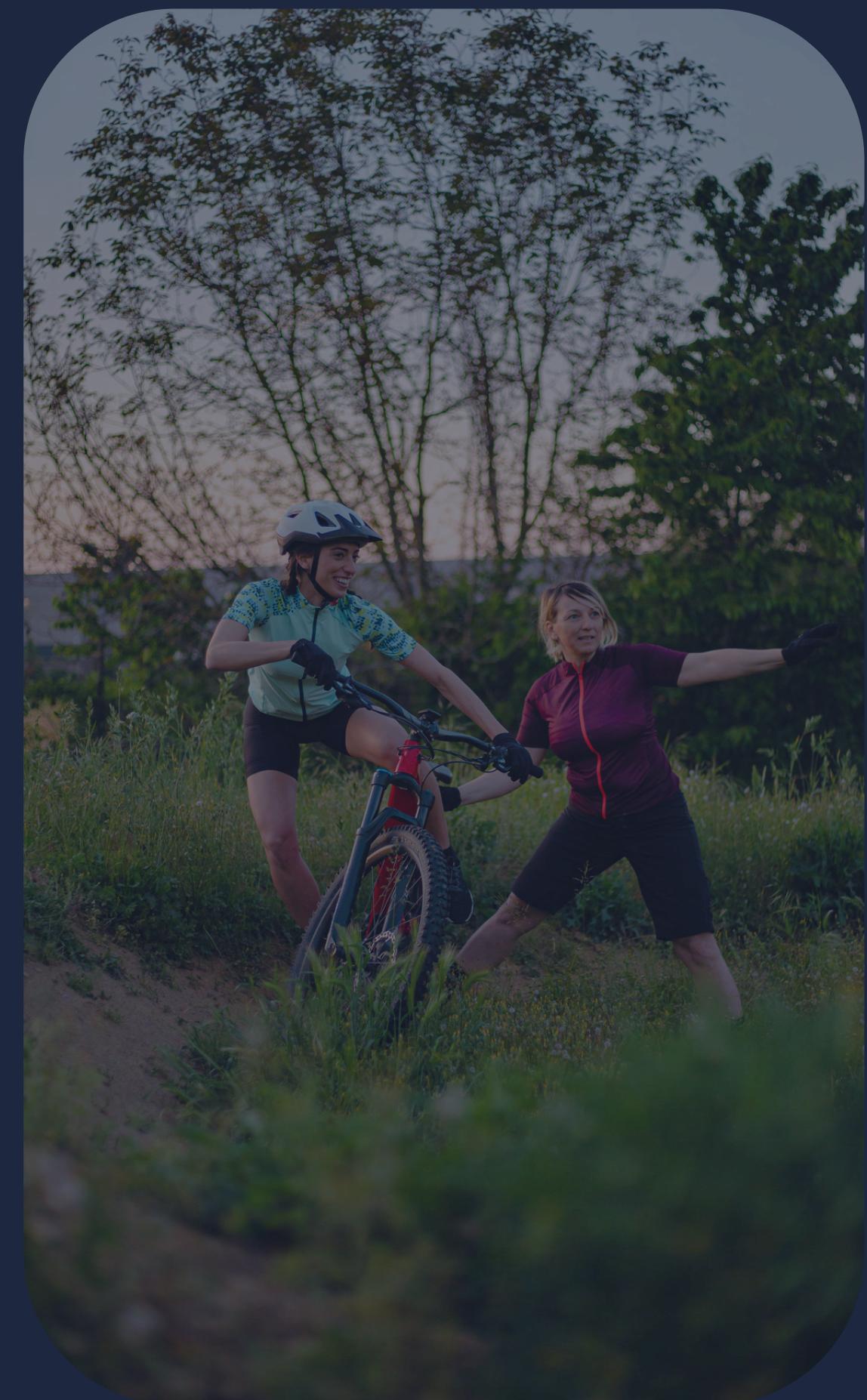
**Find the product with the highest total sales
(quantity * price) for each category.**

```
with a as (select categories.category_id, categories.category_name,
products.product_id, products.product_name,
sum(order_items.quantity * (order_items.list_price - order_items.discount)) sales
from order_items join products
on products.product_id = order_items.product_id
join categories
on categories.category_id = products.category_id
group by categories.category_id, categories.category_name,
products.product_id, products.product_name)

select category_id, category_name, product_id, product_name, sales from
(select *, rank() over(partition by category_id order by sales desc) as rnk
from a) b
where rnk = 1;
```

Result

category_id	category_name	product_id	product_name	sales
1	Children Bicycles	23	Electra Girl's Hawaii 1 (20-inch) - 2015/2016	4619278.00
2	Comfort Bicycles	26	Electra Townie Original 7D EQ - 2016	8039320.00
3	Cruisers Bicycles	16	Electra Townie Original 7D EQ - 2016	9359304.00
4	Cyclocross Bicycles	11	Surly Straggler 650b - 2016	25382383.00
5	Electric Bikes	9	Trek Conduit+ - 2016	43499347.00
6	Mountain Bikes	7	Trek Slash 8 275 - 2016	61599226.00
7	Road Bikes	56	Trek Domane SLR 6 Disc - 2017	23649774.00





Inventory Management

Find the highest-priced product for each category name.

```
select * from
(select categories.category_id, categories.category_name,
products.product_name, products.list_price,
rank() over(partition by categories.category_id order by products.list_price desc) rnk
from products join categories
on products.category_id = categories.category_id) a
where rnk = 1;
```

Result

category_id	category_name	product_name	list_price	rnk
1	Children Bicycles	Electra Straight 8 3i (20-inch) - Boy's - 2017	48999.00	1
1	Children Bicycles	Electra Townie 3i EQ (20-inch) - Boys' - 2017	48999.00	1
1	Children Bicycles	Trek Superfly 24 - 2017/2018	48999.00	1
2	Comfort Bicycles	Electra Townie Go! 8i - 2017/2018	259999.00	1
3	Cruisers Bicycles	Electra Townie Commute Go! - 2018	299999.00	1
3	Cruisers Bicycles	Electra Townie Commute Go! Ladies' - 2018	299999.00	1
4	Cyclocross Bicycles	Trek Boone 7 Disc - 2018	399999.00	1
5	Electric Bikes	Trek Powerfly 8 FS Plus - 2017	499999.00	1
5	Electric Bikes	Trek Powerfly 7 FS - 2018	499999.00	1
5	Electric Bikes	Trek Super Commuter+ 8S - 2018	499999.00	1
6	Mountain Bikes	Trek Fuel EX 98 275 Plus - 2017	529999.00	1
6	Mountain Bikes	Trek Remedy 98 - 2017	529999.00	1
7	Road Bikes	Trek Domane SLR 9 Disc - 2018	1199999.00	1

Inventory Management

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

```
with a as (select products.product_id,  
products.product_name,sum(order_items.quantity) total_quantity  
from order_items join products  
on order_items.product_id = products.product_id  
group by products.product_id, products.product_name)  
  
select product_name from  
(select *, rank() over(order by total_quantity desc) rnk  
from a) b  
where rnk <=3;
```

Result

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





Inventory Management

Find the median value of the price list.

```
with a as (select list_price,  
row_number() over(order by list_price) position,  
count(*) over() n from order_items)  
  
select case  
when n%2 = 0 then (select avg(list_price) from a  
where position in ((n/2),(n/2)+1))  
else ( select list_price from a where position = ((n+1)/2))  
end as median from a limit 1;
```

Result

median
59999.000000

Inventory Management

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

```
select products.product_id,  
products.product_name from products  
where not exists ( select product_id from order_items  
where products.product_id = order_items.product_id);
```

Result

product_id	product_name
1	Trek 820 - 2016
121	Surly Krampus Frameset - 2018
125	Trek Kids' Dual Sport - 2018
154	Trek Domane SLR 6 Disc Women's - 2018
195	Electra Townie Go! 8i Ladies' - 2018
267	Trek Precaliber 12 Girl's - 2018
284	Electra Savannah 1 (20-inch) - Girl's - 2018
291	Electra Sweet Ride 1 (20-inch) - Girl's - 2018
316	Trek Checkpoint ALR 4 Women's - 2019
317	Trek Checkpoint ALR 5 - 2019





Thank You



www.linkedin.com/in/isthatmanik