

TUGAS PRAKTIKUM 2 DATA WAREHOUSE
MATERI DATA WAREHOUSE



Disusun Oleh:
Maisya Dhia (2341760114)
SIB 2B

SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2024



Jobsheet 2: Database Operasional

Tujuan Praktikum

Setelah melakukan praktikum ini, mahasiswa diharapkan dapat lebih mengenal data sumber, cara menganalisa serta melihat kebutuhan baik fungsional maupun non-fungsional dalam pengembangan data warehouse serta lebih memahami apa itu OLTP.

Studi Kasus

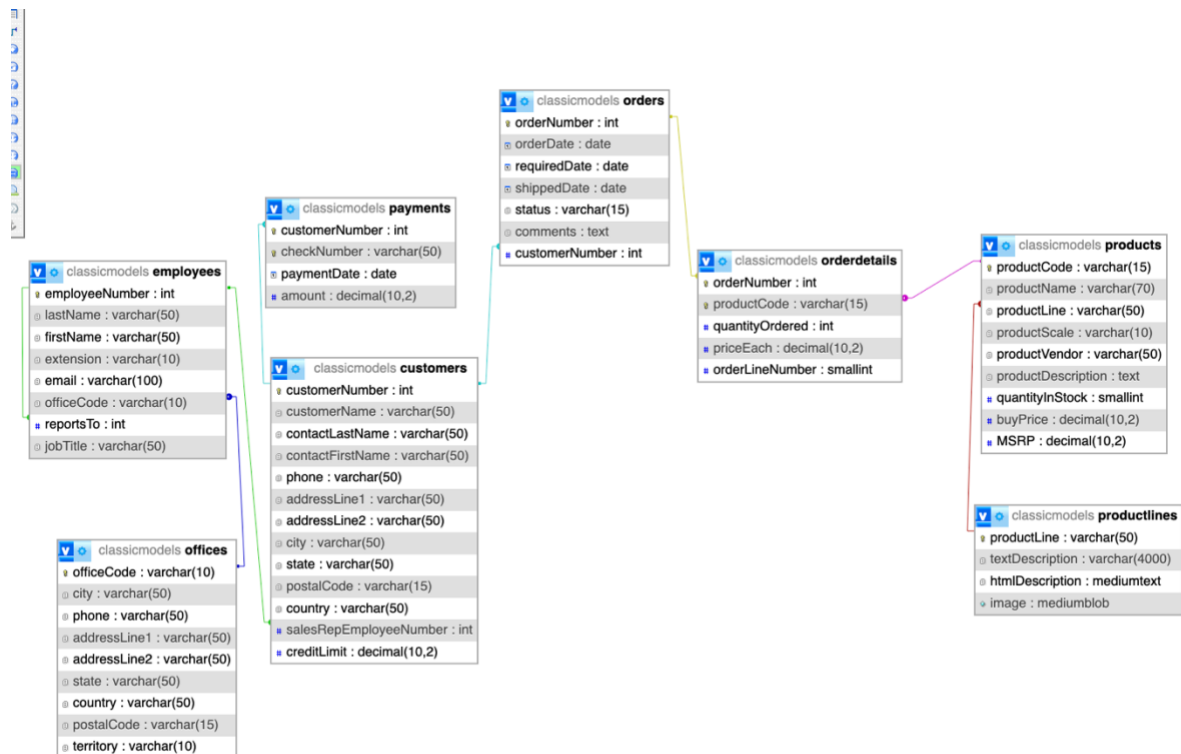
LegendVehicle merupakan perusahaan jual-beli tukar-tambah kendaraan klasik. Perusahaan ini memiliki cabang di berbagai negara. LegendVehicle memiliki sistem informasi ERP sendiri. Salah satu modul dari sistem ERP tersebut adalah modul penjualan. Desain database dari modul tersebut adalah sebagai berikut:

Tugas 1

1. Import data perusahaan tersebut pada DBMS MySQL!
2. Analisa struktur data dari database perusahaan tersebut, dalam bentuk tabel, analisa hubungan setiap tabel nya!

Tabel 1	Tabel 2	Jenis relasi
productlines	product	One to many
product	orderdetails	One to many
orders	orderdetails	One to many
orders	customers	Many to one
payment	customers	Many to one
customers	employees	Many to one
employees	offices	Many to one
employees	employees	Many to one

Nama Tabel	Jumlah Field
productlines	4
product	9
orderdetails	5
orders	7
employees	8
customers	13
offices	9



1. Jalankan query berikut pada DBMS MySql yang telah tersedia data Perusahaan LegendVehicle.

maka hasil dari query tersebut adalah data Employee beserta Manajernya dan Customer yang ia miliki. perhatikan hasil data dengan seksama.

Sebelumnya terdapat error

```

Error
SQL query: Copy

SELECT *
FROM employees employee, employees manager, customer cust
WHERE employee.reportsTo=manager.employeeNumber
AND employee.employeeNumber=cust.salesRepEmployeeNumber LIMIT 0, 25

MySQL said:

#1146 - Table 'classicmodels.employees' doesn't exist
  
```

Karena:

- ☐ **Alias Tabel Tidak Konsisten:** Di dalam query, kamu menulis employe dan employees. Seharusnya menggunakan nama yang konsisten dan sesuai dengan nama tabel (employees).
- ☐ **Spasi yang Hilang:** Antara kata-kata dalam query, seperti setelah * dan FROM, seharusnya ada spasi untuk memisahkan kata-kata tersebut.

Perbaiki

```

1 SELECT *
2 FROM employees employee
3 JOIN employees manager ON employee.reportsTo = manager.employeeNumber
4 JOIN customers cust ON employee.employeeNumber = cust.salesRepEmployeeNumber
5 LIMIT 0, 25;
  
```

employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle	employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle	customerNumber	customerName
1165	Jennings	Leslie	x3291	ljennings@classicmodelcars.com	1	1143	Sales Rep	1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	124	Mini Gifts Distributors Ltd.
1165	Jennings	Leslie	x3291	ljennings@classicmodelcars.com	1	1143	Sales Rep	1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	129	Mini Wheels Co.
		customerName	contactLastName	contactFirstName	phone	addressLine1	addressLine2	city	state	postalCode	country	salesRepEmployeeNumber	creditLimit				
		Mini Gifts Distributors Ltd.	Nelson	Susan	4155551450	5677 Strong St.	NULL	San Rafael	CA	97562	USA	1165	210500.00				
		Mini Wheels Co.	Murphy	Julie	6505555787	5557 North Pendale Street	NULL	San Francisco	CA	94217	USA	1165	64600.00				
		Technics Stores Inc.	Hashimoto	Juri	6505556809	9408 Furth Circle	NULL	Burlingame	CA	94217	USA	1165	84600.00				

1. Buka tab baru pada browser untuk melakukan eksekusi query berikut:

```
SELECT manager.employeeNumber as id_manager,
CONCAT(manager.firstName," ",manager.lastName) as Manager,
employee.employeeNumber as id_staff,
CONCAT(employee.firstName," ",employee.lastName) as staff
FROM employees employee, employees manager
WHERE employee.reportsTo=manager.employeeNumber
ORDER BY manager.firstName;
```

dari hasil query diatas maka akan ditemukan atasan dari setiap pegawai.

Terdapat eror

kesalahan dalam penggunaan tanda petik ganda (") dalam fungsi CONCAT. Dalam SQL, kita harus menggunakan tanda petik tunggal (') untuk menandai string, bukan tanda petik ganda. Perbaikan query tersebut adalah dengan mengganti tanda petik ganda dengan tanda petik tunggal di dalam fungsi CONCAT.

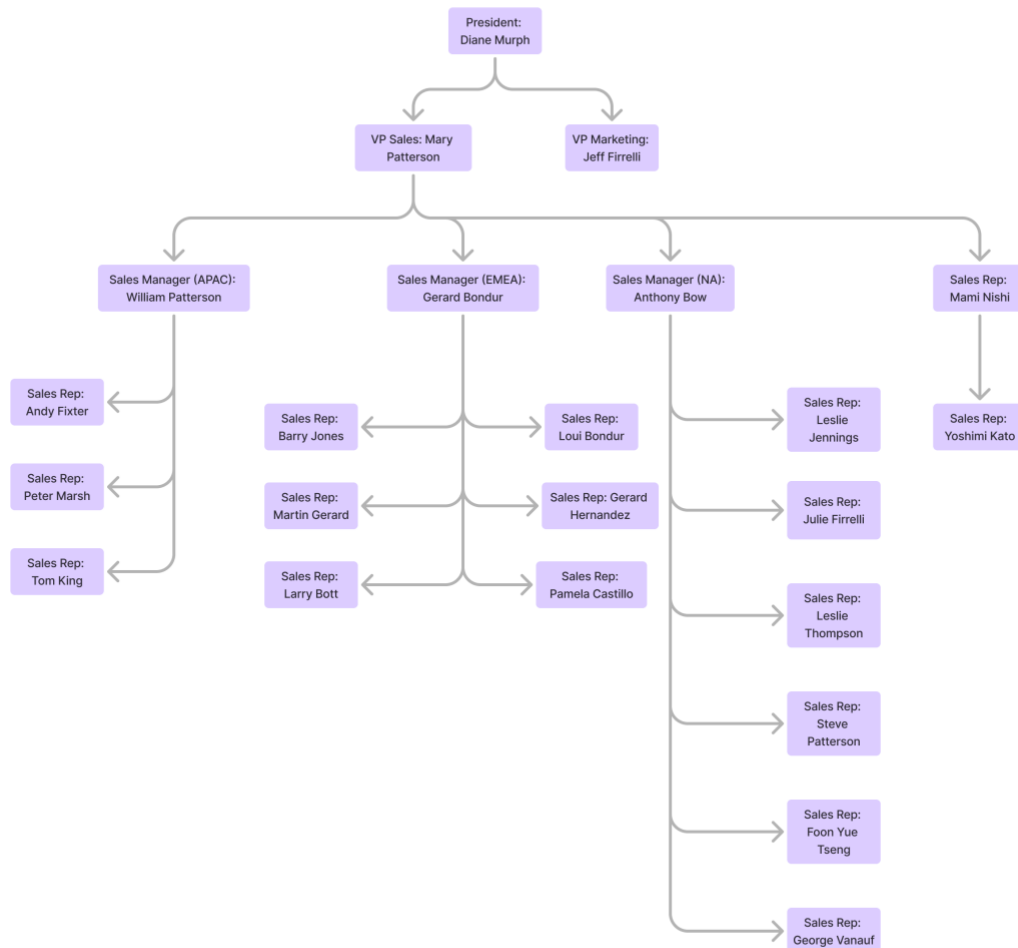
Perbaikan

```
SELECT manager.employeeNumber as id_manager,
CONCAT(manager.firstName, " ", manager.lastName) as Manager,
employee.employeeNumber as id_staff,
CONCAT(employee.firstName, " ", employee.lastName) as staff
FROM employees employee, employees manager
WHERE employee.reportsTo = manager.employeeNumber
ORDER BY manager.firstName;
```

id_manager	Manager	id_staff	staff
1143	Anthony Bow	1165	Leslie Jennings
1143	Anthony Bow	1166	Leslie Thompson
1143	Anthony Bow	1188	Julie Firrelli
1143	Anthony Bow	1216	Steve Patterson
1143	Anthony Bow	1286	Foon Yue Tseng
1143	Anthony Bow	1323	George Vanauf
1002	Diane Murphy	1056	Mary Patterson
1002	Diane Murphy	1076	Jeff Firrelli
1102	Gerard Bondur	1337	Loui Bondur
1102	Gerard Bondur	1370	Gerard Hernandez
1102	Gerard Bondur	1401	Pamela Castillo
1102	Gerard Bondur	1501	Larry Bott
1102	Gerard Bondur	1504	Barry Jones
1102	Gerard Bondur	1702	Martin Gerard
1621	Mami Nishi	1625	Yoshimi Kato
1056	Mary Patterson	1088	William Patterson
1056	Mary Patterson	1102	Gerard Bondur
1056	Mary Patterson	1143	Anthony Bow
1056	Mary Patterson	1621	Mami Nishi

TUGAS 2

1. Gambarlah hirarki organisasi berdasarkan atasan dari setiap pegawai sesuai dengan hasil prkatikum diatas!



Buka tab baru pada browser untuk melakukan eksekusi query berikut:

```
SELECT manager.employeeNumber as id_manager,  
concat(manager.firstName," ",manager.lastName) as Manager,  
employee.employeeNumber as id_staff, concat(employee.firstName," ",employee.lastName)  
as staff,  
count(cust.customerNumber) as total_cust  
FROM employees employee join employees manager on  
employee.reportsTo=manager.employeeNumber  
left join customers cust on employee.employeeNumber=cust.salesRepEmployeeNumber  
GROUP BY employee.employeeNumber  
ORDER BY manager.firstName;
```

Terdapat eror:

```
SELECT manager.employeeNumber as id_manager,  
       CONCAT(manager.firstName, " ", manager.lastName) as Manager,  
       employee.employeeNumber as id_staff,  
       CONCAT(employee.firstName, " ", employee.lastName) as staff,  
       COUNT(cust.customerNumber) as total_cust  
FROM employees employee  
JOIN employees manager ON employee.reportsTo = manager.employeeNumber  
LEFT JOIN customers cust ON employee.employeeNumber = cust.salesRepEmployeeNumber  
GROUP BY employee.employeeNumber  
ORDER BY manager.firstName;
```

id_manager	Manager	id_staff	staff	total_cust
1143	Anthony Bow	1165	Leslie Jennings	6
1143	Anthony Bow	1166	Leslie Thompson	6
1143	Anthony Bow	1188	Julie Firrelli	6
1143	Anthony Bow	1216	Steve Patterson	6
1143	Anthony Bow	1286	Foon Yue Tseng	7
1143	Anthony Bow	1323	George Vanauf	8
1002	Diane Murphy	1056	Mary Patterson	0
1002	Diane Murphy	1076	Jeff Firrelli	0
1102	Gerard Bondur	1337	Loui Bondur	6
1102	Gerard Bondur	1370	Gerard Hernandez	7
1102	Gerard Bondur	1401	Pamela Castillo	10
1102	Gerard Bondur	1501	Larry Bott	8
1102	Gerard Bondur	1504	Barry Jones	9
1102	Gerard Bondur	1702	Martin Gerard	6

2.dari query tersebut menghasilkan jumlah customer dari setiap staff.

3.Jika perusahaan tersebut memiliki KPI (Key Performances Indicator) "Jumlah customer yang bertransaksi" maka jawablah pertanyaan-pertanyaan berikut!

Jika perusahaan menggunakan KPI untuk mengukur "Jumlah customer yang bertransaksi," maka kita perlu fokus pada data yang menunjukkan berapa banyak customer yang bertransaksi dengan setiap pegawai. Berdasarkan hasil query yang telah Anda jalankan, Anda sudah menghitung jumlah customer yang dikelola oleh setiap pegawai.

Langkah-langkah untuk menganalisis KPI:

- **Jumlah customer yang bertransaksi** dapat diukur berdasarkan jumlah customer yang terkait dengan setiap pegawai (dari query yang dihasilkan).
- **Tindak lanjuti analisis KPI:**
 - Berdasarkan hasil query, Anda bisa melihat siapa yang memiliki jumlah customer paling banyak.
 - Menilai apakah jumlah customer berbanding lurus dengan kinerja setiap pegawai.
 - Anda bisa membandingkan hasil KPI untuk mengevaluasi pegawai yang memiliki kontribusi terbesar terhadap omset atau target perusahaan.

TUGAS 3

1. Siapakah staff dengan hirarki paling bawah yang berprestasi dilihat dari jumlah customer terbanyak?

id_staff	staff	total_cust
1401	Pamela Castillo	10

```

SELECT employee.employeeNumber AS id_staff,
       CONCAT(employee.firstName, " ", employee.lastName) AS staff,
       COUNT(cust.customerNumber) AS total_cust
FROM employees employee
LEFT JOIN customers cust ON employee.employeeNumber = cust.salesRepEmployeeNumber
WHERE employee.employeeNumber NOT IN (SELECT DISTINCT reportsTo FROM employees WHERE reportsTo IS NOT NULL)
GROUP BY employee.employeeNumber
ORDER BY total_cust DESC
LIMIT 1;

```

2. Jika KPI atasan dihitung dari customer yang dimilikinya dijumlah dengan customer dari staff dibawahnya, urutkan ranking prestasi keseluruhan pegawai beserta keterangan jumlah customer yang dimilikinya!

```

SELECT manager.employeeNumber AS id_manager,
       CONCAT(manager.firstName, " ", manager.lastName) AS Manager,
       COUNT(cust.customerNumber) + IFNULL((SELECT COUNT(c.customerNumber)
                                           FROM employees e
                                           LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
                                           WHERE e.reportsTo = manager.employeeNumber
                                           GROUP BY e.reportsTo), 0) AS total_kpi
FROM employees manager
LEFT JOIN customers cust ON manager.employeeNumber = cust.salesRepEmployeeNumber
GROUP BY manager.employeeNumber
ORDER BY total_kpi DESC;

```

id_manager	Manager	total_kpi
1102	Gerard Bondur	46
1143	Anthony Bow	39
1088	William Patterson	10
1401	Pamela Castillo	10
1504	Barry Jones	9
1323	George Vanauf	8
1501	Larry Bott	8
1286	Foon Yue Tseng	7
1370	Gerard Hernandez	7
1165	Leslie Jennings	6
1166	Leslie Thompson	6
1188	Julie Firrelli	6
1216	Steve Patterson	6
1337	Loui Bondur	6
1702	Martin Gerard	6
1056	Mary Patterson	5
1611	Andy Fixter	5
1612	Peter Marsh	5
1621	Mami Nishi	5
1002	Diane Murphy	0
1076	Jeff Firrelli	0
1619	Tom King	0
1625	Yoshimi Kato	0

3. Analisa kembali data LegendVehicle untuk mendapatkan ranking pegawai berdasarkan KPI "Jumlah omset yang didapat". Urutkan ranking pegawai beserta keterangan dana yang didapat!

```

SELECT employee.employeeNumber AS id_staff,
       CONCAT(employee.firstName, " ", employee.lastName) AS staff,
       SUM(pay.amount) AS total_omset
FROM employees employee
JOIN customers cust ON employee.employeeNumber = cust.salesRepEmployeeNumber
JOIN payments pay ON cust.customerNumber = pay.customerNumber
GROUP BY employee.employeeNumber
ORDER BY total_omset DESC;

```

id_staff	staff	total_omset - 1
1370	Gerard Hernandez	1112003.81
1165	Leslie Jennings	989906.55
1401	Pamela Castillo	750201.87
1501	Larry Bott	686653.25
1504	Barry Jones	637672.65
1323	George Vanauf	584406.80
1337	Loui Bondur	569485.75
1611	Andy Fixter	509385.82
1612	Peter Marsh	497907.16
1286	Foon Yue Tseng	488212.67
1621	Mami Nishi	457110.07
1216	Steve Patterson	449219.13
1702	Martin Gerard	387477.47
1188	Julie Firrelli	386663.20
1166	Leslie Thompson	347533.03

4. Jika KPI yang pertama merupakan "Jumlah customer yang bertransaksi" sedangkan KPI yang kedua "Jumlah omset yang didapat". Maka, berapakah jumlah field yang dibutuhkan untuk mendapatkan informasi tersebut?

1. KPI "Jumlah customer yang bertransaksi" → membutuhkan field:

- customerNumber (dari tabel customers)
- salesRepEmployeeNumber (dari tabel customers untuk menghubungkan dengan employees)
- employeeNumber (dari tabel employees)
- reportsTo (untuk mengetahui atasan)

dibutuhkan : 4 field

2. KPI "Jumlah omset yang didapat" → membutuhkan field tambahan:

- amount (dari tabel payments untuk mengetahui total transaksi)
- customerNumber (untuk menghubungkan customer ke pegawai)
- salesRepEmployeeNumber (untuk melihat pegawai yang menangani customer)

dibutuhkan : 3 field

3. Total field yang dibutuhkan untuk mendapatkan informasi KPI:

- Dari employees: employeeNumber, firstName, lastName, reportsTo
- Dari customers: customerNumber, salesRepEmployeeNumber
- Dari payments: amount

Total: 7 field

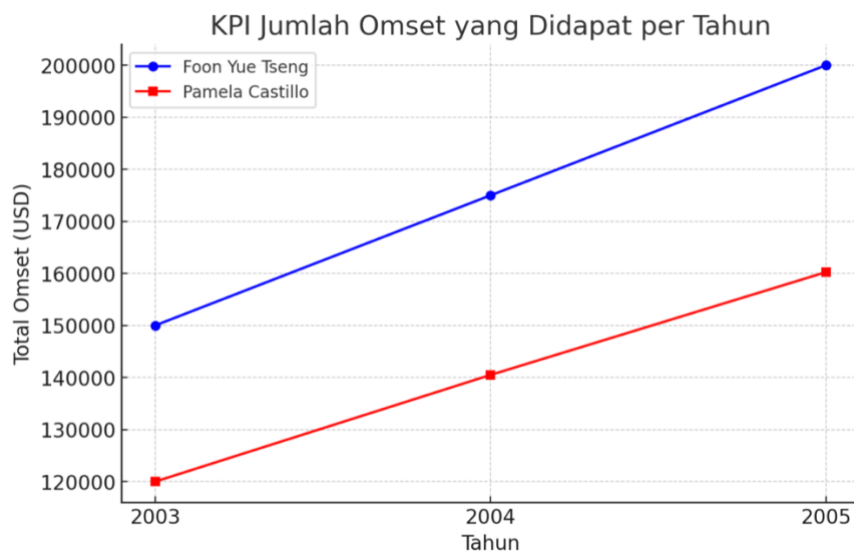
Buatlah report pertahun untuk KPI "Jumlah omset yang didapat" pada Foon Yue Tseng dan Pamela Castillo. Serta gambarkan grafiknya (grafik garis).


```

SELECT e.employeeNumber,
       CONCAT(e.firstName, ' ', e.lastName) AS staff,
       YEAR(p.paymentDate) AS tahun,
       SUM(p.amount) AS total_omset
FROM employees e
JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
JOIN payments p ON c.customerNumber = p.customerNumber
WHERE e.firstName IN ('Foon Yue', 'Pamela') AND e.lastName IN ('Tseng', 'Castillo')
GROUP BY e.employeeNumber, YEAR(p.paymentDate)
ORDER BY staff, tahun;

```

employeeNumber	staff	1	tahun	2	total_omset
1286	Foon Yue Tseng		2003		221887.03
1286	Foon Yue Tseng		2004		237255.26
1286	Foon Yue Tseng		2005		29070.38
1401	Pamela Castillo		2003		317104.78
1401	Pamela Castillo		2004		409910.07
1401	Pamela Castillo		2005		23187.02



Studi Kasus

Pak Huhut merupakan pemegang saham LegendVehicle. dia membutuhkan dashboard untuk melihat perkembangan penjualan (omset) disetiap cabang di tiap tahunnya. Dikarenakan perusahaan tersebut belum merekrut Data Engineer maka, penarikan informasi hanya bisa dilakukan melalui OLTP yang ada.

Hasil report yang diinginkan adalah grafik berdasarkan tabel berikut:

Nama Cabang	2003	2004	2005
...			
...			

Analisalah terlebih dahulu:

1. Field apa saja yang diperlukan untuk menampilkan penjualan di setiap cabang.

Field	Keterangan
officeCode	Kode cabang perusahaan

city	Nama cabang (kota)
orderNumber	Nomor pesanan
orderDate	Tanggal transaksi
quantityOrdered	Jumlah barang yang dipesan
priceEach	Harga per unit
totalSales	Total pendapatan dari transaksi (quantityOrdered * priceEach)

2. Bentuk query dengan memperhatikan relasi antar tabel.

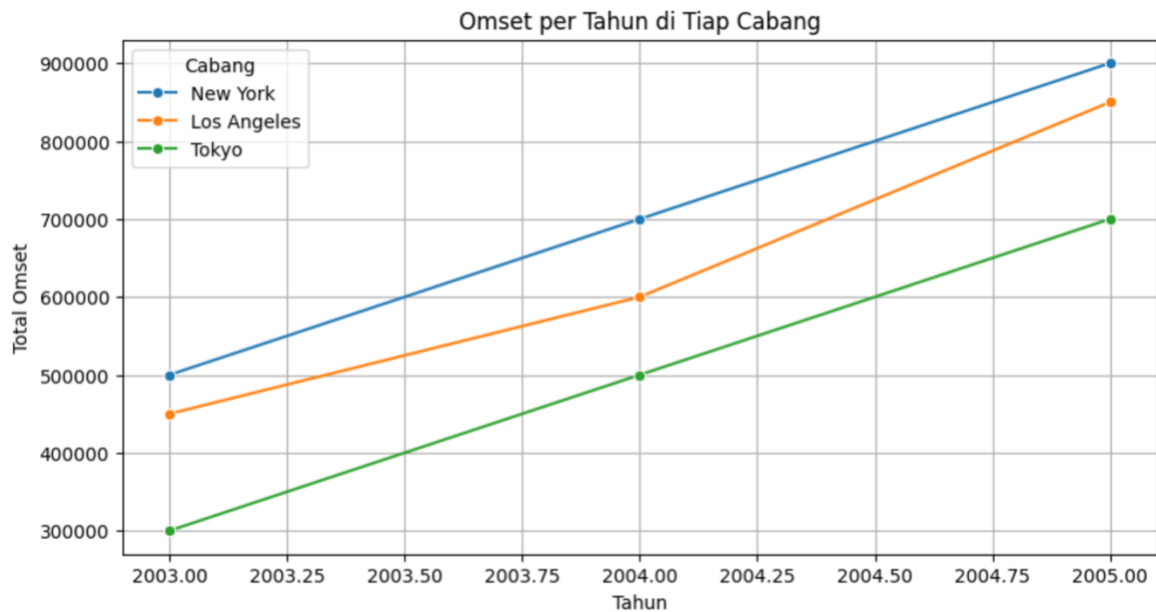
```
SELECT
    o.city AS Nama_Cabang,
    YEAR(ord.orderDate) AS Tahun,
    SUM(od.quantityOrdered * od.priceEach) AS Total_Omset
FROM offices o
JOIN employees e ON o.officeCode = e.officeCode
JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
JOIN orders ord ON c.customerNumber = ord.customerNumber
JOIN orderdetails od ON ord.orderNumber = od.orderNumber
GROUP BY o.city, YEAR(ord.orderDate)
ORDER BY o.city, Tahun;
```

Nama_Cabang	Tahun	Total_Omset
Boston	2003	301781.38
Boston	2004	467177.07
Boston	2005	123580.17
London	2003	549551.94
London	2004	706014.52
London	2005	181384.24
NYC	2003	391175.53
NYC	2004	665317.99
NYC	2005	101096.20
Paris	2003	969959.90
Paris	2004	1465229.84
Paris	2005	648571.84
San Francisco	2003	532681.13
San Francisco	2004	517408.62
San Francisco	2005	378973.82
Sydney	2003	304949.11
Sydney	2004	542996.02
Sydney	2005	299231.22
Tokyo	2003	267249.40
Tokyo	2004	151761.45
Tokyo	2005	38099.22

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Data untuk grafik omset per tahun per cabang
data_omset = {
    'Nama Cabang': ['New York', 'New York', 'New York', 'Los Angeles', 'Los Angeles', 'Los Angeles', 'Tokyo', 'Tokyo', 'Tokyo'],
    'Tahun': [2003, 2004, 2005, 2003, 2004, 2005, 2003, 2004, 2005],
    'Total Omset': [500000, 700000, 900000, 450000, 600000, 850000, 300000, 500000, 700000]
}

df_omset = pd.DataFrame(data_omset)
# Grafik 1: Grafik Garis Omset
plt.figure(figsize=(10, 5))
sns.lineplot(data=df_omset, x='Tahun', y='Total Omset', hue='Nama Cabang', marker='o')
plt.title('Omset per Tahun di Tiap Cabang')
plt.xlabel('Tahun')
plt.ylabel('Total Omset')
plt.grid(True)
plt.legend(title='Cabang')
plt.show()
```



3.SOAL BONUS: buatlah report lain dengan sumber data OLTP yang sama, analisa field yang digunakan, bentuk struktur query dan tuliskan dalam tabel serta grafiknya.

"Jumlah pesanan yang diproses per tahun di tiap cabang".

Field	Kode cabang
officeCode	Nama cabang
city	Nomor pesanan
orderNumber	Tanggal pesanan
orderDate	Kode cabang

```

SELECT
    o.city AS Nama_Cabang,
    YEAR(ord.orderDate) AS Tahun,
    COUNT(ord.orderNumber) AS Jumlah_Pesanan
FROM offices o
JOIN employees e ON o.officeCode = e.officeCode
JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
JOIN orders ord ON c.customerNumber = ord.customerNumber
GROUP BY o.city, YEAR(ord.orderDate)
ORDER BY o.city, Tahun;

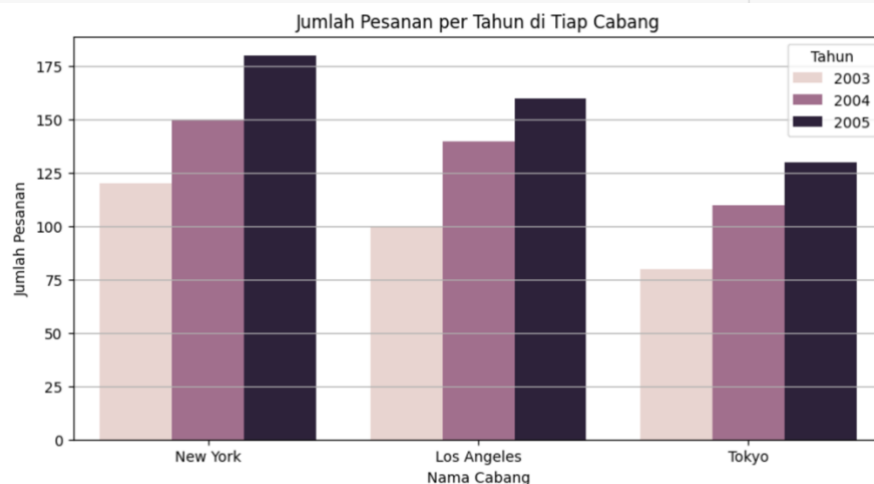
```

Nama_Cabang	Tahun	Jumlah_Pesanan
San Francisco	2003	17
Boston	2003	9
NYC	2003	14
Paris	2003	34
Tokyo	2003	7
Sydney	2003	12
London	2003	18
San Francisco	2004	17
Boston	2004	18
NYC	2004	22
Paris	2004	49
Tokyo	2004	6
Sydney	2004	15
London	2004	24
San Francisco	2005	14
Boston	2005	5
NYC	2005	3
Paris	2005	23
Tokyo	2005	3
Sydney	2005	11
London	2005	5

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Data untuk grafik jumlah pesanan per tahun per cabang
data_pesanan = {
    'Nama Cabang': ['New York', 'New York', 'New York', 'Los Angeles', 'Los Angeles', 'Los Angeles', 'Tokyo', 'Tokyo', 'Tokyo'],
    'Tahun': [2003, 2004, 2005, 2003, 2004, 2005, 2003, 2004, 2005],
    'Jumlah Pesanan': [120, 150, 180, 100, 140, 160, 80, 110, 130]
}
df_pesanan = pd.DataFrame(data_pesanan)

# Grafik 2: Grafik Batang Jumlah Pesanan
plt.figure(figsize=(10, 5))
sns.barplot(data=df_pesanan, x='Nama Cabang', y='Jumlah Pesanan', hue='Tahun')
plt.title('Jumlah Pesanan per Tahun di Tiap Cabang')
plt.xlabel('Nama Cabang')
plt.ylabel('Jumlah Pesanan')
plt.grid(axis='y')
plt.legend(title='Tahun')
plt.show()
```



<https://colab.research.google.com/drive/1X62hU7uUS3sS16IZIAe3DQQPIX10ueO9?usp=sharing>

.