

BUSINESS INTELLIGENCE

Laporan Final

Dosen Pembimbing:

Ibu Endah Septa Sintiya, S.pd., M.Kom.



Disusun oleh:

Duta Kurnia Ardhani	2341720139
Luqman Ananta A. H.	2341720077
Muhammad Zaki	2341720052
Valentina Santi Grehasta	2341720016

**JURUSAN TEKNOLOGI INFORMASI
PROGRAM STUDI D-IV TEKNIK INFORMATIKA
POLITEKNIK NEGERI MALANG
2025**

A. PENDAHULUAN

Business Intelligence (BI) merupakan proses pengolahan data mentah menjadi informasi yang bernilai guna mendukung pengambilan keputusan. Dalam implementasinya, BI membutuhkan data yang terstruktur dengan baik, umumnya disimpan dalam sebuah Data Warehouse.

Pada laporan ini, dilakukan pembahasan mengenai dataset Bike Store Sample Database yang bersumber dari Kaggle, kemudian dilanjutkan dengan perancangan star schema sebagai model data warehouse. Dataset ini dipilih karena merepresentasikan proses bisnis ritel sepeda secara lengkap, meliputi penjualan, produk, pelanggan, toko, dan stok.

B. DATA SET

1. Sumber Dataset

Dataset yang digunakan pada proyek Business Intelligence ini bersumber dari platform Kaggle dengan judul *Bike Store Sample Database*. Dataset ini diunggah oleh Dillon Myrick dan merupakan versi contoh database relasional yang banyak digunakan untuk pembelajaran SQL dan analisis data.

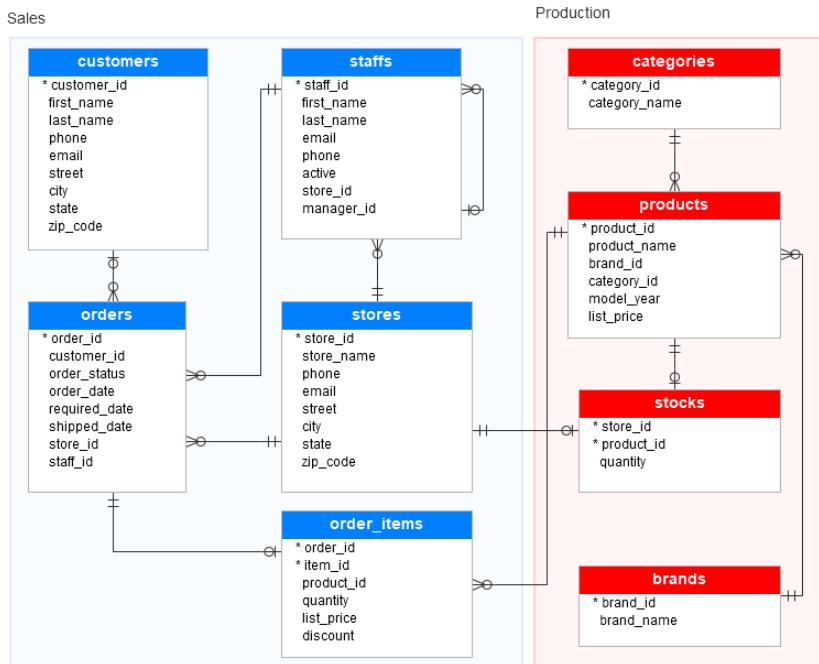
Dataset ini merepresentasikan data operasional (OLTP) dari sebuah toko sepeda, sehingga masih berorientasi pada transaksi harian dan belum dioptimalkan untuk analisis Business Intelligence.

2. Gambaran Umum Dataset

Dataset Bike Store terdiri dari beberapa tabel relasional yang saling terhubung melalui primary key dan foreign key. Dataset ini mencakup data master dan data transaksi yang menggambarkan aktivitas bisnis secara menyeluruh.

Data yang tersedia mencakup informasi mengenai produk sepeda, merek, kategori, pelanggan, staf, toko, transaksi penjualan, detail penjualan, serta stok produk.

3. Struktur Tabel Dataset



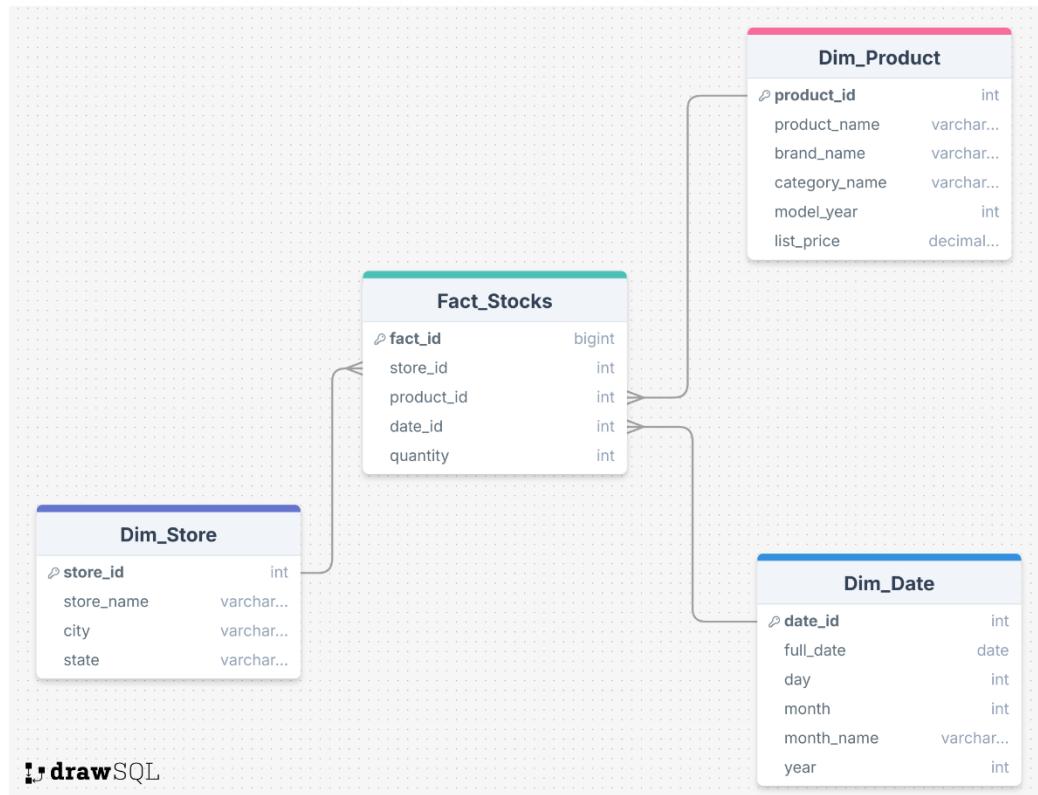
Berikut adalah tabel-tabel utama yang terdapat dalam dataset Bike Store:

- a. **Brand**: Menyimpan data merek sepeda.
- b. **Categories**: Menyimpan kategori produk sepeda.
- c. **Products**: Berisi informasi detail produk sepeda, seperti nama produk, merek, kategori, tahun model, dan harga.
- d. **Customers**: Menyimpan data pelanggan yang melakukan pembelian.
- e. **Stores**: Menyimpan informasi toko sepeda.
- f. **Staffs**: Menyimpan data pegawai yang bekerja di toko.
- g. **Orders**: Menyimpan data transaksi penjualan (header order).
- h. **Order_Items**: Menyimpan detail item produk yang dibeli dalam setiap transaksi.
- i. **Stocks**: Menyimpan informasi jumlah stok produk di masing-masing toko.

C. DESAIN STAR SCHEMA

1. Pengertian Star Schema

Star schema merupakan model perancangan Data Warehouse yang terdiri dari satu tabel fakta sebagai pusat data dan beberapa tabel dimensi yang mengelilinginya. Struktur ini menyerupai bentuk bintang, sehingga disebut star schema. Model star schema dipilih karena sederhana, mudah dipahami, dan sangat efektif untuk kebutuhan analisis Business Intelligence.



2. Tabel Dimensi

Tabel dimensi yang digunakan dalam star schema Bike Store meliputi:

a. Dim_Date

Tabel Dim_Date menyimpan informasi waktu untuk keperluan analisis temporal.

Atribut utama:

- date_id
- full_date
- day
- month
- month_name
- year

Dimensi waktu sangat penting untuk analisis tren dan pola perubahan stok dari waktu ke waktu.

b. Dim_Product

Tabel Dim_Product menyimpan informasi detail terkait produk sepeda.

Atribut utama:

- product_id

- product_name
- brand_name
- category_name
- model_year
- list_price

Dimensi ini digunakan untuk menganalisis stok berdasarkan produk, merek, kategori, dan tahun model.

c. Dim_Store

Tabel **Dim_Store** menyimpan informasi mengenai toko sepeda.

Atribut utama:

- store_id
- store_name
- city
- state

Dimensi ini memungkinkan analisis stok dan performa berdasarkan lokasi toko.

3. Tabel Fakta

Tabel Fact_Stocks merupakan tabel inti yang menyimpan data kuantitatif yang dapat dianalisis. Tabel ini merepresentasikan kondisi stok produk pada setiap toko dan waktu tertentu. Tabel ini berisi data kuantitatif yang dapat dianalisis, antara lain:

- fact_id (primary key)
- store_id (foreign key ke Dim_Store)
- product_id (foreign key ke Dim_Product)
- date_id (foreign key ke Dim_Date)
- quantity (jumlah produk terjual)

Tabel ini memungkinkan analisis seperti jumlah stok produk per toko, per produk, dan per periode waktu.

D. PROSES ETL

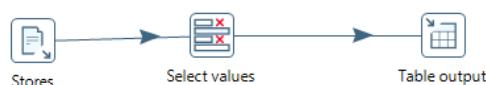
1. Dim_Date



- a. CSV file input: Membaca data tanggal dari file CSV tabel orders.
- b. Select Values: Memilih kolom yang akan dipakai yaitu order_date yang direname menjadi full_date.
- c. Unique rows: Menghilangkan tanggal yang duplikat.
- d. Add Sequence: Membuat date_id sebagai primary key dim_date.
- e. Calculator: Mengambil bagian tanggal dari full_date yaitu day, month, dan year.
- f. Value mapper: Mengubah angka bulan menjadi nama bulan sehingga menjadi kolom baru bernama month_name.
- g. Select Values: Merapikan struktur akhir sesuai tabel dim_date.
- h. Table Output: Menyimpan hasilnya ke database MySQL pada tabel dim_date.
- i. Hasil Akhir

date_id	full_date	day	month	month_name	year
1	2016-01-01	1	1	January	2016
2	2016-01-02	2	1	January	2016
3	2016-01-03	3	1	January	2016
4	2016-01-04	4	1	January	2016
5	2016-01-05	5	1	January	2016
6	2016-01-06	6	1	January	2016
7	2016-01-08	8	1	January	2016
8	2016-01-09	9	1	January	2016
9	2016-01-12	12	1	January	2016
10	2016-01-14	14	1	January	2016

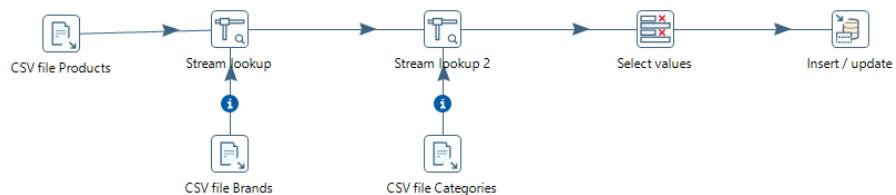
2. Dim_Store



- a. CSV File Input: Membaca data store dari file CSV tabel stores.
- b. Select Values: Memilih kolom yang akan dipakai yaitu store_name, city, dan state.
- c. Table Output: Menyimpan hasilnya ke database MySQL pada tabel dim_store.
- d. Hasil Akhir

store_id	store_name	city	state
1	Santa Cruz Bikes	Santa Cruz	CA
2	Baldwin Bikes	Baldwin	NY
3	Rowlett Bikes	Rowlett	TX

3. Dim_Product



- a. CSV File Products: Membaca data produk sepeda dari file CSV tabel products.
- b. CSV File Brands: Membaca data brand produk dari file CSV tabel brands.
- c. CSV File Products: Membaca data kategori produk dari file CSV tabel categories.
- d. Stream Lookup 1: Melakukan lookup ke file CSV tabel brands untuk mendapatkan brand_name berdasarkan brand_id.
- e. Stream Lookup 2: Melakukan lookup ke file CSV tabel categories untuk mendapatkan category_name berdasarkan category_id.
- f. Select Values: Memilih dan merapikan kolom yang akan digunakan dalam tabel dimensi yaitu product_id, product_name, brand_name, category_name, model_year, list_price.
- g. Insert/Update: Menyimpan hasil transformasi ke database MySQL pada tabel dim_product.
- h. Hasil Akhir

product_id	product_name	brand_name	category_name	model_year	list_price
1	Trek 820 - 2016	Trek	Mountain Bikes	2016	379.99
2	Ritchey Timberwolf Frameset - 2016	Ritchey	Mountain Bikes	2016	749.99
3	Surly Wednesday Frameset - 2016	Surly	Mountain Bikes	2016	999.99
4	Trek Fuel EX 8 29 - 2016	Trek	Mountain Bikes	2016	2899.99
5	Heller Shagamaw Frame - 2016	Heller	Mountain Bikes	2016	1320.99
6	Surly Ice Cream Truck Frameset - 2016	Surly	Mountain Bikes	2016	469.99
7	Trek Slash 8 27.5 - 2016	Trek	Mountain Bikes	2016	3999.99
8	Trek Remedy 29 Carbon Frameset - 2016	Trek	Mountain Bikes	2016	1799.99
9	Trek Conduit+ - 2016	Trek	Electric Bikes	2016	2999.99
10	Surly Straggler - 2016	Surly	Cyclocross Bicycles	2016	1549.00

4. Fact_Stocks



- CSV File Input: Membaca data stok dari file CSV tabel stocks yaitu store id, product_id, dan quantity.
- Database Lookup Store: Melakukan lookup ke tabel dim_store di database MySQL untuk memastikan store_id yang berasal dari CSV sesuai dengan data pada tabel dimensi store.
- Database Lookup Product: Melakukan lookup ke tabel dim_product di database MySQL untuk memvalidasi product_id yang berasal dari CSV.
- Add Constants: Menambahkan nilai konstan untuk kolom date_id. Karena data stok bersifat dummy dan menggunakan satu tanggal tertentu (misalnya tahun 2018), maka date_id diisi dengan nilai tetap yang mengacu ke tabel dim_date.
- Database Lookup Date: Melakukan lookup ke tabel dim_date untuk mendapatkan date_id berdasarkan tanggal yang sudah ditentukan sebelumnya.
- Table Output: Menyimpan hasil transformasi ke database MySQL pada tabel fact_stocks.
- Hasil Akhir

fact_id	store_id	product_id	date_id	quantity
1	1	1	722	27
2	1	2	722	5
3	1	3	722	6
4	1	4	722	23
5	1	5	722	22
6	1	6	722	0
7	1	7	722	8
8	1	8	722	0
9	1	9	722	11
10	1	10	722	15