JOBSHEET 2

DATABASE OPERASIONAL

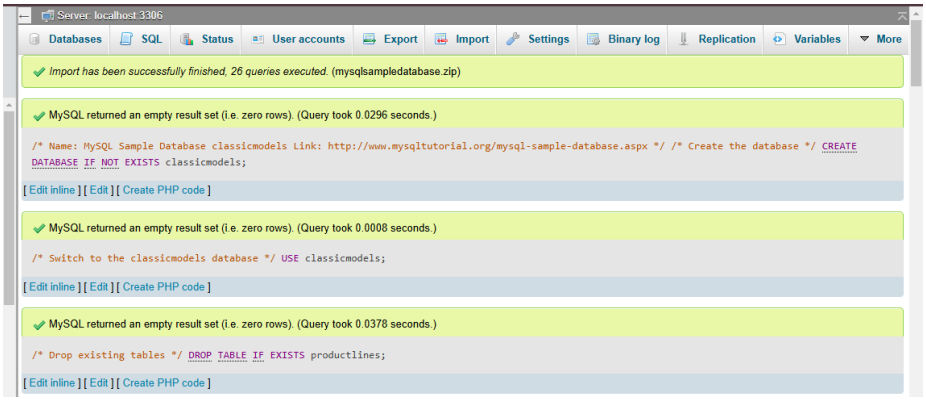
Nama : Dahniar Davina

Kelas : SIB – 2A

No Absen : 09

A. TUGAS 1

1. Import data perusahaan tersebut pada DBMS MySQL!



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

   - 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)
```

- 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 (0.00 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)
```

TABEL 1	TABEL 2	RELASI
Productlines	Products	One to many
Customers	Orders	One to many
Orders	Orderdetails	One to many
Products	Orderdetails	One to many
Customers	Payments	One to many
Offices	Employees	One to many

3. Analisa jumlah field pada tiap tabel!

NAMA TABEL	JUMLAH FIELD
productlines	4
products	9
orderdetails	5
offices	9
customers	13
payments	4
orders	7
employees	8

B. ANALISIS DATA

a) Praktikum 1

1. Jalankan query data Perusahaan LegendVehicle

```
mysql> SELECT * FROM
    -> employees employe, employees manager, customers cust
    -> WHERE employe.reportsTo = manager.employeeNumber
    -> AND employe.employeeNumber = cust.salesRepEmployeeNumber;
+----------------+----------+-----------+-----------+---------------------------------+------------+-----------+----------+----------------+----------+-----------+--
--------+---------------------------------+------------+-----------+----------------------+----------------+------------------------------------+-----------------+----
-----------+-------------------+------------------------------+------------------------------+------------------+--------------+-------------+-----------+--------
-------------+-------------+
| employeeNumber | lastName  | firstName | extension | email                           | officeCode | reportsTo | jobTitle  | employeeNumber | lastName  | firstName | e
xtension | email                           | officeCode | reportsTo | jobTitle             | customerNumber | customerName                       | contactLastName | con
tactFirstName | phone             | addressLine1                 | addressLine2                 | city             | state        | postalCode | country   | salesRepE
mployeeNumber | creditLimit |
+----------------+----------+-----------+-----------+---------------------------------+------------+-----------+----------+----------------+----------+-----------+--
--------+---------------------------------+------------+-----------+----------------------+----------------+------------------------------------+-----------------+----
-----------+-------------------+------------------------------+------------------------------+------------------+--------------+-------------+-----------+--------
-------------+-------------+
|           1165 | Jennings  | Leslie    | x3291     | ljennings@classicmodelcars.com  | 1          |      1143 | Sales Rep |           1143 | Bow       | Anthony   | x
5428     | abow@classicmodelcars.com       | 1          |      1056 | Sales Manager (NA)   |            124 | Mini Gifts Distributors Ltd.       | Nelson          | Sus
an            | 4155551450        | 5677 Strong St.              | NULL                         | San Rafael       | CA           | 97562       | USA       |
         1165 |   210500.00 |
|           1165 | Jennings  | Leslie    | x3291     | ljennings@classicmodelcars.com  | 1          |      1143 | Sales Rep |           1143 | Bow       | Anthony   | x
5428     | abow@classicmodelcars.com       | 1          |      1056 | Sales Manager (NA)   |            129 | Mini Wheels Co.                    | Murphy          | Jul
ie            | 6505555787        | 5557 North Pendale Street    | NULL                         | San Francisco    | CA           | 94217       | USA       |
         1165 |    64600.00 |
```

```
5408     | gbondur@classicmodelcars.com    | 4          |      1056 | Sale Manager (EMEA)  |            458 | Corrida Auto Replicas, Ltd         | Sommer          | Mar
tín           | (91) 555 22 82    | C/ Araquil, 67               | NULL                         | Madrid           | NULL         | 28023       | Spain     |
         1702 |   104600.00 |
|           1702 | Gerard    | Martin    | x2312     | mgerard@classicmodelcars.com    | 4          |      1102 | Sales Rep |           1102 | Bondur    | Gerard    | x
5408     | gbondur@classicmodelcars.com    | 4          |      1056 | Sale Manager (EMEA)  |            484 | Iberia Gift Imports, Corp.         | Roel            | Jos
é Pedro       | (95) 555 82 82    | C/ Romero, 33                | NULL                         | Sevilla          | NULL         | 41101       | Spain     |
         1702 |    65700.00 |
+----------------+----------+-----------+-----------+---------------------------------+------------+-----------+----------+----------------+----------+-----------+--
--------+---------------------------------+------------+-----------+----------------------+----------------+------------------------------------+-----------------+----
-----------+-------------------+------------------------------+------------------------------+------------------+--------------+-------------+-----------+--------
-------------+-------------+
100 rows in set (0.00 sec)
```

2. Buka tab baru pada browser dan eksekusi query

```
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)
```

b) TUGAS 2

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

```
mysql> SELECT
    -> e1.employeeNumber AS id_staff,
    -> CONCAT(e1.firstName, ' ', e1.lastName) AS Nama_Staff,
    -> (SELECT CONCAT(e2.firstName, ' ', e2.lastName)
    -> FROM employees e2
    -> WHERE e2.employeeNumber = e1.reportsTo) AS Nama_Manager
    -> FROM employees e1
    -> ORDER BY Nama_Manager, Nama_Staff;
+----------+-------------------+-------------------+
| id_staff | Nama_Staff        | Nama_Manager      |
+----------+-------------------+-------------------+
|     1002 | Diane Murphy      | NULL              |
|     1286 | Foon Yue Tseng    | Anthony Bow       |
|     1323 | George Vanauf     | Anthony Bow       |
|     1188 | Julie Firrelli    | Anthony Bow       |
|     1165 | Leslie Jennings   | Anthony Bow       |
|     1166 | Leslie Thompson   | Anthony Bow       |
|     1216 | Steve Patterson   | Anthony Bow       |
|     1076 | Jeff Firrelli     | Diane Murphy      |
|     1056 | Mary Patterson    | Diane Murphy      |
|     1504 | Barry Jones       | Gerard Bondur     |
|     1370 | Gerard Hernandez  | Gerard Bondur     |
|     1501 | Larry Bott        | Gerard Bondur     |
|     1337 | Loui Bondur       | Gerard Bondur     |
|     1702 | Martin Gerard     | Gerard Bondur     |
|     1401 | Pamela Castillo   | Gerard Bondur     |
|     1625 | Yoshimi Kato      | Mami Nishi        |
|     1143 | Anthony Bow       | Mary Patterson    |
|     1102 | Gerard Bondur     | Mary Patterson    |
|     1621 | Mami Nishi        | Mary Patterson    |
|     1088 | William Patterson | Mary Patterson    |
|     1611 | Andy Fixter       | William Patterson |
|     1612 | Peter Marsh       | William Patterson |
|     1619 | Tom King          | William Patterson |
+----------+-------------------+-------------------+
23 rows in set (0.00 sec)
```

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,
    -> 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, employee.employeeNumber
    -> 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 |
|       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 |
+------------+------------------+----------+------------------+------------+
22 rows in set (0.00 sec)
```

dan 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!

c) TUGAS 3

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

```
mysql> SELECT
    -> employee.employeeNumber AS id_staff,
    -> CONCAT(employee.firstName, " ", employee.lastName) AS staff,
    -> COUNT(cust.customerNumber) AS total_cust
    -> FROM employees employee
    -> LEFT JOIN employees sub_employee
    -> ON employee.employeeNumber = sub_employee.reportsTo
    -> LEFT JOIN customers cust
    -> ON employee.employeeNumber = cust.salesRepEmployeeNumber
    -> WHERE sub_employee.employeeNumber IS NULL
    -> GROUP BY employee.employeeNumber
    -> ORDER BY total_cust DESC
    -> LIMIT 1;
+----------+-----------------+------------+
| id_staff | staff           | total_cust |
+----------+-----------------+------------+
|     1401 | Pamela Castillo |         10 |
+----------+-----------------+------------+
1 row in set (0.00 sec)
```

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!

```
mysql> WITH TotalCustomer AS (
    -> SELECT
    -> e.employeeNumber AS id_pegawai,
    -> CONCAT(e.firstName, ' ', e.lastName) AS Nama_Pegawai,
    -> e.reportsTo AS id_atasan,
    -> COUNT(c.customerNumber) AS customer_pribadi
    -> FROM employees e
    -> LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
    -> GROUP BY e.employeeNumber
    -> ),
    ->
```

```
    -> CustomerHierarchy AS (
    -> SELECT
    -> tc1.id_pegawai,
    -> tc1.Nama_Pegawai,
    -> tc1.customer_pribadi + IFNULL(SUM(tc2.customer_pribadi), 0) AS total_custome
r_hierarchy
    -> FROM TotalCustomer tc1
    -> LEFT JOIN TotalCustomer tc2 ON tc2.id_atasan = tc1.id_pegawai
    -> GROUP BY tc1.id_pegawai
    -> )
    ->
    -> SELECT
    -> id_pegawai,
    -> Nama_Pegawai,
    -> total_customer_hierarchy
    -> FROM CustomerHierarchy
    -> ORDER BY total_customer_hierarchy DESC;
```

```
+------------+--------------------+---------------------------+
| id_pegawai | Nama_Pegawai       | total_customer_hierarchy  |
+------------+--------------------+---------------------------+
|       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 |
|       1056 | Mary Patterson     |                         5 |
|       1611 | Andy Fixter        |                         5 |
|       1612 | Peter Marsh        |                         5 |
|       1621 | Mami Nishi         |                         5 |
|       1002 | Diane Murphy       |                         0 |
|       1076 | Jeff Firrelli      |                         0 |
|       1619 | Tom King           |                         0 |
|       1625 | Yoshimi Kato       |                         0 |
+------------+--------------------+---------------------------+
23 rows in set (0.01 sec)
```

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

```
mysql> WITH TotalOmset AS (
    -> SELECT
    -> e.employeeNumber AS id_pegawai,
    -> CONCAT(e.firstName, ' ', e.lastName) AS Nama_Pegawai,
    -> e.reportsTo AS id_atasan,
    -> SUM(p.amount) AS omset_pribadi
    -> FROM employees e
    -> LEFT JOIN customers c ON e.employeeNumber = c.salesRepEmployeeNumber
    -> LEFT JOIN payments p ON c.customerNumber = p.customerNumber
    -> GROUP BY e.employeeNumber
    -> ),
    ->
    -> OmsetHierarchy AS (
    -> SELECT
    -> t1.id_pegawai,
    -> t1.Nama_Pegawai,
    -> t1.omset_pribadi + IFNULL(SUM(t2.omset_pribadi), 0) AS total_omset_hierarchy

    -> FROM TotalOmset t1
    -> LEFT JOIN TotalOmset t2 ON t2.id_atasan = t1.id_pegawai
    -> GROUP BY t1.id_pegawai
    -> )
    ->
    -> SELECT
    -> id_pegawai,
    -> Nama_Pegawai,
    -> total_omset_hierarchy AS total_omset,
    -> RANK() OVER (ORDER BY total_omset_hierarchy DESC) AS ranking
    -> FROM OmsetHierarchy
    -> ORDER BY total_omset_hierarchy DESC;
```

id_pegawai	Nama_Pegawai	total_omset	ranking
1370	Gerard Hernandez	1112003.81	1
1165	Leslie Jennings	989906.55	2
1401	Pamela Castillo	750201.87	3
1501	Larry Bott	686653.25	4
1504	Barry Jones	637672.65	5
1323	George Vanauf	584406.80	6
1337	Loui Bondur	569485.75	7
1611	Andy Fixter	509385.82	8
1612	Peter Marsh	497907.16	9
1286	Foon Yue Tseng	488212.67	10
1621	Mami Nishi	457110.07	11
1216	Steve Patterson	449219.13	12
1702	Martin Gerard	387477.47	13
1188	Julie Firrelli	386663.20	14
1166	Leslie Thompson	347533.03	15
1002	Diane Murphy	NULL	16
1056	Mary Patterson	NULL	16
1076	Jeff Firrelli	NULL	16
1088	William Patterson	NULL	16
1102	Gerard Bondur	NULL	16
1143	Anthony Bow	NULL	16
1619	Tom King	NULL	16
1625	Yoshimi Kato	NULL	16

23 rows in set (0.00 sec)

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?

   a. Jumlah customer yang bertransaksi = 5 field

No	Field	Sumber	Keterangan
1	Id_pegawai	Employees	ID unik pegawai
2	Nama_pegawai	Employees	Gabungan (firstname+lastname)
3	Id_atasan	Employees	ID atasan pegawai
4	Customer_pribadi	Customers	Jumlah cust langsung yang dimiliki pegawai
5	Total_customer_hierarchy	Hasil perhitungan	Total customer pegawai+bawahan

   b. Jumlah omset yang didapat = 6 field

No	Field	Sumber	Keterangan
1	Id_pegawai	Employees	ID unik pegawai
2	Nama_pegawai	Employees	Gabungan (firstname+lastname)
3	Id_atasan	Employees	ID unik atasan / manager
4	Omset_pribadi	Payments	Omset langsung dari cust yang ditangani manager
5	Total_omset_hierarchy	Hasil perhitungan	Omset pegawai+omset bawahan
6	rangking	Hasil perhitungan	Rangking pegawai berdasar total omset

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

```
mysql> SELECT
    -> e.employeeNumber AS id_pegawai,
    -> CONCAT(e.firstName, ' ', e.lastName) AS Nama_Pegawai,
    -> YEAR(p.paymentDate) AS Tahun,
    -> SUM(p.amount) AS Total_Omset
    -> 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(p.paymentDate)
    -> ORDER BY Tahun, Nama_Pegawai;
```

id_pegawai	Nama_Pegawai	Tahun	Total_Omset
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	23187.02

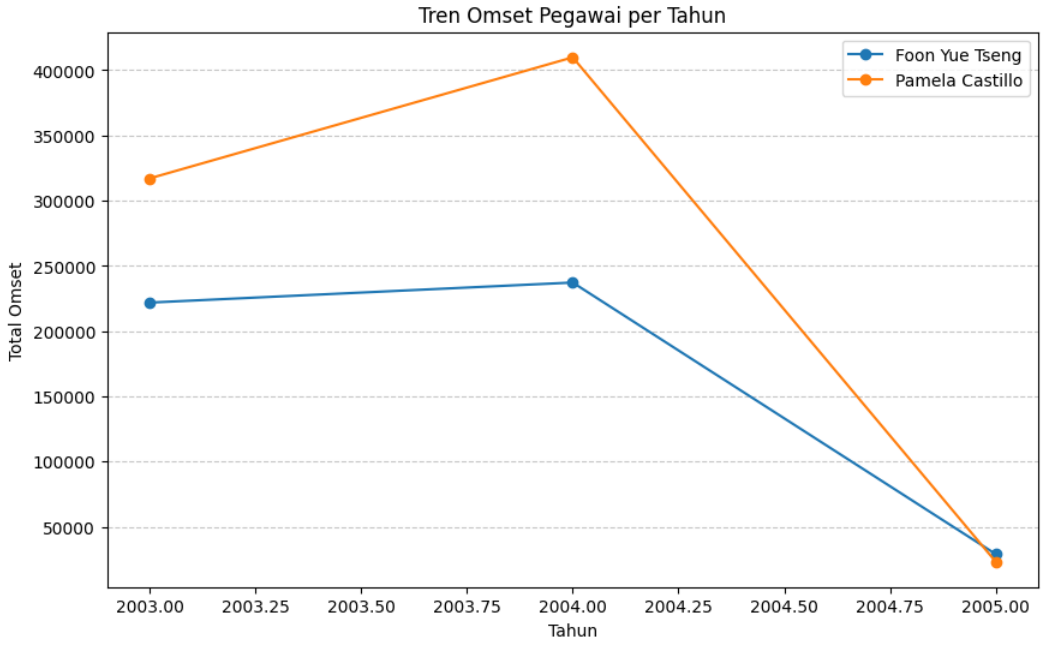
6 rows in set (0.01 sec)

```python
plt.figure(figsize=(10, 6))

# Buat line chart
for name in df["Nama_Pegawai"].unique():
    subset = df[df["Nama_Pegawai"] == name]
    plt.plot(subset["Tahun"], subset["Total_Omset"], marker="o", linestyle="-", label=name)

# Tambahkan label dan judul
plt.xlabel("Tahun")
plt.ylabel("Total Omset")
plt.title("Tren Omset Pegawai per Tahun")
plt.legend()
plt.grid(axis="y", linestyle="--", alpha=0.7)

# Tampilkan plot
plt.show()
```

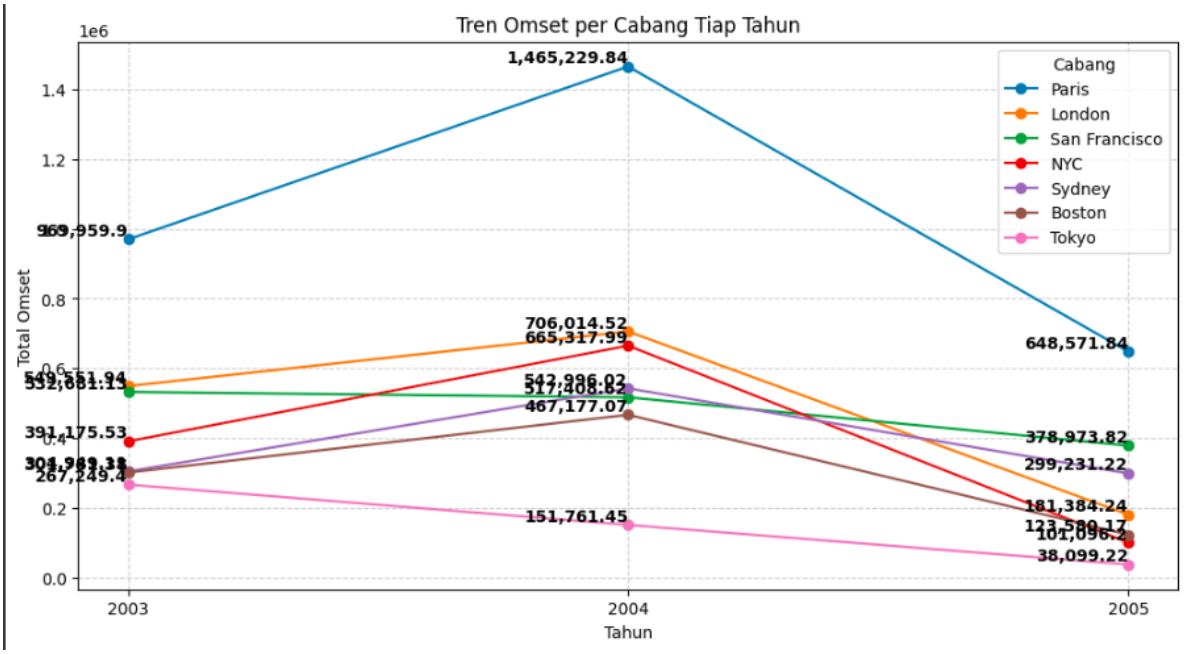


C. STUDI KASUS

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:

Cabang	2003	2004	2005
Paris	969959.90	1465229.84	648571.84
London	549551.94	706014.52	181384.24
San Francisco	532681.13	517408.62	378973.82
NYC	391175.53	665317.99	101096.20
Sydney	304949.11	542996.02	299231.22
Boston	301781.38	467177.07	123580.17
Tokyo	267249.40	151761.45	38099.22



Analisalah terlebih dahulu :

1. Field apa saja yang diperlukan untuk menampilkan penjualan di setiap cabang. Untuk melihat perkembangan penjualan (omset) di setiap cabang per tahun, field yang diperlukan :
   - ➢ offices.city → Nama cabang (lokasi kantor).

- orders.orderDate → Tanggal pemesanan untuk menentukan tahun.
- orderdetails.quantityOrdered → Jumlah barang yang dibeli.
- orderdetails.priceEach → Harga per unit barang.
- (Omset dihitung sebagai quantityOrdered * priceEach)

2. Bentuk query dengan memperhatikan relasi antar tabel.

```
mysql> SELECT
    ->     o.city AS cabang,
    ->     YEAR(od.orderDate) AS tahun,
    ->     SUM(odt.quantityOrdered * odt.priceEach) AS total_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 tahun, total_omset DESC;
+---------------+-------+-------------+
| cabang        | tahun | total_omset |
+---------------+-------+-------------+
| Paris         |  2003 |   969959.90 |
| London        |  2003 |   549551.94 |
| San Francisco |  2003 |   532681.13 |
| NYC           |  2003 |   391175.53 |
| Sydney        |  2003 |   304949.11 |
| Boston        |  2003 |   301781.38 |
| Tokyo         |  2003 |   267249.40 |
| Paris         |  2004 |  1465229.84 |
| London        |  2004 |   706014.52 |
| NYC           |  2004 |   665317.99 |
| Sydney        |  2004 |   542996.02 |
| San Francisco |  2004 |   517408.62 |
| Boston        |  2004 |   467177.07 |
| Tokyo         |  2004 |   151761.45 |
| Paris         |  2005 |   648571.84 |
| San Francisco |  2005 |   378973.82 |
| Sydney        |  2005 |   299231.22 |
| London        |  2005 |   181384.24 |
| Boston        |  2005 |   123580.17 |
| NYC           |  2005 |   101096.20 |
| Tokyo         |  2005 |    38099.22 |
+---------------+-------+-------------+
21 rows in set (0.08 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.

PRODUK TERLARIS BERDASARKAN OMSET (5 TAHUN TERAKHIR)

- Laporan ini bertujuan untuk mengidentifikasi **10 produk dengan total omset tertinggi dalam 5 tahun terakhir** berdasarkan data OLTP. Analisis ini

memberikan wawasan tentang **produk dengan permintaan tertinggi** serta membantu dalam strategi pemasaran dan perencanaan stok.

- Tabel yang Digunakan =

1. products → Menyimpan informasi produk seperti kode dan nama.
2. orders → Menyimpan data pesanan pelanggan, termasuk tanggal pemesanan.
3. orderdetails → Menghubungkan pesanan dengan produk dan mencatat jumlah unit serta harga satuan.

- Field yang digunakan dalam Query =

Field	Deskripsi
productCode	Kode unik produk
productName	Nama produk
quantityOrdered	Jumlah unit produk yang dipesan
priceEach	Harga perunit dari produk
orderDate	Tanggal pesanan dibuat
Total_terjual	Total unit terjual (SUM(quantityOrdered))
Total_omset	Total pendapatan (SUM(quantityOrdered * priceEach))

- Query SQL

```
mysql> SELECT
    -> p.productCode AS kode_produk,
    -> p.productName AS nama_produk,
    -> SUM(od.quantityOrdered) AS total_terjual,
    -> SUM(od.quantityOrdered * od.priceEach) AS total_omset
    -> FROM orderdetails od
    -> JOIN products p ON od.productCode = p.productCode
    -> JOIN orders o ON od.orderNumber = o.orderNumber
    -> WHERE o.orderDate >= '2003-01-01' -- Sesuaikan dengan tahun awal yang tersedia
    -> GROUP BY p.productCode, p.productName
    -> ORDER BY total_omset DESC
    -> LIMIT 10;
+-------------+----------------------------------+--------------+-------------+
| kode_produk | nama_produk                      | total_terjual | total_omset |
+-------------+----------------------------------+--------------+-------------+
| S18_3232    | 1992 Ferrari 360 Spider red      |         1808 |   276839.98 |
| S12_1108    | 2001 Ferrari Enzo                |         1019 |   190755.86 |
| S10_1949    | 1952 Alpine Renault 1300         |          961 |   190017.96 |
| S10_4698    | 2003 Harley-Davidson Eagle Drag Bike |          985 |   170686.00 |
| S12_1099    | 1968 Ford Mustang                |          933 |   161531.48 |
| S12_3891    | 1969 Ford Falcon                 |          965 |   152543.02 |
| S18_1662    | 1980s Black Hawk Helicopter      |         1040 |   144959.91 |
| S18_2238    | 1998 Chrysler Plymouth Prowler   |          986 |   142530.63 |
| S18_1749    | 1917 Grand Touring Sedan         |          918 |   140535.60 |
| S12_2823    | 2002 Suzuki XREO                 |         1028 |   135767.03 |
+-------------+----------------------------------+--------------+-------------+
10 rows in set (0.04 sec)
```

➢ Insight Data

1. Ferrari 360 Spider Red (1992) adalah Produk dengan Omset Tertinggi

   Terjual 1.088 unit dengan total omset $276,839.98

   Menunjukkan bahwa mobil sport klasik Ferrari tetap diminati dalam 5 tahun terakhir.

2. Ferrari Enzo (2001) & Alpine Renault (1952) Juga Masuk Peringkat Atas

   Ferrari Enzo terjual 1.019 unit dengan omset $190,755.86, menunjukkan minat tinggi terhadap Ferrari modern.

   Alpine Renault 1300 membuktikan bahwa mobil klasik tetap populer di kalangan kolektor.

3. Ford Mustang & Ford Falcon Mendapat Peminat Tinggi

   Ford Mustang (1968) terjual 933 unit, membuktikan bahwa model muscle car klasik masih diminati.

   Ford Falcon (1969) juga menempati posisi tinggi dengan 965 unit terjual.

4. Produk Non-Mobil Juga Masuk Daftar

   Black Hawk Helicopter (1980s) menunjukkan bahwa mainan replika helikopter juga memiliki pasar yang kuat dengan omset $144,959.11.

➢ Visualisasi Data

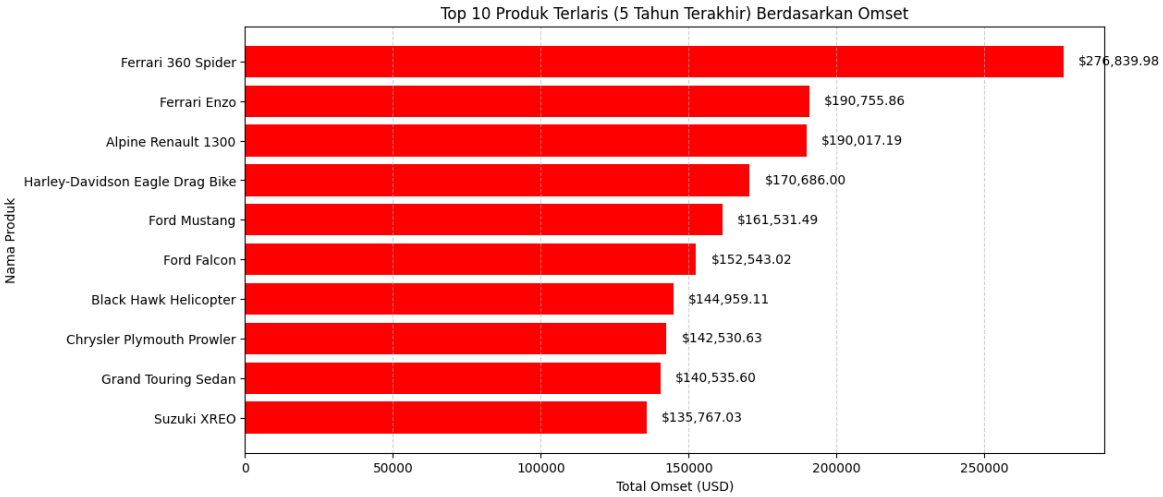
```python
import matplotlib.pyplot as plt

# Data dari tabel di atas
produk = ["Ferrari 360 Spider", "Ferrari Enzo", "Alpine Renault 1300", "Harley-Davidson Eagle Drag Bike",
          "Ford Mustang", "Ford Falcon", "Black Hawk Helicopter", "Chrysler Plymouth Prowler",
          "Grand Touring Sedan", "Suzuki XREO"]
omset = [276839.98, 190755.86, 190017.19, 170686.00, 161531.49, 152543.02, 144959.11, 142530.63, 140535.60, 135767.03]

# Membuat bar chart
plt.figure(figsize=(12, 6))
plt.barh(produk[::-1], omset[::-1], color='red')  # Dibalik agar produk teratas ada di atas
plt.xlabel("Total Omset (USD)")
plt.ylabel("Nama Produk")
plt.title("Top 10 Produk Terlaris (5 Tahun Terakhir) Berdasarkan Omset")
plt.grid(axis='x', linestyle='--', alpha=0.6)

# Menampilkan nilai omset di setiap bar
for i, v in enumerate(omset[::-1]):
    plt.text(v + 5000, i, f"${v:,.2f}", va='center', fontsize=10)

plt.show()
```



- Penjelasan Grafik

1. Produk dengan omset tertinggi (Ferrari 360 Spider) berada di bagian atas.
2. Semua produk memiliki omset di atas $130,000, menunjukkan konsistensi dalam penjualan tinggi.