

JOB SHEET 2

Data Operasional

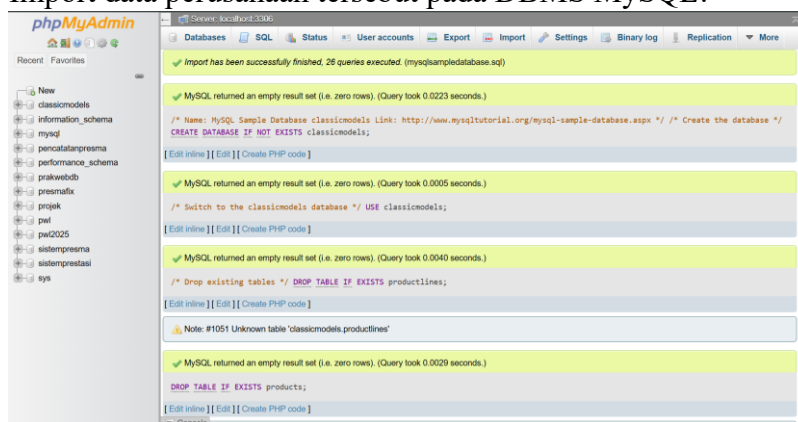
Nama : Reza Angelina Febriyanti

Kelas / No Absen : SIB 2A / 27

NIM : 2341760015

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!

Struktur Data dari Database Berdasarkan tabel berikut

- customers

```
mysql> DESC customers;
```

Field	Type	Null	Key	Default	Extra
customerNumber	int	NO	PRI	NULL	
customerName	varchar(50)	NO		NULL	
contactLastName	varchar(50)	NO		NULL	
contactFirstName	varchar(50)	NO		NULL	
phone	varchar(50)	NO		NULL	
addressLine1	varchar(50)	NO		NULL	
addressLine2	varchar(50)	YES		NULL	
city	varchar(50)	NO		NULL	
state	varchar(50)	YES		NULL	
postalCode	varchar(15)	YES		NULL	
country	varchar(50)	NO		NULL	
salesRepEmployeeNumber	int	YES	MUL	NULL	
creditLimit	decimal(10,2)	YES		NULL	

13 rows in set (0.04 sec)

- employees

```
mysql> DESC employees;
```

Field	Type	Null	Key	Default	Extra
employeeNumber	int	NO	PRI	NULL	
lastName	varchar(50)	NO		NULL	
firstName	varchar(50)	NO		NULL	
extension	varchar(10)	NO		NULL	
email	varchar(100)	NO		NULL	
officeCode	varchar(10)	NO	MUL	NULL	
reportsTo	int	YES	MUL	NULL	
jobTitle	varchar(50)	NO		NULL	

8 rows in set (0.00 sec)

- offices

```
mysql> DESC offices;
```

Field	Type	Null	Key	Default	Extra
officeCode	varchar(10)	NO	PRI	NULL	
city	varchar(50)	NO		NULL	
phone	varchar(50)	NO		NULL	
addressLine1	varchar(50)	NO		NULL	
addressLine2	varchar(50)	YES		NULL	
state	varchar(50)	YES		NULL	
country	varchar(50)	NO		NULL	
postalCode	varchar(15)	NO		NULL	
territory	varchar(10)	NO		NULL	

9 rows in set (0.00 sec)

- orderdetails

```
mysql> DESC orderdetails;
```

Field	Type	Null	Key	Default	Extra
orderNumber	int	NO	PRI	NULL	
productCode	varchar(15)	NO	PRI	NULL	
quantityOrdered	int	NO		NULL	
priceEach	decimal(10,2)	NO		NULL	
orderLineNumber	smallint	NO		NULL	

5 rows in set (0.00 sec)

- orders

```
mysql> DESC orders;
```

Field	Type	Null	Key	Default	Extra
orderNumber	int	NO	PRI	NULL	
orderDate	date	NO		NULL	
requiredDate	date	NO		NULL	
shippedDate	date	YES		NULL	
status	varchar(15)	NO		NULL	
comments	text	YES		NULL	
customerNumber	int	NO	MUL	NULL	

7 rows in set (0.00 sec)

- payments

```
mysql> DESC payments;
```

Field	Type	Null	Key	Default	Extra
customerNumber	int	NO	PRI	NULL	
checkNumber	varchar(50)	NO	PRI	NULL	
paymentDate	date	NO		NULL	
amount	decimal(10,2)	NO		NULL	

4 rows in set (0.00 sec)

- productlines

```
mysql> DESC productlines;
```

Field	Type	Null	Key	Default	Extra
productLine	varchar(50)	NO	PRI	NULL	
textDescription	varchar(4000)	YES		NULL	
htmlDescription	mediumtext	YES		NULL	
image	mediumblob	YES		NULL	

4 rows in set (0.00 sec)

- products

```
mysql> DESC products;
```

Field	Type	Null	Key	Default	Extra
productCode	varchar(15)	NO	PRI	NULL	
productName	varchar(70)	NO		NULL	
productLine	varchar(50)	NO	MUL	NULL	
productScale	varchar(10)	NO		NULL	
productVendor	varchar(50)	NO		NULL	
productDescription	text	NO		NULL	
quantityInStock	smallint	NO		NULL	
buyPrice	decimal(10,2)	NO		NULL	
MSRP	decimal(10,2)	NO		NULL	

```
9 rows in set (0.00 sec)
```

Berdasarkan hasil analisa dari key constrain pada tabel, berikut hubungan relasi setiap tabel

Tabel 1	Tabel 2	Jenis Relasi
customers	orders	One to Many (Satu customers dapat memiliki banyak orders)
orders	ordersdetails	One to Many (Satu orders dapat memiliki beberapa ordersdetails)
products	ordersdetails	One to Many (Satu products memiliki beberapa orderdetails)
customers	payments	One to Many (Satu customers dapat melakukan beberapa payments)
employees	offices	Many to One (Banyak karyawan bekerja pada satu kantor)
productlines	products	One to Many (Productlines dapat memiliki beberapa products)

3. Analisa jumlah field pada setiap tabel!

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

PRAKTIKUM

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

```
mysql> SELECT * FROM
-> employees employee, employees manager, customers cust
-> WHERE employee.reportsTo = manager.employeeNumber
-> AND employee.employeeNumber = cust.salesRepEmployeeNumber;
```

employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle	employeeNumber	lastName	firstName	extension	email	officeCode	reportsTo	jobTitle	customerNumber	customerName	city	contactLastName	contactFirstName	phone	addressLine1	salesRepEmployeeNumber	creditLimit
1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	1143	Mini Gifts Distributors Ltd.	San Rafael	Nelson	Susan	4155551450	5677 Strong St.	1165	210500.00
1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	1143	Mini Wheels Co.	San Francisco	Murphy	Julie	6505555787	5557 North Pendale Street	1165	64600.00
1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	1143	Bow	Anthony	x5428	abow@classicmodelcars.com	1	1056	Sales Manager (NA)	1143	Technics Stores Inc.	Burlingame	Hashimoto	Juri	6505556809	9408 Furth Circle	1165	84600.00
1102	Bondur	Gerard	x5408	gbondur@classicmodelcars.com	4	1056	Sale Manager (EMEA)	1102	Bondur	Gerard	x5408	gbondur@classicmodelcars.com	4	1056	Sale Manager (EMEA)	1102	Iberia Gift Imports, Corp.	Sevilla	Roel	José Pedro	(95) 555 82 82	C/ Romero, 33	1702	65700.00

100 rows in set (0.00 sec)

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

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

```
mysql> 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
-> JOIN employees manager ON employee.reportsTo = manager.employeeNumber
-> ORDER BY manager.firstName;
```

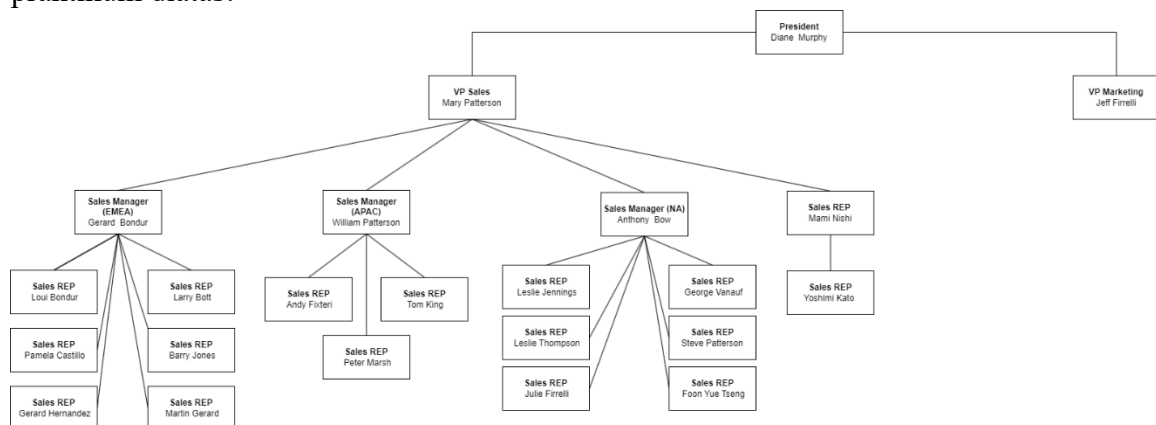
id_manager	Manager	id_staff	staff
1143	Anthony Bow	1165	Leslie Jennings
1143	Anthony Bow	1166	Leslie Thompson
1143	Anthony Bow	1188	Julie Firrelli
1143	Anthony Bow	1216	Steve Patterson
1143	Anthony Bow	1286	Foon Yue Tseng
1143	Anthony Bow	1323	George Vanauf
1002	Diane Murphy	1056	Mary Patterson
1002	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

22 rows in set (0.00 sec)

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

Tugas 2

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



2. Buka **tab baru** pada browser untuk melakukan eksekusi **query** berikut: dari query tersebut menghasilkan jumlah **customer** dari setiap **staff**. Jika perusahaan tersebut memiliki **KPI (Key Performances Indicator)**

```

mysql> 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
-> manager.employeeNumber, manager.firstName, manager.lastName,
-> employee.employeeNumber, employee.firstName, employee.lastName
-> 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
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
1088	William 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

22 rows in set (0.00 sec)

"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?
Jika diranking 3 besar staff berprestasi berdasarkan customer terbanyak diantaranya Pamela Castillo dengan customer 10 orang, George Vanauf dengan customer 8 orang, Foon Yue Tseng dan Gerard Hernandez dengan customer 7 orang
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!

Query untuk menampilkan jumlah customers setiap pegawai

```
mysql> SELECT
-> CONCAT(e.firstName, ' ', e.lastName) AS Nama_Pegawai,
-> COUNT(c.customerNumber) AS Jumlah_Customer
-> FROM employees e
-> LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmp
-> ORDER BY Nama_Pegawai;ber
```

Nama_Pegawai	Jumlah_Customer
Andy Fixter	5
Anthony Bow	0
Barry Jones	9
Diane Murphy	0
Foon Yue Tseng	7
George Vanauf	8
Gerard Bondur	0
Gerard Hernandez	7
Jeff Firrelli	0
Julie Firrelli	6
Larry Bott	8
Leslie Jennings	6
Leslie Thompson	6
Loui Bondur	6
Mami Nishi	5
Martin Gerard	6
Mary Patterson	0
Pamela Castillo	10
Peter Marsh	5
Steve Patterson	6
Tom King	0
William Patterson	0
Yoshimi Kato	0

Berikut ini ranking pegawai dengan jabatan manager dengan perolehan customer terbanyak

- Gerard Bondur = 46 cutomers (ranking 1)
 - William Patterson = 10 customers (ranking 3)
 - Anthony Bow = 39 customers (ranking 2)
3. Analisa kembali data LegendVehicle untuk mendapatkan ranking pegawai berdasarkan KPI "**Jumlah omset yang didapat**". Urutkan ranking pegawai beserta keterangan dana yang didapat!

Query untuk menampilkan omset pegawai berdasarkan pesanan yang dibuat pelanggan

```
mysql> SELECT CONCAT(e.firstName, ' ', e.lastName) AS Nama_Pegawai,
-> COALESCE(SUM(od.quantityOrdered * od.priceEach), 0) AS Total_Omset
-> FROM employees e
-> LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmp
-> LEFT JOIN orders o ON c.customerNumber = o.customerNumber
-> ORDER BY Total_Omset DESC;ON o.orderNumber = od.orderNumber
```

Nama_Pegawai	Total_Omset
Gerard Hernandez	1258577.81
Leslie Jennings	1081530.54
Pamela Castillo	868220.55
Larry Bott	732096.79
Barry Jones	704853.91
George Vanauf	669377.05
Peter Marsh	584593.76
Loui Bondur	569485.75
Andy Fixter	562582.59
Steve Patterson	505875.42
Foon Yue Tseng	488212.67
Mami Nishi	457110.07
Martin Gerard	387477.47
Julie Firrelli	386663.20
Leslie Thompson	347533.03
Diane Murphy	0.00
Mary Patterson	0.00
Jeff Firrelli	0.00
William Patterson	0.00
Gerard Bondur	0.00
Anthony Bow	0.00
Tom King	0.00
Yoshimi Kato	0.00

23 rows in set (0.01 sec)

Rangking 1 : Gerard Hernandez dengan omset 1258577.81

Rangking 2 : Leslie Jennings dengan omset 1081530.54

Rangking 3 : Pamela Castillo dengan omset 868220.55

Ranking 4 : Larry Bott dengan omset 732096.79

Rangking 5 : Barry Jones dengan omset 704853.91

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?

Untuk menampilkan jumlah customer dibutuhkan 5 field diantaranya:

- Field firstName, lastName dari tabel customers untuk menampilkan nama pegawai
- Field customerNumber (untuk menghitung jumlah customers) dari tabel customers
- Field salesRepEmployeeNumber dari tabel customers & field employeeNumber dari tabel employee (untuk menghubungkan tabel employee ke tabel customers)

Untuk menampilkan jumlah omset dibutuhkan 10 field diantaranya:

- Field firstName, lastName dari tabel customers untuk menampilkan nama pegawai
- Field quantityOrdered, priceEach (untuk menghitung omset) dari tabel orderdetails
- Field employeeNumber dari tabel employee
- Field salesRepEmployeeNumber dari tabel customers
- Field customerNumber dari tabel orders dan tabel customers
- Field orderNumber dari tabel orders dan orderdetails (untuk menghubungkan tabel orders dan tabel orderdetails)

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

Query untuk menampilkan omset per tahun dari **Foon Yue Tseng**

```
mysql> SELECT
-> CONCAT(e.firstName, ' ', e.lastName) AS Nama_Pegawai,
-> YEAR(o.orderDate) AS Tahun,
-> COALESCE(SUM(od.quantityOrdered * od.priceEach), 0) 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 CONCAT(e.firstName, ' ', e.lastName) = 'Foon Yue Tseng'
-> GROUP BY Nama_Pegawai, Tahun
-> ORDER BY Tahun;
```

Nama_Pegawai	Tahun	Total_Omset
Foon Yue Tseng	2003	221887.03
Foon Yue Tseng	2004	237255.26
Foon Yue Tseng	2005	29070.38

3 rows in set (0.00 sec)

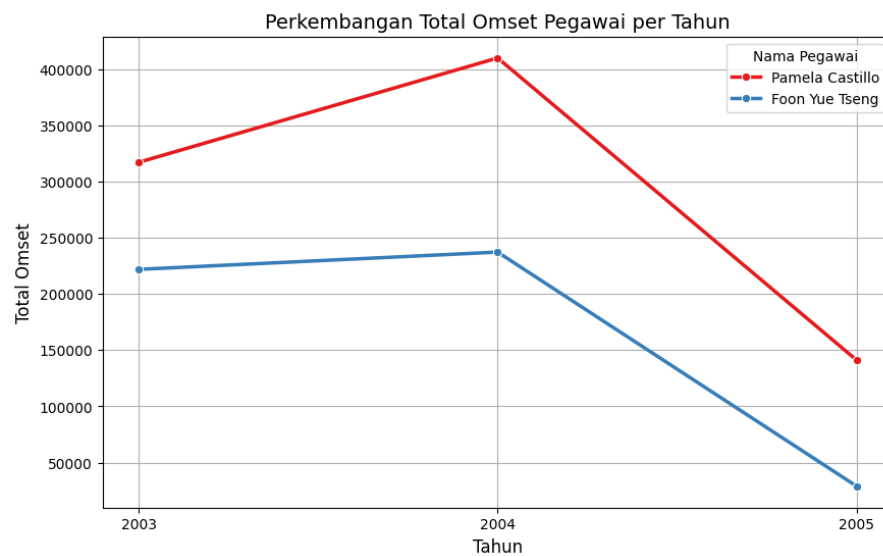
Query untuk menampilkan omset per tahun dari **Pamela Castillo**

```
mysql> SELECT
-> CONCAT(e.firstName, ' ', e.lastName) AS Nama_Pegawai,
-> YEAR(o.orderDate) AS Tahun,
-> COALESCE(SUM(od.quantityOrdered * od.priceEach), 0) 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 CONCAT(e.firstName, ' ', e.lastName) = 'Pamela Castillo'
-> GROUP BY Nama_Pegawai, Tahun
-> ORDER BY Tahun;
```

Nama_Pegawai	Tahun	Total_Omset
Pamela Castillo	2003	317104.78
Pamela Castillo	2004	409910.07
Pamela Castillo	2005	141205.70

3 rows in set (0.00 sec)

Grafik garis kedua pegawai



Analisis

Terjadi kenaikan omset yang drastis pada Pamela sebesar 92805,29 ditahun 2004 dan penurunan omset sebesar 268.704,37 pada tahun 2005.

Terjadi kenaikan omset sedikit sebesar 15368,23 pada Foon ditahun 2004 dan penurunan drastis sebesar 208184,88 ditahun 2005

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.

Analisalah terlebih dahulu:

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

Tabel offices: city, officeCode

Tabel orders: orderDate, orderNumber, customerNumber

Tabel customers: customerNumber, salesRepEmployeeNumber

Tabel employees: employeeNumber, officeCode

Tabel orderdetails: quantityOrdered, priceEach, orderNumber

2. Bentuk query dengan memperhatikan relasi antar tabel

```
SELECT
```

```
  o.city AS `Nama Cabang`,
```

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

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

```
  SUM(CASE WHEN YEAR(odr.orderDate) = 2005 THEN od.quantityOrdered *  
od.priceEach ELSE 0 END) AS `2005`
```

```
FROM orderdetails od
```

```
JOIN orders odr ON od.orderNumber = odr.orderNumber
```

```
JOIN customers c ON odr.customerNumber = c.customerNumber
```

```
JOIN employees e ON c.salesRepEmployeeNumber = e.employeeNumber
```

```
JOIN offices o ON e.officeCode = o.officeCode
```

```
GROUP BY o.city
```

```
ORDER BY o.city;
```

Nama 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

7 rows in set (0.01 sec)

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

1. Menampilkan jumlah product yang terjual pertahun 1936 Mercedes-Benz 500K Special Roadster

```
SELECT
    YEAR(o.orderDate) AS tahun,
    p.productName,
    SUM(od.quantityOrdered) AS total_terjual
FROM orderdetails od
JOIN orders o ON od.orderNumber = o.orderNumber
JOIN products p ON od.productCode = p.productCode
WHERE p.productName = '1936 Mercedes-Benz 500K Special Roadster'
GROUP BY tahun, p.productName
ORDER BY tahun ASC;
```

tahun	productName	total_terjual
2003	1936 Mercedes-Benz 500K Special Roadster	429
2004	1936 Mercedes-Benz 500K Special Roadster	363
2005	1936 Mercedes-Benz 500K Special Roadster	168

3 rows in set (0.00 sec)

2. Menampilkan jumlah product yang terjual pertahun 1974 Ducati 350 Mk3 Desmo

```
SELECT
    YEAR(o.orderDate) AS tahun,
    p.productName,
    SUM(od.quantityOrdered) AS total_terjual
FROM orderdetails od
JOIN orders o ON od.orderNumber = o.orderNumber
JOIN products p ON od.productCode = p.productCode
WHERE p.productName = '1974 Ducati 350 Mk3 Desmo'
GROUP BY tahun, p.productName
ORDER BY tahun ASC;
```

tahun	productName	total_terjual
2003	1974 Ducati 350 Mk3 Desmo	265
2004	1974 Ducati 350 Mk3 Desmo	430
2005	1974 Ducati 350 Mk3 Desmo	203

3 rows in set (0.00 sec)

3. Menampilkan jumlah product yang terjual pertahun 1997 BMW R 1100 S

```
SELECT  
YEAR(o.orderDate) AS tahun,  
p.productName,  
SUM(od.quantityOrdered) AS total_terjual  
FROM orderdetails od  
JOIN orders o ON od.orderNumber = o.orderNumber  
JOIN products p ON od.productCode = p.productCode  
WHERE p.productName = '1997 BMW R 1100 S'  
GROUP BY tahun, p.productName  
ORDER BY tahun ASC;
```

tahun	productName	total_terjual
2003	1997 BMW R 1100 S	344
2004	1997 BMW R 1100 S	418
2005	1997 BMW R 1100 S	271

3 rows in set (0.00 sec)

Dari ketiga query diatas dibutuhkan beberapa field

Tabel order: orderDate, orderNumber

Tabel orderdetails: quantityOrdered, orderNumber, productCode

Tabel products: productName, productCode

Grafik Line Chart untuk menampilkan tren penjualan mobil dari tahun ke tahun berdasarkan data diatas

