# Business Case: Bicycle Sales Data Warehouse

## Introduction

The data warehouse for Bicycle Sales aims to consolidate sales, customer, inventory, and staff data into a unified platform for comprehensive analysis and decision-making. By integrating various data sources, the company will be able to optimize sales, manage inventory effectively, and enhance customer experience. This report discusses the business case, key business questions, the data warehouse answers, the rationale behind its schema design, and the data sources and integration methods employed.

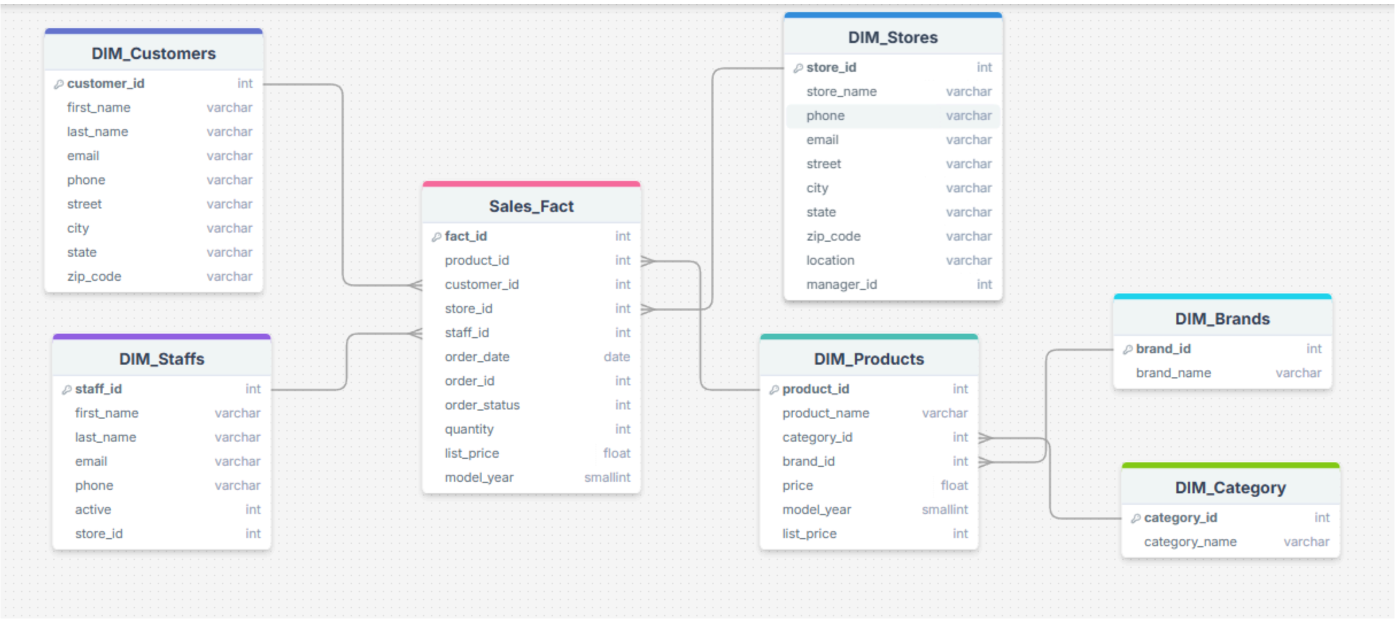
## The Business Case for the Data Warehouse

The bicycle sales company operates across multiple sales channels, offering various types of bicycles (e.g., road bikes, mountain bikes, and e-bikes). With the increasing volume of sales transactions, customer interactions, and product offerings, the company faces challenges in managing data from disparate systems. These challenges include limited visibility into sales performance, inventory management issues, and the inability to target customers effectively with personalized marketing strategies.

A data warehouse solves these problems by centralizing the data from multiple sources, such as sales transactions, customer databases, and inventory systems, into one cohesive platform. The data warehouse will enable better decision-making, streamline reporting, improve forecasting, and help the company respond more effectively to market trends. The integration of data from multiple channels will allow the company to understand sales patterns, optimize inventory, and increase revenue by leveraging data insights.

## SnowFlake Schema design :

The schema design of the bicycle sales data warehouse is structured around a SnowFlake schema, which is ideal for analytical queries and decision-making. The fact table (Sales_Fact) acts as the central hub, linking to various dimension tables (Products, Customers, Stores, Staffs, and Categories). This design facilitates efficient querying, as fact tables store transactional data, and dimension tables store descriptive attributes.



| DIM_Customers | |
|---|---|
| customer_id | int |
| first_name | varchar |
| last_name | varchar |
| email | varchar |
| phone | varchar |
| street | varchar |
| city | varchar |
| state | varchar |
| zip_code | varchar |

| DIM_Staffs | |
|---|---|
| staff_id | int |
| first_name | varchar |
| last_name | varchar |
| email | varchar |
| phone | varchar |
| active | int |
| store_id | int |

| Sales_Fact | |
|---|---|
| fact_id | int |
| product_id | int |
| customer_id | int |
| store_id | int |
| staff_id | int |
| order_date | date |
| order_id | int |
| order_status | int |
| quantity | int |
| list_price | float |
| model_year | smallint |

| DIM_Stores | |
|---|---|
| store_id | int |
| store_name | varchar |
| phone | varchar |
| email | varchar |
| street | varchar |
| city | varchar |
| state | varchar |
| zip_code | varchar |
| location | varchar |
| manager_id | int |

| DIM_Products | |
|---|---|
| product_id | int |
| product_name | varchar |
| category_id | int |
| brand_id | int |
| price | float |
| model_year | smallint |
| list_price | int |

| DIM_Brands | |
|---|---|
| brand_id | int |
| brand_name | varchar |

| DIM_Category | |
|---|---|
| category_id | int |
| category_name | varchar |

### Key features of the schema design:

- **Fact Table (Sales_Fact):** The fact table stores the core transactional data, including sales transactions, order dates, quantities, product details, store details, and staff involved in the sale. This table is connected to dimension tables through foreign keys.
- **Dimension Tables:** These tables provide detailed context for the sales data:
  - Products Dimension: Contains product details, including name, price, category, and brand.
  - Customers Dimension: Stores customer demographic information, enabling customer segmentation.
  - Stores Dimension: Details of each store, including location and management.
  - Staffs Dimension: Contains staff details to analyze performance and track sales by staff members.
  - Categories and Brands Dimensions: Used to group products by category (e.g., road bikes, mountain bikes) and brand, enabling detailed performance analysis.

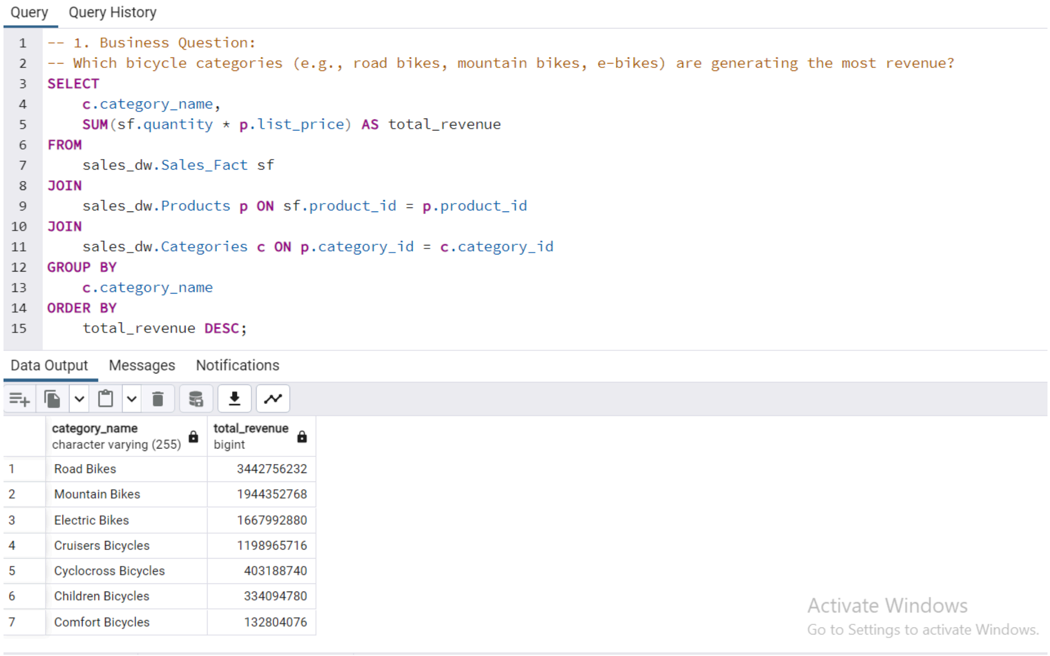
## Inside the zip file:

- Raw data files in the attached folder
- Python file that loads data into PostgreSQL.
- Files to create original tables and snowflake schema tables.
- Answers to business questions.

## Business Questions Answered

The data warehouse answers key business questions that drive operational improvements and growth. These questions include:

- **Which bicycle categories are generating the most revenue?** The warehouse aggregates sales data by product category (e.g., road bikes, mountain bikes) and provides the total revenue generated by each category. This insight helps the company understand which product lines are most profitable and decide where to focus marketing and sales efforts.

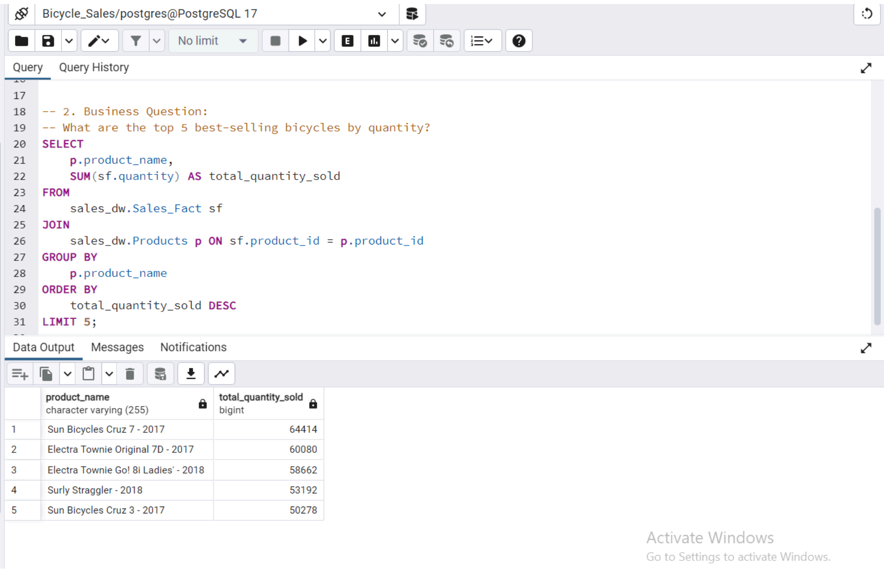


```sql
-- 1. Business Question:
-- Which bicycle categories (e.g., road bikes, mountain bikes, e-bikes) are generating the most revenue?
SELECT
    c.category_name,
    SUM(sf.quantity * p.list_price) AS total_revenue
FROM
    sales_dw.Sales_Fact sf
JOIN
    sales_dw.Products p ON sf.product_id = p.product_id
JOIN
    sales_dw.Categories c ON p.category_id = c.category_id
GROUP BY
    c.category_name
ORDER BY
    total_revenue DESC;
```

| | category_name<br>character varying (255) | total_revenue<br>bigint |
|---|---|---|
| 1 | Road Bikes | 3442756232 |
| 2 | Mountain Bikes | 1944352768 |
| 3 | Electric Bikes | 1667992880 |
| 4 | Cruisers Bicycles | 1198965716 |
| 5 | Cyclocross Bicycles | 403188740 |
| 6 | Children Bicycles | 334094780 |
| 7 | Comfort Bicycles | 132804076 |

- **What are the top-selling bicycles by quantity?**

- The data warehouse allows the company to identify the top-selling bicycles by quantity over a specific time period. This helps with stock planning, ensuring that popular bikes are always in stock while underperforming models can be phased out or promoted.

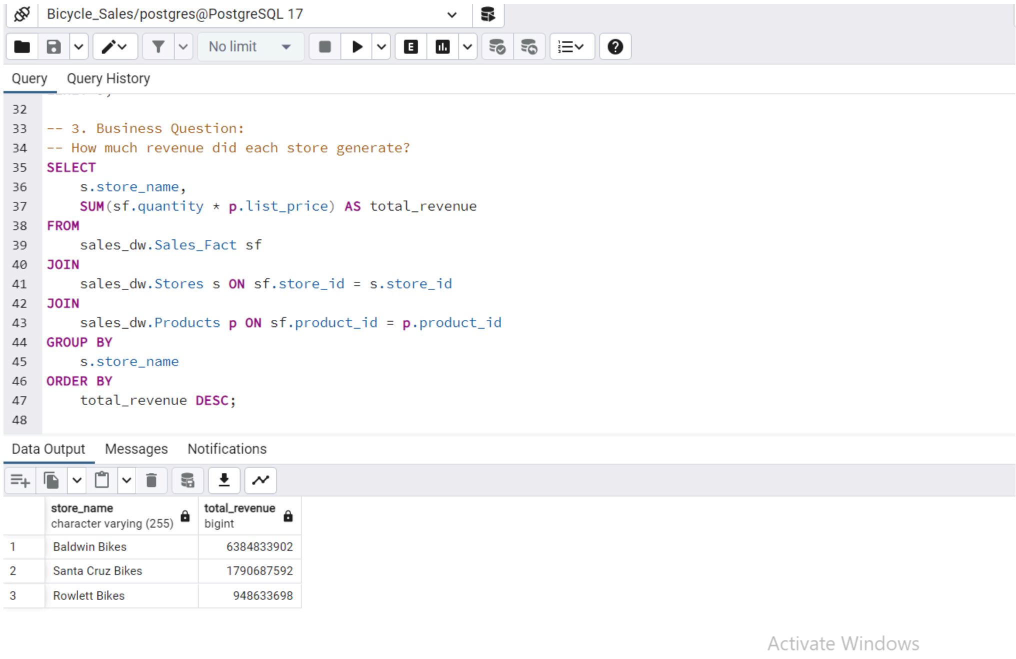


```sql

-- 2. Business Question:
-- What are the top 5 best-selling bicycles by quantity?
SELECT
    p.product_name,
    SUM(sf.quantity) AS total_quantity_sold
FROM
    sales_dw.Sales_Fact sf
JOIN
    sales_dw.Products p ON sf.product_id = p.product_id
GROUP BY
    p.product_name
ORDER BY
    total_quantity_sold DESC
LIMIT 5;
```

| | product_name<br>character varying (255) | total_quantity_sold<br>bigint |
|---|---|---|
| 1 | Sun Bicycles Cruz 7 - 2017 | 64414 |
| 2 | Electra Townie Original 7D - 2017 | 60080 |
| 3 | Electra Townie Go! 8i Ladies' - 2018 | 58662 |
| 4 | Surly Straggler - 2018 | 53192 |
| 5 | Sun Bicycles Cruz 3 - 2017 | 50278 |

- **How much revenue is generated by each store?** By linking sales data with store locations, the data warehouse provides detailed performance metrics at the store level. This information helps identify high-performing stores and areas that require additional support or promotional efforts.



```sql

-- 3. Business Question:
-- How much revenue did each store generate?
SELECT
    s.store_name,
    SUM(sf.quantity * p.list_price) AS total_revenue
FROM
    sales_dw.Sales_Fact sf
JOIN
    sales_dw.Stores s ON sf.store_id = s.store_id
JOIN
    sales_dw.Products p ON sf.product_id = p.product_id
GROUP BY
    s.store_name
ORDER BY
    total_revenue DESC;

```

| | store_name<br>character varying (255) | total_revenue<br>bigint |
|---|---|---|
| 1 | Baldwin Bikes | 6384833902 |
| 2 | Santa Cruz Bikes | 1790687592 |
| 3 | Rowlett Bikes | 948633698 |