

# Analisis Data dwh\_phi Menggunakan SQL



**BY: SAFFA RAHMATULLAH**

## Link Dataset :

[\*https://bit.ly/dataset\\_dwi\\_phi\\_analyst\*](https://bit.ly/dataset_dwi_phi_analyst)

## Tools yang digunakan :



**Project ini menggunakan Query  
SQL Basic to Advance**

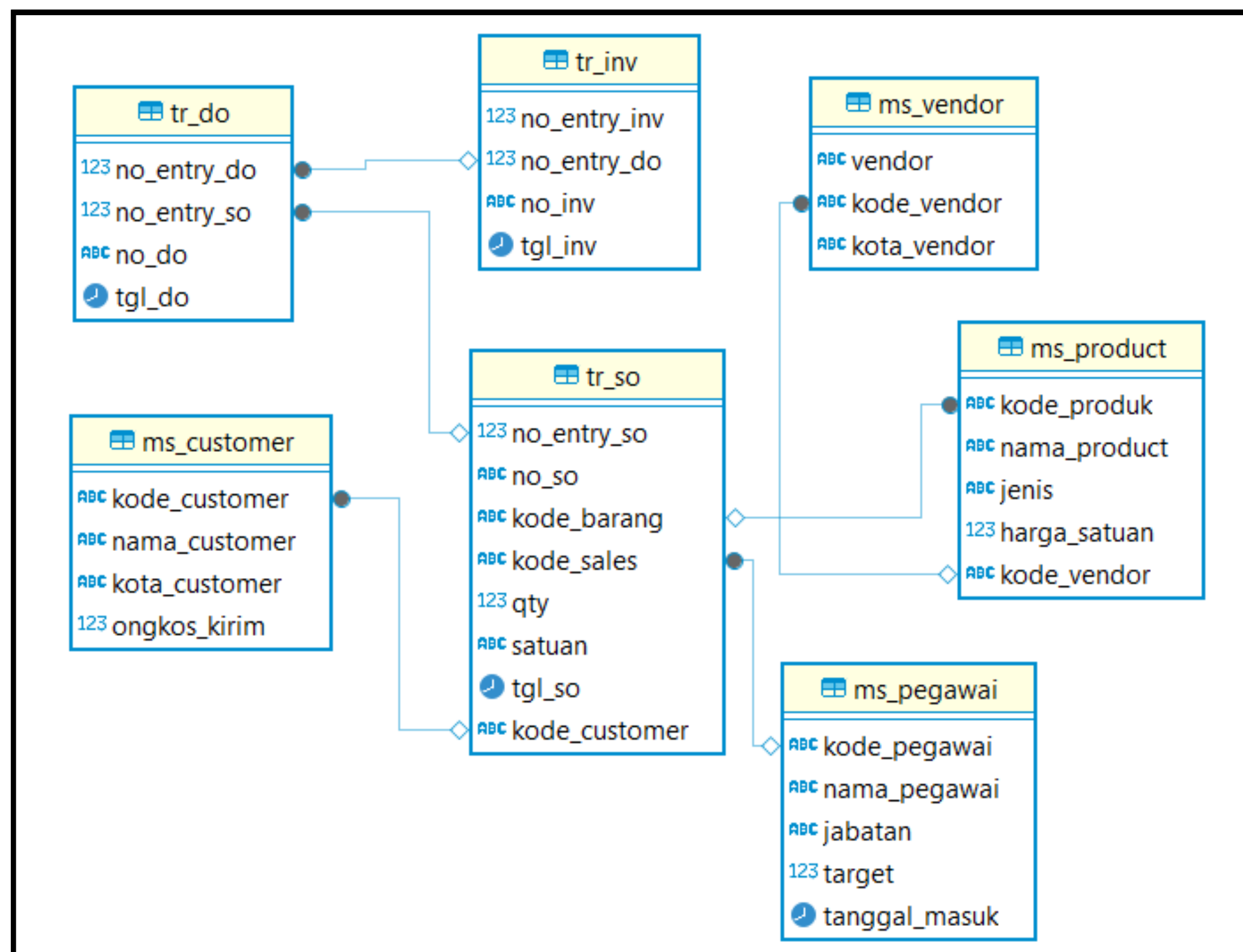
## Link Github :

**Untuk melihat secara lengkap analisis  
project ini, dapat diakses pada link  
github berikut:**

[\*https://bit.ly/github\\_dwi\\_phi\\_analyst\*](https://bit.ly/github_dwi_phi_analyst)

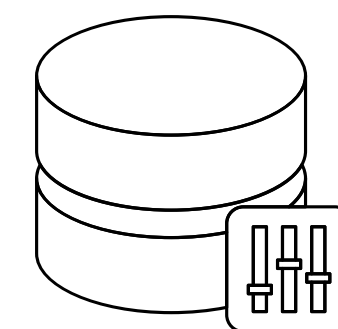


# ER Diagram



# Data Description

Column Name	Description
no_entry_do	nomor entry delivery order
no_entry_so	nomor entry sales order
no_do	nomor delivery order
tgl_do	tanggal delivery order
no_entry_inv	nomorentry invoice
no_inv	nomor invoice
tgl_inv	tanggal invoice
vendor	vendor
kode_vendor	kode vendor
kota_vendor	kota vendor
kode_customer	kode customer
nama_customer	nama customer
kota_customer	kota customer
ongkos_kirim	ongkos kirim
no_so	nomos sales order
kode_barang	kode barang
kode_sales	kode sales
qty	quantity
satuan	satuan
tgl_so	tanggal sales order
kode_produk	kode produk
nama_product	nama product
jenis	jenis product
harga_satuan	harga satuan
kode_pegawai	kode pegawai
jabatan	jabatan pegawai
target	target pegawai
tanggal_masuk	tanggal masuk pegawai



## 1

# Produktivitas Sales

Soal : Hitunglah produktivitas sales dari banyaknya SO yang didapatkan pada bulan Januari, dimana satu nomor SO dapat mengandung banyak barang.



## Query

```

2 • SELECT
3     p.nama_pegawai,
4     COUNT(DISTINCT s.no_so) AS jumlah_so,
5     p.target,
6     CASE
7         WHEN COUNT(DISTINCT s.no_so) < p.target THEN 'ya'
8         ELSE 'tidak'
9     END AS kurang_dari_target
10  FROM
11     tr_so s
12  LEFT JOIN
13     ms_pegawai p on s.kode_sales = p.kode_pegawai
14     AND month(s.tgl_so) = 01
15  GROUP BY
16     p.kode_pegawai
17  ORDER BY
18     p.nama_pegawai;
19

```

## Output

Result Grid    Filter Rows: <input type="text"/>   Export: 				
	nama_pegawai	jumlah_so	target	kurang_dari_target
▶	Andi	2	5	ya
	Ayu	2	2	tidak
	Budi	3	2	tidak
	Joko	1	2	ya
	Susi	3	2	tidak

- Column **jumlah\_so** memiliki nilai yang diperoleh dari jumlah column distinct no\_so
- Column **kurang\_dari\_target** terdiri dari 2 pernyataan yaitu "ya" atau "tidak" dengan syarat  $\text{target} > \text{jumlah\_so}$  dan menggunakan query kondisional CASE WHEN



## 2

# Laporan Delivery Order

Soal : Table yang dibutuhkan untuk membentuk report delivery order adalah table **tr\_do**, nilai value yang ditampilkan adalah **harga + PPn + ongkos jalan**. Catatan : PPn = 10%

Harga quantity pada SO harus dikonversi terlebih dahulu untuk mendapat nilai satuannya, lalu dikalikan dengan harga satuan produk dengan ketentuan berikut

Satuan	Qty
Krat	24
Dus	30
Botol	1

## Output

### Query

```

1  •  SELECT
2      d.no_do,
3      s.kode_customer,
4      d.tgl_do,
5      CASE
6          WHEN s.satuan = 'Krat' THEN (s.qty * 24)
7          WHEN s.satuan = 'Dus' THEN (s.qty * 30)
8          ELSE s.qty
9      END AS qty,
10     ROUND(SUM(
11         CASE
12             WHEN s.satuan = 'Krat' THEN ((s.qty * 24) * p.harga_satuan)
13             WHEN s.satuan = 'Dus' THEN ((s.qty * 30) * p.harga_satuan)
14             ELSE (s.qty * p.harga_satuan)
15         END * 1.1
16     ) + c.ongkos_kirim, 0) AS amount
17 FROM
18     tr_do d
19 LEFT JOIN
20     tr_so s ON d.no_entry_so = s.no_entry_so

```

```

21 INNER JOIN
22     ms_product p ON s.kode_barang = p.kode_produk
23 INNER JOIN
24     ms_customer c ON s.kode_customer = c.kode_customer
25 GROUP BY
26     d.no_do, s.kode_customer, d.tgl_do
27 ORDER BY
28     d.no_do;

```

no_do	kode_customer	tgl_do	qty	amount
DO001	C001	2018-01-03	36	500200
DO002	C002	2018-01-08	750	3530000
DO003	C001	2018-01-09	18	108160
DO004	C003	2018-01-10	930	3615500
DO005	C003	2018-01-15	1560	20627000
DO006	C002	2018-01-15	630	4916000
DO007	C001	2018-01-08	21	152050
DO008	C001	2018-01-08	240	1345000
DO009	C002	2018-01-12	48	329000
DO010	C004	2018-01-20	1224	6083800
DO011	C003	2018-01-27	76	453000
DO012	C004	2018-01-30	82	521100
DO013	C002	2018-01-20	192	1332200

- Column **qty** diperoleh dari hasil konversi nilai satuannya menggunakan query kondisional CASE WHEN
- Column **amount** diperoleh dari hasil perhitungan total ((konversi qty \* harga\_satuan) \* 1.1 ) + ongkos\_kirim menggunakan query kondisional CASE WHEN

### 3 Laporan Umur Utang Pelanggan




Soal : Utang pelanggan adalah DO yang belum terbayarkan di table invoice. Tentukan umur utang pertanggal 1 Februari 2018

Ketentuan : Urutkan baris berdasarkan kolom **aging** dari besar ke kecil, dan kolom **no\_do** dari kecil ke besar

#### Query

```
1 • SELECT
2     d.no_do,
3     nama_customer,
4     d.tgl_do,
5     '2018-02-01' AS date_measurement,
6     DATEDIFF('2018-02-01', d.tgl_do) AS aging
7 FROM
8     tr_inv i
9 RIGHT JOIN
10    tr_do d ON i.no_entry_do = d.no_entry_do
11 INNER JOIN
12    tr_so s ON d.no_entry_so = s.no_entry_so
13 INNER JOIN
14    ms_customer c ON s.kode_customer = c.kode_customer
15 WHERE
16     i.no_inv IS NULL
17 ORDER BY
18     aging DESC, d.no_do;
```

#### Output

Result Grid     Filter Rows: <input type="text"/>   Export:    Wrap Cell					
	no_do	nama_customer	tgl_do	date_measurement	aging
▶	DO001	Warung bu Indah	2018-01-03	2018-02-01	29
	DO008	Warung bu Indah	2018-01-08	2018-02-01	24
	DO009	Toko Budi	2018-01-12	2018-02-01	20
	DO010	Toko Bu Endah	2018-01-20	2018-02-01	12
	DO013	Toko Budi	2018-01-20	2018-02-01	12
	DO011	Toko Pak Imin	2018-01-27	2018-02-01	5

- Column **aging** diperoleh dari perbedaan hari antara date\_measurement dengan tgl\_do menggunakan fungsi DATEDIFF

## 4

# Tiga Produk Paling Laris Berdasarkan Kuantitas

Soal : Dengan menggunakan query dan table **tr\_so**, buatlah output dengan kolom **nama\_product** dan **qty**.

Ketentuan : Batasi record hanya 3 top produk saja, urutkan baris berdasarkan kolom **qty** dari besar ke kecil, dan nama kolom **nama\_product** dari kecil ke besar



## Query

```

1  SELECT
2      p.nama_product,
3      SUM(CASE
4          WHEN s.satuan = 'Krat' THEN s.qty * 24
5          WHEN s.satuan = 'Dus' THEN s.qty * 30
6          ELSE s.qty
7      END) AS qty
8  FROM
9      tr_so s
10 INNER JOIN
11     ms_product p ON s.kode_barang = p.kode_produk
12 GROUP BY
13     p.nama_product
14 ORDER BY
15     qty DESC,
16     nama_product ASC
17 LIMIT 3;

```

## Output

Result Grid     Filter Rows		
	nama_product	qty
▶	Soda Segar	4663
	Soda gembira ria	3846
	Teh daun pucuk	1224

- Column **qty** diperoleh dari total konversi nilai qty berdasarkan satuannya pada masing - masing nama\_product

5

## Tiga Vendor Paling Banyak Menjual

Soal : Dengan menggunakan query dan table tr\_inv, buatlah output dengan kolom **vendor** dan **amount**.

Ketentuan : Batasi record hanya 3 top vendor saja, urutkan baris berdasarkan kolom **amount** dari besar ke kecil, dan nama kolom **vendor** dari kecil ke besar



### Query

```

1 • SELECT
2     v.vendor,
3     SUM(
4         CASE
5             WHEN s.satuan = 'Krat' THEN s.qty * 24 * p.harga_satuan
6             WHEN s.satuan = 'Dus' THEN s.qty * 30 * p.harga_satuan
7             ELSE s.qty * p.harga_satuan
8         END
9     ) AS amount
10 FROM
11     tr_inv i
12 INNER JOIN
13     tr_do d ON i.no_entry_do = d.no_entry_do
14 INNER JOIN
15     tr_so s ON d.no_entry_so = s.no_entry_so
16 INNER JOIN
17     ms_product p ON s.kode_barang = p.kode_produk
18 INNER JOIN
19     ms_vendor v ON p.kode_vendor = v.kode_vendor
20 GROUP BY
21     v.vendor
22 ORDER BY
23     amount DESC,
24     v.vendor ASC
25 LIMIT 3;

```

### Output

Result Grid     Filter Rows: <input type="text"/>		
	vendor	amount
▶	PT Alam Maju	19286500
	PT Berjaya	4410000
	PT Maju Berkarya	3255000

- Column **amount** diperoleh dari total konversi nilai qty \* harga\_satuan menggunakan fungsi SUM dan CASE WHEN



# Thanks

## Lets Connect



**saffarahmatullah@gmail.com**



**<https://linkedin.com/in/saffarahmatullah>**



**<https://github.com/RSaff>**