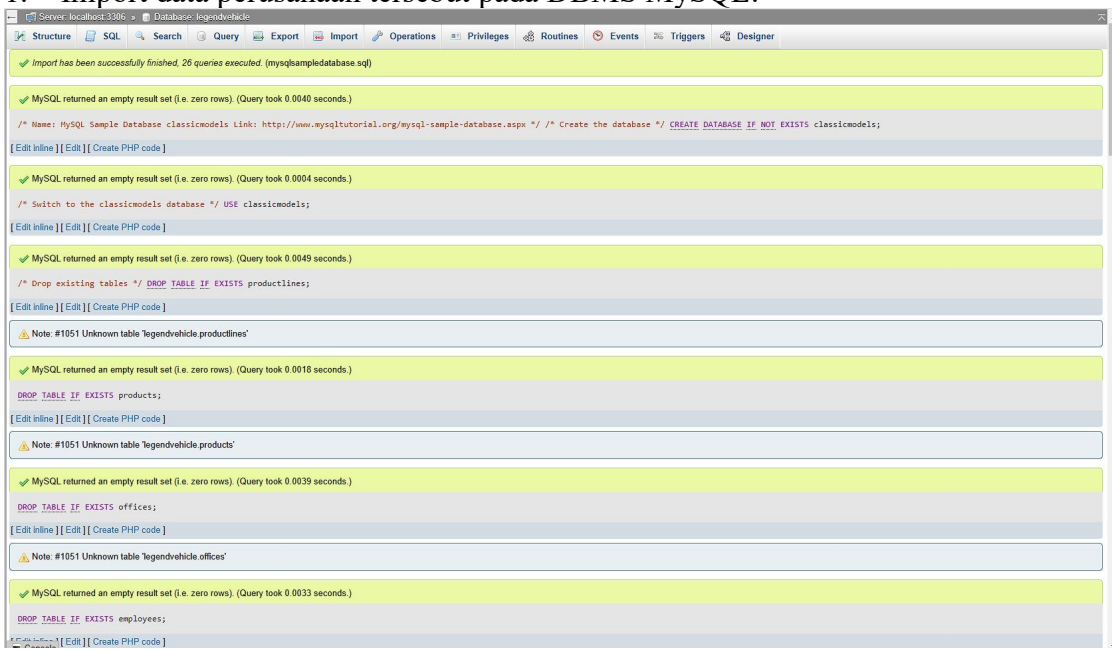
JOBSHEET 2
DATA WAREHOUSE
KHUZAMA FILLA JANUARTHA

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!

T2	T1	Relasi
Customers	Orders	One to Many
Orders	OrderDetails	One to Many
OrderDetails	Products	Many to One
Products	ProductLines	Many to One
Customers	Payments	One to Many
Employees	Offices	Many to One
Customers	Employees	Many to One
Employees	Employees	One to Many

3. Analisa jumlah field pada setiap tabel!

Tabel	Field
Customers	13
Employees	8
Offices	9
Orderdetails	5
Orders	7
Payment	4
Productlines	4
Products	9

PRAKTIKUM 1

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.

```
SELECT * FROM employees AS employe JOIN employees AS manager ON employe.reportsTo = manager.employeeNumber JOIN customers AS cust ON employe.employeeNumber = cust.salesRepEmployeeNumber;
```

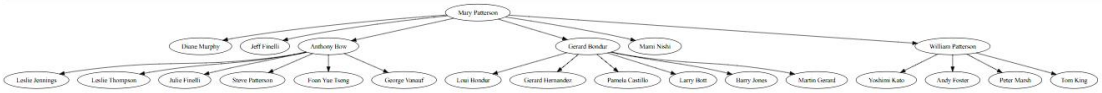
2. Buka tab baru pada browser untuk melakukan eksekusi query berikut, dari hasil query diatas maka akan ditemukan atasan dari setiap pegawai.

```
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
1088	William Patterson	1611	Andy Fixter
1088	William Patterson	1612	Peter Marsh
1088	William Patterson	1619	Tom King

TUGAS 2

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



2. Buka tab baru pada browser untuk melakukan eksekusi query berikut, dari query tersebut menghasilkan jumlah customer dari setiap staff.

Showing rows 0 - 21 (22 total, Query took 0.0024 seconds.)

```
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	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
1621	Mami Nishi	1625	Yoshimi Kato	0
1056	Mary Patterson	1088	William Patterson	0
1056	Mary Patterson	1102	Gerard Bondur	0
1056	Mary Patterson	1143	Anthony Bow	0
1056	Mary Patterson	1621	Mami Nishi	5
1088	William Patterson	1611	Andy Fixter	5
1088	William Patterson	1612	Peter Marsh	5
1088	William Patterson	1619	Tom King	0

TUGAS 3

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

```
WITH StaffTanpaBawahan AS ( SELECT e.employeeNumber, CONCAT(e.firstName, ' ', e.lastName) AS employee_name, COUNT(c.customerNumber) AS total_customers FROM employees e LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber WHERE e.employeeNumber NOT IN (SELECT DISTINCT reportsTo FROM employees WHERE reportsTo IS NOT NULL) GROUP BY e.employeeNumber, e.firstName, e.lastName ) SELECT * FROM StaffTanpaBawahan ORDER BY total_customers DESC LIMIT 1;
```

employeeNumber	employee_name	total_customers
1401	Pamela Castillo	10

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!

Showing rows 0 - 22 (23 total, Query took 0.0027 seconds.)

```
WITH RECURSIVE EmployeeHierarchy AS ( SELECT e.employeeNumber, e.firstName, e.lastName, e.reportsTo, COUNT(c.customerNumber) AS total_customer FROM employees e LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber GROUP BY e.employeeNumber, e.firstName, e.lastName, e.reportsTo UNION ALL SELECT e.employeeNumber, e.firstName, e.lastName, e.reportsTo, eh.total_customer FROM employees e JOIN EmployeeHierarchy eh ON e.employeeNumber = eh.reportsTo ) SELECT employeeNumber, firstName, lastName, SUM(total_customer) AS total_customer_akhir FROM EmployeeHierarchy GROUP BY employeeNumber, firstName, lastName ORDER BY total_customer_akhir DESC;
```

employeeNumber	firstName	lastName	total_customer_akhir
1002	Diane	Murphy	100
1056	Mary	Patterson	100
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
1611	Andy	Fixter	5
1612	Peter	Marsh	5
1621	Mami	Nishi	5
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!

Showing rows 0 - 14 (15 total, Query took 0.0199 seconds.)

```
WITH SalesPerEmployee AS ( SELECT e.employeeNumber, CONCAT(e.firstName, ' ', e.lastName) AS employee_name, SUM(od.quantityOrdered * od.priceEach) AS total_omset FROM
employees e JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber JOIN orders o ON c.customerNumber = o.customerNumber JOIN orderdetails od ON o.orderNumber =
od.orderNumber GROUP BY e.employeeNumber, e.firstName, e.lastName ) SELECT employeeNumber, employee_name, total_omset, RANK() OVER (ORDER BY total_omset DESC) AS ranking
FROM SalesPerEmployee;
```

employeeNumber	employee_name	total_omset	ranking
1370	Gerard Hernandez	1258577.81	1
1165	Leslie Jennings	1081530.54	2
1401	Pamela Castillo	868220.55	3
1501	Larry Bott	732096.79	4
1504	Barry Jones	704853.91	5
1323	George Vanauf	669377.05	6
1612	Peter Marsh	584593.76	7
1337	Loui Bondur	569485.75	8
1611	Andy Fixter	562582.59	9
1216	Steve Patterson	505875.42	10
1286	Foon Yue Tseng	488212.67	11
1621	Mami Nishi	457110.07	12
1702	Martin Gerard	387477.47	13
1188	Julie Firrelli	386663.20	14
1166	Leslie Thompson	347533.03	15

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	JUMLAH FIELD
Jumlah customer yang berinteraksi	3 field (employeeNumber, EmployeeName, total_customer)
Jumlah omset yang didapat	4 field(employeeNumber, employee_name, total_orders [pegawai], total_omset)

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

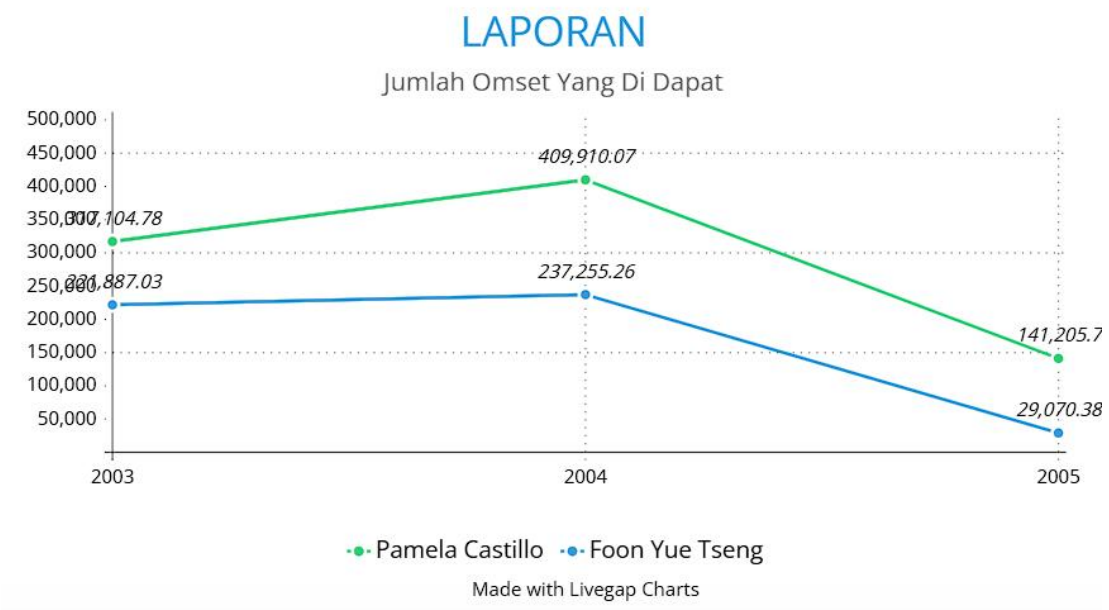
QUERY

```
SELECT e.employeeNumber, CONCAT(e.firstName, ' ', e.lastName) AS employee_name, YEAR(o.orderDate) AS tahun, SUM(od.quantityOrdered * od.priceEach) AS total_omset FROM
employees e JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber JOIN orders o ON c.customerNumber = o.customerNumber JOIN orderdetails od ON o.orderNumber =
od.orderNumber WHERE e.firstName IN ('Foon Yue', 'Pamela') AND e.lastName IN ('Tseng', 'Castillo') GROUP BY e.employeeNumber, tahun ORDER BY tahun, employee_name;
```

TABEL

Nama	2003	2004	2005
Foon Yue Tseng	221887.03	237255.26	29070.38
Pamela Castillo	317104.78	409910.07	141205.70

GRAFIK



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.
Analisalah terlebih dahulu:

1. Field apa saja yang diperlukan untuk menampilkan penjualan di setiap cabang.
Nama Cabang : Biasanya dari tabel offices (misalnya, o.city atau o.officeName)
Tahun : Bisa diambil dari YEAR(o.orderDate) dari tabel orders
Total Omzet : Dihitung dari SUM(od.quantityOrdered * od.priceEach) dari tabel orderdetails

2. Bentuk query dengan memperhatikan relasi antar tabel.

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

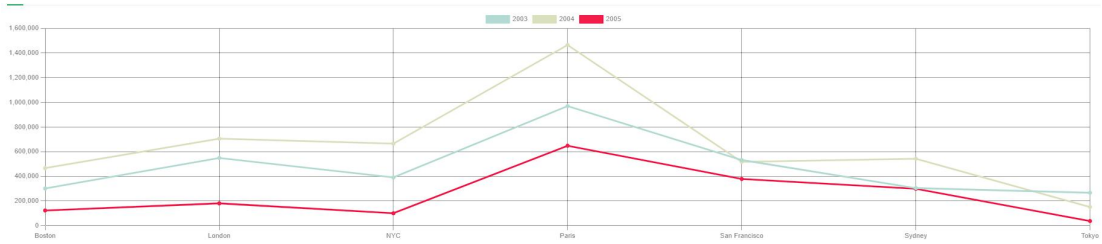
NamaCabang	Tahun ▲ 2	Omset
Boston	2003	301781.38
Boston	2004	467177.07
Boston	2005	123580.17
London	2003	549551.94
London	2004	706014.52
London	2005	181384.24
NYC	2003	391175.53
NYC	2004	665317.99
NYC	2005	101096.20
Paris	2003	969959.90
Paris	2004	1465229.84
Paris	2005	648571.84
San Francisco	2003	532681.13
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

TABEL

Cabang	2003	2004	2005
Boston	301781.38	467177.07	123580.17
London	549551.94	706014.52	181384.24
NYC	391175.53	665317.99	101096.20
Paris	969959.90	1465229.84	648571.84
San Francisco	532681.13	517408.62	378973.82

Sydney	304949.11	542996.02	299231.22
Tokyo	267249.40	151761.45	38099.22

GRAFIK



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

FIELD

Laporan ini akan menampilkan **jumlah total order yang dilakukan di setiap cabang per tahun.** Ini berguna untuk melihat tren peningkatan atau penurunan jumlah transaksi di setiap cabang.

Field	Tabel	Deskripsi
o.city	offices	Nama cabang berdasarkan lokasi kantor
YEAR(od.orderDate)	orders	Tahun transaksi terjadi
COUNT(DISTINCT od.orderNumber)	orders	Total jumlah order di setiap cabang tiap tahun

QUERY

Showing rows 0 - 20 (21 total, Query took 0.0078 seconds.)

```
SELECT o.city AS NamaCabang, YEAR(od.orderDate) AS Tahun, COUNT(DISTINCT od.orderNumber) AS TotalOrder FROM orders od JOIN customers c ON od.customerNumber =
c.customerNumber JOIN employees e ON c.salesRepEmployeeNumber = e.employeeNumber JOIN offices o ON e.officeCode = o.officeCode GROUP BY o.city, YEAR(od.orderDate) ORDER BY
o.city, Tahun;
```

TABEL

NamaCabang	Tahun ▲ 2	TotalOrder
Boston	2005	5
London	2003	18
London	2004	24
London	2005	5
NYC	2003	14
NYC	2004	22
NYC	2005	3
Paris	2003	34
Paris	2004	49
Paris	2005	23
San Francisco	2003	17
San Francisco	2004	17
San Francisco	2005	14
Sydney	2003	12
Sydney	2004	15
Sydney	2005	11
Tokyo	2003	7
Tokyo	2004	6
Tokyo	2005	3

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

GRAFIK