

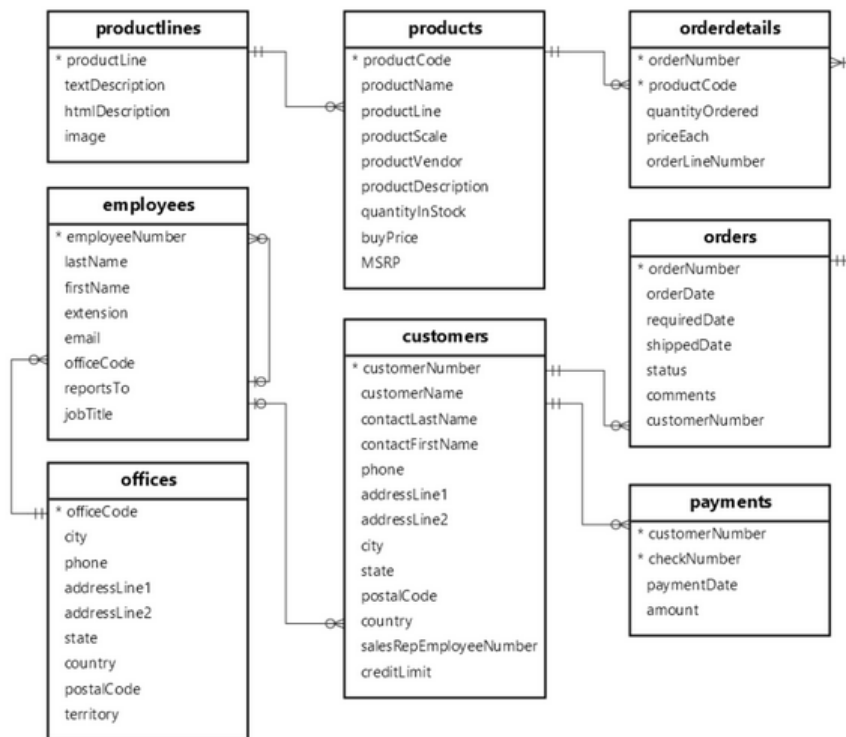
DATA WAREHOUSE

JOBSHEET 2

DATABASE OPERATIONAL

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:

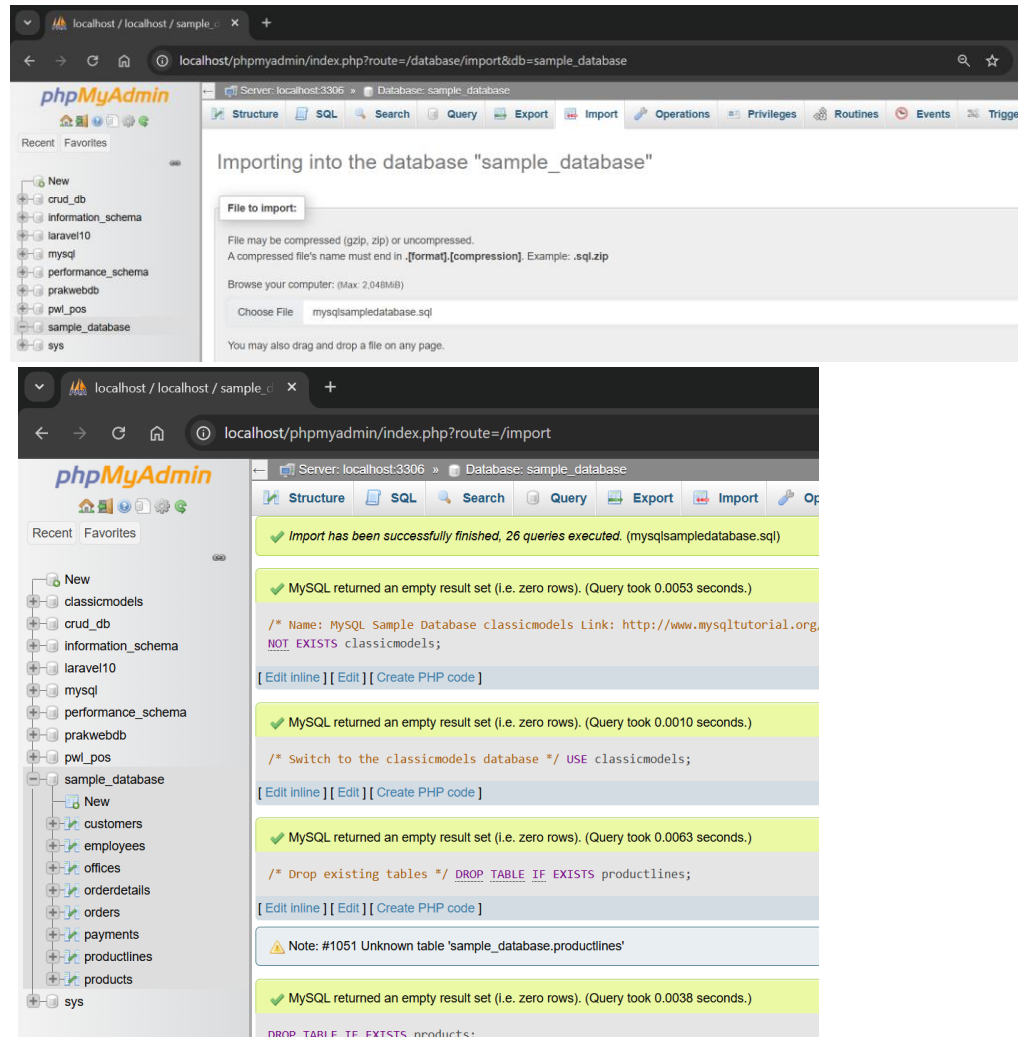


Selain itu proses penjualan kendaraan pada perusahaan tersebut bukan hanya melalui showroom cabang, melainkan reseller-reseller bebas lainnya.

Tugas 1

1. Import data perusahaan tersebut pada DBMS MySQL!

⇒ Import data ke mysql



2. Analisa struktur data dari database perusahaan tersebut, dalam bentuk tabel, analisa hubungan setiap tabel nya!

⇒ Relation tiap table

| Table 1 | Table 2 | Relation |
|--------------|--------------|---|
| productlines | products | One to many (1 productline punya banyak product) |
| products | orderdetails | One to many (1 products bisa muncul di banyak order) |
| orders | orderdetails | One to many (1 order bisa punya banyak detail barang) |

| | | |
|-----------|-----------|---|
| customers | orders | One to many (1 customer bisa punya banyak order) |
| employees | customers | One to many (1 employee, bisa handle banyak customer) |
| employees | employees | Self join (karyawan memiliki atasan) |
| offices | employees | One to many (1 kantor memiliki banyak karyawan) |

3. Analisa jumlah field pada setiap tabel!

⇒ Jumlah field pada setiap table

| Table | Jumlah Field |
|--------------|--------------|
| productlines | 4 |
| products | 9 |
| orders | 7 |
| orderdetails | 5 |
| customers | 13 |
| employees | 8 |
| offices | 9 |

PRAKTIKUM 1

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

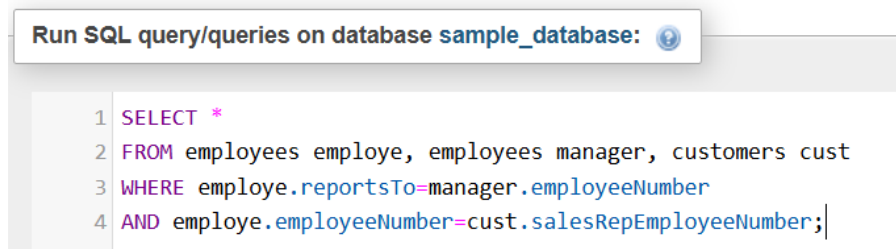
```
1 SELECT *
2 FROM employees employe, employes manager, customer cust
3 WHERE employee.reportsTo=manager.employeeNumber
4 AND employee.employeeNumber=cust.salesRepEmployeeNumber;
```

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

⇒ Hasil execute query



Perbaiki



Kesalahan pada query sebelumnya adalah pada penulisan employee dan customers yang salah

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

```
1 SELECT manager.employeeNumber as id_manager,
2 CONCAT(manager.firstName," ",manager.lastName) as Manager,
3 employee.employeeNumber as id_staff,
4 CONCAT(employee.firstName," ",employee.lastName) as staff
5 FROM employees employee, employees manager
6 WHERE employee.reportsTo=manager.employeeNumber
7 ORDER BY manager.firstName;
```

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

⇒ Error, perbaikan query ;

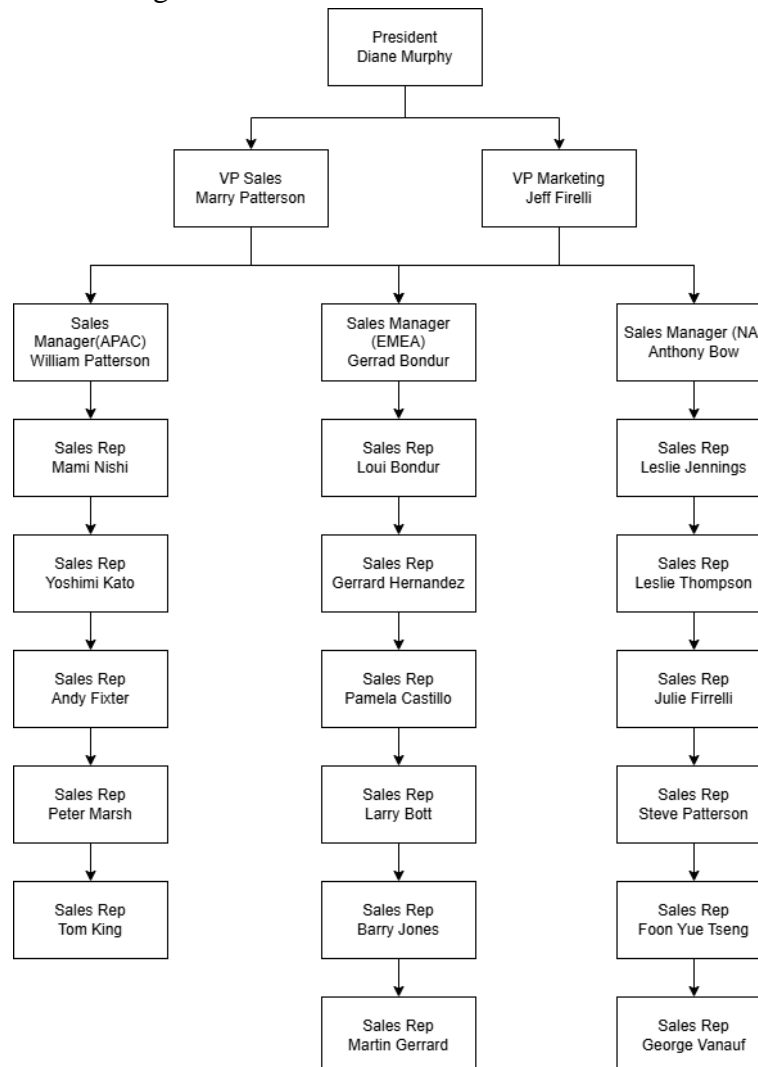
Run SQL query/queries on database sample_database: ?

```
1 SELECT manager.employeeNumber AS id_manager,  
2 CONCAT(manager.firstName, " ", manager.lastName) AS Manager,  
3 employee.employeeNumber AS id_staff,  
4 CONCAT(employee.firstName, " ", employee.lastName) AS staff  
5 FROM employees employee, employees manager  
6 WHERE employee.reportsTo = manager.employeeNumber  
7 ORDER BY manager.firstName;
```

Tugas 2

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

⇒ Hirearki organisasi



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

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

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

⇒ Terjadi error

Perbaiki;

```
1 SELECT manager.employeeNumber as id_manager,  
2 CONCAT(manager.firstName, " ", manager.lastName) as Manager,  
3 employee.employeeNumber as id_staff,  
4 CONCAT(employee.firstName, " ", employee.lastName) as staff,  
5 COUNT(cust.customerNumber) as total_cust  
6 FROM employees employee  
7 JOIN employees manager ON employee.reportsTo = manager.employeeNumber  
8 LEFT JOIN customers cust ON employee.employeeNumber = cust.salesRepEmployeeNumber  
9 GROUP BY employee.employeeNumber  
0 ORDER BY manager.firstName;
```

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

Tugas 3

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

⇒ Pamela Castillo, employee number : 1401, with 10 total customers

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!

⇒ Rank KPI atasan

| Rank | Employee | Total Customer | Keterangan |
|------|-------------------|----------------|--------------------------------------|
| 1 | Gerrard Bondur | 46 | 6 bawahan (6 + 7 + 10 + 8 + 9 + 6) |
| 2 | Anthony Bow | 33 | 6 Bawahan (6 + 6 + 0 + 6 + 7 + 8) |
| 3 | William Patterson | 15 | 5 bawahan (5 + 0 + 5 + 5 + 0) |

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

⇒ Query

```

1 SELECT
2     e.employeeNumber,
3     CONCAT(e.firstName, " ", e.lastName) AS employee_name,
4     SUM(od.quantityOrdered * od.priceEach) AS total_sales
5 FROM employees e
6 JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
7 JOIN orders o ON c.customerNumber = o.customerNumber
8 JOIN orderdetails od ON o.orderNumber = od.orderNumber
9 GROUP BY e.employeeNumber
10 ORDER BY total_sales DESC;

```

Hasil eexecute

| employeeNumber | employee_name | total_sales | 1 |
|----------------|------------------|-------------|---|
| 1370 | Gerard Hernandez | 1258577.81 | |
| 1165 | Leslie Jennings | 1081530.54 | |
| 1401 | Pamela Castillo | 868220.55 | |
| 1501 | Larry Bott | 732096.79 | |
| 1504 | Barry Jones | 704853.91 | |
| 1323 | George Vanauf | 669377.05 | |
| 1612 | Peter Marsh | 584593.76 | |
| 1337 | Loui Bondur | 569485.75 | |
| 1611 | Andy Fixter | 562582.59 | |
| 1216 | Steve Patterson | 505875.42 | |
| 1286 | Foon Yue Tseng | 488212.67 | |
| 1621 | Mami Nishi | 457110.07 | |
| 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?

⇒ KPI 1 (Jumlah customer): cukup pakai **nomor customer** (1 field).

KPI 2 (Jumlah omset): perlu tahu:

- Berapa barang yang dibeli? (quantityOrdered)
- Berapa harga per barang? (priceEach)

Total: **3 field** (customerNumber, quantityOrdered, priceEach)

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

⇒ Step yang diperlukan untuk membuat report

- Filter pegawai berdasarkan nama.
- Ambil transaksi mereka (join customers → orders → orderdetails).
- Group berdasarkan tahun.
- Hitung total omset per tahun

Query

```

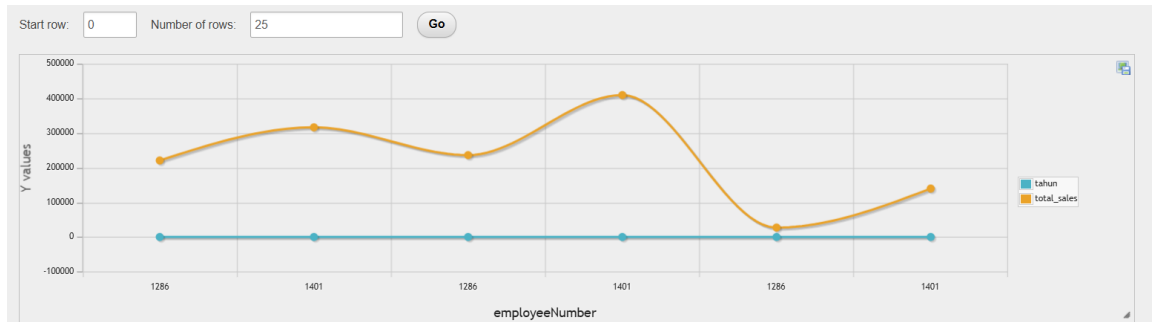
1 SELECT e.employeeNumber, e.firstName, e.lastName, YEAR(o.orderDate) AS tahun,
2 SUM(od.quantityOrdered * od.priceEach) AS total_sales
3 FROM employees e
4 JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
5 JOIN orders o ON c.customerNumber = o.customerNumber
6 JOIN orderdetails od ON o.orderNumber = od.orderNumber
7 WHERE e.firstName IN ('Foon Yue', 'Pamela') AND e.lastName IN ('Tseng', 'Castillo')
8 GROUP BY e.employeeNumber, tahun
9 ORDER BY tahun, e.firstName;

```

Hasil execute

| employeeNumber | firstName | lastName | tahun | total_sales |
|----------------|-----------|----------|-------|-------------|
| 1286 | Foon Yue | Tseng | 2003 | 221887.03 |
| 1401 | Pamela | Castillo | 2003 | 317104.78 |
| 1286 | Foon Yue | Tseng | 2004 | 237255.26 |
| 1401 | Pamela | Castillo | 2004 | 409910.07 |
| 1286 | Foon Yue | Tseng | 2005 | 29070.38 |
| 1401 | Pamela | Castillo | 2005 | 141205.70 |

Chart



Penjelasan grafik

Performa Karyawan 1286 (Foon Yue Tseng)

- Tahun 2003: Penjualan sebesar 221,887.03.
- Tahun 2004: Penjualan meningkat menjadi 237,255.26.
- Tahun 2005: Penjualan turun drastis menjadi 29,070.38.

Performa Karyawan 1401 (Pamela Castillo)

- Tahun 2003: Penjualan sebesar 317,104.78.
- Tahun 2004: Penjualan meningkat signifikan menjadi 409,910.07.
- Tahun 2005: Penjualan turun menjadi 141,205.70.

Grafik ini menunjukkan pola "peak and drop," yaitu kinerja puncak di tengah periode (2004) diikuti oleh penurunan drastis di akhir periode (2005). Penurunan ini kemungkinan dipengaruhi oleh faktor eksternal seperti strategi perusahaan atau kondisi pasar.

Studi Kasus

Pak Huhut merupakan pemegang saham LegendVehicle. dia membutuhkan dashboard untuk melihat perkembangan penjualan (omset) di setiap 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 yang diperlukan

| Field | Keterangan |
|-------------|---------------------------|
| branchName | Nama cabang |
| orderDate | Tanggal / Waktu transaksi |
| orderAmount | Nilai total transaksi |
| branchID | Id tiap cabang |

2. Bentuk query dengan memperhatikan relasi antar tabel.

⇒ Query

```
Run SQL query/queries on database sample_database: ⓘ  
  
1 SELECT o.city AS branchName,  
2 YEAR(p.paymentDate) AS Tahun,  
3 SUM(p.amount) AS Total_Omset  
4 FROM payments p JOIN customers c  
5 ON p.customerNumber = c.customerNumber  
6 JOIN employees e  
7 ON c.salesRepEmployeeNumber = e.employeeNumber  
8 JOIN offices o  
9 ON e.officeCode = o.officeCode  
10 GROUP BY o.city, YEAR(p.paymentDate)  
11 ORDER BY o.city, YEAR(p.paymentDate);
```

Hasil Execute

| branchName | Tahun | Total_Omset |
|---------------|-------|-------------|
| Boston | 2003 | 301781.38 |
| Boston | 2004 | 467177.07 |
| Boston | 2005 | 66923.88 |
| London | 2003 | 505384.85 |
| London | 2004 | 674815.75 |
| London | 2005 | 144125.30 |
| NYC | 2003 | 391175.53 |
| NYC | 2004 | 623872.78 |
| NYC | 2005 | 57571.16 |
| Paris | 2003 | 969959.90 |
| Paris | 2004 | 1368458.96 |
| Paris | 2005 | 480750.04 |
| San Francisco | 2003 | 532681.13 |
| San Francisco | 2004 | 517408.62 |
| San Francisco | 2005 | 287349.83 |
| Sydney | 2003 | 281985.51 |
| Sydney | 2004 | 509833.62 |
| Sydney | 2005 | 215473.85 |
| Tokyo | 2003 | 267249.40 |
| Tokyo | 2004 | 151761.45 |
| Tokyo | 2005 | 38099.22 |

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

Report : Performa Produk Terlaris per Cabang per Tahun

| Field | Table | Kegunaan |
|------------------------|--------------|------------------------------------|
| officeCode | Offices | Identitas cabang |
| city | Offices | Nama cabang |
| employeeNumber | employees | Karyawan penanggung jawab customer |
| SalesRepEmployeeNumber | customers | Karyawan yang handle customer |
| customerNumber | customers | Identitas customer |
| orderNumber | orders | Kunci transaksi |
| Orderdate | orders | Tanggal Transaksi |
| Productcode | orderdetails | Produk yang dipesan |
| Quantityordered | Orderdetails | Berapa unit produk yang dipesan |
| Productname | Products | Nama produk |
| productline | products | Kategori produk |

Query

```

1 SELECT o.officeCode AS Kode_Cabang,
2       o.city AS Nama_Cabang,
3       YEAR(od.orderDate) AS Tahun,
4       p.productName AS Produk,
5       SUM(odd.quantityOrdered) AS Total_Unit_Terjual
6 FROM offices o
7 JOIN employees e ON o.officeCode = e.officeCode
8 JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
9 JOIN orders od ON c.customerNumber = od.customerNumber
10 JOIN orderdetails odd ON od.orderNumber = odd.orderNumber
11 JOIN products p ON odd.productCode = p.productCode
12 GROUP BY o.officeCode, o.city, Tahun, Produk
13 ORDER BY o.officeCode, Tahun, Total_Unit_Terjual DESC;
14

```

Hasil Execute

| Kode_Cabang | Nama_Cabang | Tahun | Produk | Total_Unit_Terjual ▲ 1 |
|-------------|---------------|-------|---|------------------------|
| 4 | Paris | 2005 | 1968 Dodge Charger | 10 |
| 4 | Paris | 2005 | 1913 Ford Model T Speedster | 10 |
| 3 | NYC | 2005 | American Airlines: B767-300 | 11 |
| 6 | Sydney | 2005 | 1936 Mercedes Benz 500k Roadster | 15 |
| 5 | Tokyo | 2005 | 1936 Mercedes Benz 500k Roadster | 15 |
| 6 | Sydney | 2005 | 1937 Horch 930V Limousine | 18 |
| 2 | Boston | 2003 | 1962 City of Detroit Streetcar | 20 |
| 4 | Paris | 2003 | 1969 Chevrolet Camaro Z28 | 20 |
| 7 | London | 2003 | 1970 Dodge Coronet | 20 |
| 7 | London | 2004 | 1966 Shelby Cobra 427 S/C | 20 |
| 1 | San Francisco | 2003 | 1992 Ferrari 360 Spider red | 20 |
| 3 | NYC | 2003 | 1997 BMW F650 ST | 20 |
| 7 | London | 2003 | America West Airlines B757-200 | 20 |
| 5 | Tokyo | 2003 | 1982 Camaro Z28 | 20 |
| 1 | San Francisco | 2005 | 1954 Greyhound Scenicruiser | 20 |
| 1 | San Francisco | 2005 | 1958 Setra Bus | 20 |
| 5 | Tokyo | 2004 | 1996 Peterbilt 379 Stake Bed with Outrigger | 20 |
| 2 | Boston | 2004 | 1971 Alpine Renault 1600s | 20 |
| 6 | Sydney | 2005 | 1998 Chrysler Plymouth Prowler | 20 |
| 5 | Tokyo | 2003 | 1962 LanciaA Delta 16V | 20 |
| 3 | NYC | 2004 | HMS Bounty | 20 |
| 6 | Sydney | 2004 | 1965 Aston Martin DB5 | 20 |
| 4 | Paris | 2003 | 1949 Jaguar XK 120 | 20 |
| 1 | San Francisco | 2005 | Boeing X-32A JSF | 20 |
| 4 | Paris | 2005 | 1982 Ducati 900 Monster | 20 |