**LAPORAN PRAKTIKUM**

**JOBSHEET 02**

**DATABASE OPERASIONAL**

Disusun untuk memenuhi nilai tugas

Mata Kuliah : Data Warehouse



Oleh :

Aqueena Regita Hapsari

2341760096

SIB-2B - 03

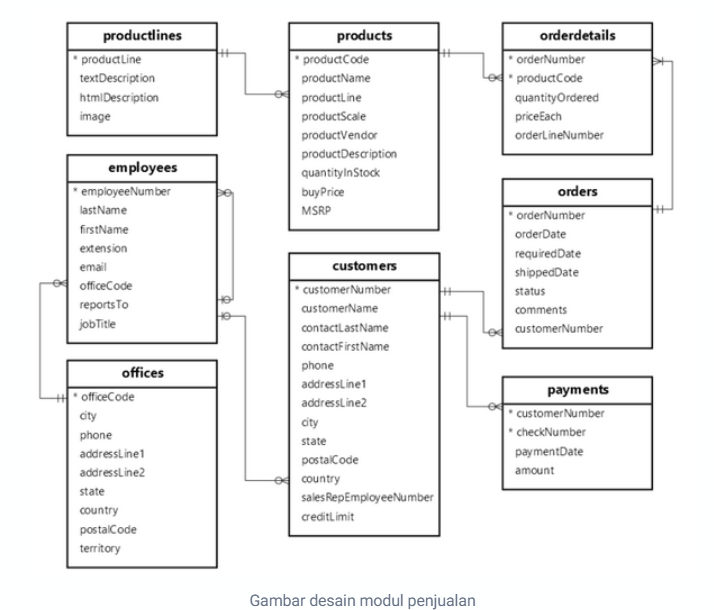
**PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS**

**POLITEKNIK NEGERI MALANG**

**TAHUN AJARAN 2024/2025**

**STUDI KASUS**

LegendVehicle merupakan perusahan 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:

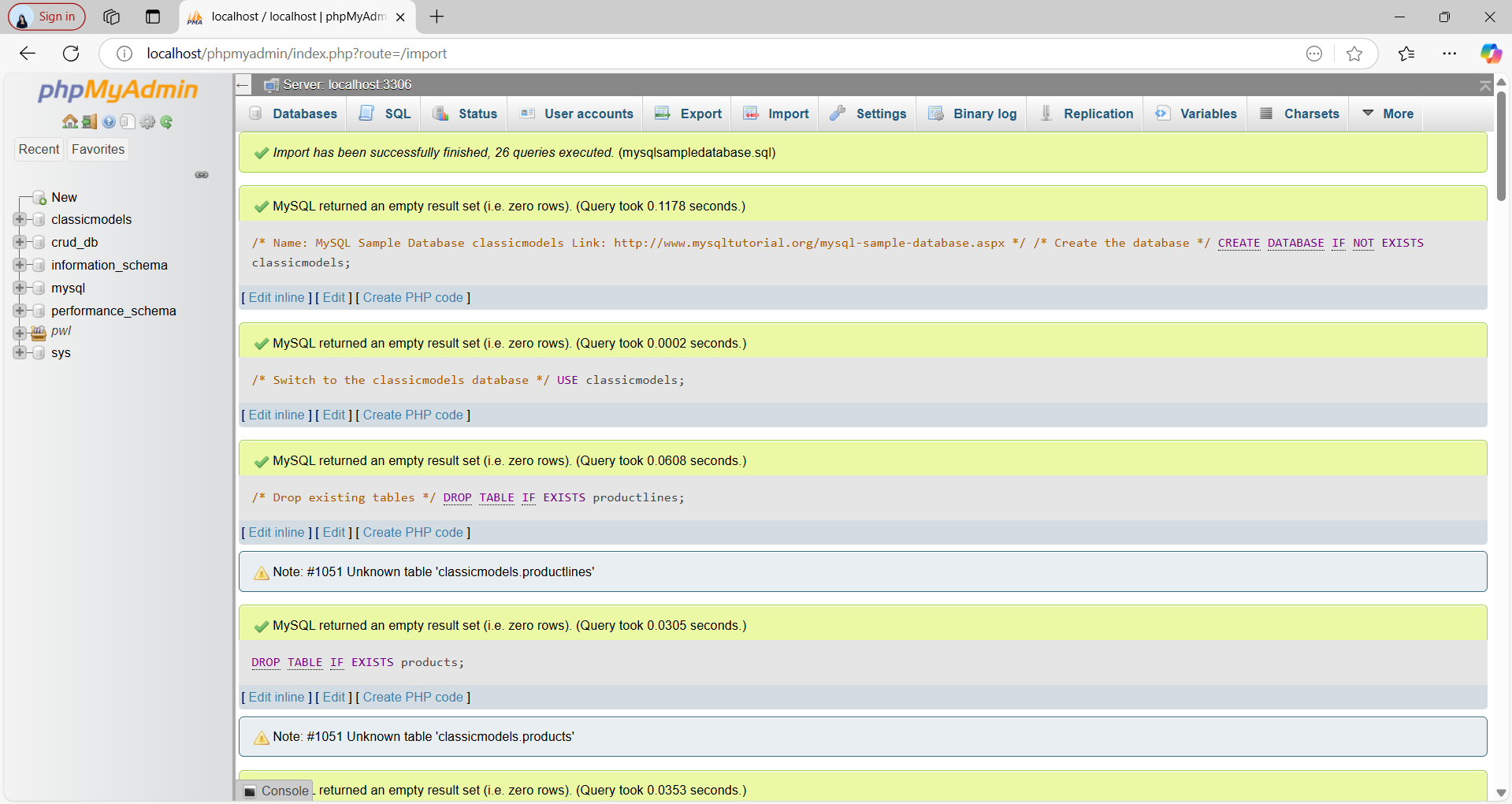


Selain itu proses penjualan kendaraan pada perusahaan tersebut bukan hanya melalui showroom cabang, melainkan reseller-reseller bebas lainnya.

**TUGAS 1**

1. Import data perusahaan tersebut pada DBMS MySQL!

**Hasil :**



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

**Jawaban :**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Tabel 1 | Tabel 2 | Jenis Relasi |
|  | productlines | products | One to Many |
|  | customers | orders | One to Many |
|  | orders | orderdetails | One to Many |
|  | products | orderdetails | One to Many |
|  | employees | customers | One to Many |
|  | offices | employees | One to Many |

1. Analisa jumlah field pada setiap tabel!

**Jawaban :**

|  |  |  |
| --- | --- | --- |
| No. | Nama Tabel | Jumlah Field |
|  | productlines | 4 |
|  | products | 9 |
|  | offices | 9 |
|  | employees | 8 |
|  | customers | 13 |
|  | payments | 4 |
|  | orders | 7 |
|  | orderdetails | 5 |

**A. ANALISA DATA**

**PRAKTIKUM 1**

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

SELECT \*

FROM employees employe, employes manager, customer cust

WHERE employee.reportsTo=manager.employeeNumber

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.

**Jawaban :**

Perbaikan :

SELECT \*

FROM employees employee, employees manager, customers cust

WHERE employee.reportsTo = manager.employeeNumber

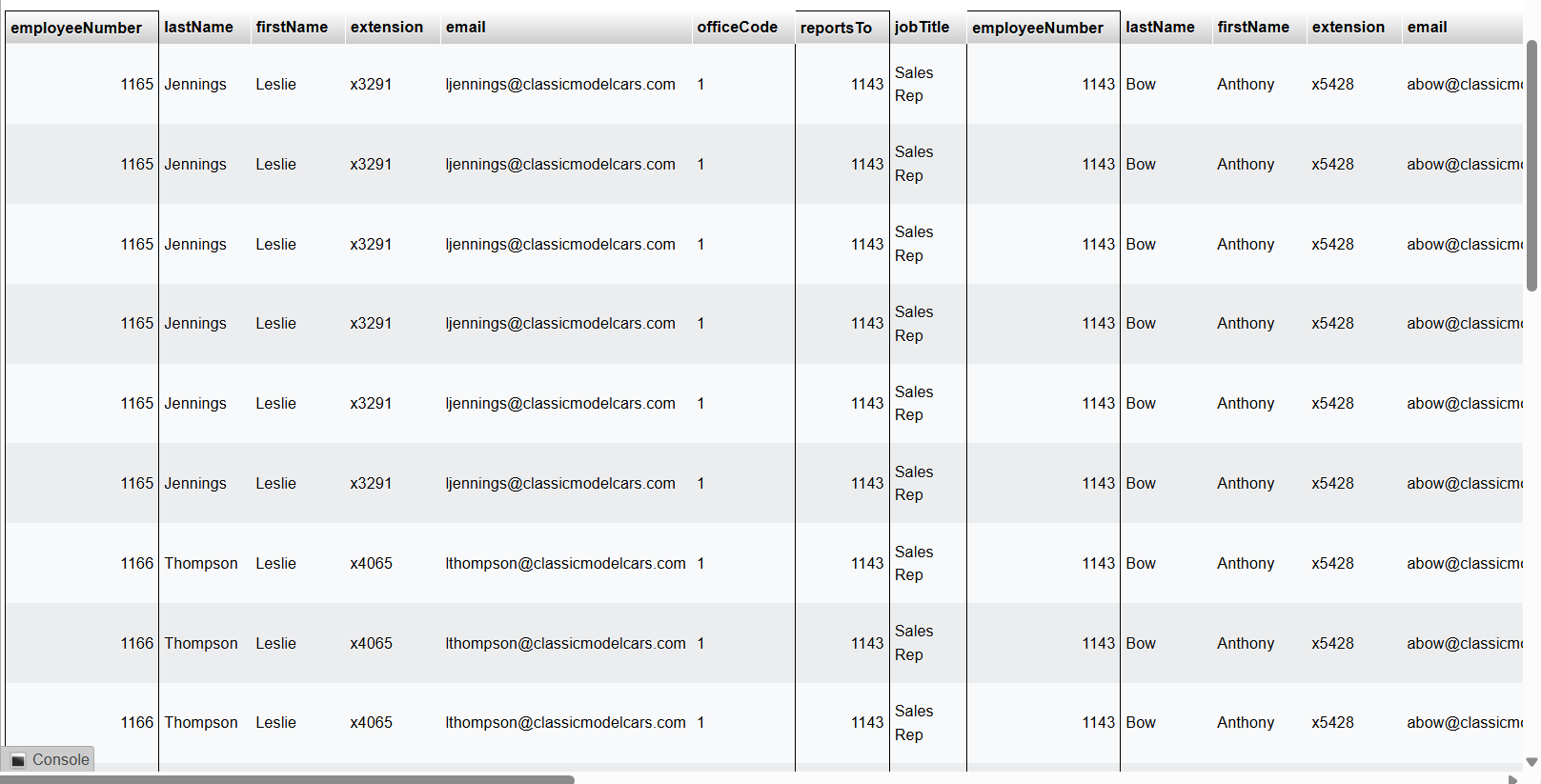
AND employee.employeeNumber = cust.salesRepEmployeeNumber;

Analisis query :

1. Query ini menggunakan self-join pada tabel employees, di mana:

* employee adalah staf.
* manager adalah atasannya (reportsTo merujuk ke employeeNumber seorang manajer).

1. Kemudian, query juga menghubungkan employee dengan customer berdasarkan salesRepEmployeeNumber.
2. Hasil query ini akan menampilkan daftar pegawai beserta manajernya dan customer yang dimiliki pegawai tersebut.



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.

**Jawaban :**

Perbaikan :

SELECT

manager.employeeNumber AS id\_manager,

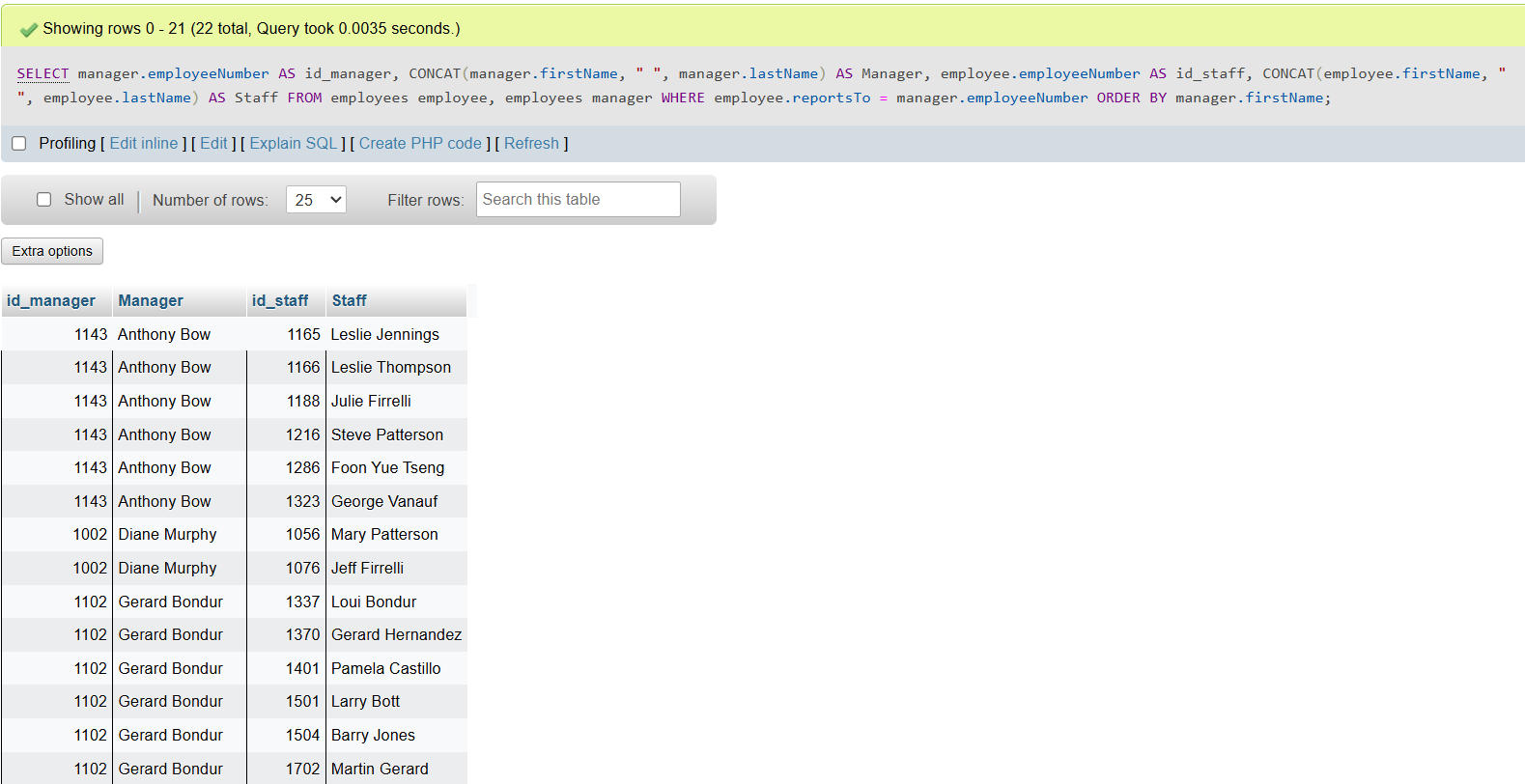
CONCAT(manager.firstName, " ", manager.lastName) AS Manager, employee.employeeNumber AS id\_staff,

CONCAT(employee.firstName, " ", employee.lastName) AS Staff

FROM employees employee, employees manager WHERE employee.reportsTo = manager.employeeNumber ORDER BY manager.firstName;

Analisis Query :

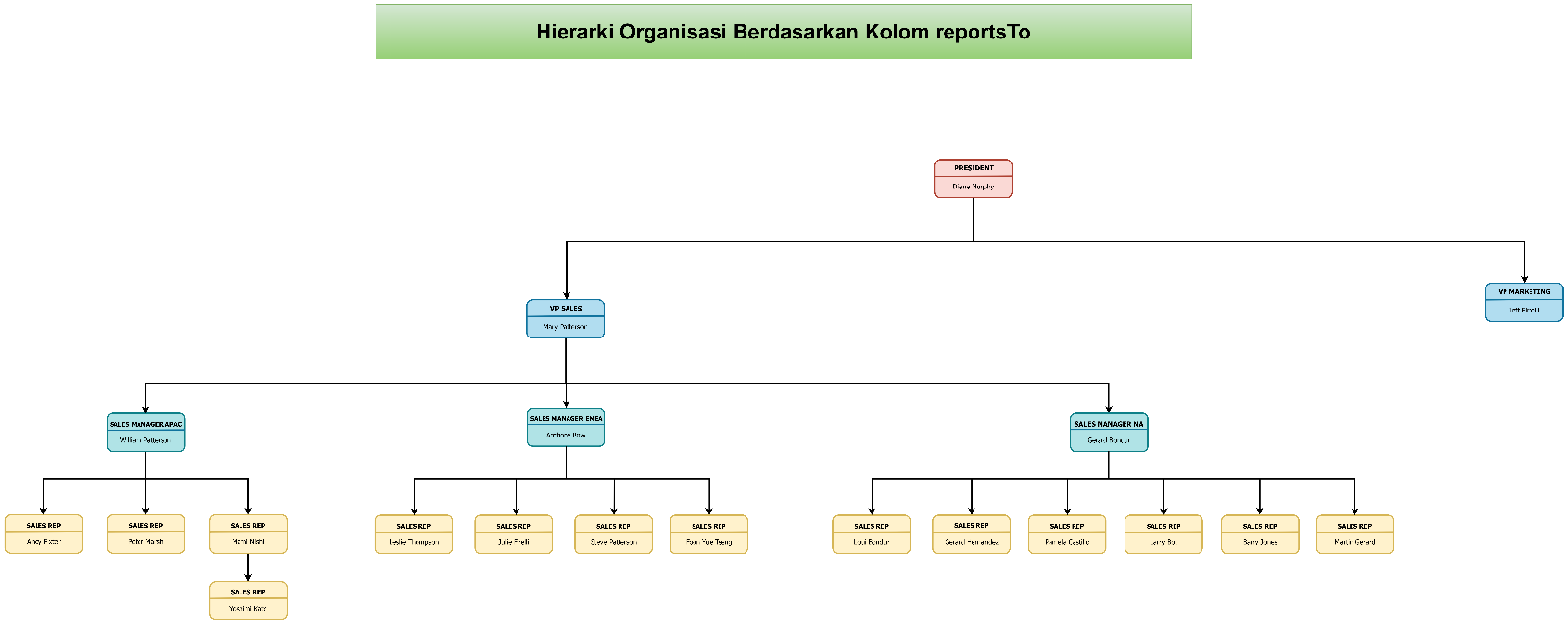
1. Query ini menggunakan self-join pada tabel employees untuk menampilkan hierarki pegawai.
2. Kolom reportsTo menunjukkan bahwa seorang pegawai memiliki atasan (manager.employeeNumber).
3. Hasilnya berupa daftar pegawai dengan ID manajer mereka, serta nama manajer dan staf.
4. ORDER BY manager.firstName akan mengurutkan berdasarkan nama depan manajer.



**TUGAS 2**

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

**Jawaban :**



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,

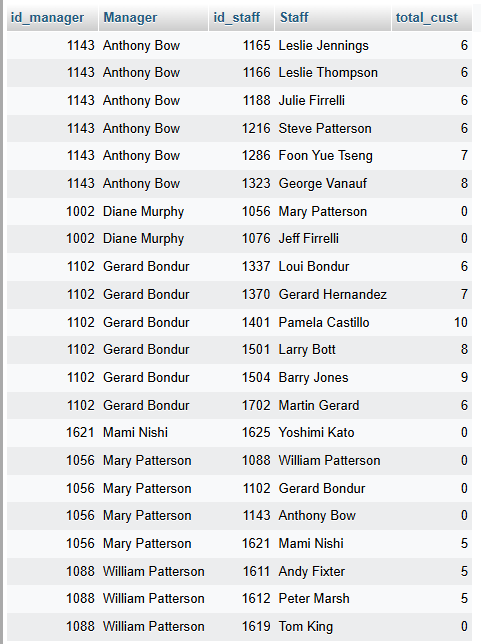
count(cust.customerNumber) as total\_cust

FROM employees employee join employees manager on employee.reportsTomanager.employeeNumber

left join customers cust on employee.employeeNumber=cust.salesRepEmployeeNumber GROUP BY employee.employeeNumber

ORDER BY manager.firstName;

dari query tersebut menghasilkan jumlah customer dari setiap staff.



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?

**Jawaban :**

Menggunakan query ORDER BY total\_cust DESC untuk melihat staff dengan pelanggna terbanyak di level bawah.

SELECT

employee.employeeNumber,

CONCAT(employee.firstName, " ", employee.lastName) AS Staff,

COUNT(cust.customerNumber) AS total\_cust

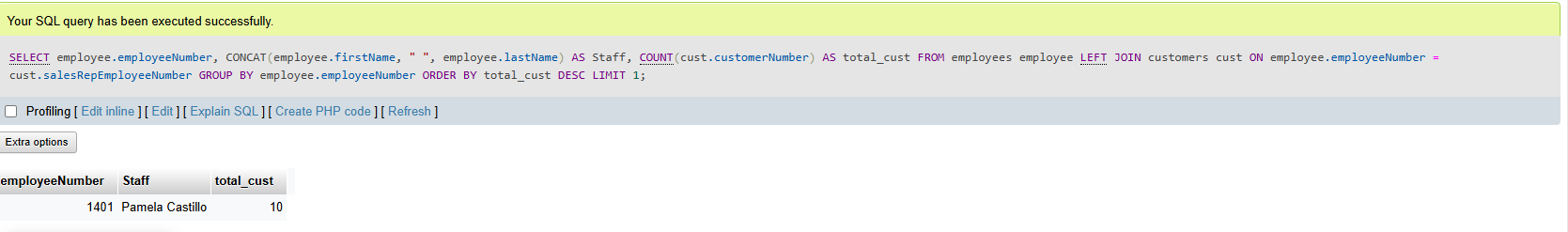
FROM employees employee

LEFT JOIN customers cust ON employee.employeeNumber = cust.salesRepEmployeeNumber

GROUP BY employee.employeeNumber

ORDER BY total\_cust DESC

LIMIT 1;

****

1. 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!

**Jawaban :**

Jika KPI atasan dihitung dari jumlah customer pribadinya + jumlah customer dari staff di bawahnya maka ranking pegawai berdasarkan jumlah customer total (termasuk bawahan).

WITH RECURSIVE EmployeeHierarchy AS (

SELECT

employeeNumber, reportsTo, firstName, lastName

FROM employees

WHERE reportsTo IS NOT NULL

UNION ALL

SELECT

e.employeeNumber, e.reportsTo, e.firstName, e.lastName

FROM employees e

INNER JOIN EmployeeHierarchy eh ON e.reportsTo = eh.employeeNumber

)

SELECT

e.employeeNumber AS id\_employee,

CONCAT(e.firstName, " ", e.lastName) AS Employee,

COUNT(c.customerNumber) AS direct\_customers,

(COUNT(c.customerNumber) + COALESCE(SUM(sub.total\_cust), 0)) AS total\_kpi\_customers

FROM employees e

LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber

LEFT JOIN (

SELECT salesRepEmployeeNumber, COUNT(customerNumber) AS total\_cust

FROM customers

GROUP BY salesRepEmployeeNumber

) sub ON e.employeeNumber = sub.salesRepEmployeeNumber

GROUP BY e.employeeNumber, e.firstName, e.lastName

ORDER BY total\_kpi\_customers DESC;



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

**Jawaban :**

query ini untuk menghitung omset berdasarkan total pembayaran yang diterima oleh setiap pegawai

SELECT

e.employeeNumber AS id\_employee,

CONCAT(e.firstName, " ", e.lastName) AS Employee,

SUM(p.amount) AS total\_revenue

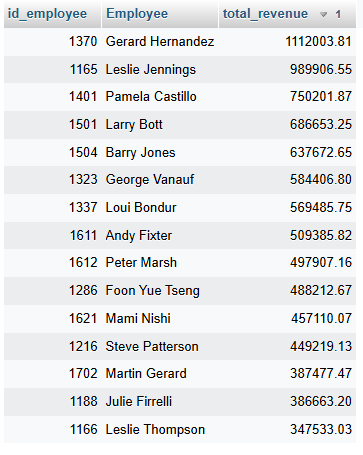
FROM employees e

JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber

JOIN payments p ON c.customerNumber = p.customerNumber

GROUP BY e.employeeNumber

ORDER BY total\_revenue DESC;



1. 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?

**Jawaban :**

|  |  |
| --- | --- |
| KPI | Jumlah Field yang Dibutuhkan |
| Jumlah customer yang bertransaksi | customerNumber, salesRepEmployeeNumber |
| Jumlah omset yang didapat | amount, customerNumber, salesRepEmployeeNumber |

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

**Jawaban :**

Tabel laporan per tahun untuk Foon Yue Tsend dan Pamela Castillo, query :

SELECT

e.employeeNumber AS id\_employee,

CONCAT(e.firstName, " ", e.lastName) AS Employee,

YEAR(p.paymentDate) AS year,

SUM(p.amount) AS total\_revenue

FROM employees e

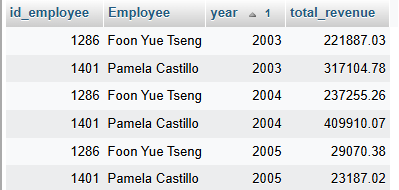
JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber

JOIN payments p ON c.customerNumber = p.customerNumber

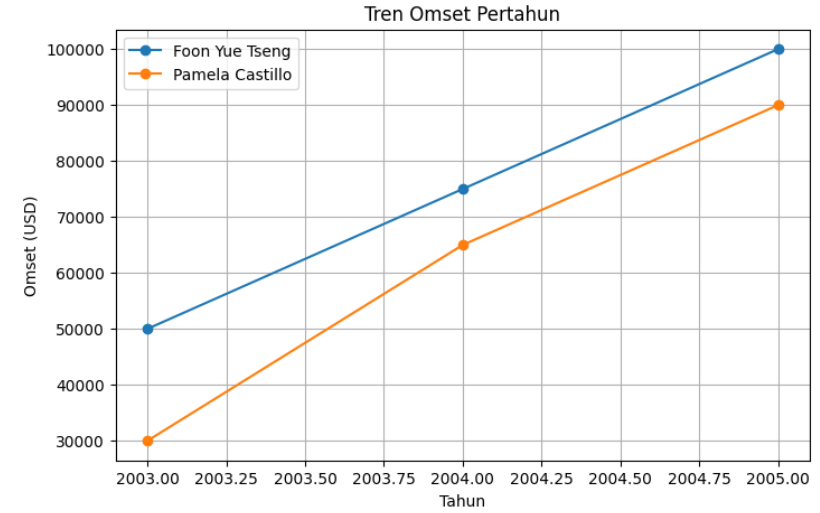
WHERE e.firstName IN ('Foon Yue', 'Pamela') AND e.lastName IN ('Tseng', 'Castillo')

GROUP BY e.employeeNumber, year

ORDER BY year ASC;



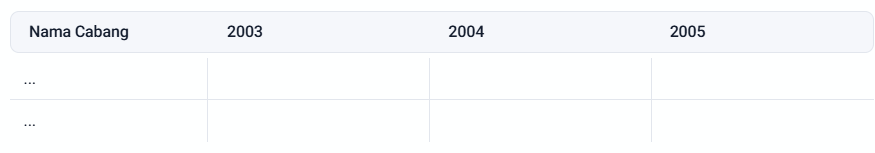
Grafik garis dari jumlah omset yang didapat oleh Foon Yue Tseng dan Pamela Casillo



**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 melaluai OLTP yang ada.

Hasil report yang diinginkan adalah grafik berdasarkan tabel berikut:



Analisalah terlebih dahulu:

1. Field apa saja yang diperlukan untuk menampilkan penjualan di setiap cabang.
2. Bentuk query dengan memperhatikan relasi antar tabel.

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

**Jawaban :**

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

Dilihat dari struktur database **classicmodels**, tabel yang dapat digunakan adalah:

1. **offices** → Menyimpan data cabang (lokasi kantor).
2. **employees** → Menyimpan informasi karyawan yang bisa dikaitkan dengan cabang.
3. **customers** → Menyimpan informasi pelanggan.
4. **orders** → Menyimpan informasi transaksi, termasuk tanggal order.
5. **orderdetails** → Menyimpan rincian produk yang dipesan, termasuk jumlah dan harga.

Field yang digunakan dalam query:

1. Nama Cabang → **o.city** (dari tabel **offices**)
2. Tahun Transaksi → **YEAR**(**orders.orderDate)**
3. Total Omset → **SUM**(**orderdetails.quantityOrdered \* orderdetails.priceEach)**
4. Bentuk query dengan memperhatikan relasi antar tabel.
   1. Query untuk Laporan Omset per Tahun per Cabang

SELECT

o.city AS Nama\_Cabang,

SUM(CASE WHEN YEAR(od.orderDate) = 2003 THEN (odt.quantityOrdered \* odt.priceEach) ELSE 0 END) AS '2003',

SUM(CASE WHEN YEAR(od.orderDate) = 2004 THEN (odt.quantityOrdered \* odt.priceEach) ELSE 0 END) AS '2004',

SUM(CASE WHEN YEAR(od.orderDate) = 2005 THEN (odt.quantityOrdered \* odt.priceEach) ELSE 0 END) AS '2005'

FROM offices o

JOIN employees e ON o.officeCode = e.officeCode

JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber

JOIN orders od ON c.customerNumber = od.customerNumber

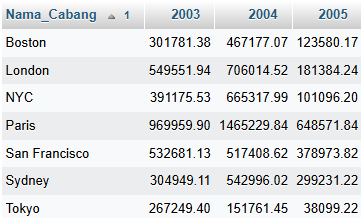
JOIN orderdetails odt ON od.orderNumber = odt.orderNumber

GROUP BY o.city

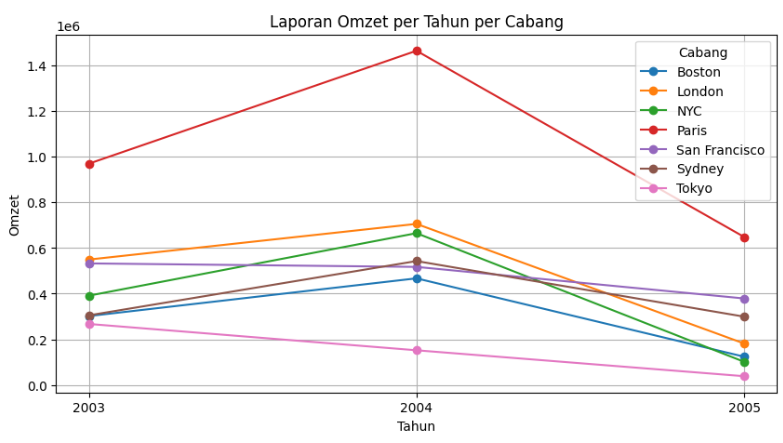
ORDER BY Nama\_Cabang;

Penjelasan query :

1. Menghubungkan tabel
2. offices → Menyediakan data cabang.
3. employees → Menghubungkan cabang dengan pelanggan.
4. customers → Menyimpan pelanggan dan sales representative.
5. orders → Menyimpan order beserta tanggalnya.
6. orderdetails → Menghitung total harga dari setiap pesanan.
7. Menggunakan SUM(CASE WHEN YEAR(od.orderDate) = X THEN od.total ELSE 0 END) artinya menghitung total omset berdasarkan tahun tertentu.
8. GROUP BY o.city artinya mengelompokkan hasil berdasarkan cabang.



* 1. Visualisasi data (dengan python namun tidak connect DB)



1. Buatlah report lain dengan sumber data OLTP yang sama, analisa field yang digunakan, bentuk struktur query dan tuliskan dalam tabel serta grafiknya.
2. **Report : Laporan Total Pembayaran (Revenue) per Tahun** berdasarkan tabel **payments**.
3. Adapun field yang digunakan adalah:
4. payments.customerNumber → Untuk menghubungkan pelanggan dengan transaksi.
5. payments.amount → Jumlah pembayaran yang dilakukan pelanggan.
6. payments.paymentDate → Untuk mengekstrak tahun pembayaran.
7. Struktur Query untuk report total pembayaran (revenue) per tahun

SELECT

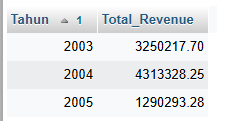
YEAR(paymentDate) AS Tahun,

SUM(amount) AS Total\_Revenue

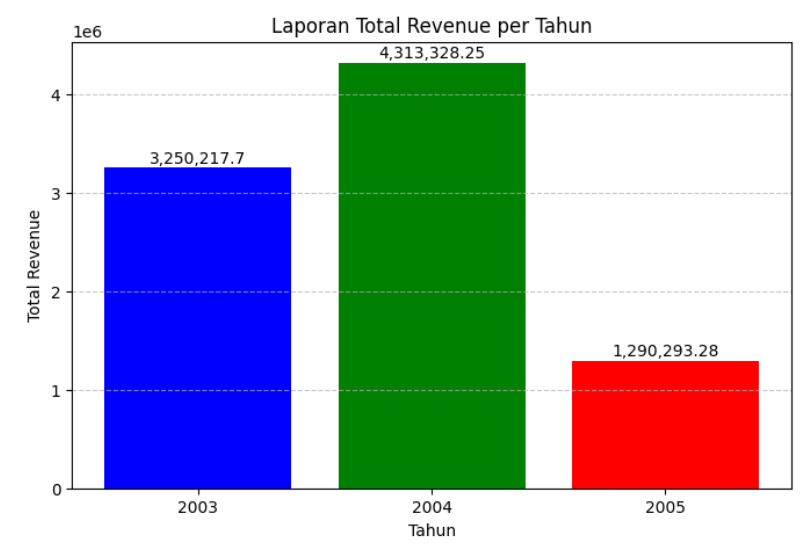
FROM payments

GROUP BY Tahun

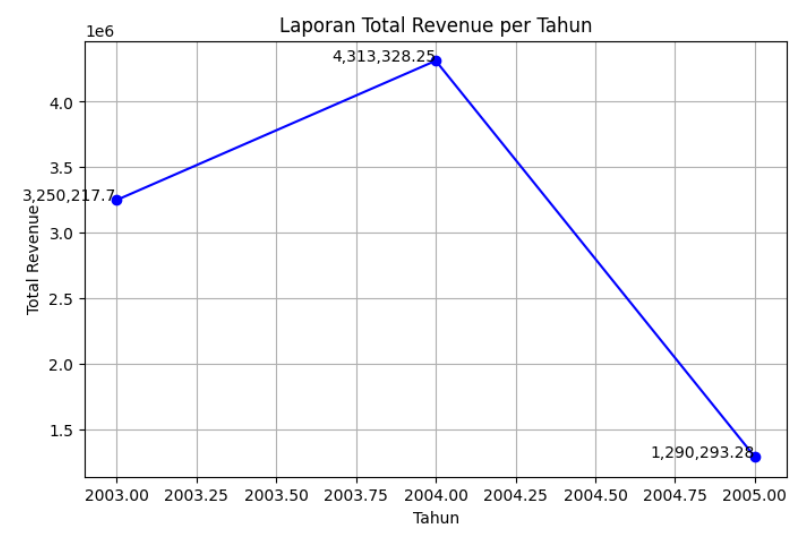
ORDER BY Tahun;



1. Visualisasi Grafik (tidak connect database)



Gambar 1 Grafik Batang Report Revenue per Tahun



Gambar 2 Grafik Garis Revenue per Tahun