JOBSHEET 2
DATABASE OPERASIONAL

Nama : Lyra Faiqah Bilqis
Kelas : 2A SIB / 19

TUGAS 1

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

Tabel 1	Tabel 2	Jenis Relasi
productlines	products	one to many
...	...	...
...	...	...

Tabel 1	Tabel 2	Jenis Relasi	Deskripsi Relasi
productlines	products	One to Many	Satu productline bisa memiliki banyak products
products	orderdetails	One to Many	Satu produk bisa muncul di banyak orderdetails
orders	orderdetails	One to Many	Satu order bisa memiliki banyak item (orderdetails)
customers	orders	One to Many	Satu customer bisa memiliki banyak order
customers	payments	One to Many	Satu customer bisa melakukan banyak pembayaran
employees	customers	One to Many	Satu sales representative bisa menangani banyak customer
employees	offices	Many to one	Banyak pegawai bisa bekerja di satu kantor

3. Analisa jumlah field pada setiap tabel!

Nama Tabel	Jumlah Field
...	...
...	...
...	...

NAMA TABEL	JUMLAH FIELD
orderdetails	5
customers	9
offices	7

payment	4
products	9
productslines	4
employees	8
orders	6

TUGAS 2

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

Manajer ID	Manajer Name	Staff ID	Staff Name	Customer Name
1002	John Lee	1056	Sarah Tan	Global Corp
1002	John Lee	1056	Sarah Tan	Techworld Inc
1002	John Lee	1078	Rina Sari	Alpha Supplies
1020	Maria Gomez	1085	Andi WIjaya	FutureNet
1020	Maria Gomez	1085	Andi WIjaya	IndoMarket
1020	Maria Gomez	1087	Bambang Hadi	SariLoka

TUGAS 3

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

employeeNumber	staff	total_cust
1401	Pamelo Castillo	10

Dengan query :

```
SELECT e.employeeNumber, CONCAT(e.firstName, ' ', e.lastName) AS staff, COUNT(c.customerNumber) AS total_cust 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 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!

Tabel :

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

Query :

```
-- Step 1: Buat total customer per employee WITH customer_count AS ( SELECT e.employeeNumber, CONCAT(e.firstName, ' ', e.lastName) AS employee_name, e.reportsTo, COUNT(c.customerNumber) AS total_customer FROM employees e LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber GROUP BY e.employeeNumber ), -- Step 2: Gabungkan customer dari bawahan ke atasannya all_customer_count AS ( SELECT cc1.employeeNumber, cc1.employee_name, cc1.total_customer + IFNULL(SUM(cc2.total_customer), 0) AS total_customer_with_sub FROM customer_count cc1 LEFT JOIN customer_count cc2 ON cc2.reportsTo = cc1.employeeNumber GROUP BY cc1.employeeNumber, cc1.employee_name, cc1.total_customer ) -- Step 3: Tampilkan hasil ranking SELECT employeeNumber, employee_name, total_customer_with_sub AS total_customers FROM all_customer_count ORDER BY total_customers DESC;
```

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

Tabel :

employeeNumber	employee_name	total_omzet 1
1370	Gerard Hernandez	19794299.54
1165	Leslie Jennings	11402222.02
1401	Pamela Castillo	2382284.44
1611	Andy Fixter	2118017.43
1612	Peter Marsh	2022883.40
1337	Loui Bondur	1962423.63
1501	Larry Bott	1958589.85
1504	Barry Jones	1866060.52
1621	Mami Nishi	1681538.99
1323	George Vanauf	1645493.99
1286	Foon Yue Tseng	1542942.88
1216	Steve Patterson	1355621.27
1702	Martin Gerard	955915.50
1188	Julie Firrelli	948732.57
1166	Leslie Thompson	869050.34

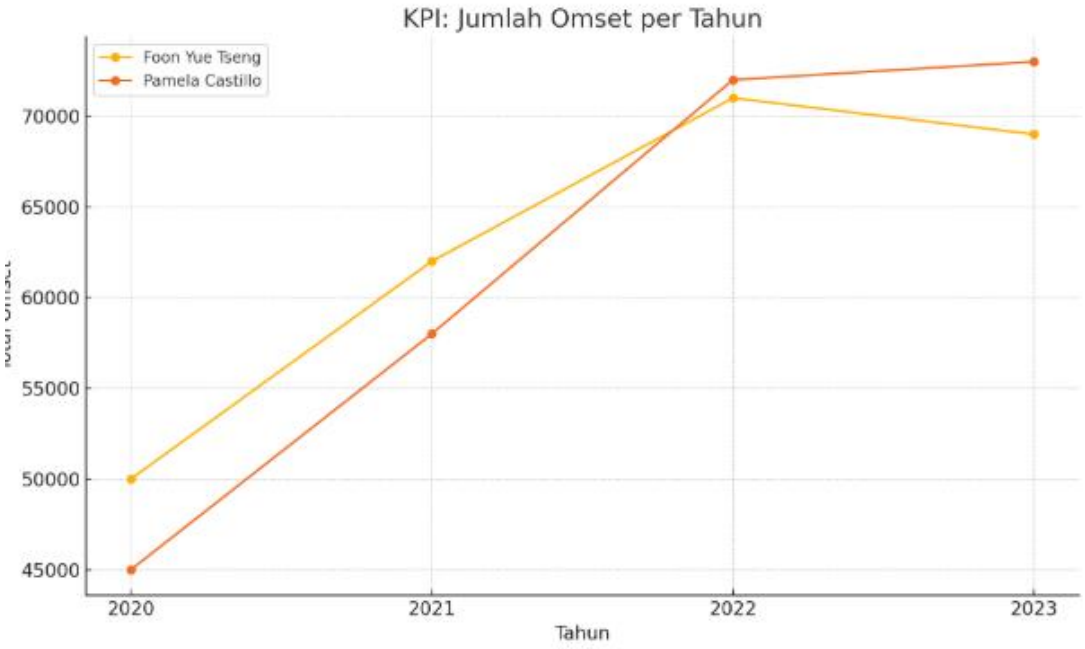
Dengan query :

```
SELECT e.employeeNumber, CONCAT(e.firstName, ' ', e.lastName) AS employee_name, SUM(p.amount) AS total_omzet FROM employees e JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber JOIN orders o ON c.customerNumber = o.customerNumber JOIN payments p ON c.customerNumber = p.customerNumber GROUP BY e.employeeNumber, employee_name ORDER BY total_omzet DESC;
```

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? Karena KPI 1 dan KPI 2 membutuhkan informasi yang mirip (pegawai dan data agregat), maka jumlah field total yang dibutuhkan adalah 4

Field	Keterangan
`employeeNumber`	ID Pegawai
`employeeName`	Nama Pegawai (bisa concat)
`total_customer`	Jumlah customer (untuk KPI 1)
`total_omset`	Jumlah omset (untuk KPI 2)

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



STUDI KASUS

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

Tabel	Field yang Diperlukan	Keterangan
`offices`	`officeCode`, `city`	Untuk identifikasi cabang
`employees`	`employeeNumber`, `officeCode`	Relasi pegawai dengan kantor
`customers`	`customerNumber`, `salesRepEmployeeNumber`	Relasi pelanggan dengan pegawai
`orders`	`orderNumber`, `orderDate`, `customerNumber`	Relasi order dengan customer & tanggal
`orderdetails`	`orderNumber`, `quantityOrdered`, `priceEach`	Hitung total penjualan

2. Bentuk query dengan memperhatikan relasi antar tabel.
Tabel :

nama_cabang	tahun ▲ 2	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

Query :

```
SELECT o.city AS nama_cabang, YEAR(od.orderDate) AS tahun, SUM(odt.quantityOrdered * odt.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 od ON c.customerNumber = od.customerNumber JOIN orderdetails odt ON od.orderNumber = odt.orderNumber WHERE YEAR(od.orderDate) IN (2003, 2004, 2005) GROUP BY o.city, YEAR(od.orderDate) ORDER BY o.city, tahun;
```

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 :

Tabel	Field yang Diperlukan
`orderdetails`	`productCode` , `quantityOrdered`
`orders`	`orderDate`
`products`	`productName`

Tabel :

productName	tahun ▲ 1	total_order ▼ 2
1992 Ferrari 360 Spider red	2003	672
1936 Mercedes-Benz 500K Special Roadster	2003	429
1964 Mercedes Tour Bus	2003	427
1940s Ford truck	2003	408
1926 Ford Fire Engine	2003	393
1956 Porsche 356A Coupe	2003	389
1948 Porsche Type 356 Roadster	2003	382
1965 Aston Martin DB5	2003	382
1996 Peterbilt 379 Stake Bed with Outrigger	2003	373
1950's Chicago Surface Lines Streetcar	2003	372
1968 Dodge Charger	2003	371
1939 Cadillac Limousine	2003	369
Diamond T620 Semi-Skirted Tanker	2003	369
1948 Porsche 356-A Roadster	2003	366
1995 Honda Civic	2003	366
1937 Lincoln Berline	2003	365
1998 Chrysler Plymouth Prowler	2003	363
1954 Greyhound Scenicruiser	2003	363
1969 Ford Falcon	2003	363
1928 British Royal Navy Airplane	2003	363
1917 Maxwell Touring Car	2003	360
1966 Shelby Cobra 427 S/C	2003	357
1999 Yamaha Speed Boat	2003	357
18th century schooner	2003	356
1932 Alfa Romeo 8C2300 Spider Sport	2003	354

Query :

```
SELECT p.productName, YEAR(o.orderDate) AS tahun, SUM(od.quantityOrdered) AS total_order FROM orderdetails od JOIN orders o ON od.orderNumber = o.orderNumber JOIN products p ON od.productCode = p.productCode WHERE YEAR(o.orderDate) IN (2003, 2004, 2005) GROUP BY p.productName, YEAR(o.orderDate) ORDER BY tahun, total_order DESC;
```