**Report on Database Project**

**1. Introduction**

The purpose of this project is to design and implement a database system for a bike store. The database manages customers, staffs, stores, orders, and products. The design is based on an ERD (Entity Relationship Diagram) that shows the relationship between tables in the **Sales** and **Production** modules.

**Database Setup**

**1. Database Creation**

First, we created a new database named **BikeStore**:

CREATE DATABASE BikeStore;

USE BikeStore;

**2. Table Creation**

We created tables for both **Sales** and **Production** modules.

**Example: Customers Table**

CREATE TABLE customers (

customer\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(100),

last\_name VARCHAR(100),

phone VARCHAR(20),

email VARCHAR(150),

street VARCHAR(255),

city VARCHAR(100),

state VARCHAR(100),

zip\_code VARCHAR(20)

);

**Example: Products Table**

CREATE TABLE products (

product\_id INT PRIMARY KEY AUTO\_INCREMENT,

product\_name VARCHAR(255),

brand\_id INT,

category\_id INT,

model\_year YEAR,

list\_price DECIMAL(10,2),

FOREIGN KEY (brand\_id) REFERENCES brands(brand\_id),

FOREIGN KEY (category\_id) REFERENCES categories(category\_id)

);

We repeated this process for **all tables** (customers, staffs, stores, orders, order\_items, categories, brands, products, stocks).

**3. Relationships**

After creating tables, we added **primary keys** to uniquely identify each record and **foreign keys** to connect tables.

* Customers are linked to Orders.
* Orders are linked to Order Items.
* Order Items are linked to Products.
* Products are linked to Brands and Categories.
* Stocks link Products with Stores.
* Staffs are linked to Stores and Orders.

Example:

ALTER TABLE orders

ADD FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id),

ADD FOREIGN KEY (store\_id) REFERENCES stores(store\_id),

ADD FOREIGN KEY (staff\_id) REFERENCES staffs(staff\_id);

**Import Data into Tables**

After creating the database and tables, the next step was to insert the data.  
I used two different methods:

**2.1 Import Data Using CSV Files**

For most tables (such as categories, brands, products, customers, stores, orders, and order\_items), I used **MySQL Workbench Table Data Import Wizard**.

Steps:

1. Right-click the table.
2. Select **Table Data Import Wizard**.
3. Choose the CSV file from the project folder.
4. Map the columns with the table structure.
5. Click **Finish** to insert the data.

This method successfully imported all data from CSV files into the respective tables.

**2.2 Insert Staff Data Manually**

For the staffs table, the import wizard did not work correctly.  
Therefore, I inserted the staff data manually using an **INSERT statement**

This ensured that all staff records were inserted correctly into the database.

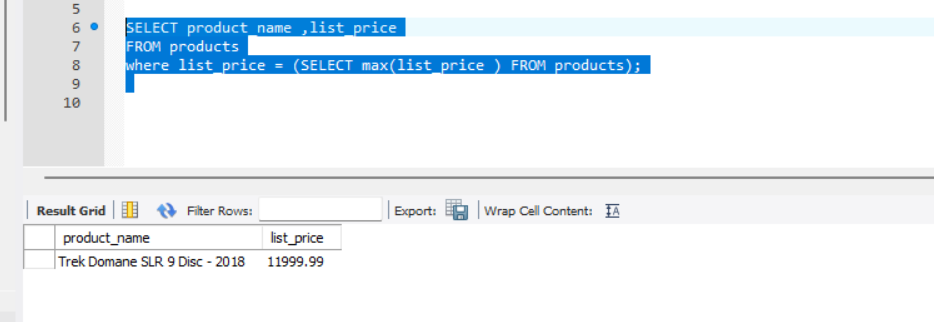
**Queries\_Report**

1. Which bike is most expensive? What could be the motive behind pricing this bike at the high price?

**SELECT product\_name ,list\_price**

**FROM products**

**where list\_price = (SELECT max(list\_price ) FROM products);**



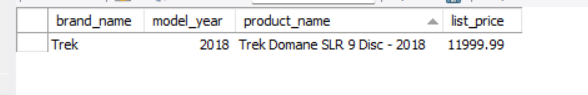
select br.brand\_name , pro.model\_year,pro.product\_name ,pro.list\_price

from products pro

join brands br

on br.brand\_id = pro.brand\_id

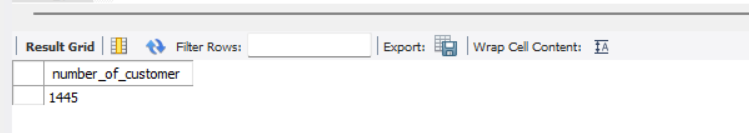
where pro.product\_name = 'Trek Domane SLR 9 Disc - 2018';



The most expensive bike is *Trek Domane SLR 9 Disc - 2018* (Brand: Trek, Year: 2018) with a price of 11,999.99.  
The high price can be justified by the strong Trek brand reputation, use of advanced technology (carbon frame, disc brakes), and the fact that it targets professional cyclists rather than casual customers.

1. How many total customers does BikeStore have? Would you consider people with order status 3 as customers substantiate your answer?

select count(customer\_id) as number\_of\_customer from customers;

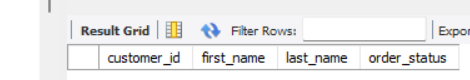


SELECT DISTINCT c.customer\_id, c.first\_name, c.last\_name, o.order\_status

FROM customers c

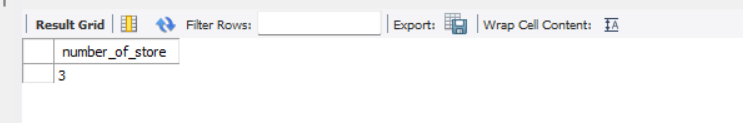
JOIN orders o ON c.customer\_id = o.customer\_id

WHERE o.order\_status = 3 ;



BikeStore has **1445 registered customers**.  
When checking for order status = 3, there were no results. This suggests either that status 3 is not used in the current dataset or that there are no customers with orders in this status. Therefore, all 1445 customers are registered, but the meaning of status 3 should be clarified with the business context.

3) How many stores does BikeStore have?

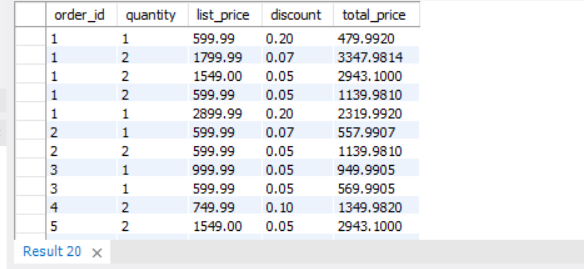
select count(\*) as number\_of\_store from stores;  


1. What is the total price spent per order?

select order\_id, quantity, list\_price, discount,

(list\_price \*quantity\*(1-discount)) as total\_price

from order\_items;

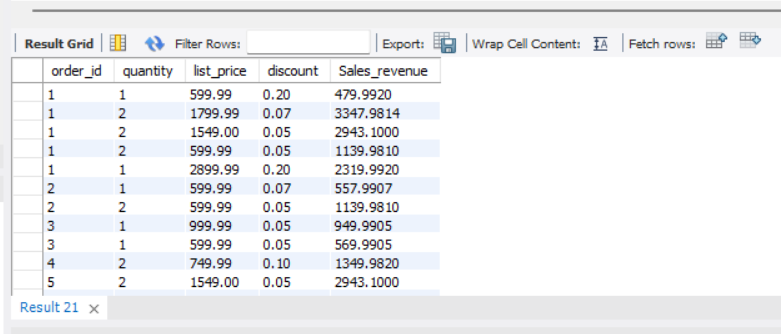


1. What’s the sales/revenue per store? Hint: Sales revenue = ([list\_price] \*[quantity]\*(1-[discount]))

select order\_id, quantity, list\_price, discount,

(list\_price \*quantity\*(1-discount)) as Sales\_revenue

from order\_items;



1. Which category is most sold?

select ca.category\_name , sum( ord.quantity) as total\_sold

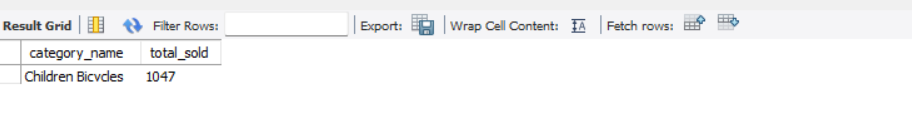
from categories ca

join products pr on ca.category\_id = pr.category\_id

join order\_items ord on ord.product\_id = pr.product\_id

group by(pr.category\_id)

limit 1;



1. Which category rejected more orders?

SELECT

p.product\_name,

SUM(oi.quantity) AS total\_sold

FROM order\_items oi

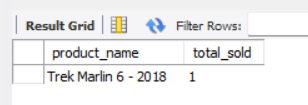
JOIN products p

ON oi.product\_id = p.product\_id

GROUP BY p.product\_name

ORDER BY total\_sold ASC

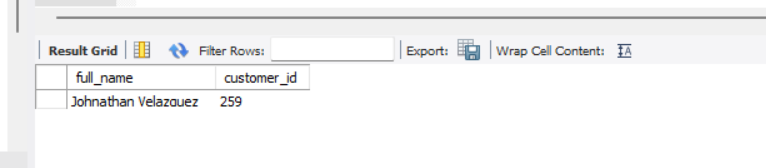
LIMIT 1;



9) What’s the full name of a customer with ID 259?

select concat(first\_name,' ',last\_name) as full\_name ,customer\_id

from customers

where customer\_id = 259 ;  


10) What did the customer on question 9 buy and when? What’s the

status of this order?  
SELECT

c.customer\_id,

c.first\_name,

c.last\_name,

p.product\_name,

oi.quantity,

o.order\_date,

o.order\_status

FROM customers c

JOIN orders o

ON c.customer\_id = o.customer\_id

JOIN order\_items oi

ON o.order\_id = oi.order\_id

JOIN products p

ON oi.product\_id = p.product\_id

WHERE c.customer\_id = 259; 