

**MODUL PRAKTIKUM MATA KULIAH
DATA WAREHOUSE**

DATA WAREHOUSE FOR BUSINESS INTELLIGENCE

**Disusun Oleh:
Joko Suntoro, M.Kom.**



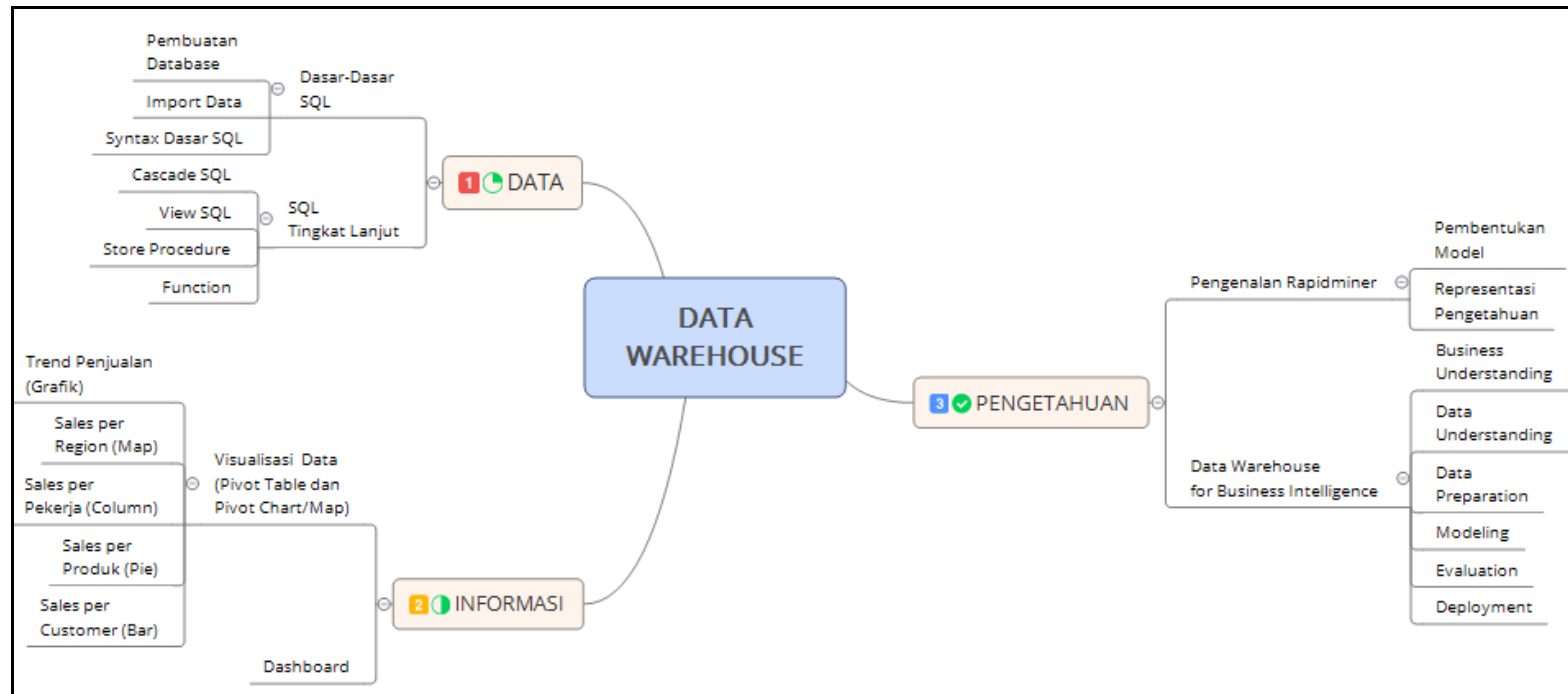
**FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI
UNIVERSITAS SEMARANG
SEMARANG
2019**

DAFTAR ISI

| | |
|---|----|
| DAFTAR ISI..... | 2 |
| MIND MAP PRAKTIKUM DATA WAREHOUSE | 4 |
| BAB 1 DASAR-DASAR SQL | 5 |
| 1.1 Pembuatan Database | 5 |
| 1.2 Import Data | 7 |
| 1.3 Syntax Dasar MySQL..... | 9 |
| 1.3.1 Melihat Struktur Tabel | 9 |
| 1.3.2 Menampilkan Data | 11 |
| 1.3.3 Menambahkan Data | 14 |
| 1.3.4 Mengubah Data | 16 |
| 1.3.5 Menghapus data | 17 |
| BAB 2 SQL TINGKAT LANJUT | 18 |
| 2.1 Cascade pada MySQL..... | 18 |
| 2.2 View pada MySQL | 20 |
| BAB 3 VISUALISASI DATA (PIVOT TABLE, CHART, MAP DAN DASHBOARD)..... | 21 |
| 3.1 Pivot Table dan Pivot Chart Trend Penjualan..... | 21 |
| 3.2 Pivot Table dan Maps Chart Sales per Region | 25 |
| 3.3 Pivot Table dan Pivot Chart Sales per Pekerja | 29 |
| 3.4 Pivot Table dan Pivot Chart Sales per Produk..... | 30 |
| 3.5 Pivot Table dan Pivot Chart Sales per Customer..... | 32 |
| 3.6 Pembuatan Dashboard | 33 |
| BAB 4 PENGENALAN RAPIDMINER DAN REPRESENTASI PENGETAHUAN..... | 37 |
| 4.1 Pengenalan Rapidminer | 37 |

| | | |
|-----------------|--|----|
| 4.2 | Representasi Pengetahuan..... | 37 |
| 4.2.1 | Estimasi..... | 37 |
| 4.2.2 | Prediksi..... | 39 |
| 4.2.3 | Klasifikasi | 40 |
| 4.2.4 | Klustering..... | 41 |
| BAB 5 | DATA WAREHOUSE FOR BUSINESS INTELLIGENCE (Studi Kasus: Klustering Data Pekerja) | 43 |
| 5.1 | Konteks dan Perspektif | 43 |
| 5.2 | Studi Kasus | 43 |
| 5.2.1 | Business Understanding..... | 43 |
| 5.2.2 | Data Understanding..... | 44 |
| 5.2.3 | Data Preparation..... | 44 |
| 5.2.4 | Modeling | 45 |
| 5.2.5 | Evaluation | 46 |
| 5.2.6 | Deployment..... | 47 |
| TENTANG PENULIS | | 48 |

MIND MAP PRAKTIKUM DATA WAREHOUSE



Catatan: modul ini bersifat general dan ada beberapa langkah yang akan dijelaskan secara rinci saat di kelas teori dan praktikum

BAB 1

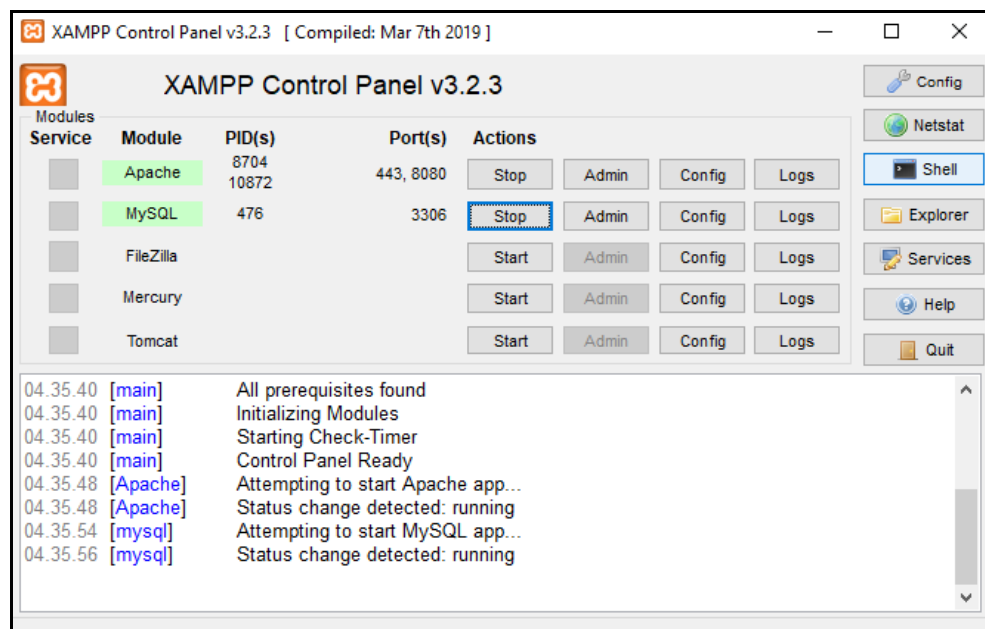
DASAR-DASAR SQL

Tujuan dari data warehouse adalah melakukan *collecting* data dari berbagai *source system/database* untuk memudahkan pengaksesan informasi dan membantu proses analisis data guna pengambilan keputusan. Data warehouse bersifat OLAP (*On Line Analytical Processing*), yang menekankan pada analisis data. Sedangkan database bersifat OLTP (*On Line Transaction Processing*), yang menekankan pada pemrosesan data transaksional. Dalam bab ini akan direfresh kembali materi kuliah pemrograman database tentang pembuatan database, import data, query-query dasar dalam MySQL.

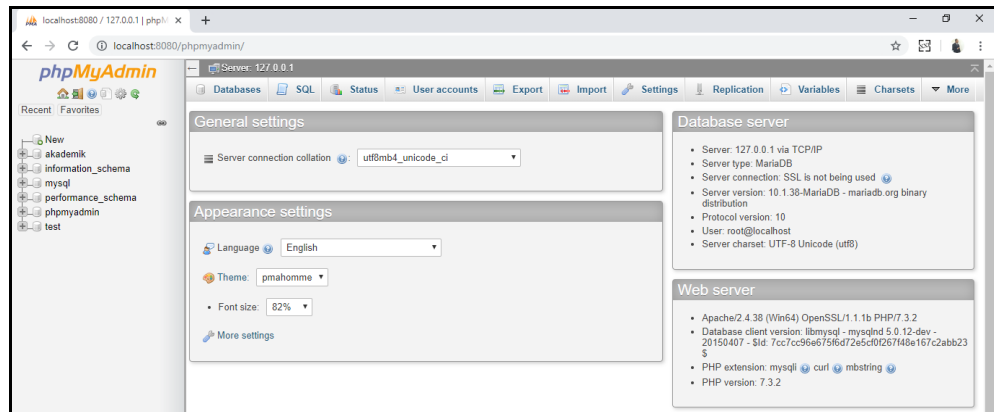
1.1 Pembuatan Database

Database management system yang digunakan pada praktikum ini adalah MySQL. Pada sub-bab ini akan digunakan program bantu PHPMyAdmin. Berikut adalah langkah-langkah pembuatan database:

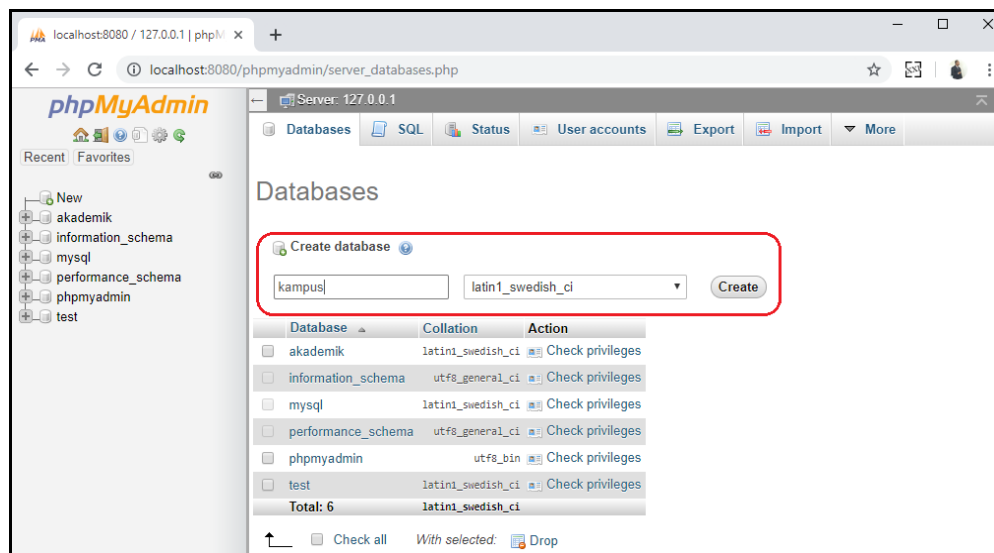
1. Aktifkan xampp control panel.



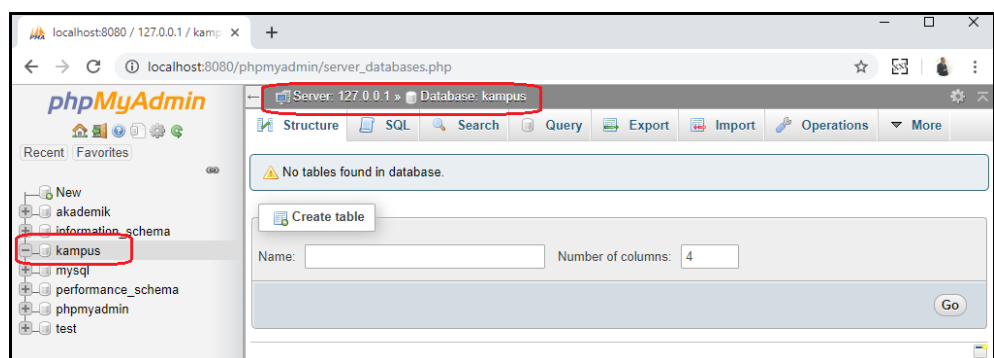
2. Buka browser, kemudian ketikkan url <http://localhost/phpmyadmin/> (apabila menggunakan port 8080, ubah menjadi <http://localhost:8080/phpmyadmin/>).



3. Klik menu Databases, pada bagian Create database, ketikkan nama database yaitu **kampus**, kemudian klik tombol Create.



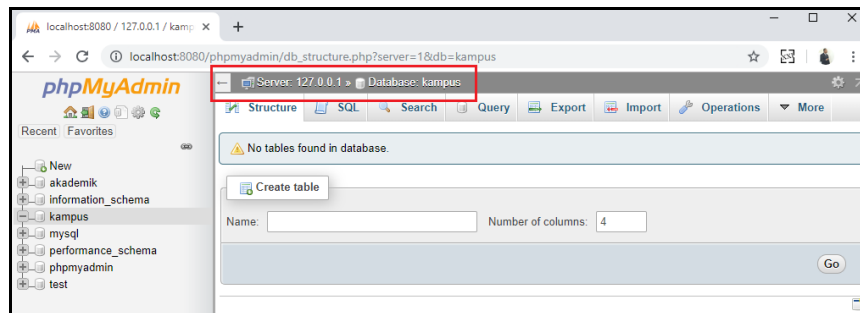
4. Pada bagian tab menu sebelah kanan, akan muncul database dengan nama kampus.



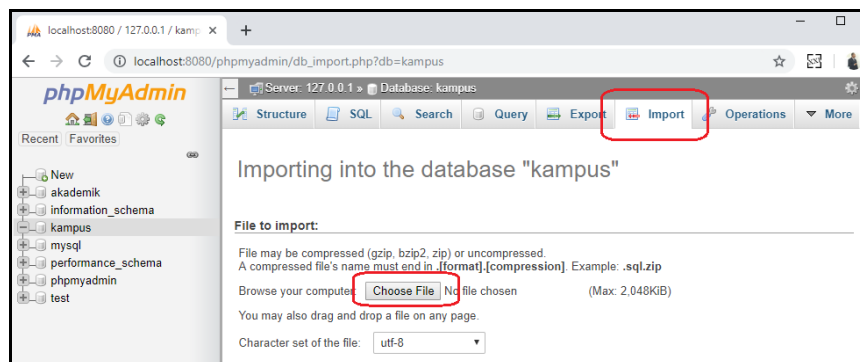
1.2 Import Data

Setelah berhasil membuat database pada sub bab sebelumnya, langkah selanjutnya adalah import data (tabel) ke dalam database kampus. Berikut adalah langkah-langkah melakukan import data:

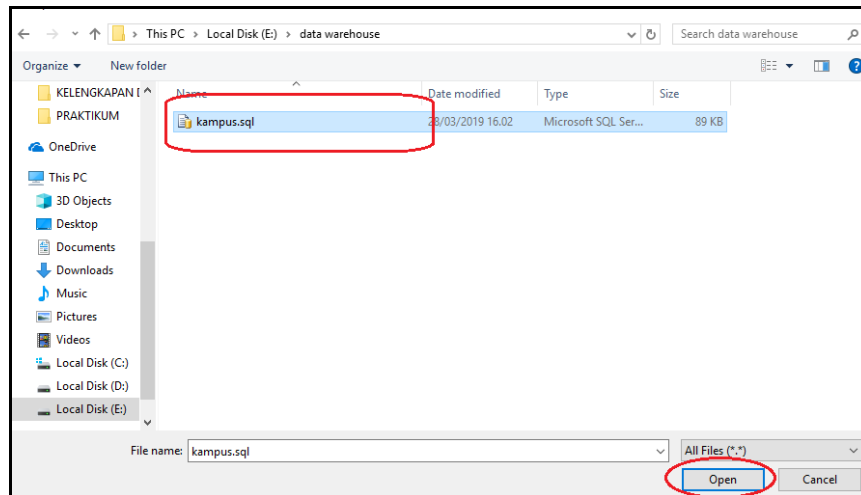
1. Masuk ke dalam database kampus, pastikan (menu bar atas berada pada database kampus)



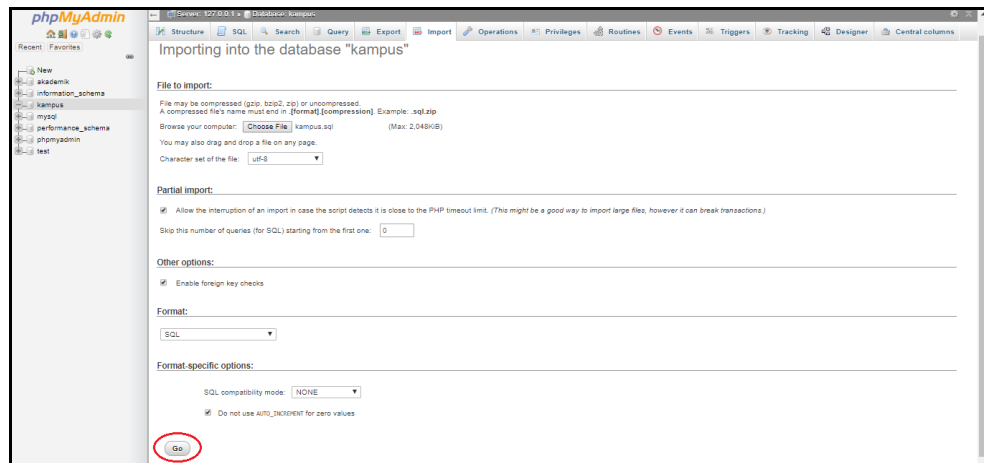
2. Klik menu Import, pada bagian File to import, lalu klik Choose File



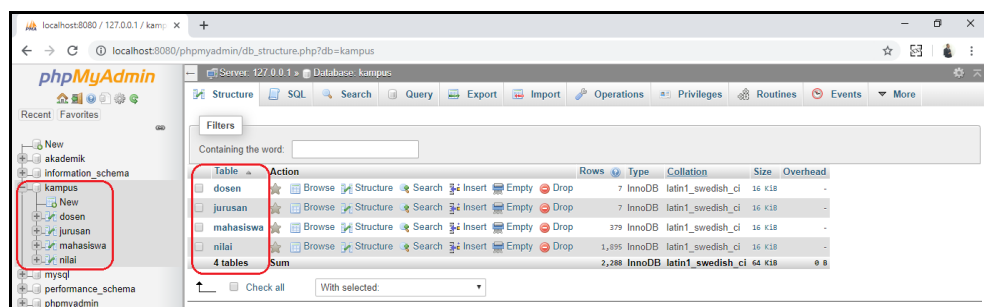
3. Pilih lokasi direktori penyimpanan file kampus.sql, kemudian klik Open



4. Klik tombol Go



5. Pada database kampus akan bertambah empat tabel, yaitu dosen, jurusan, mahasiswa, dan nilai. Sampai dengan tahap ini, kita telah berhasil melakukan import data.



1.3 Syntax Dasar MySQL

1.3.1 Melihat Struktur Tabel

Struktur tabel berisi field-field (nama kolom) pada tabel dan tipe data masing-masing field. Dalam MySQL untuk melihat struktur tabel digunakan perintah *desc*. Pada sub bab ini digunakan command prompt untuk penulisan syntax MySQL, berikut adalah langkah-langkah melihat struktur dalam tabel:

1. Masuk ke dalam MySQL command prompt, ketikkan *show databases;* untuk melihat nama-nama database

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 263
Server version: 10.1.38-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| akademik |
| information_schema |
| kampus |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
+-----+
7 rows in set (0.03 sec)
```

2. Gunakan database kampus, dengan ketikan *use kampus;*

```
MariaDB [kampus]> use kampus;
Database changed
MariaDB [kampus]>
```

3. Tampilkan semua tabel yang berada di dalam database kampus

```
MariaDB [kampus]> show tables;
+-----+
| Tables_in_kampus |
+-----+
| dosen |
| jurusan |
| mahasiswa |
| nilai |
+-----+
4 rows in set (0.00 sec)
```

4. Lihat struktur tabel pada tabel dosen

```

MariaDB [kampus]> desc dosen;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(10) | NO | PRI | NULL | auto_increment |
| kode_dosen | varchar(10) | NO | | NULL | |
| nama_dosen | varchar(50) | NO | | NULL | |
| kode_jur | varchar(5) | NO | | NULL | |
| alamat | text | NO | | NULL | |
| pendidikan | enum('S1','S2','S3') | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.02 sec)

```

Dari struktur tabel dosen di atas dapat dijabarkan bahwa:

- tabel dosen memiliki enam field
- field id merupakan primary key dan auto increment (otomatis terisi) dengan bertipe integer.
- field kode_dosen, nama_dosen, dan kode_jur bertipe varchar
- field alamat bertipe text
- field pendidikan bertipe enum, dengan value: S1, S2, dan S3

5. Gunakan syntax `desc < nama tabel >` untuk melihat struktur pada tabel jurusan, mahasiswa, dan nilai.

```

MariaDB [kampus]> desc jurusan;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id     | int(8) | NO   | PRI | NULL    | auto_increment |
| kode_jur | varchar(10) | NO   |     | NULL    |                 |
| nama_jur | varchar(50) | NO   |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.02 sec)

MariaDB [kampus]> desc mahasiswa;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id     | int(25) | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(25) | NO   |     | NULL    |                 |
| nama_mhs | varchar(50) | NO   |     | NULL    |                 |
| jk     | enum('L','P') | NO   |     | NULL    |                 |
| status_kerja | enum('bekerja','mahasiswa') | NO   |     | NULL    |                 |
| usia   | int(3) | NO   |     | NULL    |                 |
| status_nikah | enum('belum menikah','menikah') | NO   |     | NULL    |                 |
| kode_jur | varchar(5) | NO   |     | NULL    |                 |
| kode_dosen | varchar(10) | NO   |     | NULL    |                 |
| kelulusan | enum('terlambat','tepat') | NO   |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.01 sec)

MariaDB [kampus]> desc nilai;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id     | int(10) | NO   | PRI | NULL    | auto_increment |
| nim    | varchar(10) | NO   |     | NULL    |                 |
| nilai  | float  | NO   |     | NULL    |                 |
| semester | int(5) | NO   |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)

```

1.3.2 Menampilkan Data

Syntax yang digunakan untuk menampilkan data adalah *select * from nama_tabel*. Tanda * diartikan sebagai menampilkan semua field yang ada dalam tabel. Sedangkan untuk menampilkan beberapa field dalam tabel bisa digunakan syntax *select field1, field2, ... field n from nama_tabel*.

Berikut adalah beberapa contoh kasus menampilkan data pada MySQL:

1. **Menampilkan seluruh data dalam satu tabel.** Arsyia adalah kepala bagian akademik di Universitas XYZ, dan Zizi adalah seorang database administrator di Universitas XYZ. Arsyia meminta tolong kepada Zizi untuk menampilkan semua data jurusan di Universitas XYZ. Berikut adalah syntax yang ditulis oleh Zizi

```
MariaDB [kampus]> select * from jurusan;
```

| id | kode_jur | nama_jur |
|----|----------|--------------------|
| 1 | TI | TEKNIK INFORMATIKA |
| 2 | SI | SISTEM INFORMASI |
| 3 | IK | ILMU KOMUNIKASI |
| 4 | AK | AKUNTANSI |
| 5 | MAN | MANAJEMEN |
| 6 | IH | ILMU HUKUM |
| 7 | PSI | PSIKOLOGI |

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

2. **Menampilkan field-field tertentu dalam satu tabel.** Arsyia meminta tolong kepada Zizi untuk menampilkan No. Induk Dosen, Nama Dosen, dan Pendidikan Terakhir. Berikut adalah syntax yang ditulis oleh Zizi

```
MariaDB [kampus]> select kode_dosen as "No. Induk Dosen",
-> nama_dosen as "Nama Dosen",
-> pendidikan as "Pendidikan Terakhir"
-> from dosen;
```

| No. Induk Dosen | Nama Dosen | Pendidikan Terakhir |
|-----------------|-----------------------------|---------------------|
| DOS01 | JOKO SUNTORO, M.Kom. | S2 |
| DOS02 | PRIND TRIAJENG P., M.Kom. | S2 |
| DOS03 | Dr. IKA MERDEKAWATI, M.Psi. | S3 |
| DOS04 | Dr. RUTIN FISEKATI | S3 |
| DOS05 | SURONO, M.I.Kom. | S2 |
| DOS06 | AAN CHUNAIFI, M.H. | S2 |
| DOS07 | ALI MOH. RONDI, M.M., Akt. | S2 |

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

3. **Menampilkan field-field tertentu dalam satu tabel dengan kondisi tertentu.** Ketua program studi ilmu komunikasi (kode_jur = IK) Universitas XYZ akan mengadakan program beasiswa kuliah gratis untuk melanjutkan studi S2 di luar negeri bagi para alumnus. Syarat program beasiswa kuliah gratis tersebut adalah alumnus tersebut belum bekerja (status_kerja = mahasiswa), belum menikah (status_nikah = belum menikah), usia kurang dari dua puluh lima tahun, dan saat studi S1 lulus tepat waktu (≤ 4 tahun). Ketua program studi ilmu komunikasi ingin melihat nim alumnus, nama alumnus, dan jenis kelamin alumnus yang masuk ke dalam daftar penerima program beasiswa dengan syarat yang telah disebutkan.

```

MariaDB [kampus]> select nim as NIM, nama_mhs as "Nama Mahasiswa", jk as "Jenis Kelamin"
-> from mahasiswa
-> where kode_jur="IK" and status_kerja="mahasiswa"
-> and status_nikah="belum menikah" and usia < 25 and kelulusan="tepat";
+-----+-----+-----+
| NIM    | Nama Mahasiswa          | Jenis Kelamin |
+-----+-----+-----+
| IK.147  | KARSONO HENDI           | L             |
| IK.153  | HERU DARMAWAN           | L             |
| IK.161  | DEWI KARYASARI          | P             |
| IK.162  | ROSIDI                  | L             |
| IK.167  | MUHAMMAD TAHRIR FATHONY | L             |
| IK.184  | ARDI SETIAWAN           | L             |
| IK.194  | SITI CHODIROH           | P             |
| IK.202  | ANWAR KHOERUDIN         | L             |
| IK.207  | HANIK ROSIDAH           | P             |
| IK.208  | ATIQOH                  | P             |
| IK.210  | INGGIT BHINTAR HISKYATITIS | P             |
| IK.213  | ARIS SETYAWAN           | L             |
+-----+-----+-----+
12 rows in set (0.00 sec)

```

4. **Menampilkan data dari dua atau lebih relasi tabel.** Melanjutkan dari contoh studi kasus no. 2, dari data dosen tersebut Arsyia meminta tolong kembali kepada Zizi untuk menambahkan data homebase masing-masing dosen (relasi tabel dosen dan tabel jurusan).

```

MariaDB [kampus]> select a.kode_dosen as "NIDN", a.nama_dosen as "Nama Dosen",
-> a.pendidikan as "Pendidikan Terakhir", b.nama_jur as "Homebase"
-> from dosen a, jurusan b
-> where a.kode_jur = b.kode_jur;
+-----+-----+-----+-----+
| NIDN   | Nama Dosen              | Pendidikan Terakhir | Homebase      |
+-----+-----+-----+-----+
| DOS01  | JOKO SUNTORO, M.Kom.    | S2                  | TEKNIK INFORMATIKA |
| DOS02  | PRIND TRIAJENG P., M.Kom. | S2                  | SISTEM INFORMASI    |
| DOS05  | SURONO, M.I.Kom.        | S2                  | ILMU KOMUNIKASI     |
| DOS07  | ALI MOH. RONDI, M.M., Akt. | S2                  | AKUNTANSI           |
| DOS04  | Dr. RUTIN FISEKATI      | S3                  | MANAJEMEN           |
| DOS06  | AAN CHUNAIFI, M.H.      | S2                  | ILMU HUKUM          |
| DOS03  | Dr. IKA MERDEKAWATI, M.Psi. | S3                  | PSIKOLOGI           |
+-----+-----+-----+-----+
7 rows in set (0.00 sec)

```

5. **Menampilkan data dari dua atau lebih relasi tabel dengan kondisi tertentu.** Arsyia sebagai kepala bagian akademik ingin mengetahui data NIM, nama alumni, nama jurusan, dan dosen wali dengan kriteria bahwa alumni tersebut sudah bekerja dan sudah menikah (relasi tabel mahasiswa, tabel jurusan, dan tabel dosen).

```

MariaDB [kampus]> select a.nim as "NIM", a.nama_mhs as "Nama Alumnus",
-> b.nama_jur as "Nama Jurusan", c.nama_dosen as "Dosen Wali"
-> from mahasiswa a, jurusan b, dosen c
-> where a.kode_jur = b.kode_jur
-> and a.kode_dosen = c.kode_dosen
-> and a.status_kerja = "bekerja"
-> and a.status_nikah = "menikah";
+-----+-----+-----+-----+
| NIM   | Nama Alumnus   | Nama Jurusan   | Dosen Wali      |
+-----+-----+-----+-----+
| TI.9   | HENDRIK MULIYANTO | TEKNIK INFORMATIKA | JOKO SUNTORO, M.Kom. |
| TI.25  | MARTHINA FLAVERIA | TEKNIK INFORMATIKA | JOKO SUNTORO, M.Kom. |
| TI.36  | SUGIYANTO       | TEKNIK INFORMATIKA | JOKO SUNTORO, M.Kom. |
| TI.38  | SUTOPO          | TEKNIK INFORMATIKA | JOKO SUNTORO, M.Kom. |
| SI.72  | HAROZUM         | SISTEM INFORMASI  | PRIND TRIAJENG P., M.Kom. |
| IK.155 | PURWADI         | ILMU KOMUNIKASI   | SURONO, M.I.Kom.    |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)

```

1.3.3 Menambahkan Data

Syntax yang digunakan untuk menambahkan data ke dalam tabel adalah *insert into namaTabel values (value 1, value 2, ..., value n)*. Berikut adalah contoh studi kasus menambahkan data:

1. **Menambahkan satu data.** Universitas XYZ menambahkan jurusan baru yaitu teknik elektronika. Jurusan sebelum ditambahkan jurusan teknik elektro berjumlah tujuh jurusan yaitu

```

MariaDB [kampus]> select * from jurusan;
+-----+-----+-----+
| id | kode_jur | nama_jur      |
+-----+-----+-----+
| 1  | TI       | TEKNIK INFORMATIKA |
| 2  | SI       | SISTEM INFORMASI  |
| 3  | IK       | ILMU KOMUNIKASI   |
| 4  | AK       | AKUNTANSI         |
| 5  | MAN      | MANAJEMEN         |
| 6  | IH       | ILMU HUKUM        |
| 7  | PSI      | PSIKOLOGI         |
+-----+-----+-----+
7 rows in set (0.00 sec)

```

Arsya sebagai kepala bagian akademik meminta bantuan kepada Zizi sebagai database administrator untuk menambahkan jurusan baru tersebut ke dalam database. Berikut adalah syntax yang ditulis oleh Zizi untuk menambah jurusan teknik elektro ke dalam database (dalam hal ini tabel jurusan)

```

MariaDB [kampus]> insert into jurusan values ('', 'TE', 'Teknik Elektronika');
Query OK, 1 row affected, 1 warning (0.18 sec)

```

Setelah diketikan syntax seperti di atas, maka dalam tabel jurusan akan bertambah satu jurusan baru yaitu teknik elektronika

```
MariaDB [kampus]> select * from jurusan;
```

| id | kode_jur | nama_jur |
|----|----------|--------------------|
| 1 | TI | TEKNIK INFORMATIKA |
| 2 | SI | SISTEM INFORMASI |
| 3 | IK | ILMU KOMUNIKASI |
| 4 | AK | AKUNTANSI |
| 5 | MAN | MANAJEMEN |
| 6 | IH | ILMU HUKUM |
| 7 | PSI | PSIKOLOGI |
| 8 | TE | Teknik Elektronika |

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

2. **Menambahkan lebih dari satu data.** Universitas XYZ telah menambahkan jurusan baru yaitu teknik elektronika, sehingga Universitas XYZ membutuhkan dosen baru untuk mengajar di jurusan teknik elektronika tersebut. Nila sebagai kepala bagian kepegawaian telah meng-hire beberapa dosen yang siap mengajar di jurusan teknik elektronika. Guna pendataan daftar dosen, Nila meminta tolong kepada Zizi untuk menambahkan beberapa nama dosen yang mengajar di teknik elektronika Universitas XYZ. Syntax yang digunakan oleh Zizi untuk menambahkan daftar nama dosen teknik elektronika adalah sebagai berikut

```
MariaDB [kampus]> insert into dosen values
-> ('', 'DOS08', 'ENGGAR JULIANTO, M.T.', 'TE', 'Batang', 'S2'),
-> ('', 'DOS09', 'Dr. SAYYAF', 'TE', 'Semarang', 'S3');
Query OK, 2 rows affected, 2 warnings (0.10 sec)
Records: 2 Duplicates: 0 Warnings: 2
```

```
MariaDB [kampus]> select * from dosen;
```

| id | kode_dosen | nama_dosen | kode_jur | alamat | pendidikan |
|----|------------|-----------------------------|----------|----------|------------|
| 1 | DOS01 | JOKO SUNTORO, M.Kom. | TI | Semarang | S2 |
| 2 | DOS02 | PRIND TRIAJENG P., M.Kom. | SI | Kendal | |
| 3 | DOS03 | Dr. IKA MERDEKAWATI, M.Psi. | PSI | Semarang | S3 |
| 4 | DOS04 | Dr. RUTIN FISEKATI | MAN | Batang | |
| 5 | DOS05 | SURONO, M.I.Kom. | IK | Klaten | |
| 6 | DOS06 | AAN CHUNAIFI, M.H. | IH | Semarang | S2 |
| 7 | DOS07 | ALI MOH. RONDI, M.M., Akt. | AK | Semarang | |
| 8 | DOS08 | ENGGAR JULIANTO, M.T. | TE | Batang | S2 |
| 9 | DOS09 | Dr. SAYYAF | TE | Semarang | S3 |

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

1.3.4 Mengubah Data

Data yang telah disimpan dalam database jika terjadi perubahan data, dapat dilakukan update (ubah) data. Syntax yang digunakan untuk mengubah data adalah *update namatabel set field1 = 'nilaibaru' where <kondisi>*.

Sebagai contoh, dosen bernama Enggar Julianto, M.T., dengan kode_dosen DOS08 pindah alamat rumah. Sebelumnya Enggar Julianto, M.T. beralamat di Batang, dan saat ini pindah alamat ke Jakarta. Berikut adalah syntax yang digunakan untuk mengubah data alamat Enggar Julianto, M.T.

```
MariaDB [kampus]> select *from dosen;
```

| id | kode_dosen | nama_dosen | kode_jur | alamat | pendidikan |
|----|------------|-----------------------------|----------|----------|------------|
| 1 | DOS01 | JOKO SUNTORO, M.Kom. | TI | Semarang | S2 |
| 2 | DOS02 | PRIND TRIAJENG P., M.Kom. | SI | Kendal | S2 |
| 3 | DOS03 | Dr. IKA MERDEKAWATI, M.Psi. | PSI | Semarang | S3 |
| 4 | DOS04 | Dr. RUTIN FISEKATI | MAN | Batang | S3 |
| 5 | DOS05 | SURONO, M.I.Kom. | IK | Klaten | S2 |
| 6 | DOS06 | AAN CHUNAIFI, M.H. | IH | Semarang | S2 |
| 7 | DOS07 | ALI MOH. RONDI, M.M., Akt. | AK | Semarang | S2 |
| 8 | DOS08 | ENGGAR JULIANTO, M.T. | TE | Batang | S2 |
| 9 | DOS09 | Dr. SAYYAF | TE | Semarang | S3 |

9 rows in set (0.00 sec)

```
MariaDB [kampus]> update dosen set alamat = "Jakarta" where kode_dosen = "DOS08";  
Query OK, 1 row affected (0.06 sec)  
Rows matched: 1 Changed: 1 Warnings: 0
```

```
MariaDB [kampus]> select * from dosen;
```

| id | kode_dosen | nama_dosen | kode_jur | alamat | pendidikan |
|----|------------|-----------------------------|----------|----------|------------|
| 1 | DOS01 | JOKO SUNTORO, M.Kom. | TI | Semarang | S2 |
| 2 | DOS02 | PRIND TRIAJENG P., M.Kom. | SI | Kendal | S2 |
| 3 | DOS03 | Dr. IKA MERDEKAWATI, M.Psi. | PSI | Semarang | S3 |
| 4 | DOS04 | Dr. RUTIN FISEKATI | MAN | Batang | S3 |
| 5 | DOS05 | SURONO, M.I.Kom. | IK | Klaten | S2 |
| 6 | DOS06 | AAN CHUNAIFI, M.H. | IH | Semarang | S2 |
| 7 | DOS07 | ALI MOH. RONDI, M.M., Akt. | AK | Semarang | S2 |
| 8 | DOS08 | ENGGAR JULIANTO, M.T. | TE | Jakarta | S2 |
| 9 | DOS09 | Dr. SAYYAF | TE | Semarang | S3 |

1.3.5 Menghapus data

Syntax yang digunakan untuk menghapus data dalam database adalah *delete from namatabel where <kondisi>*. Sebagai contoh Dr. Sayyaf mengundurkan diri sebagai dosen di Universitas XYZ. Berikut adalah syntax yang digunakan untuk menghapus data Dr. Sayyaf dari database

```
MariaDB [kampus]> select * from dosen;
```

| id | kode_dosen | nama_dosen | kode_jur | alamat | pendidikan |
|----|------------|-----------------------------|----------|----------|------------|
| 1 | DOS01 | JOKO SUNTORO, M.Kom. | TI | Semarang | S2 |
| 2 | DOS02 | PRIND TRIAJENG P., M.Kom. | SI | Kendal | |
| 3 | DOS03 | Dr. IKA MERDEKAWATI, M.Psi. | PSI | Semarang | S3 |
| 4 | DOS04 | Dr. RUTIN FISEKATI | MAN | Batang | |
| 5 | DOS05 | SURONO, M.I.Kom. | IK | Klaten | |
| 6 | DOS06 | AAN CHUNAIFI, M.H. | IH | Semarang | S2 |
| 7 | DOS07 | ALI MOH. RONDI, M.M., Akt. | AK | Semarang | |
| 8 | DOS08 | ENGGAR JULIANTO, M.T. | TE | Jakarta | S2 |
| 9 | DOS09 | Dr. SAYYAF | TE | Semarang | S3 |

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

```
MariaDB [kampus]> delete from dosen where kode_dosen = "DOS09";
```

Query OK, 1 row affected (0.09 sec)

```
MariaDB [kampus]> select * from dosen;
```

| id | kode_dosen | nama_dosen | kode_jur | alamat | pendidikan |
|----|------------|-----------------------------|----------|----------|------------|
| 1 | DOS01 | JOKO SUNTORO, M.Kom. | TI | Semarang | S2 |
| 2 | DOS02 | PRIND TRIAJENG P., M.Kom. | SI | Kendal | |
| 3 | DOS03 | Dr. IKA MERDEKAWATI, M.Psi. | PSI | Semarang | S3 |
| 4 | DOS04 | Dr. RUTIN FISEKATI | MAN | Batang | |
| 5 | DOS05 | SURONO, M.I.Kom. | IK | Klaten | |
| 6 | DOS06 | AAN CHUNAIFI, M.H. | IH | Semarang | S2 |
| 7 | DOS07 | ALI MOH. RONDI, M.M., Akt. | AK | Semarang | |
| 8 | DOS08 | ENGGAR JULIANTO, M.T. | TE | Jakarta | S2 |

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

BAB 2

SQL TINGKAT LANJUT

2.1 Cascade pada MySQL

Cascade adalah fungsi dalam MySQL yang berguna untuk membuat relasi antar tabel (tabel induk dan tabel anak). Apabila data pada tabel induk diupdate atau didelete, maka tabel anak yang berelasi dengan tabel induk akan diupdate/didelete secara otomatis.

Perancangan database yang baik adalah jika terjadi perubahan data pada tabel induk, maka tabel anak juga otomatis mengalami perubahan, baik itu update maupun delete. Untuk membuat otomatisasi tersebut, digunakan fungsi cascade pada MySQL. Database yang digunakan dalam bab ini adalah sales. Database sales terdiri dari empat tabel, yaitu tabel transaksi, tabel pekerja, tabel customer, dan tabel produk. Tabel transaksi merupakan tabel induk, sedangkan tabel anak yaitu tabel pekerja, tabel customer, dan tabel produk.

Berikut adalah langkah-langkah pembuatan cascade pada MySQL:

1. Buat database dengan nama sales
2. Import file sales.sql ke dalam database sales
3. Relasikan tabel transaksi dan tabel customer, ketikkan syntax berikut

```
MariaDB [sales]> alter table transaksi add foreign key (idCust) references  
-> customer (idCust) on delete cascade on update cascade;  
Query OK, 1003 rows affected (1.10 sec)  
Records: 1003 Duplicates: 0 Warnings: 0
```

4. Relasikan tabel transaksi dan tabel produksi, ketikkan syntax berikut

```
MariaDB [sales]> alter table transaksi add foreign key (idProd) references  
-> produk (idProd) on delete cascade on update cascade;  
Query OK, 1003 rows affected (2.42 sec)  
Records: 1003 Duplicates: 0 Warnings: 0
```

5. Relasikan tabel transaksi dan tabel pekerja, ketikkan syntax berikut

```
MariaDB [sales]> alter table transaksi add foreign key (idPek) references  
-> pekerja (idPek) on delete cascade on update cascade;  
Query OK, 1003 rows affected (1.25 sec)  
Records: 1003 Duplicates: 0 Warnings: 0
```

6. Sebagai contoh, akan dilakukan pengurangan satu pekerja di PT Enggar Jaya Group dengan jumlah penjualan terkecil. Langkah awal sebelum dilakukan delete data pada tabel pekerja, maka kita akan melihat jumlah penjualan masing-masing pekerja pada tabel transaksi, berikut adalah syntax untuk melihat jumlah penjualan masing-masing pekerja

```
MariaDB [sales]> select idPek, sum(qty) as jumlah from transaksi group by idPek;
```

| idPek | jumlah |
|-------|--------|
| 1 | 6790 |
| 2 | 8000 |
| 3 | 13740 |
| 4 | 3550 |
| 5 | 9200 |
| 6 | 5190 |
| 7 | 4440 |
| 8 | 20 |

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

7. Dari gambar di atas, terlihat bahwa pekerja dengan idPek = 8 adalah pekerja dengan penjualan terkecil, sehingga pekerja dengan idPek = 8 akan dihapus dari tabel pekerja. Jika data idPek = 8 dihapus dari tabel pekerja, maka secara otomatis data penjualan idPek = 8 pada tabel transaksi akan hilang, karena telah ditambahkan perintah cascade.

```
MariaDB [sales]> select idPek, sum(qty) as jumlah from transaksi group by idPek;
```

| idPek | jumlah |
|-------|--------|
| 1 | 6790 |
| 2 | 8000 |
| 3 | 13740 |
| 4 | 3550 |
| 5 | 9200 |
| 6 | 5190 |
| 7 | 4440 |
| 8 | 20 |

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

```
MariaDB [sales]> delete from pekerja where idPek=8;
```

```
Query OK, 1 row affected (0.08 sec)
```

```
MariaDB [sales]> select idPek, sum(qty) as jumlah from transaksi group by idPek;
```

| idPek | jumlah |
|-------|--------|
| 1 | 6790 |
| 2 | 8000 |
| 3 | 13740 |
| 4 | 3550 |
| 5 | 9200 |
| 6 | 5190 |
| 7 | 4440 |

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

2.2 View pada MySQL

View adalah query yang disimpan dan ketika dipanggil akan menghasilkan set hasil. View bertindak sebagai tabel virtual. Tujuan pembuatan view adalah untuk memudahkan dalam pembuatan laporan, selain itu digunakan untuk mempercepat proses menampilkan data. Secara sederhana view adalah perintah SELECT (digunakan untuk menampilkan data) yang disimpan ke dalam tabel/database.

Berikut adalah langkah-langkah pembuatan view di MySQL (melanjutkan dari sub bab sebelumnya):

8. Guna kebutuhan pembuatan dashboard penjualan (akan dibahas pada Bab 3 tentang visualisasi data), maka perlu dibuatkan view untuk memudahkan dalam collecting data dan pembuatan laporan.
9. Field yang dibutuhkan untuk pembuatan dashboard adalah nama customer, tanggal transaksi, nama pekerja, region, kode produk, dan quantity. Field-field tersebut berada di dalam tabel customer, pekerja, produk, dan transaksi.
10. Syntax yang digunakan adalah untuk pembuatan view adalah

```
MariaDB [sales]> create view v_sales
-> as
-> select a.namaCust as "Customer", b.tglTrans as "Tanggal",
-> c.namaPek as "Pekerja", c.Region as "Region",
-> d.kodeProd as "Produk", b.qty as "Qty"
-> from customer a, transaksi b, pekerja c, produk d
-> where a.idCust = b.idCust
-> and c.idPek = b.idPek
-> and d.idProd = b.idProd
-> group by b.tglTrans;
Query OK, 0 rows affected (0.09 sec)
```

11. Nama view yang dibuat adalah v_sales, untuk menampilkan data pada view v_sales, digunakan syntax berikut

```
MariaDB [sales]> select * from v_sales;
```

| Customer | Tanggal | Pekerja | Region | Produk | Qty |
|---------------------|------------|-----------------|-----------|--------|-----|
| Sony Electronic | 2017-01-01 | Surono | Vietnam | item 3 | 40 |
| Sime Darb | 2017-01-02 | Prind Tri Ajeng | Filipina | item 1 | 80 |
| Sime Darb | 2017-01-03 | Cahya Nurani | Indonesia | item 4 | 80 |
| Exxon Mobil | 2017-01-04 | Prind Tri Ajeng | Filipina | item 4 | 40 |
| Gecina | 2017-01-05 | Prind Tri Ajeng | Filipina | item 3 | 20 |
| PT Sejahtera Bareng | 2017-01-06 | Cahya Nurani | Indonesia | item 4 | 50 |
| Royal Dutch | 2017-01-07 | Prind Tri Ajeng | Filipina | item 2 | 40 |
| PT Sejahtera Bareng | 2017-01-08 | Surono | Vietnam | item 5 | 70 |
| Ventas | 2017-01-09 | Surono | Vietnam | item 4 | 90 |
| Sime Darb | 2017-01-10 | Cahya Nurani | Indonesia | item 3 | 30 |
| Sime Darb | 2017-01-11 | Surono | Vietnam | item 3 | 90 |
| Boston Properties | 2017-01-12 | Cahya Nurani | Indonesia | item 2 | 80 |
| Caltex Pasific | 2017-01-13 | Surono | Vietnam | item 3 | 50 |
| Sony Electronic | 2017-01-14 | Joko Suntoro | Thailand | item 4 | 20 |
| Sime Darb | 2017-01-15 | Prind Tri Ajeng | Filipina | item 1 | 40 |
| Gecina | 2017-01-16 | Prind Tri Ajeng | Filipina | item 1 | 20 |
| Exxon Mobil | 2017-01-17 | Surono | Vietnam | item 4 | 10 |
| Sony Electronic | 2017-01-18 | Prind Tri Ajeng | Filipina | item 4 | 80 |
| Link Reit | 2017-01-19 | Cahya Nurani | Indonesia | item 1 | 60 |
| Exxon Mobil | 2017-01-20 | Surono | Vietnam | item 4 | 40 |

BAB 3

VISUALISASI DATA

(PIVOT TABLE, CHART, MAP DAN DASHBOARD)

Setelah berhasil melakukan collecting data dari source system/database, langkah selanjutnya adalah melakukan visualisasi data guna pembuatan laporan untuk pendukung keputusan top management. Visualisasi data dalam data warehouse dibagi menjadi dua bagian yaitu dimensional data store dan multidimensional database. Dalam dimensional data store, visualisasi data dibuat dalam bentuk grafik, spreadsheets, pivot table, pivot chart, add hoc queries dan report). Sedangkan dalam multidimensional database, biasanya digunakan untuk analisis tingkat lanjut yaitu berbentuk data mining, dashboard, scorecards, dan aplikasi business intelligence lainnya).

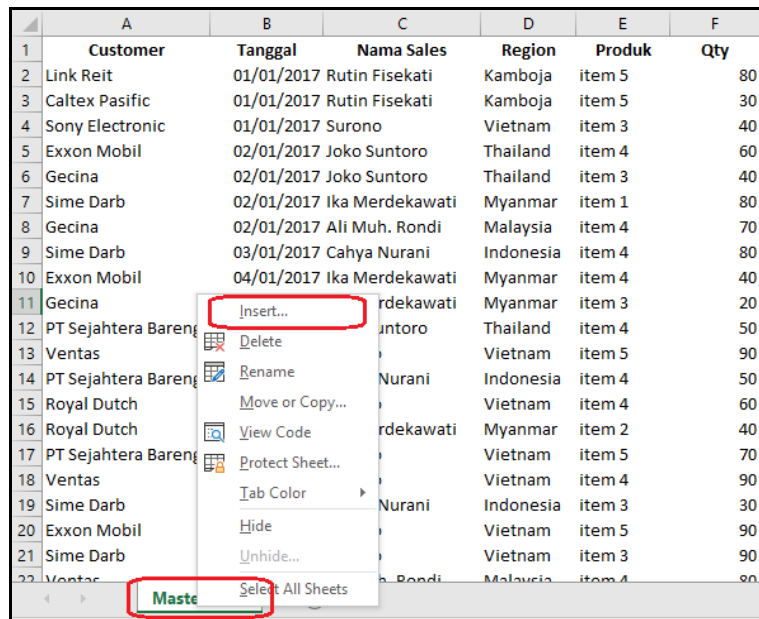
Dalam bab ini akan dibuat visualisasi data dalam bentuk pivot table, pivot chart, dan dalam bentuk dashboard. Data yang akan digunakan dalam bab ini adalah data sales.xlsx. Data sales.xlsx didapatkan dari export Views bernama view_report database sales pada sub bab sebelumnya. Berikut adalah langkah-langkah pembuatan visualisasi data guna pembuatan laporan untuk pendukung keputusan:

3.1 Pivot Table dan Pivot Chart Trend Penjualan

1. Siapkan data sales.xlsx

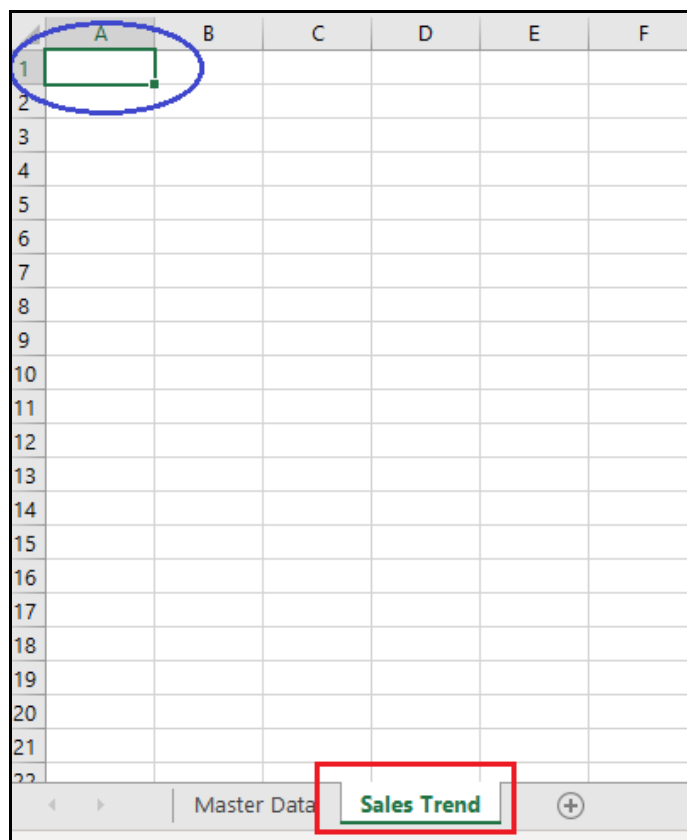
| | A | B | C | D | E | F |
|----|---------------------|------------|-----------------|-----------|--------|-----|
| 1 | Customer | Tanggal | Nama Sales | Region | Produk | Qty |
| 2 | Link Reit | 01/01/2017 | Rutin Fisekati | Kamboja | item 5 | 80 |
| 3 | Caltex Pasific | 01/01/2017 | Rutin Fisekati | Kamboja | item 5 | 30 |
| 4 | Sony Electronic | 01/01/2017 | Surono | Vietnam | item 3 | 40 |
| 5 | Exxon Mobil | 02/01/2017 | Joko Suntoro | Thailand | item 4 | 60 |
| 6 | Gecina | 02/01/2017 | Joko Suntoro | Thailand | item 3 | 40 |
| 7 | Sime Darb | 02/01/2017 | Ika Merdekawati | Myanmar | item 1 | 80 |
| 8 | Gecina | 02/01/2017 | Ali Muh. Rondi | Malaysia | item 4 | 70 |
| 9 | Sime Darb | 03/01/2017 | Cahya Nurani | Indonesia | item 4 | 80 |
| 10 | Exxon Mobil | 04/01/2017 | Ika Merdekawati | Myanmar | item 4 | 40 |
| 11 | Gecina | 05/01/2017 | Ika Merdekawati | Myanmar | item 3 | 20 |
| 12 | PT Sejahtera Bareng | 05/01/2017 | Joko Suntoro | Thailand | item 4 | 50 |
| 13 | Ventas | 05/01/2017 | Surono | Vietnam | item 5 | 90 |
| 14 | PT Sejahtera Bareng | 06/01/2017 | Cahya Nurani | Indonesia | item 4 | 50 |
| 15 | Royal Dutch | 06/01/2017 | Surono | Vietnam | item 4 | 60 |
| 16 | Royal Dutch | 07/01/2017 | Ika Merdekawati | Myanmar | item 2 | 40 |
| 17 | PT Sejahtera Bareng | 08/01/2017 | Surono | Vietnam | item 5 | 70 |
| 18 | Ventas | 09/01/2017 | Surono | Vietnam | item 4 | 90 |
| 19 | Sime Darb | 10/01/2017 | Cahya Nurani | Indonesia | item 3 | 30 |
| 20 | Exxon Mobil | 11/01/2017 | Surono | Vietnam | item 5 | 90 |
| 21 | Sime Darb | 11/01/2017 | Surono | Vietnam | item 3 | 90 |

2. Tambahkan sheet baru, dengan cara klik kanan sheet Master Data – pilih Insert – Worksheet – lalu klik OK



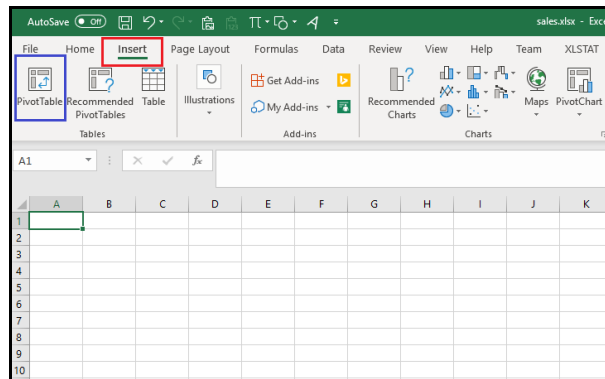
| | A | B | C | D | E | F |
|----|---------------------|------------|-----------------|-----------|--------|-----|
| 1 | Customer | Tanggal | Nama Sales | Region | Produk | Qty |
| 2 | Link Reit | 01/01/2017 | Rutin Fisekati | Kamboja | item 5 | 80 |
| 3 | Caltex Pasific | 01/01/2017 | Rutin Fisekati | Kamboja | item 5 | 30 |
| 4 | Sony Electronic | 01/01/2017 | Surono | Vietnam | item 3 | 40 |
| 5 | Exxon Mobil | 02/01/2017 | Joko Suntoro | Thailand | item 4 | 60 |
| 6 | Gecina | 02/01/2017 | Joko Suntoro | Thailand | item 3 | 40 |
| 7 | Sime Darb | 02/01/2017 | Ika Merdekawati | Myanmar | item 1 | 80 |
| 8 | Gecina | 02/01/2017 | Ali Muh. Rondi | Malaysia | item 4 | 70 |
| 9 | Sime Darb | 03/01/2017 | Cahya Nurani | Indonesia | item 4 | 80 |
| 10 | Exxon Mobil | 04/01/2017 | Ika Merdekawati | Myanmar | item 4 | 40 |
| 11 | Gecina | | rdekawati | Myanmar | item 3 | 20 |
| 12 | PT Sejahtera Bareng | | ntoro | Thailand | item 4 | 50 |
| 13 | Ventas | | | Vietnam | item 5 | 90 |
| 14 | PT Sejahtera Bareng | | Nurani | Indonesia | item 4 | 50 |
| 15 | Royal Dutch | | | Vietnam | item 4 | 60 |
| 16 | Royal Dutch | | rdekawati | Myanmar | item 2 | 40 |
| 17 | PT Sejahtera Bareng | | | Vietnam | item 5 | 70 |
| 18 | Ventas | | | Vietnam | item 4 | 90 |
| 19 | Sime Darb | | Nurani | Indonesia | item 3 | 30 |
| 20 | Exxon Mobil | | | Vietnam | item 5 | 90 |
| 21 | Sime Darb | | | Vietnam | item 3 | 90 |
| 22 | Ventas | | h. Rondi | Malaysia | item 4 | 80 |

3. Ubah nama sheet baru menjadi Sales Trend, lalu letakkan kursor pada sel A1.

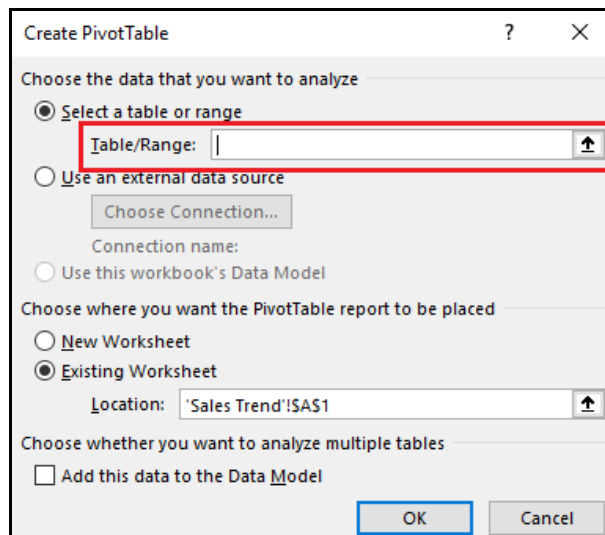


| | A | B | C | D | E | F |
|----|---|---|---|---|---|---|
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |
| 7 | | | | | | |
| 8 | | | | | | |
| 9 | | | | | | |
| 10 | | | | | | |
| 11 | | | | | | |
| 12 | | | | | | |
| 13 | | | | | | |
| 14 | | | | | | |
| 15 | | | | | | |
| 16 | | | | | | |
| 17 | | | | | | |
| 18 | | | | | | |
| 19 | | | | | | |
| 20 | | | | | | |
| 21 | | | | | | |
| 22 | | | | | | |

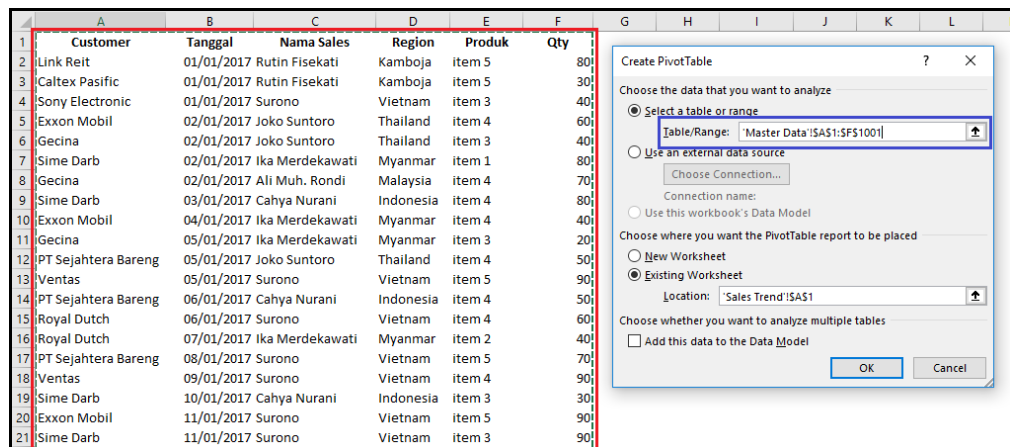
- Klik menu Insert – Pivot Table



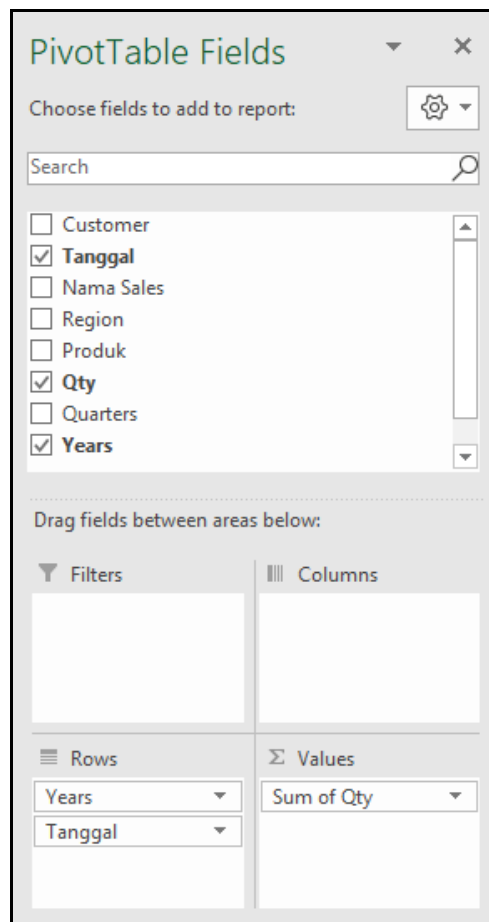
- Pada bagian Create PivotTable, letakkan pada bagian Select a table or range – Table/Range



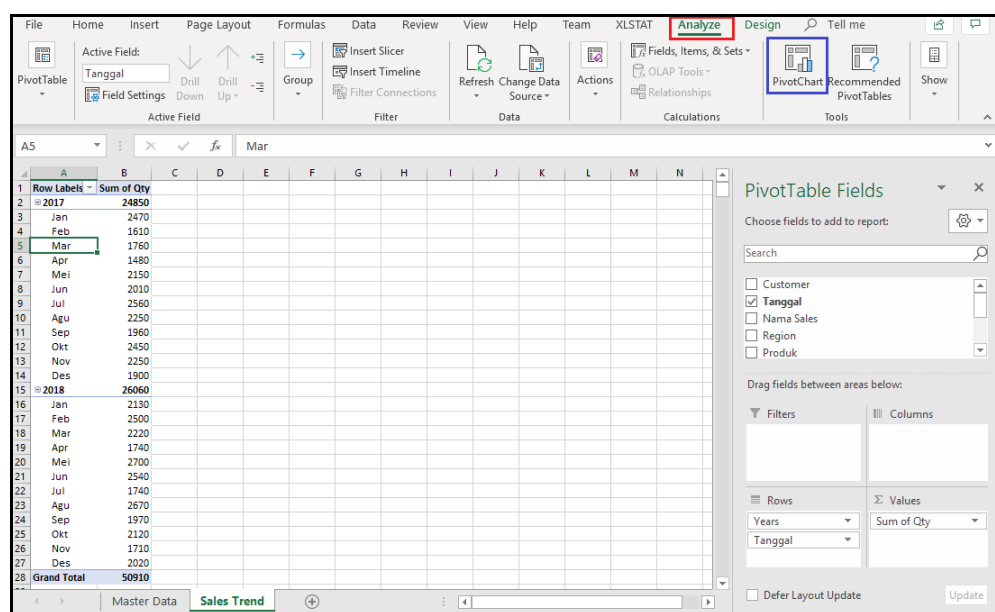
- Klik sheet Master Data, kemudian blok data dari sel A1:F1001, lalu klik OK



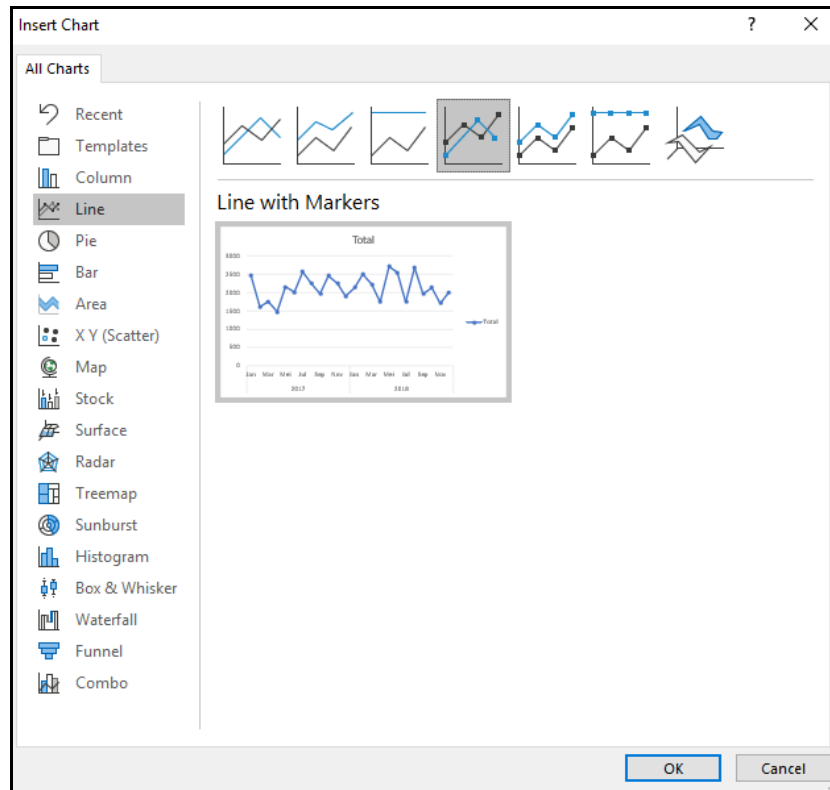
7. Pada bagian PivotTables Fields, pilih Tanggal dan Years lalu drag ke bagian Rows, dan pilih Qty lalu drag ke bagian Values



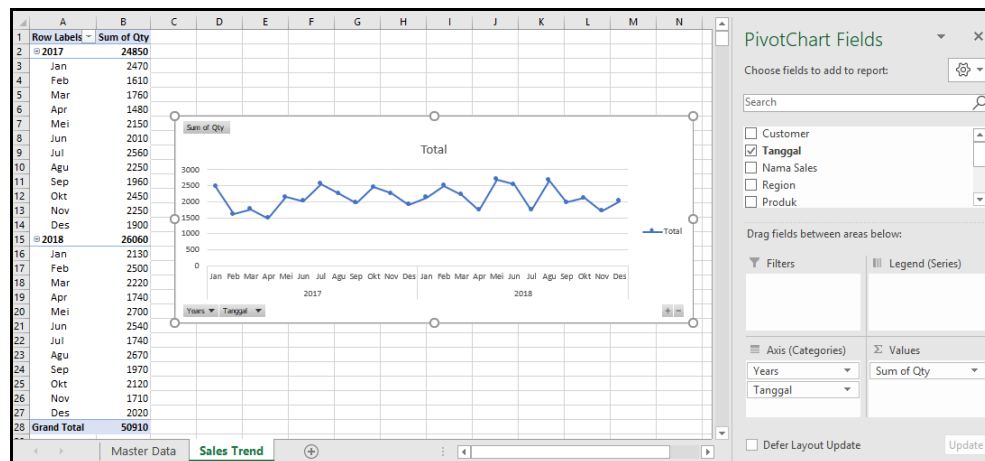
8. Letakkan kursor di sel A3, lalu pilih menu Analyze, klik PivotChart



9. Pada bagian Insert Chart pilih Line – Line with Markers, lalu klik tombol OK



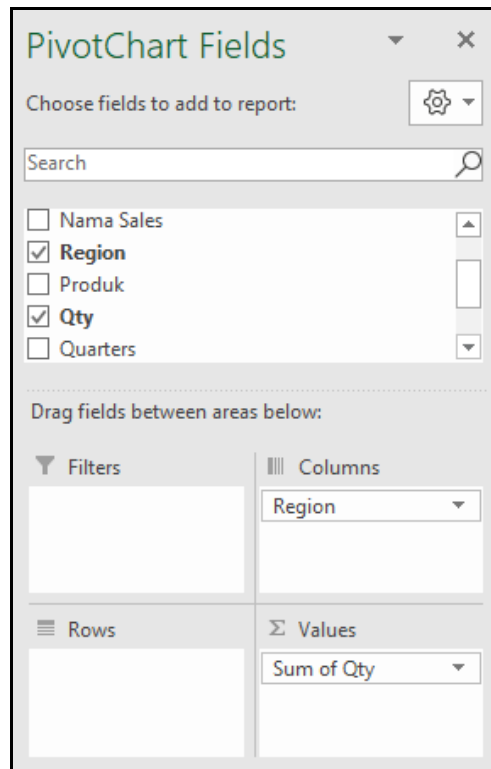
10. Maka akan terlihat hasil pivot chart (dalam bentuk grafik line) seperti gambar di bawah ini



3.2 Pivot Table dan Maps Chart Sales per Region

1. Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Sales per Region, lalu letakkan kursor pada sel A1
2. Tambahkan pivot table dengan mengambil semua data dari sheet Master Data

3. Pada bagian PivotTables Fields, pilih Region lalu drag ke bagian Columns, dan pilih Qty lalu drag ke bagian Values



4. Blok sel A2:I3, lalu copy-paste ke sel A6, ubah nama Sum of Qty pada sel A7 menjadi Quantity, dan hapus Grand Total pada sel I6:I7

| | A | B | C | D | E | F | G | H | I |
|---|------------|---------------|-----------|---------|----------|---------|----------|---------|-------------|
| 1 | | Column Labels | | | | | | | |
| 2 | | Filipina | Indonesia | Kamboja | Malaysia | Myanmar | Thailand | Vietnam | Grand Total |
| 3 | Sum of Qty | 6790 | 8000 | 13740 | 3550 | 9200 | 5190 | 4440 | 50910 |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| 6 | | Filipina | Indonesia | Kamboja | Malaysia | Myanmar | Thailand | Vietnam | |
| 7 | Quantity | 6790 | 8000 | 13740 | 3550 | 9200 | 5190 | 4440 | |

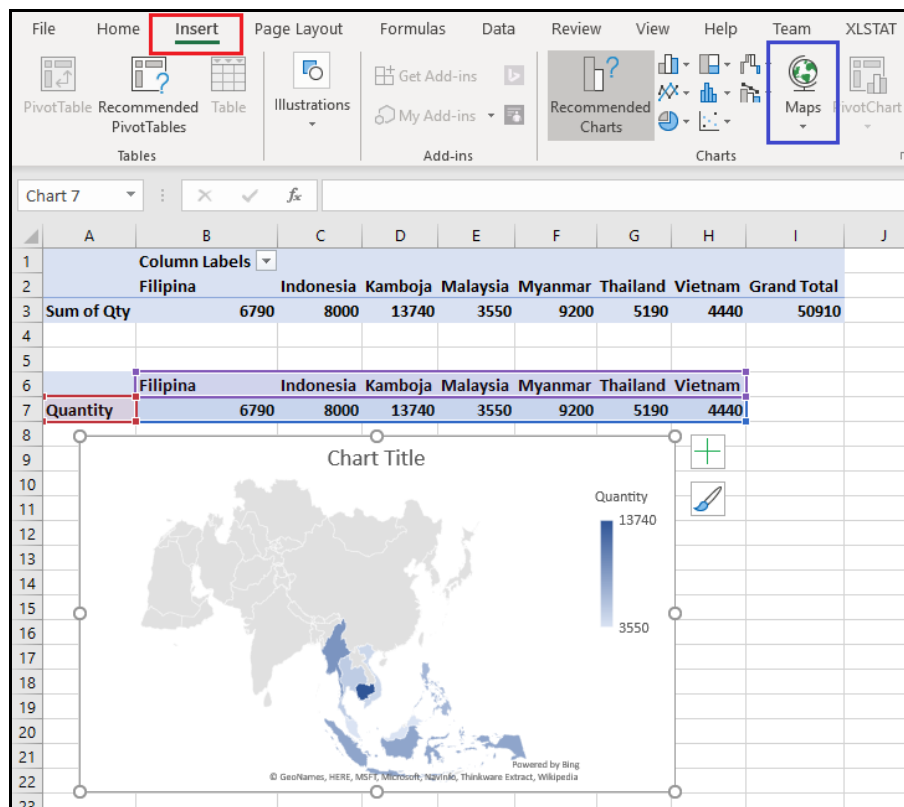
5. Klik kursos pada sel B7, lalu ketikkan formula =B3, lalu tekan tombol Enter

| | A | B | C | D | E | F | G | H | I |
|---|------------|---------------|-----------|---------|----------|---------|----------|---------|-------------|
| 1 | | Column Labels | | | | | | | |
| 2 | | Filipina | Indonesia | Kamboja | Malaysia | Myanmar | Thailand | Vietnam | Grand Total |
| 3 | Sum of Qty | 6790 | 8000 | 13740 | 3550 | 9200 | 5190 | 4440 | 50910 |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| 6 | | Filipina | Indonesia | Kamboja | Malaysia | Myanmar | Thailand | Vietnam | |
| 7 | Quantity | =B3 | 8000 | 13740 | 3550 | 9200 | 5190 | 4440 | |

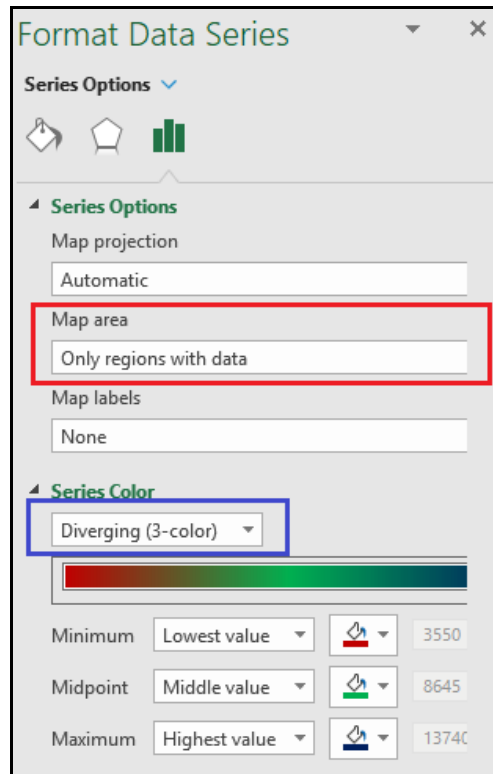
- Klik kursor pada sel B7, tekan tombol CTRL+C (copy), lalu blok sel B7:H7, setelah itu tekan tombol Enter

| | | | | | | | | | |
|---|---------------|-----------|---------|----------|---------|----------|---------|-------------|-------|
| 1 | Column Labels | | | | | | | | |
| 2 | Filipina | Indonesia | Kamboja | Malaysia | Myanmar | Thailand | Vietnam | Grand Total | |
| 3 | Sum of Qty | 6790 | 8000 | 13740 | 3550 | 9200 | 5190 | 4440 | 50910 |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| 6 | Filipina | Indonesia | Kamboja | Malaysia | Myanmar | Thailand | Vietnam | | |
| 7 | Quantity | 6790 | 8000 | 13740 | 3550 | 9200 | 5190 | 4440 | |

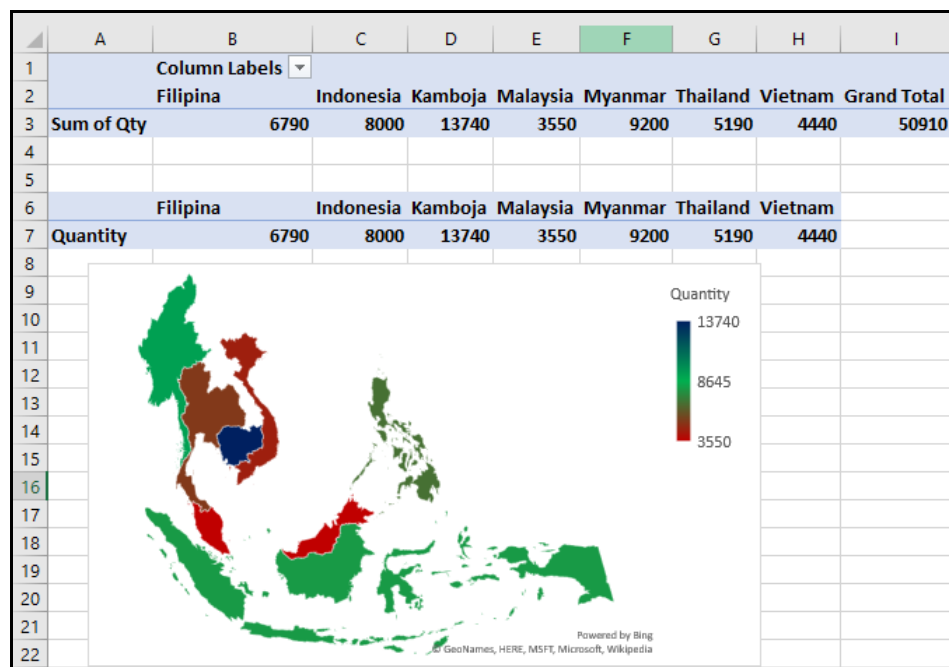
- Blok sel A6:H7, lalu klik menu Insert, kemudian pilih Maps



- Double klik, salah satu negara di dalam map, sehingga akan muncul box Format Data Series. Pada bagian Map area pilih Series Option - Only regions with data. Kemudian pilih Series Color – Diverging (3-color), lalu pada bagian Minimum, Midpoint, dan Maximum, pilih warna sesuai selera

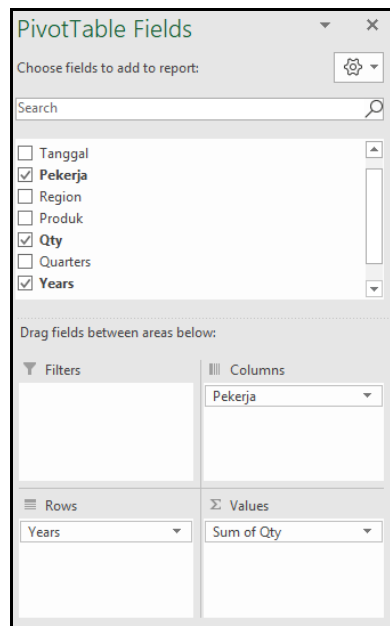


9. Hasil akhir pivot table dan map chart Sales per Region terlihat seperti gambar di bawah ini

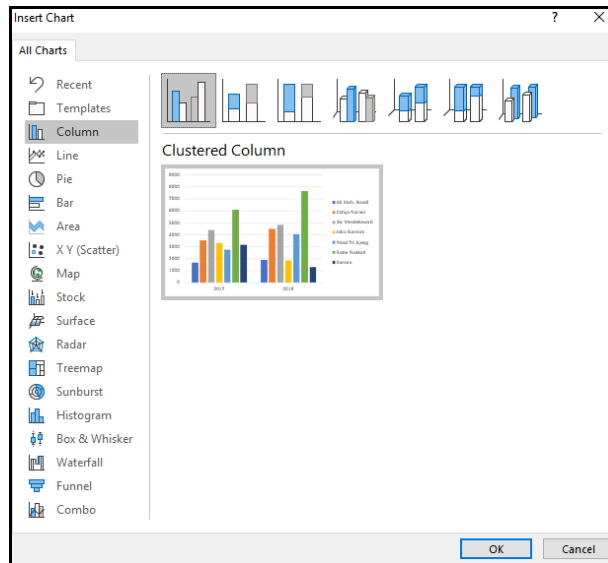


3.3 Pivot Table dan Pivot Chart Sales per Pekerja

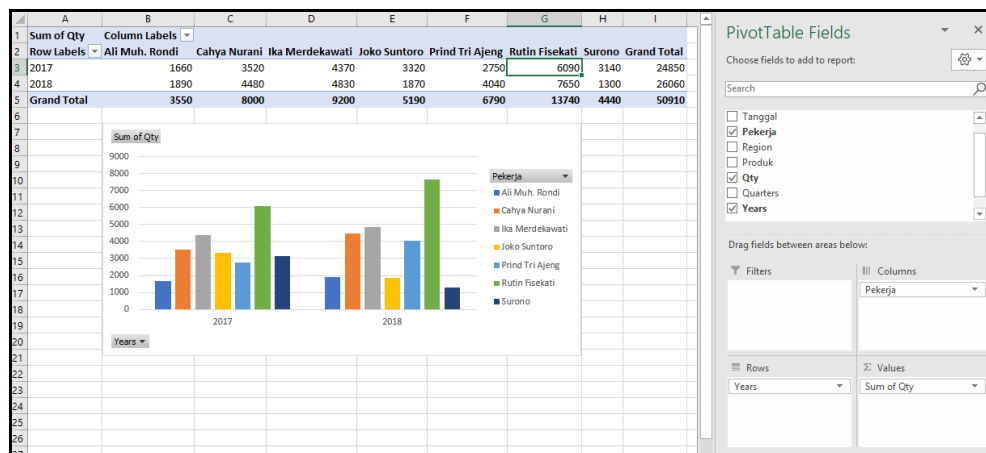
1. Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Sales per Region, lalu letakkan kursor pada sel A1
2. Tambahkan pivot table dengan mengambil semua data dari sheet Master Data
3. Pada bagian PivotTables Fields, pilih Pekerja lalu drag ke bagian Columns, pilih Years lalu drag ke Rows dan pilih Qty lalu drag ke bagian Values



4. Letakkan kursor di sel B3, lalu klik menu Analyze, klik PivotChart. Pada bagian Insert Chart pilih Column – Clustered Column, lalu klik OK

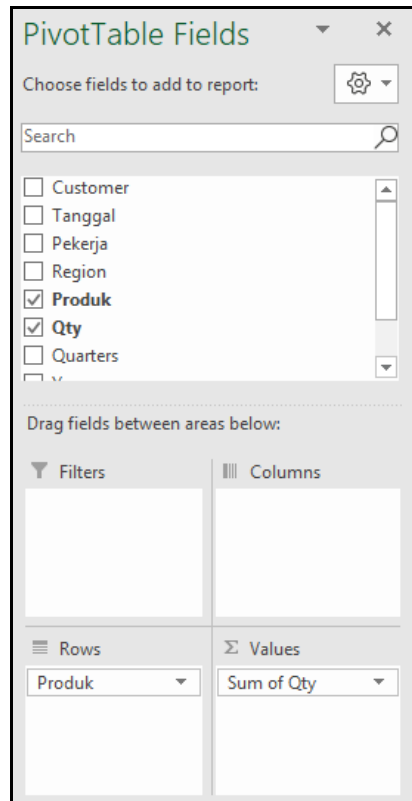


5. Hasil akhir pivot table dan pivot chart Sales per Pekerja terlihat seperti gambar di bawah ini

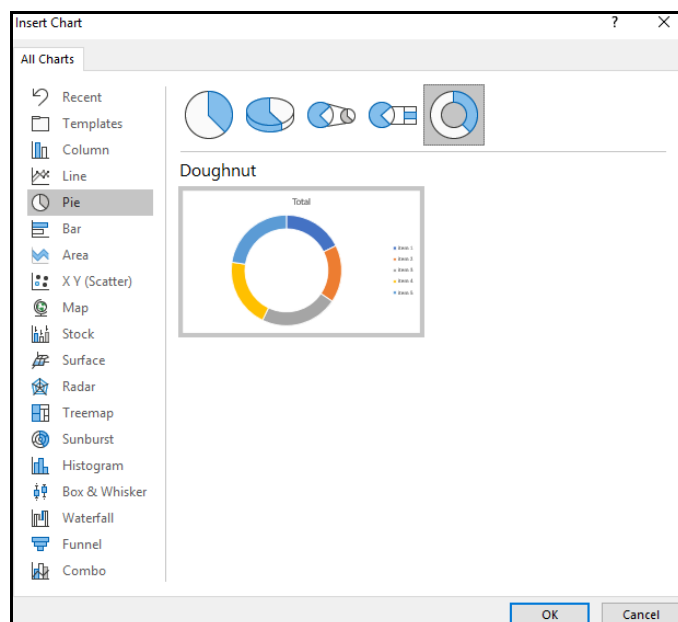


3.4 Pivot Table dan Pivot Chart Sales per Produk

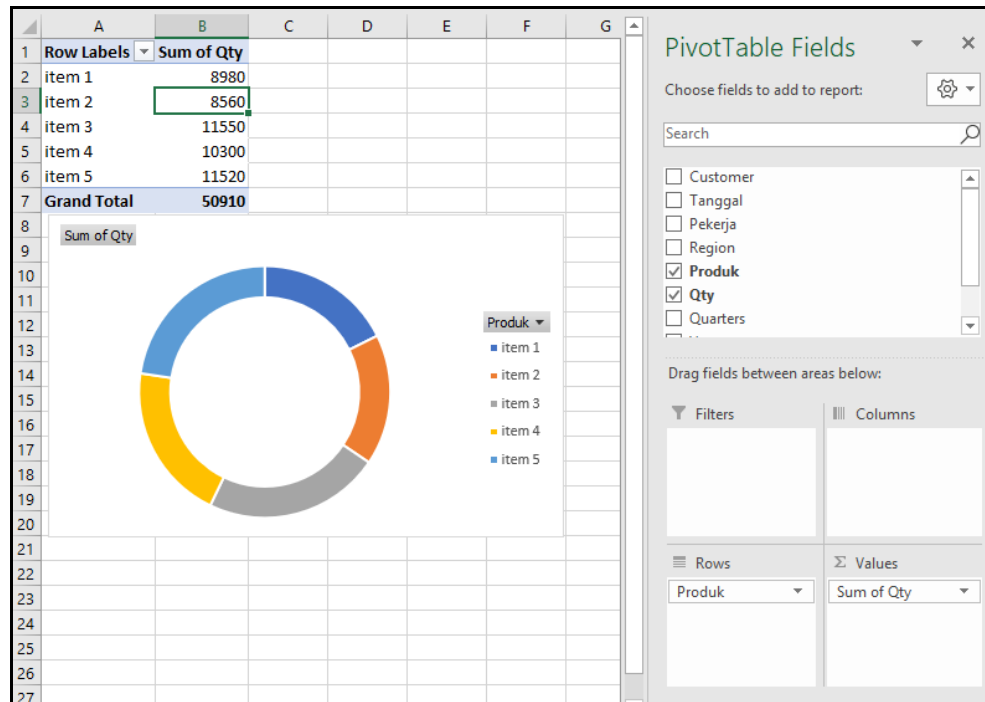
1. Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Sales per Produk, lalu letakkan kursor pada sel A1
2. Tambahkan pivot table dengan mengambil semua data dari sheet Master Data
3. Pada bagian PivotTables Fields, pilih Produk lalu drag ke Rows dan pilih Qty lalu drag ke bagian Values



4. Letakkan kursor di sel A2, lalu klik menu Analyze, klik PivotChart. Pada bagian Insert Chart pilih Pie - Doughnut, lalu klik OK

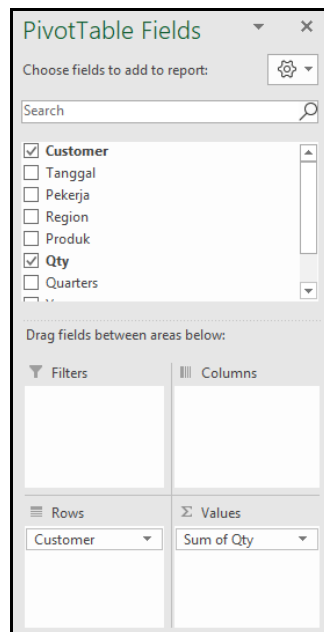


5. Hasil akhir pivot table dan pivot chart Sales per Produk terlihat seperti gambar di bawah ini

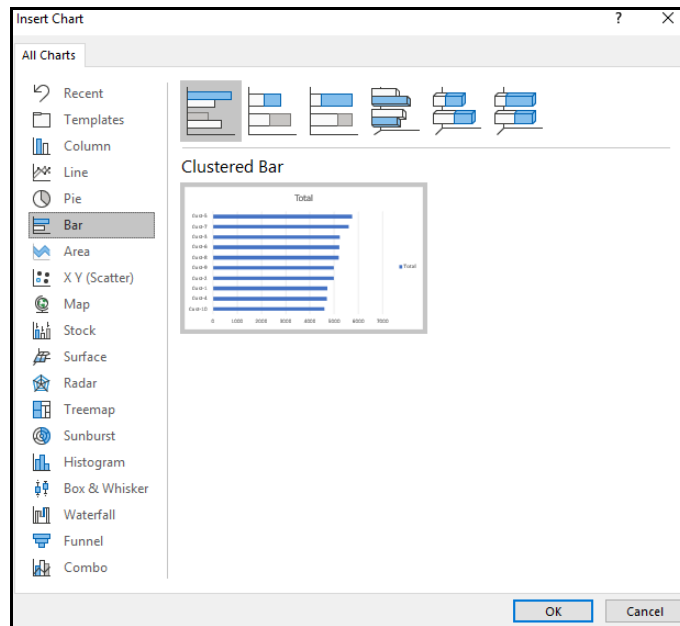


3.5 Pivot Table dan Pivot Chart Sales per Customer

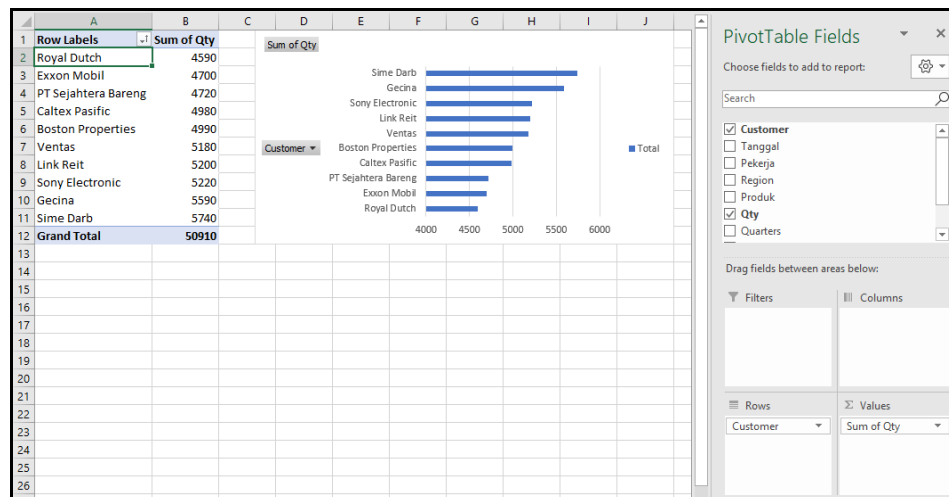
1. Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Sales per Customer, lalu letakkan kursor pada sel A1
2. Tambahkan pivot table dengan mengambil semua data dari sheet Master Data
3. Pada bagian PivotTables Fields, pilih Customer lalu drag ke Rows dan pilih Qty lalu drag ke bagian Values



- Letakkan kursor di sel A2, lalu klik menu Analyze, klik PivotChart. Pada bagian Insert Chart pilih Bar – Clustered Bar, lalu klik OK

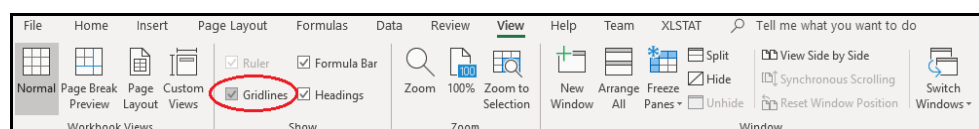


- Hasil akhir pivot table dan pivot chart Sales per Produk terlihat seperti gambar di bawah ini

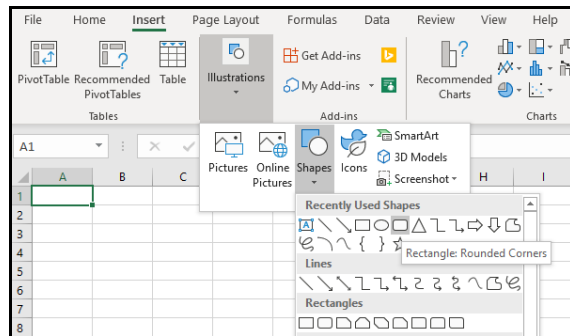


3.6 Pembuatan Dashboard

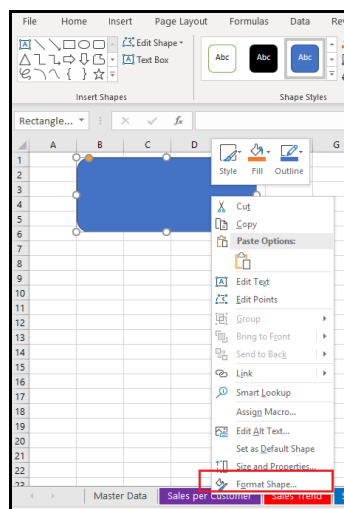
- Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Dashboard
- Klik menu View, hilangkan tanda check pada Gridlines



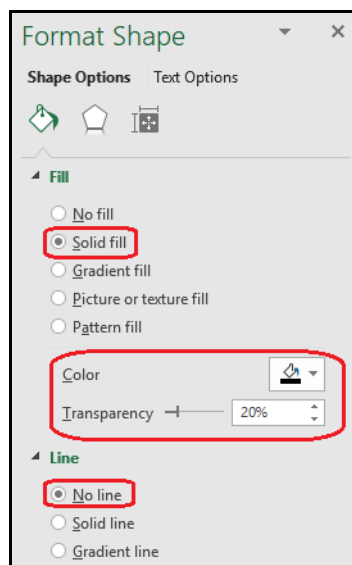
3. Klik Insert – Shape – Rectangle



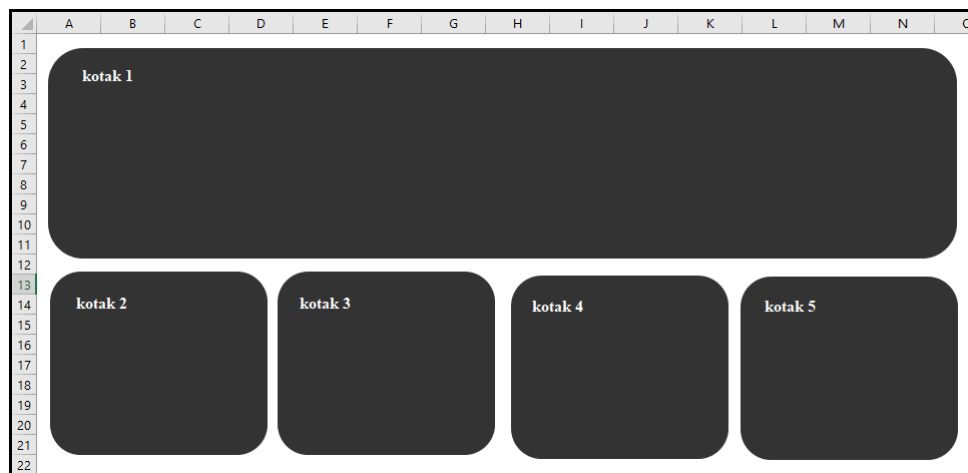
4. Klik kanan pada Shape Rectangle, kemudian pilih Format Shape



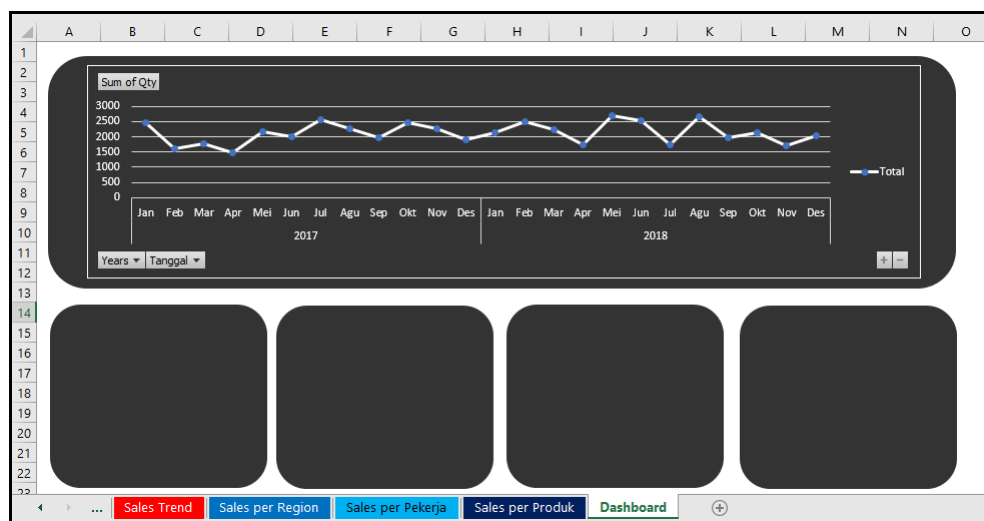
5. Pada box Format Shape ubah Fill menjadi Solid Fill, Color: hitam, dan Transparency: 20%. Kemudian ubah Line menjadi No line.



6. Buat layout seperti di bawah ini



7. Copy hasil Pivot Chart Sales Trend, lalu paste ke dalam kotak 1, lalu setting layout seperti gambar di bawah ini



8. Sama seperti langkah ketujuh, copy paste Pivot Chart Sales per Customer ke kotak 2, copy paste Pivot Chart Sales per Region ke kotak 3, copy paste Pivot Chart Sales per Pekerja ke kotak 4, dan copy paste Pivot Chart Sales per Produk ke kotak 5. Sehingga tampilan seperti gambar di bawah ini



9. Klik menu Insert – Slicer Tools, tambahkan slicer Years, Customer, Region, Pekerja dan Produk, lalu susun seperti gambar di bawah ini. Kemudian tambahkan koneksi per masing-masing slicer, agar grafik bisa berubah-ubah sesuai dengan value yang dipilih

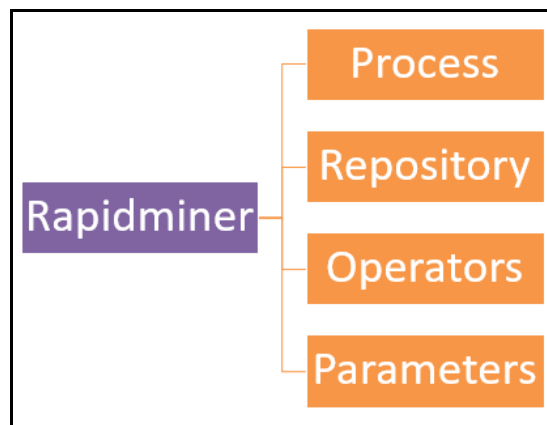


BAB 4

PENGENALAN RAPIDMINER DAN REPRESENTASI PENGETAHUAN

4.1 Pengenalan Rapidminer

Struktur Utama Rapidminer terdiri dari Process, Repository, Operators, dan Parameters. Process merupakan halaman utama dalam Rapidminer. Repository berfungsi sebagai lokasi penyimpanan terpusat untuk data dan proses analisis. Operators merupakan sekumpulan algoritme dan metode yang dibutuhkan untuk proses analisis. Parameter digunakan untuk pengaturan lebih lanjut pada bagian operator.



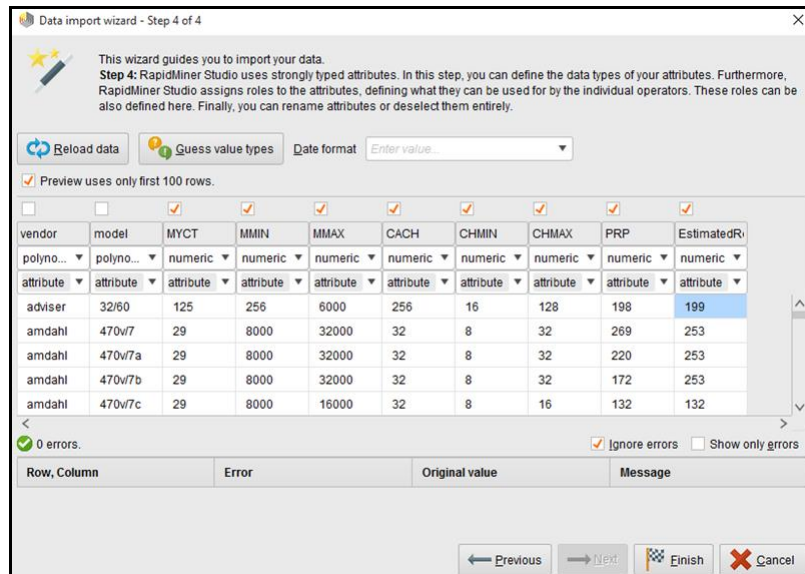
Gambar 4.1 Struktur Utama Rapidminer

4.2 Representasi Pengetahuan

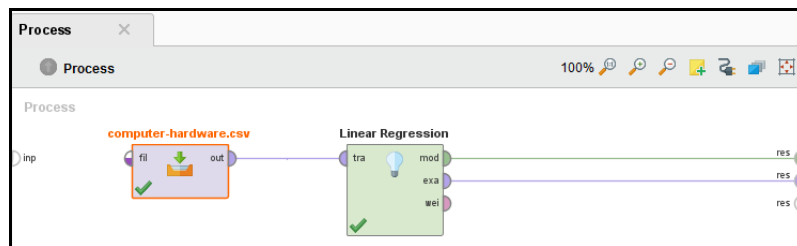
4.2.1 Estimasi

Berikut adalah langkah-langkah pembentukan model dan representasi pengetahuan dari metode estimasi:

1. Lakukan training pada dataset computer-hardware.csv



- Gunakan algoritme Linear Regression sebagai pembentuk model dalam metode estimasi



- Tampilkan representasi pengetahuan metode estimasi (dalam bentuk formula matematika)

ExampleSet (209 examples, 1 special attribute, 7 regular attributes) Filter (209 / 209 examples): all

| Row No. | EstimatedR... | MYCT | MMIN | MMAX | CACH | CHMIN | CHMAX | PRP |
|---------|---------------|------|-------|-------|------|-------|-------|------|
| 1 | 199 | 125 | 256 | 6000 | 256 | 16 | 128 | 198 |
| 2 | 253 | 29 | 8000 | 32000 | 32 | 8 | 32 | 269 |
| 3 | 253 | 29 | 8000 | 32000 | 32 | 8 | 32 | 220 |
| 4 | 253 | 29 | 8000 | 32000 | 32 | 8 | 32 | 172 |
| 5 | 132 | 29 | 8000 | 16000 | 32 | 8 | 16 | 132 |
| 6 | 290 | 26 | 8000 | 32000 | 64 | 8 | 32 | 318 |
| 7 | 381 | 23 | 16000 | 32000 | 64 | 16 | 32 | 367 |
| 8 | 381 | 23 | 16000 | 32000 | 64 | 16 | 32 | 489 |
| 9 | 749 | 23 | 16000 | 64000 | 64 | 16 | 32 | 636 |
| 10 | 1238 | 23 | 32000 | 64000 | 128 | 32 | 64 | 1144 |

LinearRegression

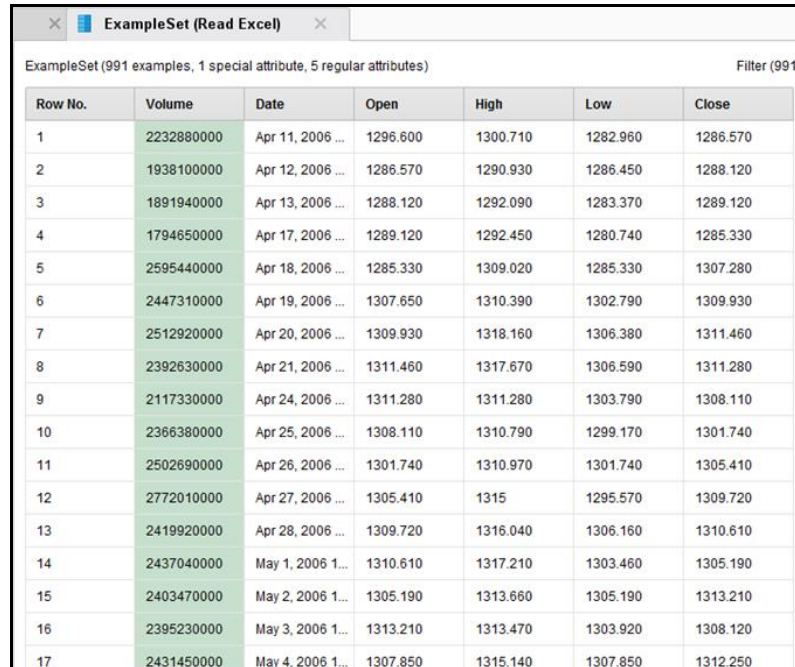
0.038 * MYCT
+ 0.005 * MMIN
+ 0.003 * MMAX
+ 0.124 * CACH
+ 0.344 * CHMAX
+ 0.577 * PRP
- 34.239

ERP=0,038*MYCT+0,005*MMIN+0,003*MMAX+0,124*CACH+0,344*CHMAX+0,577*PRP-34,239

4.2.2 Prediksi

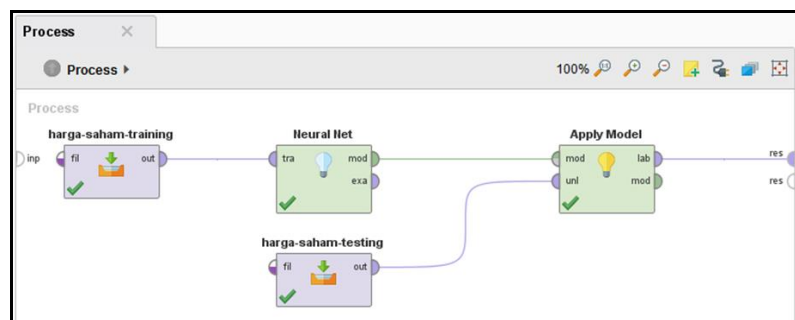
Berikut adalah langkah-langkah pembentukan model dan representasi pengetahuan dari metode prediksi:

1. Lakukan training pada dataset harga saham.xls

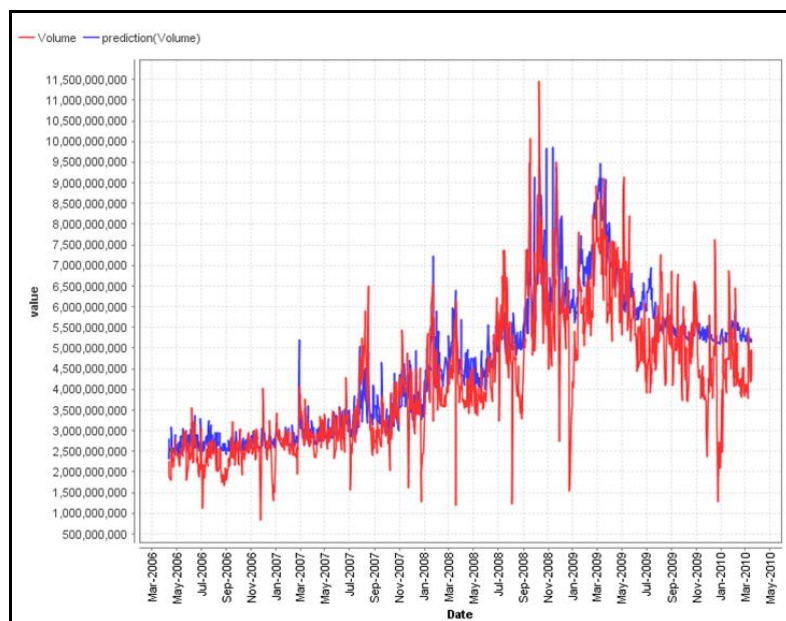


| Row No. | Volume | Date | Open | High | Low | Close |
|---------|------------|------------------|----------|----------|----------|----------|
| 1 | 2232880000 | Apr 11, 2006 ... | 1296.600 | 1300.710 | 1282.960 | 1286.570 |
| 2 | 1938100000 | Apr 12, 2006 ... | 1286.570 | 1290.930 | 1286.450 | 1288.120 |
| 3 | 1891940000 | Apr 13, 2006 ... | 1288.120 | 1292.090 | 1283.370 | 1289.120 |
| 4 | 1794650000 | Apr 17, 2006 ... | 1289.120 | 1292.450 | 1280.740 | 1285.330 |
| 5 | 2595440000 | Apr 18, 2006 ... | 1285.330 | 1309.020 | 1285.330 | 1307.280 |
| 6 | 2447310000 | Apr 19, 2006 ... | 1307.650 | 1310.390 | 1302.790 | 1309.930 |
| 7 | 2512920000 | Apr 20, 2006 ... | 1309.930 | 1318.160 | 1306.380 | 1311.460 |
| 8 | 2392630000 | Apr 21, 2006 ... | 1311.460 | 1317.670 | 1306.590 | 1311.280 |
| 9 | 2117330000 | Apr 24, 2006 ... | 1311.280 | 1311.280 | 1303.790 | 1308.110 |
| 10 | 2366380000 | Apr 25, 2006 ... | 1308.110 | 1310.790 | 1299.170 | 1301.740 |
| 11 | 2502690000 | Apr 26, 2006 ... | 1301.740 | 1310.970 | 1301.740 | 1305.410 |
| 12 | 2772010000 | Apr 27, 2006 ... | 1305.410 | 1315 | 1295.570 | 1309.720 |
| 13 | 2419920000 | Apr 28, 2006 ... | 1309.720 | 1316.040 | 1306.160 | 1310.610 |
| 14 | 2437040000 | May 1, 2006 1... | 1310.610 | 1317.210 | 1303.460 | 1305.190 |
| 15 | 2403470000 | May 2, 2006 1... | 1305.190 | 1313.660 | 1305.190 | 1313.210 |
| 16 | 2395230000 | May 3, 2006 1... | 1313.210 | 1313.470 | 1303.920 | 1308.120 |
| 17 | 2431450000 | May 4, 2006 1... | 1307.850 | 1315.140 | 1307.850 | 1312.250 |

2. Gunakan algoritme Neural Network sebagai pembentukan model dalam metode prediksi



3. Tampilkan representasi pengetahuan metode prediksi (dalam bentuk grafik)



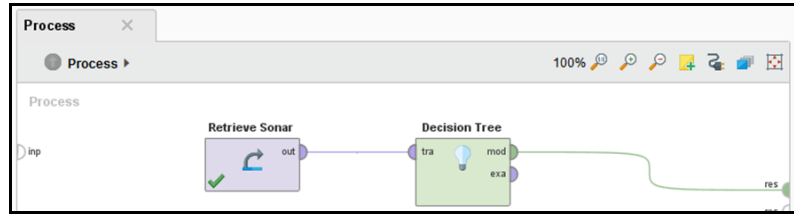
4.2.3 Klasifikasi

Berikut adalah langkah-langkah pembentukan model dan representasi pengetahuan dari metode klasifikasi:

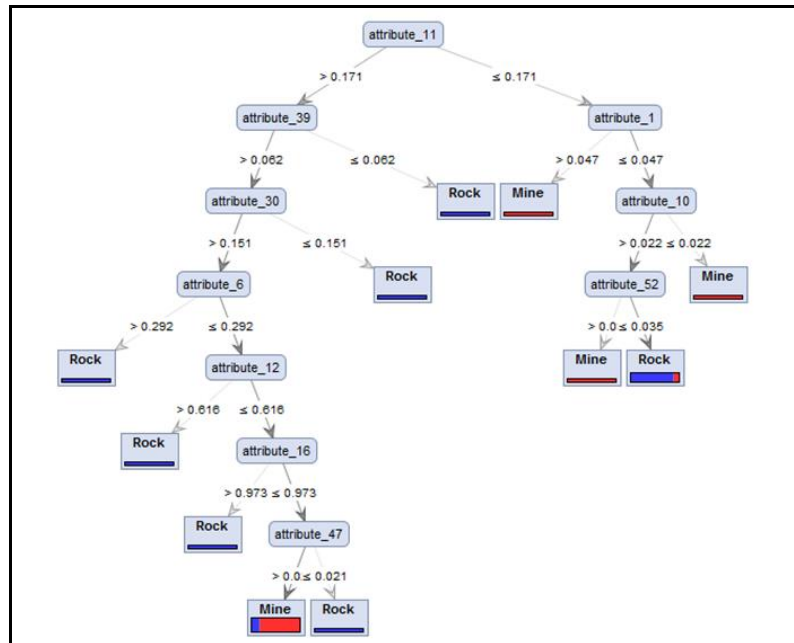
1. Lakukan training pada data sample sonar di bagian Samples Rapidminer

| ExampleSet (208 examples, 1 special attribute, 60 regular attributes) | | | | | | | | Filter (208 / 208 examples): |
|---|-------|-------------|-------------|-------------|-------------|-------------|-------------|------------------------------|
| Row No. | class | attribute_1 | attribute_2 | attribute_3 | attribute_4 | attribute_5 | attribute_6 | attribute_7 |
| 1 | Rock | 0.020 | 0.037 | 0.043 | 0.021 | 0.095 | 0.099 | 0.154 |
| 2 | Rock | 0.045 | 0.052 | 0.084 | 0.069 | 0.118 | 0.258 | 0.216 |
| 3 | Rock | 0.026 | 0.058 | 0.110 | 0.108 | 0.097 | 0.228 | 0.243 |
| 4 | Rock | 0.010 | 0.017 | 0.062 | 0.021 | 0.021 | 0.037 | 0.110 |
| 5 | Rock | 0.076 | 0.067 | 0.048 | 0.039 | 0.059 | 0.065 | 0.121 |
| 6 | Rock | 0.029 | 0.045 | 0.028 | 0.017 | 0.038 | 0.099 | 0.120 |
| 7 | Rock | 0.032 | 0.096 | 0.132 | 0.141 | 0.167 | 0.171 | 0.073 |
| 8 | Rock | 0.052 | 0.055 | 0.084 | 0.032 | 0.116 | 0.092 | 0.103 |
| 9 | Rock | 0.022 | 0.037 | 0.048 | 0.048 | 0.065 | 0.059 | 0.075 |
| 10 | Rock | 0.016 | 0.017 | 0.035 | 0.007 | 0.019 | 0.067 | 0.106 |
| 11 | Rock | 0.004 | 0.006 | 0.015 | 0.034 | 0.031 | 0.028 | 0.040 |
| 12 | Rock | 0.012 | 0.031 | 0.017 | 0.031 | 0.036 | 0.010 | 0.018 |
| 13 | Rock | 0.008 | 0.009 | 0.005 | 0.025 | 0.034 | 0.055 | 0.053 |
| 14 | Rock | 0.009 | 0.006 | 0.025 | 0.049 | 0.120 | 0.159 | 0.139 |
| 15 | Rock | 0.012 | 0.043 | 0.060 | 0.045 | 0.060 | 0.035 | 0.053 |
| 16 | Rock | 0.030 | 0.061 | 0.065 | 0.092 | 0.162 | 0.229 | 0.218 |

2. Gunakan algoritme C4.5 sebagai pembentukkan model dalam metode klasifikasi



3. Tampilkan representasi pengetahuan metode klasifikasi (dalam bentuk pohon keputusan).



4.2.4 Klustering

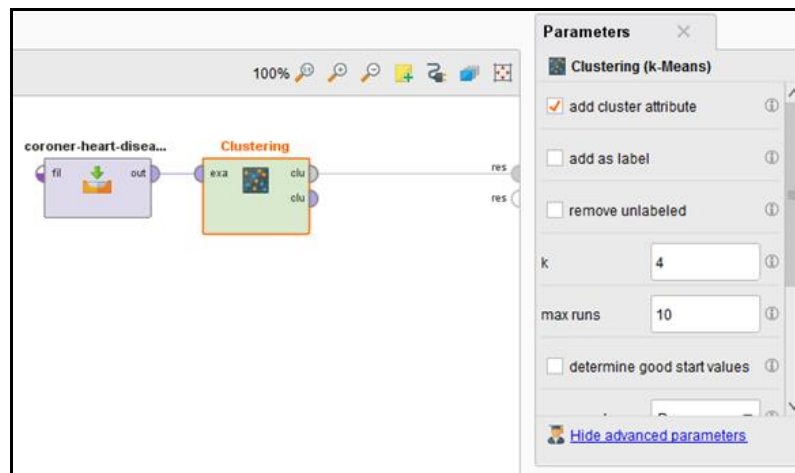
Berikut adalah langkah-langkah pembentukan model dan representasi pengetahuan dari metode klasifikasi:

1. Lakukan training pada dataset heart-disease.csv

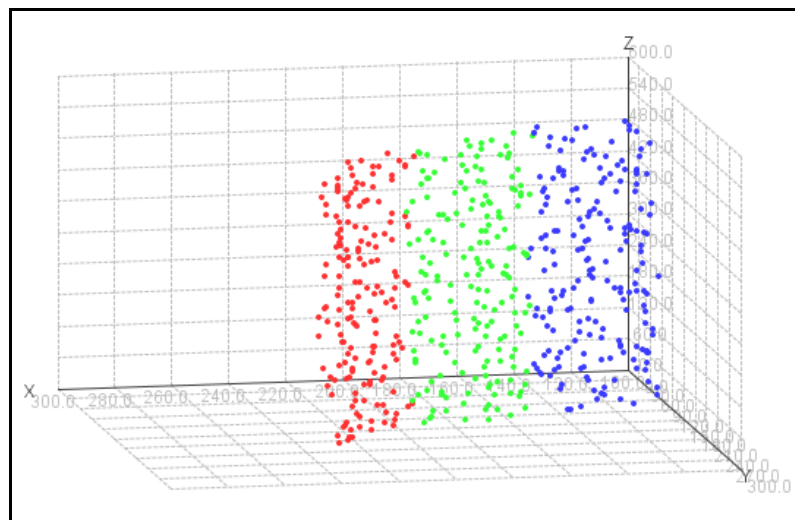
| Name | Type | Miss. | Statistics | | | | Filter (3 / 3 attributes): |
|-------------|---------|-------|------------|------------|--------------------|---------------------|----------------------------|
| Weight | Integer | 0 | Min 95 | Max 203 | Average 143.572 | Deviation 30.837 | Filter |
| Cholesterol | Integer | 0 | Min 102 | Max 235 | Average 170.433 | Deviation 39.147 | |
| Gender | Integer | 0 | Min 0 | Max 1 | Average 0.514 | Deviation 0.500 | |

| ExampleSet (547 examples, 0 special attributes, 3 regular attributes) | | | |
|---|--------|-------------|--------|
| Row No. | Weight | Cholesterol | Gender |
| 1 | 102 | 111 | 1 |
| 2 | 115 | 135 | 1 |
| 3 | 115 | 136 | 1 |
| 4 | 140 | 167 | 0 |
| 5 | 130 | 158 | 1 |
| 6 | 198 | 227 | 1 |

- Gunakan algoritme K-Means sebagai pembentuk model dalam metode klustering



- Tampilkan representasi pengetahuan metode klustering (dalam bentuk pengelompokan data).



BAB 5

DATA WAREHOUSE FOR BUSINESS INTELLIGENCE

(Studi Kasus: Klustering Data Pekerja)

5.1 Konteks dan Perspektif

Salah satu tugas pokok dari fungsi human resource development (HRD) adalah pengembangan dan training pekerja. HRD berperan aktif dalam peningkatan kinerja pekerja di salah satu perusahaan dengan cara edukasi dan training untuk karyawan.

Zizi adalah seorang manager HRD di PT Salesman, dan Nila adalah salah satu database administrator di PT Salesman. Zizi meminta tolong kepada Nila untuk mengumpulkan data-data pekerja dengan jabatan sebagai sales di PT Salesman. Zizi akan mengadakan training untuk para sales yang kurang produktif. Training Ini bertujuan agar para sales bisa lebih produktif dan mampu bekerja sesuai target yang telah ditentukan di PT Salesman.

5.2 Studi Kasus

5.2.1 Business Understanding

Zizi adalah seorang manager HRD di PT Salesman, dan Nila adalah salah satu database administrator di PT Salesman. Zizi meminta tolong kepada Nila untuk mengumpulkan data-data karyawan dengan jabatan sebagai sales di PT Salesman. Zizi akan mengadakan training untuk para sales yang kurang produktif. Training Ini bertujuan agar para sales bisa lebih produktif dan mampu bekerja sesuai target yang telah ditentukan di PT Salesman.

Nila menggunakan database MySQL untuk pengelolaan database di PT Salesman. Database yang dibangun Nila terdiri dari banyak tabel yang saling berelasi. Zizi ingin mengelompokan (klustering) data pekerja sales menjadi tiga kelompok, yaitu kelompok sangat produktif, produktif, dan kurang. Kelompok dengan hasil kurang inilah yang nantinya akan diikutkan training. Parameter yang digunakan untuk pengelompokan sales berdasarkan pada dua parameter utama yaitu capaian kerja, dan lama bekerja. Selain dua parameter utama ditambahkn pula dua

parameter tambahan untuk memudahkan proses identifikasi yaitu nomor pokok karyawan (NPK), dan jenis kelamin. Nila membuat query MySQL berisi keempat atribut tersebut, setelah Itu query yang berisi keempat atribut dikonversi ke format excel (.xlsx).

5.2.2 Data Understanding

Jumlah pekerja dengan jabatan sales pada PT Salesman sejumlah 50 (lima puluh) pekerja. Kelima puluh pekerja tersebut akan dikelompokkan menjadi tiga kelompok, yaitu kelompok sangat produktif, produktif, dan kurang. Pengelompokan data berdasarkan pada empat atribut yaitu sebagai berikut:

- a. **NPK**, berisi data nomor pokok pekerja, tipe data polynominal. NPK berfungsi sebagai Identifier (id).
- b. **Capaian**, adalah data jumlah capaian penjualan produk yang berhasil terjual dalam satu tahun. Tipe data numerik.
- c. **Waktu**, adalah data lama bekerja pekerja (sales). Tipe data numerik.
- d. **JK**, adala data jenis kelamin, berisi data: 0 untuk perempuan dan 1 untuk laki-laki.

5.2.3 Data Preparation

Berdasarkan data understanding yang sudah dijelaskan pada sub bab sebelumnya, dapat dilihat bahwa pada dataset pekerja PT Salesman dengan jabatan sebagai sales terdiri dari 50 (lima puluh) pekerja. Kelima puluh pekerja tersebut akan dikelompokkan menjadi tiga kelompok yaitu sangat produktif, produktif, dan kurang berdasarkan dengan empat parameter yaitu NPK, capaian, waktu, dan jenis kelamin.

Data import wizard - Step 4 of 4

This wizard guides you to import your data.
Step 4: RapidMiner Studio uses strongly typed attributes. In this step, you can define the data types of your attributes. Furthermore, RapidMiner Studio assigns roles to the attributes, defining what they can be used for by the individual operators. These roles can be also defined here. Finally, you can rename attributes or deselect them entirely.

Date format:

☒ Preview uses only first 100 rows.

| <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| NPK | Capaian | Waktu | JK |
| polyno... | numeric | numeric | integer |
| id | attribute | attribute | attribute |
| SLS-201... | 73 | 6 | 1 |
| SLS-201... | 99 | 7 | 0 |
| SLS-200... | 94 | 10 | 0 |
| SLS-201... | 56 | 6 | 1 |
| SLS-201... | 71 | 8 | 0 |

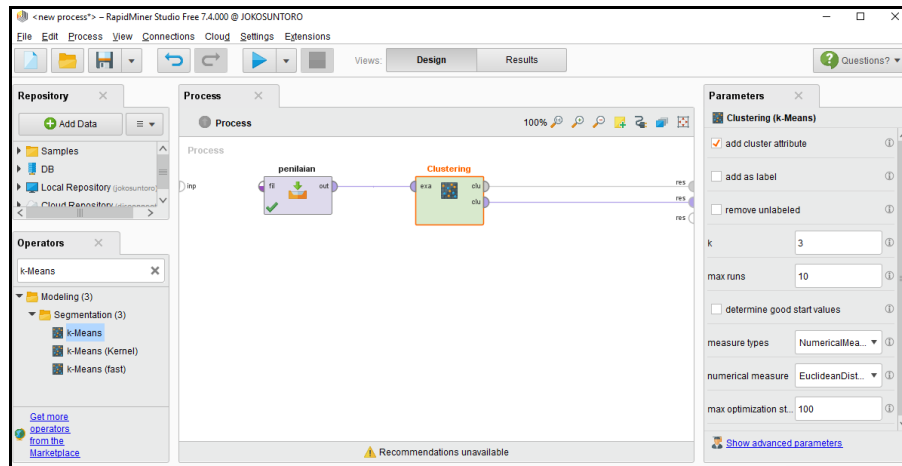
0 errors. ☒ Ignore errors ☐ Show only errors

| Row, Column | Error | Original value | Message |
|-------------|-------|----------------|---------|
|-------------|-------|----------------|---------|

5.2.4 Modeling

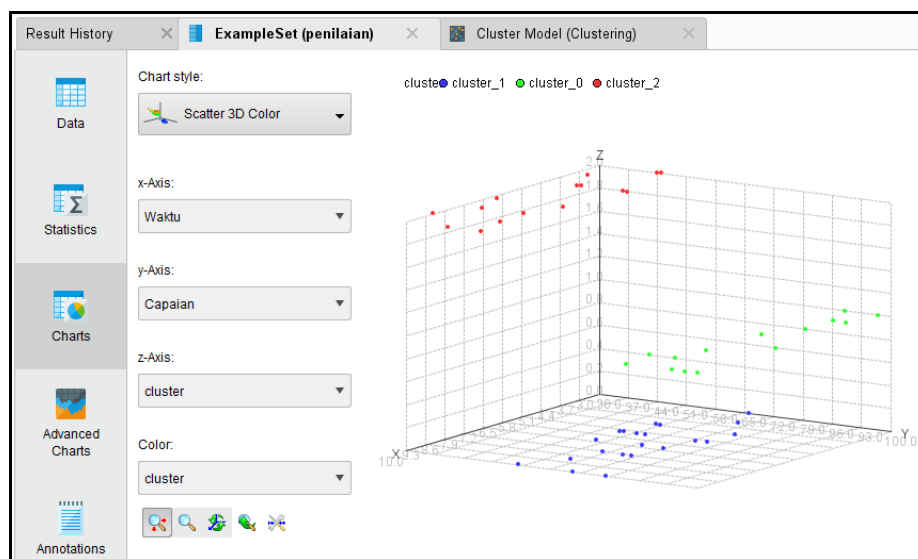
Dalam tahapan business understanding dan data understanding telah dijelaskan bahwa, Zizi yang menjabat sebagai manager HRD akan mengelompokkan pekerjaanya dengan jabatan sales menjadi tiga kelompok, yaitu kelompok sangat produktif, produktif, dan kurang. Kelompok kurang akan diajukan oleh Zizi untuk mengikuti training agar lebih produktif.

Pengelompokan pekerja digunakan peranan data mining klustering. Salah satu algoritme klustering yang banyak digunakan adalah algoritme K-Means. Algoritme K-Means dengan parameter nilai K adalah tiga (K=3) akan digunakan untuk tahapan modeling.

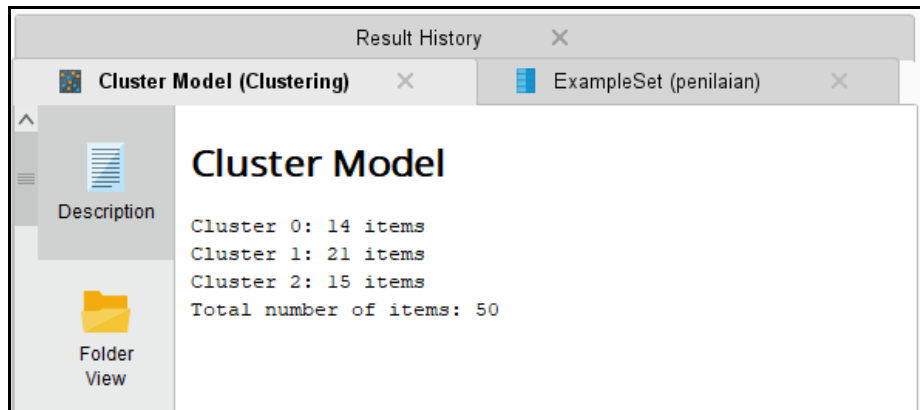


5.2.5 Evaluation

Setelah data pekerja PT Salesma dimodelkan pada sub-bab sebelumnya, langkah selanjutnya adalah dilakukan evaluasi. Hasil pengelompokan data pekerja (klustering) akan didapatkan pada tahapan evaluasi.



Untuk mengetahui jumlah data (pekerja) pada masing-masing kelompok, maka masuk ke tab Cluster Model (Clustering) - Description. Terlihat bahwa jumlah data (pekerja) pada Cluster 0 sejumlah 14 pekerja, jumlah data (pekerja) pada Cluster 1 sejumlah 21 pekerja, dan jumlah data (pekerja) pada Cluster 2 sejumlah 15 pekerja.



5.2.6 Deployment

Setelah diketahui kelompok sales dengan kategori kurang produktif, maka Nila akan membuat query SQL dengan filter kelompok kurang produktif sebagai bahan laporan untuk Zizi. Seperti yang telah dijelaskan pada tahapan business understanding, bahwa Nila menyimpan database pekerja dalam repository lokal menggunakan MySQL.

```
MariaDB [hrd]> select a.NPK, a.nama, b.capaian, b.waktu, a.penempatan
-> from pekerja a, penilaian b
-> where a.NPK = b.NPK
-> and b.capaian <= 50
-> and b.waktu <= 10;
```

| NPK | nama | capaian | waktu | penempatan |
|-------------|---------|---------|-------|------------|
| SLS-2009-40 | Cucuk | 48 | 10 | Semarang |
| SLS-2012-3 | Didik | 32 | 7 | Semarang |
| SLS-2012-6 | Dwiandi | 48 | 7 | Surabaya |
| SLS-2010-18 | Faisal | 30 | 9 | Yogyakarta |
| SLS-2010-28 | Galih | 46 | 9 | Medan |
| SLS-2009-50 | Hendi | 40 | 10 | Surabaya |
| SLS-2014-46 | Ika M | 50 | 5 | Jakarta |
| SLS-2014-22 | Irfan | 49 | 5 | Jakarta |
| SLS-2016-45 | Radit | 44 | 3 | Medan |
| SLS-2016-24 | Rani | 45 | 3 | Medan |
| SLS-2015-36 | Rendy | 34 | 4 | Medan |
| SLS-2014-13 | Riska | 39 | 5 | Semarang |
| SLS-2014-11 | Surono | 38 | 5 | Bandung |
| SLS-2011-43 | Yudi | 45 | 8 | Lombok |
| SLS-2011-48 | Zidni | 35 | 8 | Yogyakarta |

```
15 rows in set (0.07 sec)
```

TENTANG PENULIS



Joko Suntoro, menyelesaikan pendidikan SD tahun 2001 di SDN Pandean Lamper 03 Semarang, kemudian melanjutkan pendidikan di SMPN 32 Semarang, lulus tahun 2004. Setelah itu melanjutkan pendidikan di SMA Institut Indonesia, lulus tahun 2007. Memperoleh gelar S.Kom pada jurusan Teknik Informatika di Universitas Semarang pada tahun 2015, dan gelar M.Kom pada jurusan Magister Teknik Informatika di Universitas Dian Nuswantoro, Semarang tahun 2016. Sejak tahun 2008 tergabung dalam tim operasi di Domestic Gas Region IV. Penulis juga aktif sebagai dosen jurusan Teknik Informatika di Universitas Semarang dan tergabung dalam tim penelitian Intelligent Systems Research Group yang dipimpin oleh Romi Satria Wahono, Ph.D. Bidang penelitian penulis adalah data mining, software engineering, dan machine learning. Penulis dapat dihubungi melalui email jokosuntoro@usm.ac.id.