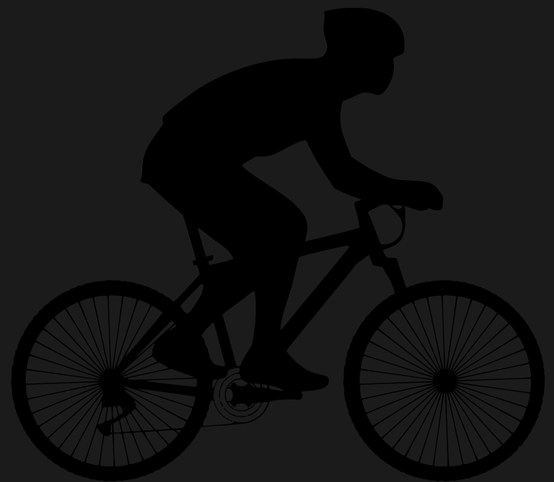
Bike Store Relational Database Project



About the data:

This specific database is a comprehensive database that holds info that may help to manage and analyze various aspects of a bike retail business. It encompasses a variety of tables that capture detailed information about products, customers, orders, staff, and stores. This database serves as a robust foundation for performing in-depth analyses and generating insights to drive business decisions and strategies.

I tried to extract as much as data by segregating them in different analytical parts through questions for the easy readability and understanding.

Sales Analysis

Which brand has the highest total sales, and what is the value of those sales?

```sql
select brands.brand_name, sum(order_items.quantity * (order_items.list_price - order_items.discount)) AS total_sales
from order_items
join products on order_items.product_id = products.product_id
join brands on products.brand_id = brands.brand_id
group by brands.brand_name
order by total_sales desc
limit 1;
```

brand_name	total_sales
Trek	5129187.519999974

Customer Insights:

Which customer has placed the highest number of orders, and what is the total value of their orders?

```
as total_orders, SUM(order_items.quantity * (order_items.list_price - order_items.discount)) as total_value
from customers
join orders on customers.customer_id = orders.customer_id
join order_items on orders.order_id = order_items.order_id
group by  customers.customer_id, customers.first_name, customers.last_name
order by total_orders desc
limit 1;
```

first_name	last_name	total_orders	total_value
Tameka	Fisher	13	26248.01999999997

Product Performance:

Which product has the highest sales volume in terms of quantity sold?

```sql
select products.product_name, sum(order_items.quantity) as total_quantity
from order_items
join products on order_items.product_id = products.product_id
group by products.product_name
order by total_quantity desc
limit 1;
```

	product_name	total_quantity
▶	Electra Cruiser 1 (24-Inch) - 2016	296

Store Performance:

Which store has generated the highest revenue from orders, and what is the total revenue?

```sql
select stores.store_name, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_revenue
from order_items
join orders on order_items.order_id = orders.order_id
join stores on orders.store_id = stores.store_id
group by stores.store_name
order by total_revenue desc
limit 1;
```

	store_name	total_revenue
▶	Baldwin Bikes	5825734.569999948

Discount Analysis:

What is the average discount given per order, and which product category has the highest average discount?

```sql
select categories.category_name, avg(order_items.discount) as avg_discount
from order_items
join products on order_items.product_id = products.product_id
join categories on products.category_id = categories.category_id
group by categories.category_name
order by avg_discount desc
limit 1;
```

	category_name	avg_discount
▶	Cydocross Bicydes	0.11132812500000015

Order Fulfillment:

What percentage of orders were shipped on or before the required date?

```
select round((count(case when orders.shipped_date <= orders.required_date then 1 end) * 100.0 / count(orders.order_id)), 2)
as on_time_percentage
from orders;
```

	on_time_percentage
▶	61.11

Staff Performance:

Which staff member has managed the highest number of orders, and what is the total value of these orders?

```sql
select staffs.first_name, staffs.last_name, count(orders.order_id)
as total_orders, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_value
from staffs
join orders on staffs.staff_id = orders.staff_id
join order_items on orders.order_id = order_items.order_id
group by staffs.staff_id, staffs.first_name, staffs.last_name
order by total_orders desc
limit 1;
```

first_name	last_name	total_orders	total_value
Marcelene	Boyer	1615	2938626.9899999793

Geographical Insights:

Which state has the highest number of customers, and what is the total revenue generated from that state?

```sql
select customers.state, count(customers.customer_id)
as total_customers, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_revenue
from customers
join orders on customers.customer_id = orders.customer_id
join order_items on orders.order_id = order_items.order_id
group by customers.state
order by total_revenue desc
limit 1;
```

	state	total_customers	total_revenue
▶	NY	3195	5825734.569999948

Customer Retention:

What is the average order frequency (number of orders per customer)?

```sql
select round(avg(order_count), 2) as avg_order_frequency
from (
select customers.customer_id, count(orders.order_id) as order_count
from customers
join orders on customers.customer_id = orders.customer_id
group by customers.customer_id
) as customer_orders;
```

avg_order_frequency
1.12

Product Popularity by Category:

Which category has the highest number of unique products sold, and what is the total quantity sold for that category?

```
select categories.category_name, count(distinct products.product_id)
as unique_products_sold, sum(order_items.quantity) as total_quantity_sold
from order_items
join products on order_items.product_id = products.product_id
join categories on products.category_id = categories.category_id
group by categories.category_name
order by total_quantity_sold desc
limit 1;
```

	category_name	unique_products_sold	total_quantity_sold
▶	Cruisers Bicycles	78	2063

Stock Management:

Which product has the lowest stock quantity across all stores, and what is the current stock level?

```sql
select products.product_name, sum(stocks.quantity) as total_stock
from stocks
join products on stocks.product_id = products.product_id
group by products.product_name
order by total_stock asc
limit 1;
```

	product_name	total_stock
▶	Trek Domane SLR Frameset - 2018	5

Revenue Distribution:

What is the distribution of total revenue across different product categories?

```sql
select categories.category_name, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_revenue
from order_items
join products on order_items.product_id = products.product_id
join categories on products.category_id = categories.category_id
group by categories.category_name
order by total_revenue desc;
```

category_name	total_revenue
Mountain Bikes	3030590.8999999766
Road Bikes	1852497.3299999984
Cruisers Bicycles	1108936.83
Electric Bikes	1020203.8500000004
Cyclocross Bicycles	799831.4400000005
Comfort Bicycles	438424.060000002
Children Bicycles	327759.39000000176

Customer Demographics:

What is the most common city among customers, and what is the total revenue generated from customers in that city?

```sql
select customers.city, count(customers.customer_id)
as total_customers, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_revenue
from customers
join orders on customers.customer_id = orders.customer_id
join order_items on orders.order_id = order_items.order_id
group by customers.city
order by total_revenue desc
limit 1;
```

	city	total_customers	total_revenue
▶	Mount Vernon	60	117001.3199999998

Order Value Distribution:

What is the average order value, and how does it vary across different stores?

```sql
select stores.store_name, avg(order_items.quantity * (order_items.list_price - order_items.discount)) as avg_order_value
from orders
join order_items on orders.order_id = order_items.order_id
join stores on orders.store_id = stores.store_id
group by stores.store_name
order by avg_order_value desc;
```

store_name	avg_order_value
Rowlett Bikes	1847.4509980806151
Baldwin Bikes	1823.3911017214234
Santa Cruz Bikes	1779.3113916500938

Product Lifecycle:

How does the sales performance of products vary by model year?

```sql
select products.model_year, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_revenue
from order_items
join products on order_items.product_id = products.product_id
group by products.model_year
order by products.model_year;
```

model_year	total_revenue
2016	3554711.9099999513
2017	3564745.519999958
2018	1458786.3699999978

Order Item Details:

Which order had the highest number of items, and what is the total value of that order?

```sql
select orders.order_id, count(order_items.item_id)
as total_items, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_value
from orders
join order_items on orders.order_id = order_items.order_id
group by orders.order_id
order by total_items desc
limit 1;
```

	order_id	total_items	total_value
▶	1	5	11397.2

Brand Market Share:

What is the market share of each brand in terms of total sales value?

```sql
select brands.brand_name, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_sales,
round(sum(order_items.quantity * (order_items.list_price - order_items.discount)) * 100.0 /
(select sum(order_items.quantity * (order_items.list_price - order_items.discount)) from order_items), 2) as market_share
from order_items
join products on order_items.product_id = products.product_id
join brands on products.brand_id = brands.brand_id
group by brands.brand_name
order by total_sales desc;
```

brand_name	total_sales	market_share
Trek	5129187.519999974	59.79
Electra	1343871.7299999967	15.67
Surly	1063039.200000001	12.39
Sun Bicycles	381842.94000000163	4.45
Haro	207060.3200000002	2.41
Heller	193782.9500000001	2.26
Pure Cycles	166126.25999999963	1.94
Ritchey	88486.01999999995	1.03

Top-selling Products:

List the top 10 best-selling products by quantity sold.

```sql
select products.product_name, sum(order_items.quantity) as total_quantity
from order_items
join products on order_items.product_id = products.product_id
group by products.product_name
order by total_quantity desc
limit 10;
```

product_name	total_quantity
Electra Cruiser 1 (24-Inch) - 2016	296
Electra Townie Original 7D EQ - 2016	290
Electra Townie Original 21D - 2016	289
Electra Girl's Hawaii 1 (16-inch) - 2015/2016	269
Surly Ice Cream Truck Frameset - 2016	167
Trek Slash 8 27.5 - 2016	154
Electra Girl's Hawaii 1 (20-inch) - 2015/2016	154
Surly Straggler 650b - 2016	151
Electra Townie Original 7D - 2015/2016	148
Surly Straggler - 2016	147

Store Sales Contribution:

What percentage of total sales does each store contribute?

```sql
select stores.store_name,
sum(order_items.quantity * (order_items.list_price - order_items.discount)) as store_sales,
round(sum(order_items.quantity * (order_items.list_price - order_items.discount)) * 100.0 /
(select sum(order_items.quantity * (order_items.list_price - order_items.discount)) from order_items), 2) as sales_contribution
from order_items
join orders on order_items.order_id = orders.order_id
join stores on orders.store_id = stores.store_id
group by stores.store_name
order by store_sales desc;
```

store_name	store_sales	sales_contribution
Baldwin Bikes	5825734.569999948	67.91
Santa Cruz Bikes	1789987.2599999944	20.87
Rowlett Bikes	962521.9700000004	11.22

Staff Efficiency:

What is the average number of orders handled by each staff member?

```
select staffs.staff_id, staffs.first_name, staffs.last_name, count(orders.order_id) as total_orders_handled
from staffs
join orders on staffs.staff_id = orders.staff_id
group by staffs.staff_id, staffs.first_name, staffs.last_name
order by total_orders_handled desc;
```

staff_id	first_name	last_name	total_orders_handled
6	Marcelene	Boyer	553
7	Venita	Daniel	540
3	Genna	Serrano	184
2	Mireya	Copeland	164
8	Kali	Vargas	88
9	Layla	Terrell	86

Discount Impact:

How do discounts impact the total sales value and order quantity?

```sql
select
case
when order_items.discount > 0 then 'Discounted'
else 'Non-Discounted'
end as discount_status,
sum(order_items.quantity * (order_items.list_price - order_items.discount)) as total_sales,
count(order_items.order_id) as total_orders
from order_items
group by discount_status;
```

	discount_status	total_sales	total_orders
▶	Discounted	8578243.79999997	4722

Category-wise Revenue:

What is the total revenue generated from each product category, and how does it compare with the overall revenue?

```sql
select categories.category_name, sum(order_items.quantity * (order_items.list_price - order_items.discount)) as category_revenue,
(sum(order_items.quantity * (order_items.list_price - order_items.discount)) * 100.0 /
(select sum(order_items.quantity * (order_items.list_price - order_items.discount)) from order_items)) as revenue_percentage
from order_items
join products on order_items.product_id = products.product_id
join categories on products.category_id = categories.category_id
group by categories.category_name
order by category_revenue desc;
```

category_name	category_revenue	revenue_percentage
Mountain Bikes	3030590.8999999766	35.32880354834386
Road Bikes	1852497.3299999984	21.59529821243837
Cruisers Bicycles	1108936.83	12.927317710415318
Electric Bikes	1020203.8500000004	11.892922068733977
Cyclocross Bicycles	799831.4400000005	9.323953231546106
Comfort Bicycles	438424.060000002	5.110883651966193
Children Bicycles	327759.39000000176	3.8208215765562987

Source of the Database::

https://www.kaggle.com/datasets/dillonmyrick/bike-store-sample-database