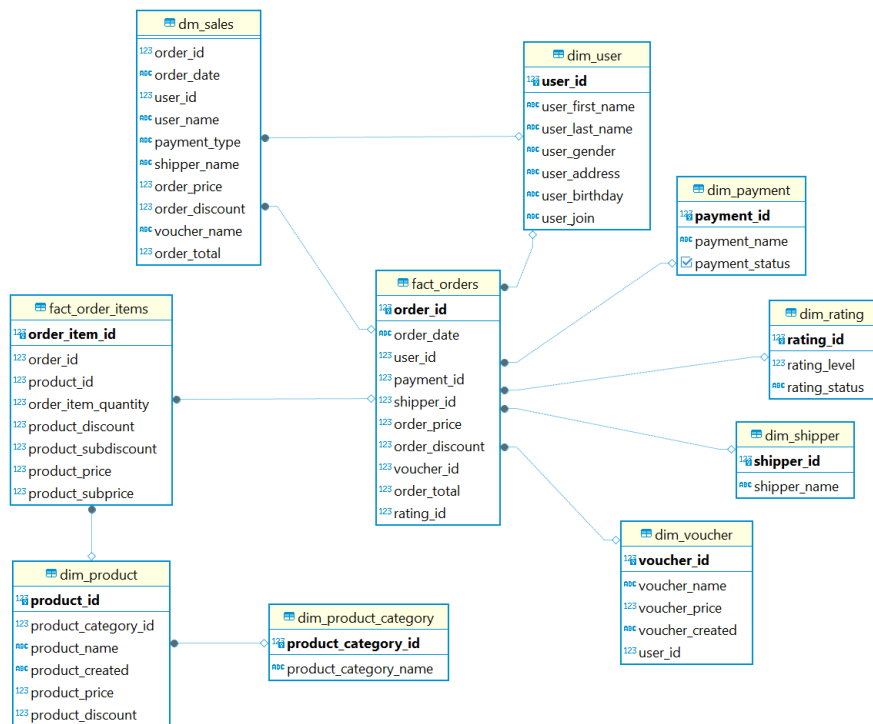# PROJECT 1 – BUSINESS INTELLIGENCE
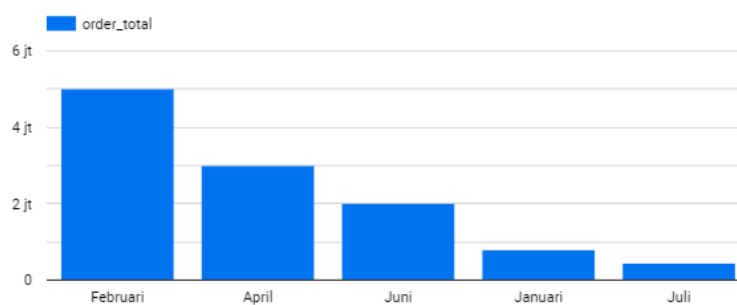

Business Requirement :
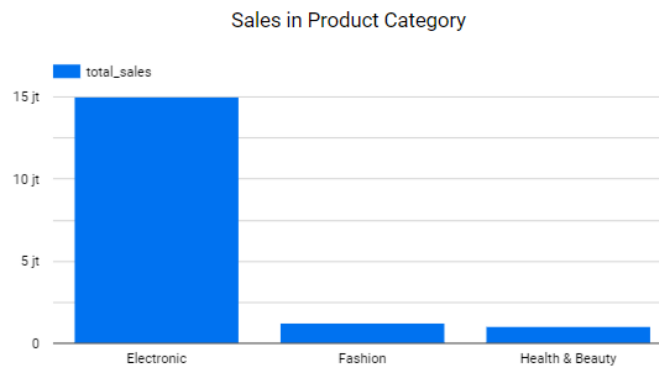
```
--1. Total Penjualan per Bulan
SELECT EXTRACT(YEAR FROM TO_DATE(order_date, 'YYYY-MM-DD')) AS "year",
       LPAD(EXTRACT(MONTH FROM TO_DATE(order_date, 'YYYY-MM-DD'))::text, 2, '0')
AS "month",
       SUM(order_total) AS total_sales_per_month
FROM dm_sales
GROUP BY
    1,2
ORDER BY
    1,2
;
```
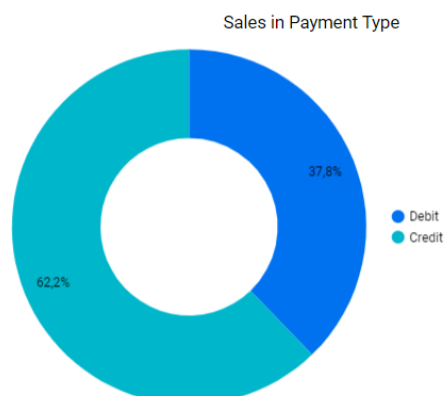

Total Order Per Month

```sql
--2. Penjualan per Kategori Produk
SELECT dpc.product_category_id,
       dpc.product_category_name,
       SUM(s.order_total) AS total_sales
FROM dm_sales s
LEFT JOIN fact_orders fo
ON s.order_id = fo.order_id
LEFT JOIN fact_order_items foi
ON fo.order_id  = foi.order_id
LEFT JOIN dim_product dp
ON foi.product_id = dp.product_id
LEFT JOIN dim_product_category dpc
ON dp.product_category_id  = dpc.product_category_id
GROUP BY
    1,2
ORDER BY
      1,2
;
```

Sales in Product Category


```sql
--3. Penjualan per Metode Pembayaran
SELECT dpm.payment_id,
       s.payment_type,
       SUM(s.order_total) AS total_sales
FROM dm_sales s
LEFT JOIN fact_orders fo
ON s.order_id = fo.order_id
LEFT JOIN dim_payment dpm
ON fo.payment_id = dpm.payment_id
GROUP BY
    1,2
ORDER BY
      1,2
;
```
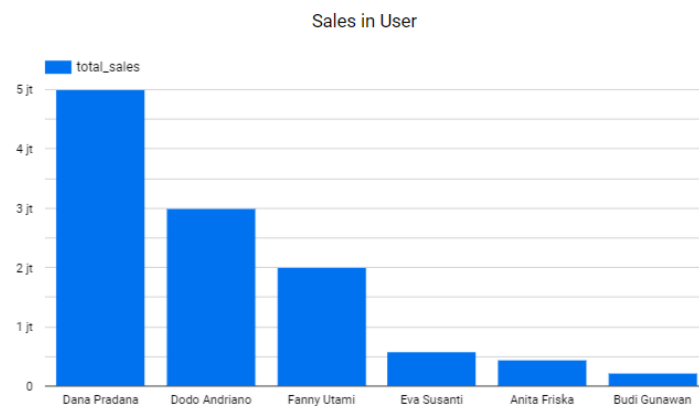
Sales in Payment Type

```sql
--4. Penjualan per Pengirim
SELECT ds.shipper_id,
       s.shipper_name,
       SUM(s.order_total) AS total_sales
FROM dm_sales s
LEFT JOIN fact_orders fo
ON s.order_id = fo.order_id
LEFT JOIN dim_shipper ds
ON fo.shipper_id = ds.shipper_id
GROUP BY
    1,2
ORDER BY
       1,2
;
```
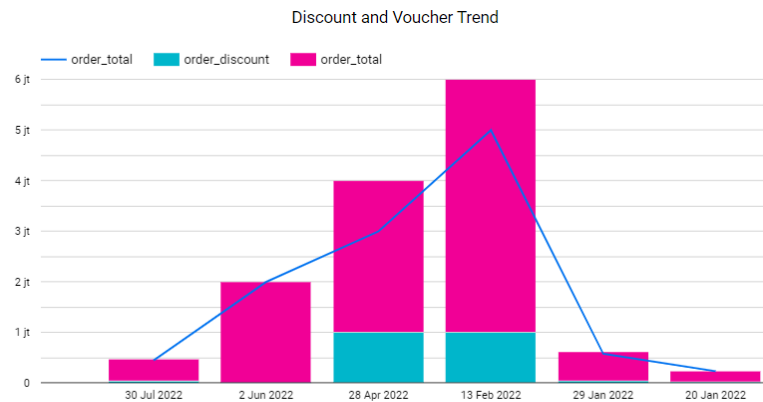
Sales in Shipper


```sql
--5. Penjualan per Pengguna
SELECT user_id,
       user_name,
       SUM(order_total) AS total_sales
FROM dm_sales
GROUP BY
    1,2
ORDER BY
       1,2
;
```
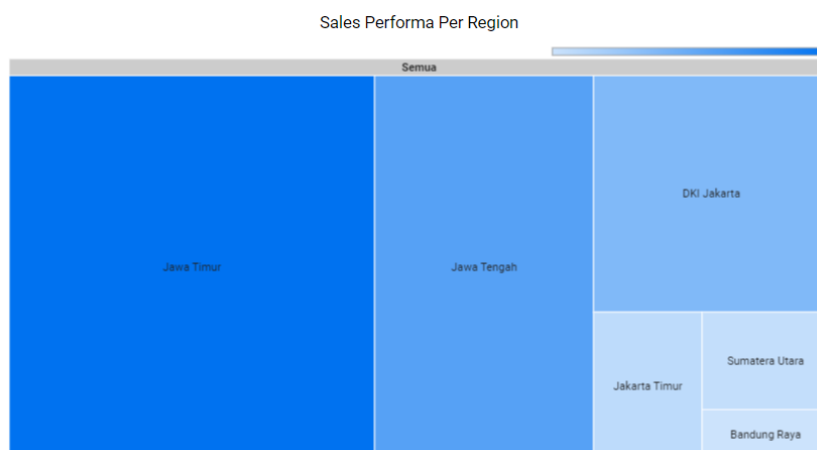
Sales in User

```sql
--6. Tren Diskon dan Voucher
SELECT TO_DATE(order_date, 'YYYY-MM-DD') AS "date",
        order_discount,
        voucher_name,
        order_total
FROM dm_sales
;
```
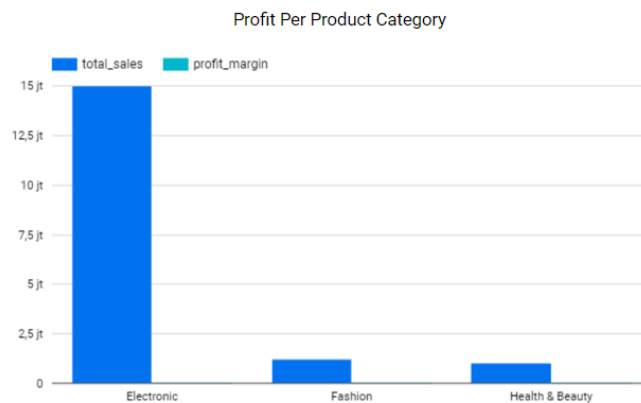

Discount and Voucher Trend

```sql
--7. Performa Penjualan per Wilayah
SELECT split_part(du.user_address, ',', 3) as region,
        SUM(s.order_total) AS total_sales
FROM dm_sales s
LEFT JOIN fact_orders fo
ON s.order_id = fo.order_id
LEFT JOIN dim_user du
ON fo.user_id = du.user_id
GROUP BY
    1
ORDER BY
    1
;
```
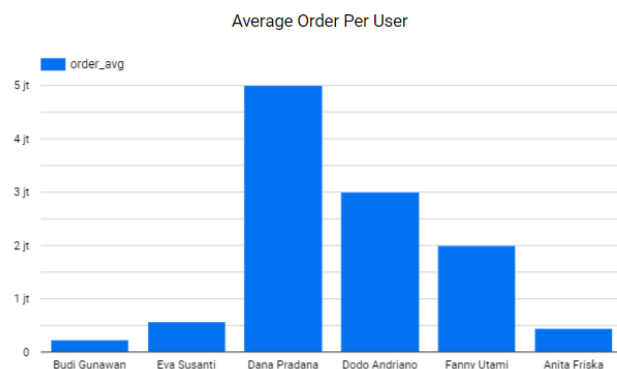

Sales Performa Per Region

```
--8. Margin Laba per Kategori Produk
SELECT dpc.product_category_id,
        dpc.product_category_name,
        SUM(s.order_price) AS total_cost,
        SUM(s.order_total) AS total_sales,
        SUM(s.order_discount) AS total_discount,
        (SUM(s.order_total) - SUM(s.order_price)- SUM(s.order_discount)) /
SUM(s.order_total) AS profit_margin
FROM dm_sales s
LEFT JOIN fact_orders fo
ON s.order_id = fo.order_id
LEFT JOIN fact_order_items foi
ON fo.order_id  = foi.order_id
LEFT JOIN dim_product dp
ON foi.product_id = dp.product_id
LEFT JOIN dim_product_category dpc
ON dp.product_category_id  = dpc.product_category_id
GROUP BY
    1,2
ORDER BY
      1,2
;
```


Profit Per Product Category

```
--9. Nilai Pesanan Rata-rata per Pengguna
SELECT user_id,
        user_name,
        AVG(order_total) AS order_avg
FROM dm_sales
GROUP BY
    1,2
ORDER BY
      1,2
;
```


Average Order Per User

```sql
--10. Tingkat Konversi Voucher
WITH voucher_recipient AS (
    SELECT
        dv.voucher_name,
        COUNT(DISTINCT dv.user_id) AS users_received
    FROM
        dim_voucher dv
    LEFT JOIN
        fact_orders fo ON dv.voucher_id = fo.voucher_id
    LEFT JOIN
        dm_sales s ON dv.voucher_name = s.voucher_name
    LEFT JOIN
        dim_user du ON fo.user_id = du.user_id
    GROUP BY
        dv.voucher_name
),
voucher_usage AS (
    SELECT
        voucher_name,
        COUNT(DISTINCT user_id) AS users_used
    FROM
        dm_sales
    WHERE
        voucher_name IS NOT NULL
    GROUP BY
        voucher_name
)
SELECT
    vr.voucher_name,
    vr.users_received,
    COALESCE(vu.users_used, 0) AS users_used,
    CASE
        WHEN vr.users_received = 0 THEN 0
        ELSE (COALESCE(vu.users_used, 0)::FLOAT / vr.users_received) * 100
    END AS conversion_rate
FROM
    voucher_recipient vr
LEFT JOIN
    voucher_usage vu ON vr.voucher_name = vu.voucher_name
ORDER BY
    vr.voucher_name
;
```

Conversion Voucher Rate