SQL Project Report

Prepared by
Vishnu Sharma

Topic
Bikestore Database

• INTRODUCTION

The BikeStores dataset is a sample retail database that contains information about a fictional bicycle store chain. It includes data related to products, orders, customers, staff, and store locations. The data spans across the years 2016 to 2018, providing a realistic view of how a retail business operates.

In this report, we performed a complete SQL-based analysis on the BikeStores database using PostgreSQL. Our objective was to explore and analyze business trends, customer behavior, staff performance, and store-level insights. The project involved:

- Writing complex SQL queries using joins, aggregates, and filters
- Creating views to simplify analysis
- Applying indexing to optimize performance
- Extracting meaningful business insights
- Answering real-world case study questions

BIKESTORES SALES ANALYSIS USING SQL

Name: Vishnu Sharma
Date: 31-march-2025
Tools: PostgreSQL

Objective of the Project

The objective of this project is to perform a complete data analysis on the BikeStores database using SQL, with a focus on extracting actionable business insights.

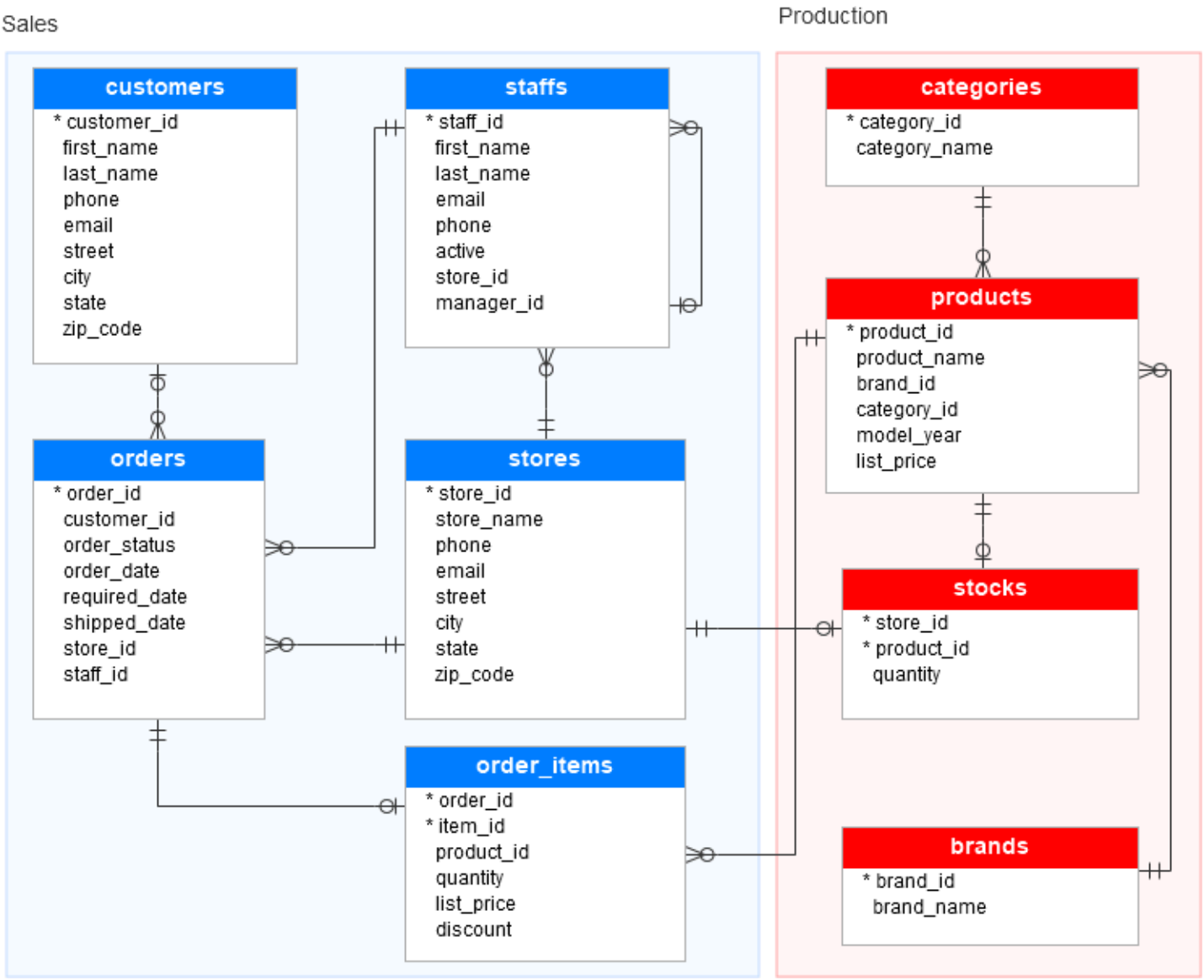
Through this project, we aim to:

- Understand sales performance across different years and stores
- Identify top-performing products and loyal customers
- Analyze staff efficiency and order handling
- Create reusable views for reporting and dashboarding
- Implement indexing to improve query performance
- Solve real-world case study questions using advanced SQL logic

This project simulates a real-world business scenario where a data analyst works with raw transactional data to generate meaningful reports and support business decisions.

Datsource: https://www.sqlservertutorial.net/getting-started/sql-server-sample-database/

THE FOLLOWING ILLUSTRATES THE BIKESTORES DATABASE DIAGRAM:



AS YOU CAN SEE FROM THE DIAGRAM, THE BIKESTORES SAMPLE DATABASE HAS TWO SCHEMAS SALES AND PRODUCTION, AND THESE SCHEMAS HAVE NINE TABLES.

CONCEPT UESD IN THIS PROJECT:

- ETL
- EDA
- INDEXS
- VIEWS
- CASE STUDY QUESTION

PRODUCTION SCHEMA – DESCRIPTION

THE PRODUCTION SCHEMA MANAGES ALL INVENTORY AND PRODUCT-RELATED DATA. IT INCLUDES:

- PRODUCTS (THE BICYCLES AND ACCESSORIES BEING SOLD)
- CATEGORIES (TYPES OF PRODUCTS, E.G., MOUNTAIN, ROAD)
- BRANDS (PRODUCT MANUFACTURERS)
- STOCKS (AVAILABLE QUANTITY OF EACH PRODUCT PER STORE)

THIS SCHEMA ENSURES THE BUSINESS CAN TRACK, MANAGE, AND ANALYZE PRODUCT AVAILABILITY, PRICING, AND MANUFACTURING DETAILS ACROSS ALL STORES.

DATALOAD_QUERY: TABLE_LOAD_QUERY.FILE

PRODUCTION.STOCKS

Column	Description
store_id	Store that holds the stock (FK)
product_id	Product in stock (FK)
quantity	Number of units available at that store

PRODUCTION.BRANDS

Column	Description
brand_id	Unique ID for each brand (Primary Key)
brand_name	Name of the brand

PRODUCTION.CATEGORIES

Column	Description
category_id	Unique ID for each category (Primary Key)
category_name	Name of the product category

PRODUCTION.PRODUCTS

Column	Description
product_id	Unique ID for each product (Primary Key)
product_name	Name of the product
brand_id	Brand associated with the product (FK)
category_id	Product category (FK)
model_year	Year of the product model
list_price	Price of the product

SALES SCHEMA – DESCRIPTION

THE SALES SCHEMA HANDLES ALL CUSTOMER-FACING AND TRANSACTIONAL OPERATIONS OF THE BIKESTORES BUSINESS.

IT INCLUDES TABLES RELATED TO:

- CUSTOMERS (WHO PLACE ORDERS)
- STORES (FROM WHERE BIKES ARE SOLD)
- STAFF (WHO MANAGE THE STORES AND HANDLE ORDERS)
- ORDERS (CUSTOMER PURCHASES)
- ORDER ITEMS (DETAILED LINE ITEMS WITHIN EACH ORDER)
-

TOGETHER, THESE TABLES REPRESENT THE SALES WORKFLOW, INCLUDING WHO BOUGHT WHAT, WHEN, FROM WHICH STORE, AND THROUGH WHICH EMPLOYEE.

DATALOAD_QUERY: SALES_TABLE_LOAD_QUERY.FILE

SALES.STORES

Column	Description
store_id	Unique ID for each store (Primary Key)
store_name	Name of the store
phone	Store contact number
email	Store email address
street	Street address of the store
city	City where store is located
state	State of the store
zip_code	ZIP/Postal code of the store

SALES.STAFFS

Column	Description
staff_id	Unique ID for each staff (Primary Key)
first_name	Staff member's first name
last_name	Staff member's last name
email	Staff email
phone	Staff phone number
active	Staff activity status (1 = active, 0 = inactive)
store_id	Store where the staff works (FK)
manager_id	Manager of the staff (self-referencing FK)

SALES.ORDERS

Column	Description
order_id	Unique ID for each order (Primary Key)
customer_id	Customer who placed the order (FK)
order_status	Status of the order (e.g., 1 = placed)
order_date	Date when the order was placed
required_date	Expected delivery date
shipped_date	Actual shipment date (can be NULL)
store_id	Store from where order was placed (FK)
staff_id	Staff who handled the order (FK)

SALES.ORDER_ITEMS

Column	Description
order_id	Associated order (FK)
item_id	Unique line item number within the order
product_id	Product being ordered (FK)
quantity	Number of units ordered
list_price	Price of the product per unit
discount	Discount applied on the product (in decimal)

SALES.CUSTOMERS

Column	Description
customer_id	Unique ID for each customer (Primary Key)
first_name	Customer's first name
last_name	Customer's last name
phone	Customer phone number
email	Customer email address
street	Customer street address
city	City of the customer
state	State of the customer
zip_code	ZIP code of the customer

EDA DESCRIPTION

IN THIS SQL PROJECT, EXPLORATORY DATA ANALYSIS (EDA) WAS PERFORMED ON THE BIKESTORES DATASET USING A VARIETY OF SQL QUERIES. THE MAIN FOCUS OF THIS PHASE WAS TO UNDERSTAND PATTERNS, TRENDS, AND ANOMALIES HIDDEN IN THE RAW TRANSACTIONAL DATA. BELOW IS A BREAKDOWN OF THE TYPES OF EDA PERFORMED:

EDA_QUERY: EDA_QUERIES.FILE

1. TIME-BASED SALES ANALYSIS

- GOAL: UNDERSTAND THE MONTHLY AND YEARLY TRENDS IN SALES VOLUME AND REVENUE.
- HOW: USED EXTRACT() AND TO_CHAR() TO GROUP ORDERS BY YEAR/MONTH AND CALCULATE TOTAL REVENUE USING SUM(QUANTITY * LIST_PRICE * (1 - DISCOUNT)).
- WHY: HELPS IN IDENTIFYING HIGH-PERFORMING MONTHS AND SALES SEASONALITY.

OUTPUT

	year numeric	month_name text	total_sales bigint
1	2016	April	118
2	2017	April	166
3	2018	April	385
4	2016	August	167
5	2017	August	191
6	2018	August	6
7	2016	December	149
8	2017	December	147

2. STORE-WISE PERFORMANCE

- GOAL: COMPARE STORES BASED ON TOTAL REVENUE AND ORDER COUNT.
- HOW: GROUPED DATA BY STORE, JOINED WITH ORDERS AND ORDER ITEMS, THEN AGGREGATED REVENUE AND ORDER COUNT.
- WHY: TO IDENTIFY WHICH STORE PERFORMS BEST IN TERMS OF SALES AND CUSTOMER ACTIVITY.

OUTPUT

	store_name character varying (255)	total_revenue numeric	orders bigint
1	Baldwin Bikes	5215751.2775	1093
2	Rowlett Bikes	867542.2436	174
3	Santa Cruz Bikes	1605823.0365	348

3. PRODUCT POPULARITY

- GOAL: IDENTIFY TOP-SELLING AND HIGHEST REVENUE-GENERATING PRODUCTS.
- HOW: GROUPED BY PRODUCT NAME AND CALCULATED QUANTITY SOLD AND REVENUE.
- WHY: HELPS IN INVENTORY PLANNING AND MARKETING FOCUS.

OUTPUT

	product_name character varying (255)	total_quantity bigint
1	Electra Cruiser 1 (24-Inch) - 2016	296
2	Electra Townie Original 7D EQ - 2016	290
3	Electra Townie Original 21D - 2016	289
4	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	269
5	Surly Ice Cream Truck Frameset - 2016	167
6	Electra Girl's Hawaii 1 (20-inch) - 2015/2016	154
7	Trek Slash 8 27.5 - 2016	154
8	Surly Straggler 650b - 2016	151
9	Electra Townie Original 7D - 2015/2016	148
10	Surly Straggler - 2016	147

4. CUSTOMER ACTIVITY

- GOAL: ANALYZE CUSTOMER ORDER FREQUENCY, CITY-WISE ORDERS, AND LOYALTY.
- HOW: JOINED CUSTOMERS WITH ORDERS, COUNTED ORDERS, AND GROUPED BY CITY AND CUSTOMER.
- WHY: UNDERSTANDING TOP CUSTOMERS AND ACTIVE REGIONS FOR BETTER TARGETING.

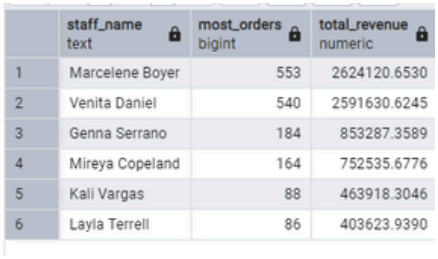
OUTPUT

	full_name text	city character varying (50)	spend_total record	spend_category text
1	Sharyn Hopkins	Baldwinsville	34807.9392,2	high
2	Pamelia Newman	Monroe	33634.2604,2	high
3	Abby Gamble	Amityville	32803.0062,2	high
4	Lyndsey Bean	Fairport	32675.0725,2	high
5	Emmitt Sanchez	New York	31925.8857,2	high
6	Melanie Hayes	Liverpool	31913.6902,2	high
7	Debra Burks	Orchard Park	27888.1834,2	high
8	Elinore Aguilar	San Angelo	25636.4531,2	high
9	Corrina Sawyer	Troy	25612.7021,2	high
10	Shena Carter	Howard Beach	24890.6244,2	high
11	Abram Copeland	Harlingen	24607.0261,2	high
12	Genoveva Baldwin	Port Washington	24433.8818,2	high
13	Robby Sykes	Hempstead	24305.1394,2	high
14	Tameka Fisher	Redondo Beach	24051.5279,2	high
15	Teofila Fischer	Huntington Station	23195.0754,2	high
16	Mozelle Carter	Houston	23049.0570,2	high

5.STAFF PERFORMANCE

- GOAL: EVALUATE STAFF BASED ON HANDLED ORDERS, REVENUE GENERATION, AND DELAYED DELIVERIES.
- HOW: JOINED ORDERS AND ORDER ITEMS WITH STAFF, THEN USED CASE AND COUNT TO FIND LATE SHIPMENTS.
- WHY: ASSISTS IN PERFORMANCE REVIEWS AND IDENTIFYING PROCESS GAPS.

OUTPUT

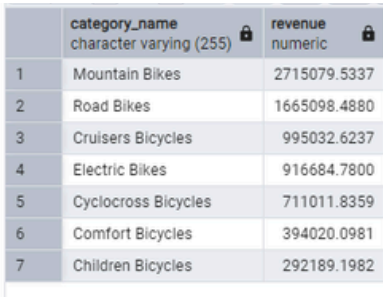


	staff_name text	most_orders bigint	total_revenue numeric
1	Marcelene Boyer	553	2624120.6530
2	Venita Daniel	540	2591630.6245
3	Genna Serrano	184	853287.3589
4	Mireya Copeland	164	752535.6776
5	Kali Vargas	88	463918.3046
6	Layla Terrell	86	403623.9390

6. 📦 PRODUCT CATEGORY TRENDS

- GOAL: EXPLORE REVENUE DISTRIBUTION ACROSS DIFFERENT PRODUCT CATEGORIES.
- HOW: AGGREGATED REVENUE AND COUNT BY CATEGORY USING PRODUCT AND ORDER ITEM JOINS.
- WHY: STRATEGIC PRODUCT PLANNING AND PROMOTIONS.

OUTPUT



	category_name character varying (255)	revenue numeric
1	Mountain Bikes	2715079.5337
2	Road Bikes	1665098.4880
3	Cruisers Bicycles	995032.6237
4	Electric Bikes	916684.7800
5	Cyclocross Bicycles	711011.8359
6	Comfort Bicycles	394020.0981
7	Children Bicycles	292189.1982

INDEXING & QUERY OPTIMIZATION – DESCRIPTION

TO ENHANCE THE PERFORMANCE OF ANALYTICAL QUERIES, ESPECIALLY THOSE INVOLVING LARGE JOINS AND FILTERS, WE IMPLEMENTED INDEXING ON KEY COLUMNS. INDEXES HELP THE DATABASE ENGINE LOCATE ROWS FASTER WITHOUT SCANNING THE ENTIRE TABLE, THUS IMPROVING SPEED SIGNIFICANTLY.

IN THE BIKESTORES PROJECT, THE FOLLOWING INDEXES WERE CREATED:

INDEX QUERY: INDEX QUERY.FILE

1. IDX_ORDERS_ORDER_DATE

TABLE: SALES.ORDERS
COLUMN: ORDER_DATE
WHY: ALMOST ALL TIME-BASED QUERIES (MONTHLY/YEARLY REVENUE) FILTER USING ORDER_DATE.
RESULT: FASTER PERFORMANCE ON QUERIES INVOLVING DATE FILTERING AND GROUPING.

OUTPUT

	QUERY PLAN text
1	Index Scan using idx_orders_order_date on orders (cost=0.28..20.12 rows=292 width=30) (actual time=0.050..0.162 rows=292 loops=...
2	Index Cond: ((order_date >= '2018-01-01'::date) AND (order_date <= '2018-12-31'::date))
3	Planning Time: 20.178 ms
4	Execution Time: 1.120 ms

2. IDX_ORDERS_CUSTOMER_ID

- TABLE: SALES.ORDERS
- COLUMN: CUSTOMER_ID
- TYPE: B-TREE INDEX
- PURPOSE: TO IMPROVE PERFORMANCE OF QUERIES THAT:
  - JOIN ORDERS WITH CUSTOMERS
  - FILTER OR GROUP ORDERS BASED ON CUSTOMER_ID
  - PERFORM CUSTOMER-BASED AGGREGATIONS OR SEGMENTATION

OUTPUT

	tablename name	indexname name	indexdef text
1	stores	stores_pkey	CREATE UNIQUE INDEX stores_pkey ON sales.stores USING btree (store_id)
2	staffs	staffs_pkey	CREATE UNIQUE INDEX staffs_pkey ON sales.staffs USING btree (staff_id)
3	staffs	staffs_email_key	CREATE UNIQUE INDEX staffs_email_key ON sales.staffs USING btree (email)
4	customers	customers_pkey	CREATE UNIQUE INDEX customers_pkey ON sales.customers USING btree (customer_id)
5	orders	orders_pkey	CREATE UNIQUE INDEX orders_pkey ON sales.orders USING btree (order_id)
6	order_items	order_items_pkey	CREATE UNIQUE INDEX order_items_pkey ON sales.order_items USING btree (order_id, item_id)
7	order_items	idx_order_items_product_id	CREATE INDEX idx_order_items_product_id ON sales.order_items USING btree (product_id)
8	orders	idx_orders_order_date	CREATE INDEX idx_orders_order_date ON sales.orders USING btree (order_date)
9	orders	idx_orders_customer_id	CREATE INDEX idx_orders_customer_id ON sales.orders USING btree (customer_id)

VIEWS

WHAT IS A VIEW?
A VIEW IS A VIRTUAL TABLE BASED ON THE RESULT OF A SQL QUERY. IT DOES NOT STORE THE DATA ITSELF BUT ALLOWS USERS TO RETRIEVE DATA AS IF THEY WERE QUERYING A PHYSICAL TABLE. VIEWS HELP IN SIMPLIFYING COMPLEX QUERIES, IMPROVING READABILITY, AND ENFORCING SECURITY BY HIDING SPECIFIC COLUMNS OR ROWS.

VIEWS_QUERY: VIEWS_QUERY.FILE

MONTHLY_STORE_PERFORMANCE_VIEW

- USEFUL FOR TIME-SERIES ANALYSIS, THIS VIEW GROUPS DATA BY MONTH AND YEAR PER STORE.
- ALLOWING FOR TREND INSIGHTS.

OUTPUT

	year numeric	month text	store_name character varying (255)	total_revenue numeric
1	2017	may	Baldwin Bikes	201407.9959
2	2017	august	Rowlett Bikes	52122.1092
3	2018	january	Baldwin Bikes	262508.3779
4	2017	february	Rowlett Bikes	56207.0087
5	2017	december	Baldwin Bikes	196824.5810
6	2017	february	Baldwin Bikes	197814.7258
7	2016	november	Rowlett Bikes	30852.2262
8	2016	october	Rowlett Bikes	13650.8501
9	2017	may	Rowlett Bikes	14727.5494
10	2018	january	Rowlett Bikes	44939.5435
11	2016	october	Baldwin Bikes	129251.7324
12	2018	march	Santa Cruz Bikes	53015.6861
13	2017	november	Santa Cruz Bikes	54307.8571
14	2016	january	Santa Cruz Bikes	71760.3064

TOP_CUSTOMERS_VIEW

- THIS VIEW FILTERS AND LISTS TOP CUSTOMERS WITH THEIR ORDER COUNTS AND EMAIL ADDRESSES.
- MAKING IT EASIER TO IDENTIFY HIGH-VALUE CLIENTS.

OUTPUT

	customer_id integer	customer_name text	total_revenue numeric
1	1007	Jimmy Russell	18278.9436
2	1006	Marjory Leonard	6723.0765
3	1186	Kiesha Bond	1228.5207
4	617	Jenna Saunders	993.2214
5	1421	Edris Barrett	17494.9405
6	1347	Kate Barber	6495.9720
7	177	Carissa Foreman	170.9910
8	91	Marvin Mullins	18856.8770
9	989	Loreen Byers	5090.4175
10	64	Bobbie Foster	8778.1311
11	1429	Lorrie Justice	2042.0634

STORE_REVENUE_VIEW

- THIS VIEW SUMMARIZES STORE-WISE REVENUE BASED ON ORDER ITEMS.
- INCLUDING THE STORE NAME AND SALES FIGURES.

OUTPUT

	stuff_name text	total_order bigint	total_revenue numeric	delayed_orders bigint
1	Genna Serrano	184	853287.3589	136
2	Kali Vargas	88	463918.3046	57
3	Layla Terrell	86	403623.9390	61
4	Marcelene Boyer	553	2624120.6530	478
5	Mireya Copeland	164	752535.6776	162
6	Venita Daniel	540	2591630.6245	457

CASE STUDY QUESTION

IN THIS SECTION OF THE PROJECT, WE APPLIED ADVANCED SQL TECHNIQUES USING CASE EXPRESSIONS TO CATEGORIZE AND LABEL DATA INTO MEANINGFUL BUSINESS SEGMENTS.

THESE CONDITIONAL LOGIC STATEMENTS ALLOW US TO:

- CONVERT RAW NUMERICAL OR DATE VALUES INTO HUMAN-READABLE INSIGHTS
- SEGMENT DATA FOR BETTER BUSINESS UNDERSTANDING
- HIGHLIGHT IMPORTANT PATTERNS OR ANOMALIES THAT WOULD BE HARD TO NOTICE WITH RAW DATA ALONE

CASE_STUDY_QUERY; CASE_STUDY_QUERY.FILE

CASE STUDY Q5:

ANALYZED STAFF PERFORMANCE BY LOOKING AT:

- ORDERS HANDLED
- REVENUE GENERATED
- COUNT OF DELAYED ORDERS

OUTPUT

	stuff_name text	total_order bigint	total_revenue numeric	delayed_orders bigint
1	Genna Serrano	184	853287.3589	136
2	Kali Vargas	88	463918.3046	57
3	Layla Terrell	86	403623.9390	61
4	Marcelene Boyer	553	2624120.6530	478
5	Mireya Copeland	164	752535.6776	162
6	Venita Daniel	540	2591630.6245	457

CASE STUDY Q4:

CALCULATED THE MONTHLY REVENUE FOR EACH STORE IN 2017.
THIS HELPS IDENTIFY SEASONAL TRENDS, HIGH AND LOW REVENUE PERIODS.

OUTPUT

	store_name character varying (255)	year numeric	month_name text	total_revenue numeric
1	Baldwin Bikes	2017	April	180553.25
2	Rowlett Bikes	2017	April	51018.54
3	Santa Cruz Bikes	2017	April	22387.39
4	Baldwin Bikes	2017	August	215006.25
5	Rowlett Bikes	2017	August	57455.73
6	Santa Cruz Bikes	2017	August	49920.43
7	Baldwin Bikes	2017	December	219273.75
8	Rowlett Bikes	2017	December	23956.37
9	Santa Cruz Bikes	2017	December	47661.21
10	Baldwin Bikes	2017	February	221792.31
11	Rowlett Bikes	2017	February	61403.42
12	Santa Cruz Bikes	2017	February	65390.13

CASE STUDY Q3:

WE LISTED THE TOP 5 CUSTOMERS BASED ON THE NUMBER OF ORDERS THEY PLACED.
WE ALSO INCLUDED THEIR EMAIL AND CITY.

OUTPUT

	full_name text	email character varying (255)	city character varying (50)	orders bigint
1	Aaron Knapp	aaron.knapp@yahoo.com	Yonkers	1
2	Abbey Pugh	abbey.pugh@gmail.com	Forest Hills	1
3	Abby Gamble	abby.gamble@aol.com	Amityville	2
4	Abram Copeland	abram.copeland@gmail.com	Harlingen	1
5	Adam Henderson	adam.henderson@hotmail.com	Los Banos	1

CONCLUSION

WHILE WORKING ON THIS SQL PROJECT USING THE BIKESTORES DATASET, I GOT A MUCH DEEPER UNDERSTANDING OF HOW REAL-WORLD RETAIL DATA LOOKS AND WORKS. I DIDN'T JUST WRITE QUERIES — I ACTUALLY EXPLORED HOW BUSINESSES TRACK PRODUCTS, STORES, CUSTOMERS, AND STAFF USING RELATIONAL DATA.

I STARTED BY LOADING THE FULL DATASET, EXPLORED IT USING SQL, AND GRADUALLY MOVED TOWARDS EXTRACTING INSIGHTS USING JOINS, GROUPINGS, CONDITIONS, AND CASE LOGIC. I ALSO CREATED VIEWS TO MAKE THINGS REUSABLE AND IMPLEMENTED INDEXES TO IMPROVE PERFORMANCE, WHICH I HAD ONLY HEARD ABOUT BEFORE THIS.

THIS PROJECT HELPED ME THINK LIKE A DATA ANALYST — NOT JUST WRITING SQL, BUT SOLVING ACTUAL BUSINESS PROBLEMS LIKE FINDING TOP CUSTOMERS, CHECKING STAFF PERFORMANCE, OR UNDERSTANDING SEASONAL TRENDS. I FEEL MORE CONFIDENT NOW IN WRITING CLEAN QUERIES, OPTIMIZING THEM, AND USING SQL NOT JUST FOR LEARNING, BUT FOR REAL DECISION-MAKING.