

**MODUL PRAKTIKUM MATA KULIAH  
DATA WAREHOUSE**

**DATA WAREHOUSE FOR BUSINESS INTELLIGENCE**

**Disusun Oleh:**

**Joko Suntoro, M.Kom.**

**Prind Triajeng P., M.Kom.**



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

## DAFTAR ISI

DAFTAR ISI.....	2
BAB 1    DASAR-DASAR SQL .....	4
1.1   Pembuatan Database .....	4
1.2   Import Data .....	6
1.3   Syntax Dasar MySQL.....	8
1.3.1   Melihat Struktur Tabel .....	8
1.3.2   Menampilkan Data .....	10
1.3.3   Menambahkan Data .....	13
1.3.4   Mengubah Data .....	15
1.3.5   Menghapus data .....	16
BAB 2    SQL TINGKAT LANJUT .....	17
2.1   Cascade pada MySQL.....	17
BAB 3    VISUALISASI DATA (PIVOT TABLE, CHART, MAP DAN DASHBOARD).....	18
3.1   Pivot Table dan Pivot Chart Trend Penjualan.....	18
3.2   Pivot Table dan Maps Chart Sales per Region .....	22
3.3   Pivot Table dan Pivot Chart Sales per Pekerja .....	26
3.4   Pivot Table dan Pivot Chart Sales per Produk.....	27
3.5   Pivot Table dan Pivot Chart Sales per Customer.....	29
BAB 4    PENGENALAN RAPIDMINER.....	31
BAB 5    DATA WAREHOUSE FOR BUSINESS INTELLIGENCE I (Studi Kasus: Prediksi Kelulusan Mahasiswa) .....	32
BAB 6    DATA WAREHOUSE FOR BUSINESS INTELLIGENCE II (Studi Kasus: Klustering Data Pekerja) .....	33



# BAB 1

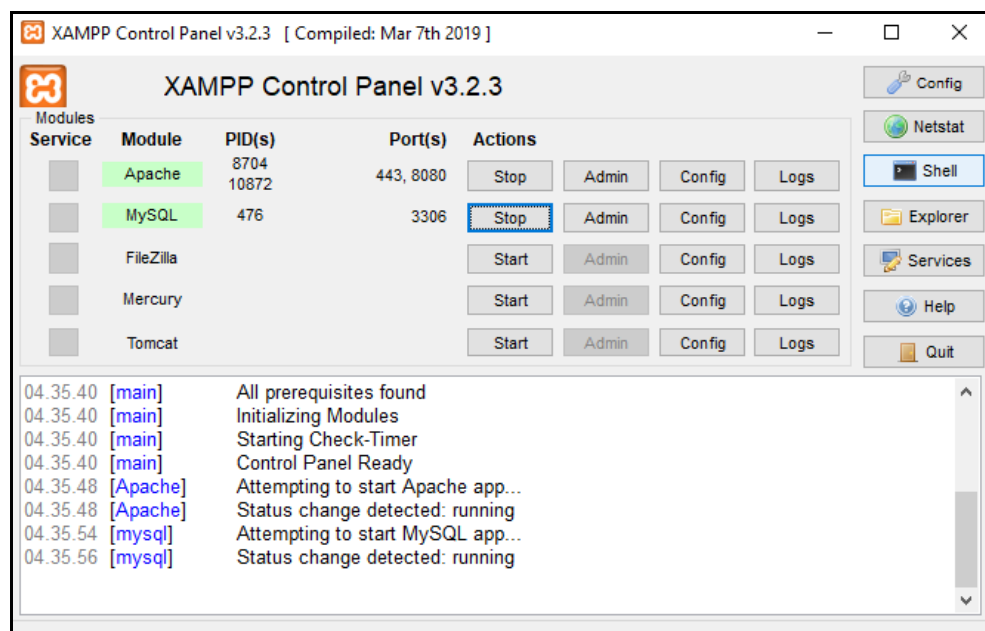
## DASAR-DASAR SQL

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.

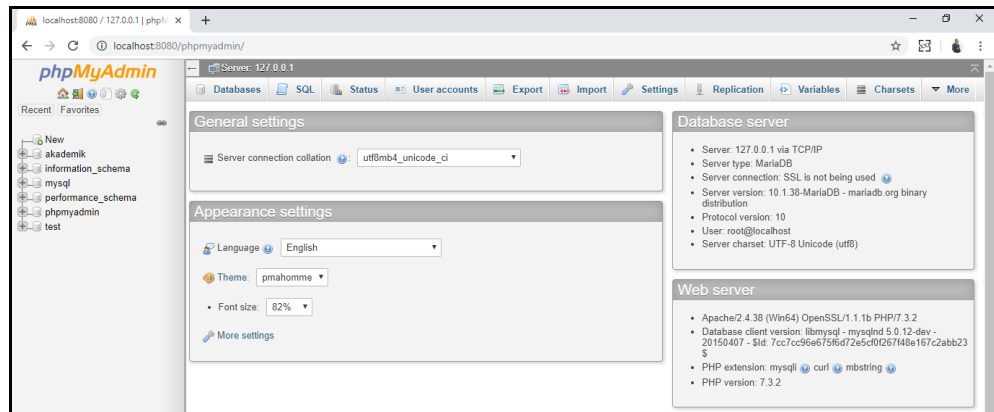
### 1.1 Pembuatan Database

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:

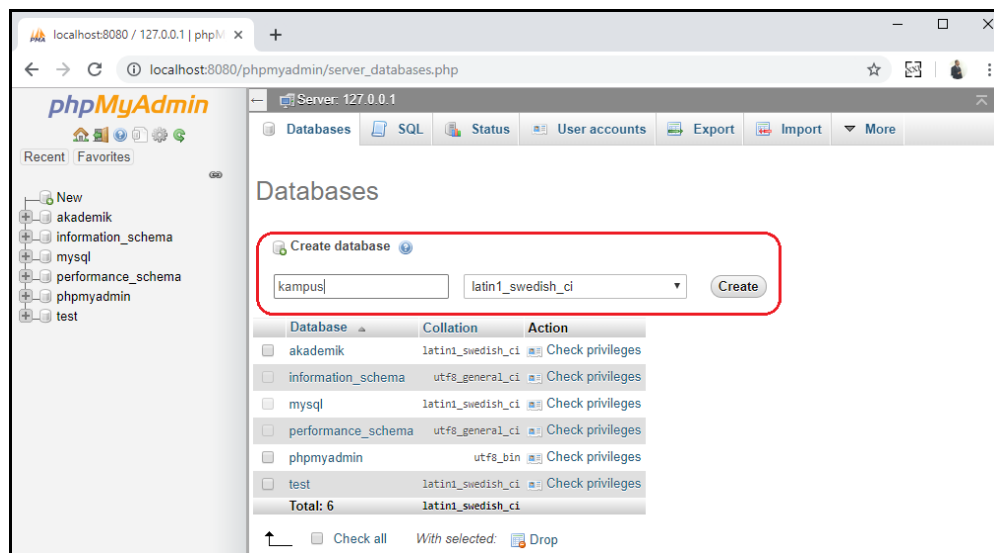
1. Aktifkan xampp control panel.



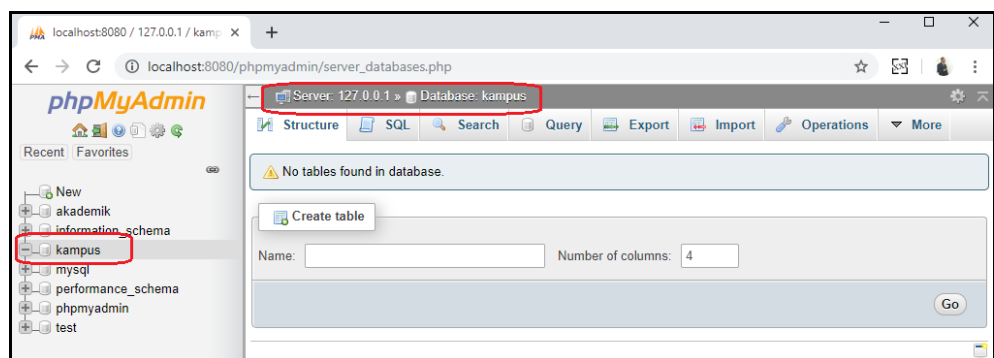
2. Buka browser, kemudian ketikkan url <http://localhost/phpmyadmin/> (apabila menggunakan port 8080, ubah menjadi <http://localhost:8080/phpmyadmin/>).



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



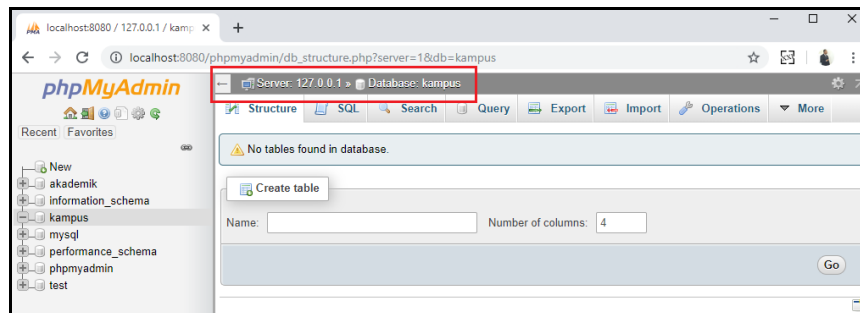
4. Pada bagian tab menu sebelah kanan, akan muncul database dengan nama kampus.



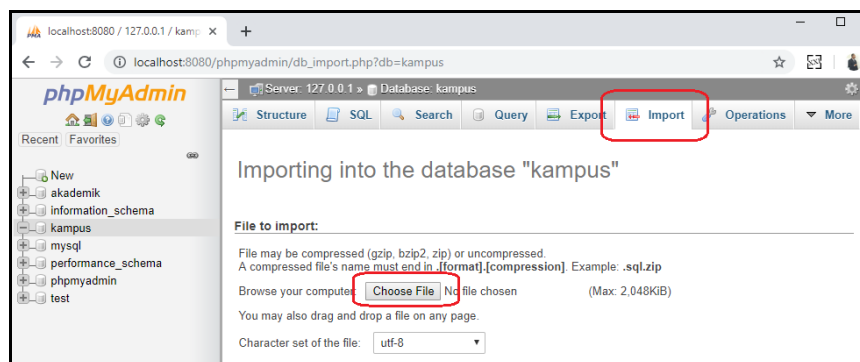
## 1.2 Import Data

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:

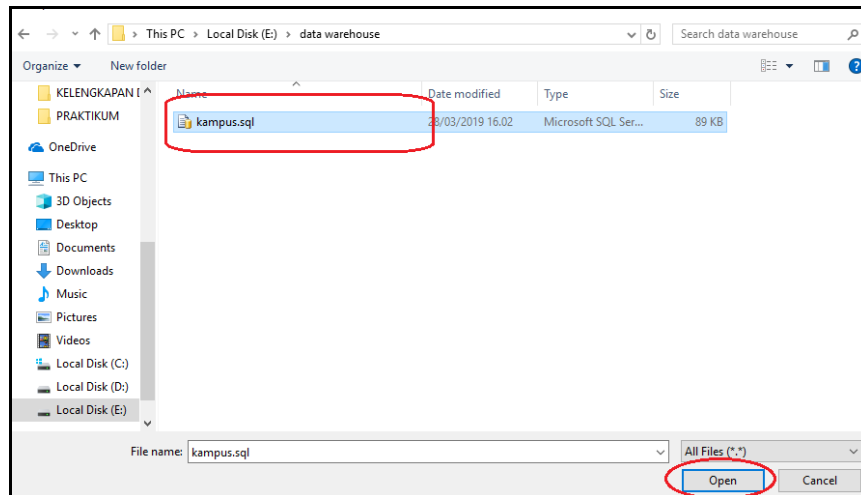
1. Masuk ke dalam database kampus, pastikan (menu bar atas berada pada database kampus)



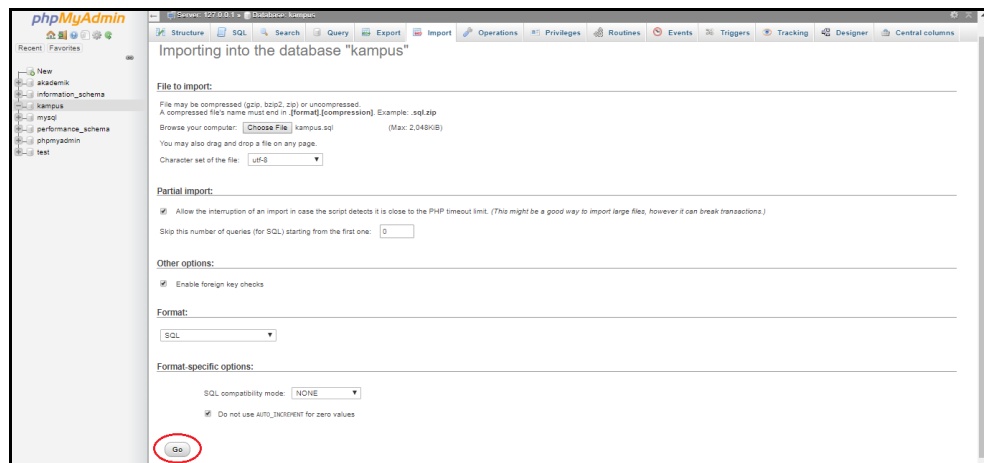
2. Klik menu Import, pada bagian File to import, lalu klik Choose File



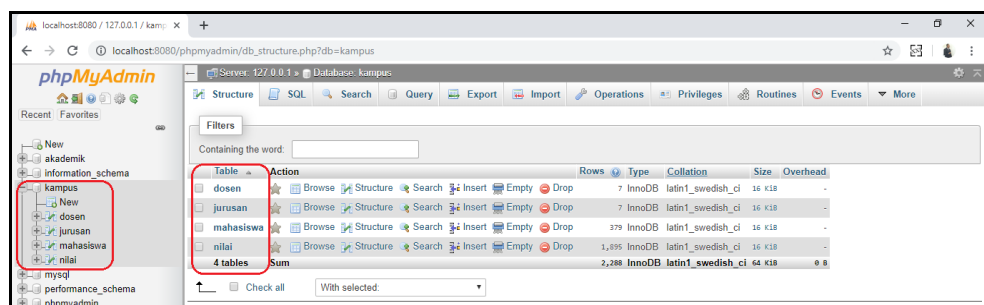
3. Pilih lokasi direktori penyimpanan file kampus.sql, kemudian klik Open



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 ( $\leq 4$  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 diketikkan 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	
	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

```
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	
	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

```
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.

## 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).

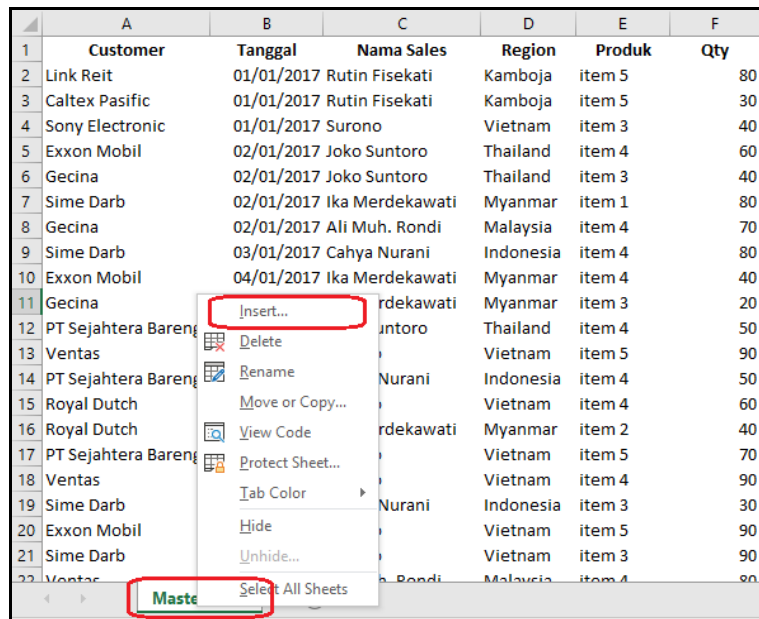
Dalam bab ini akan dibuat visualisasi data dalam bentuk pivot table, pivot chart, dan dalam bentuk dashboard. Data yang akan digunakan dalam bab ini adalah data sales.xlsx. Data sales.xlsx didapatkan dari export Views bernama view\_report database sales pada sub bab sebelumnya. Berikut adalah langkah-langkah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

### 3.1 Pivot Table dan Pivot Chart Trend Penjualan

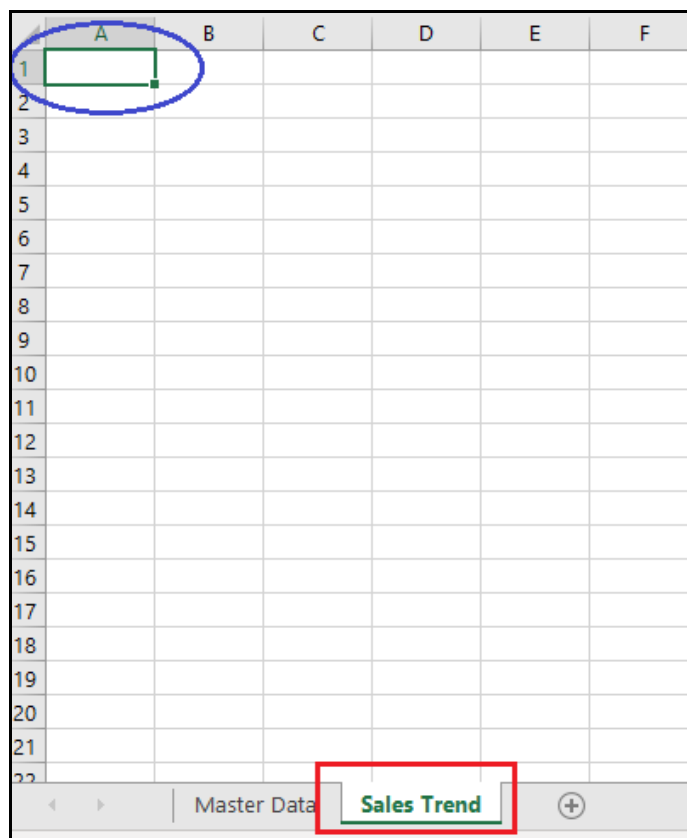
1. Siapkan data sales.xlsx

	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

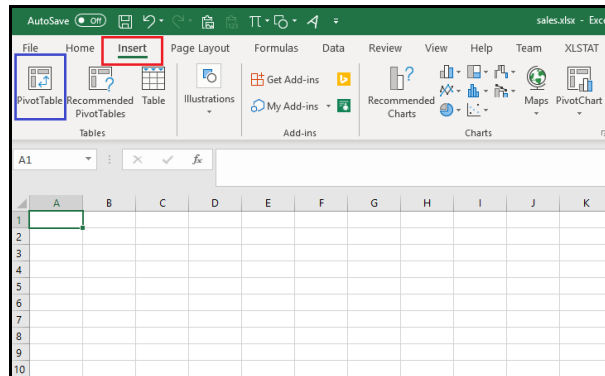
2. Tambahkan sheet baru, dengan cara klik kanan sheet Master Data – pilih Insert – Worksheet – lalu klik OK



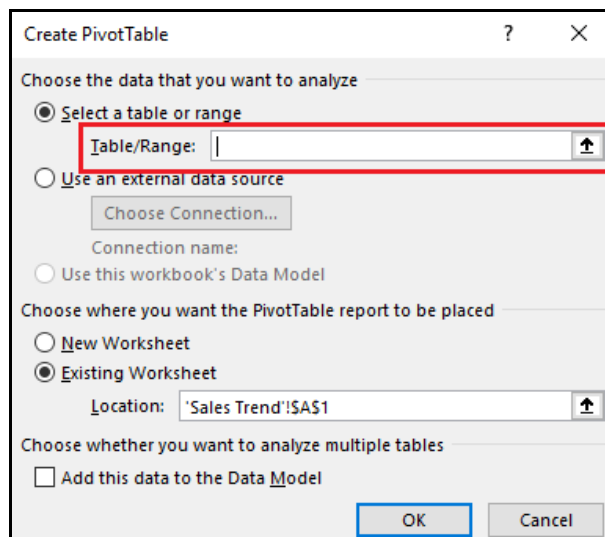
3. Ubah nama sheet baru menjadi Sales Trend, lalu letakkan kursor pada sel A1.



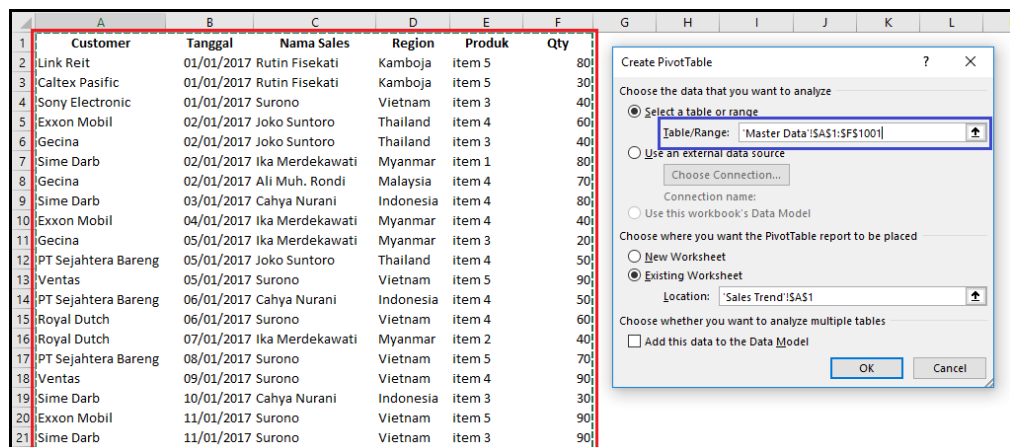
- Klik menu Insert – Pivot Table



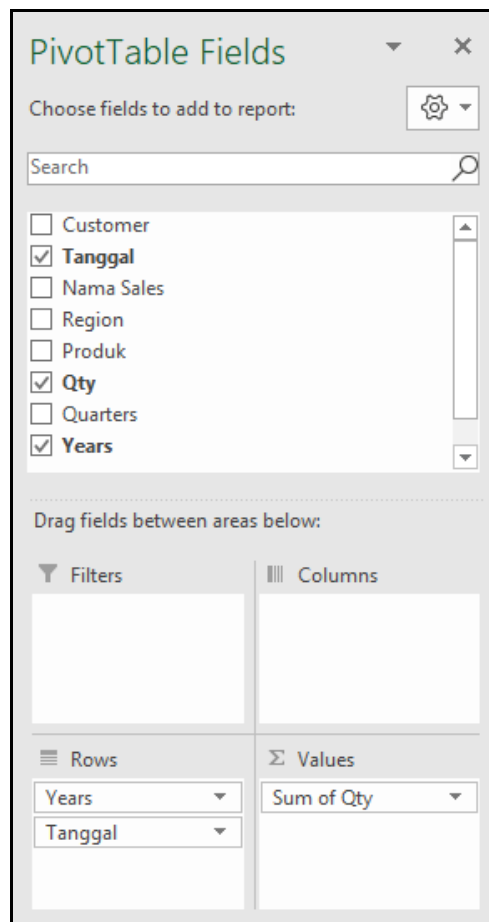
- Pada bagian Create PivotTable, letakkan pada bagian Select a table or range – Table/Range



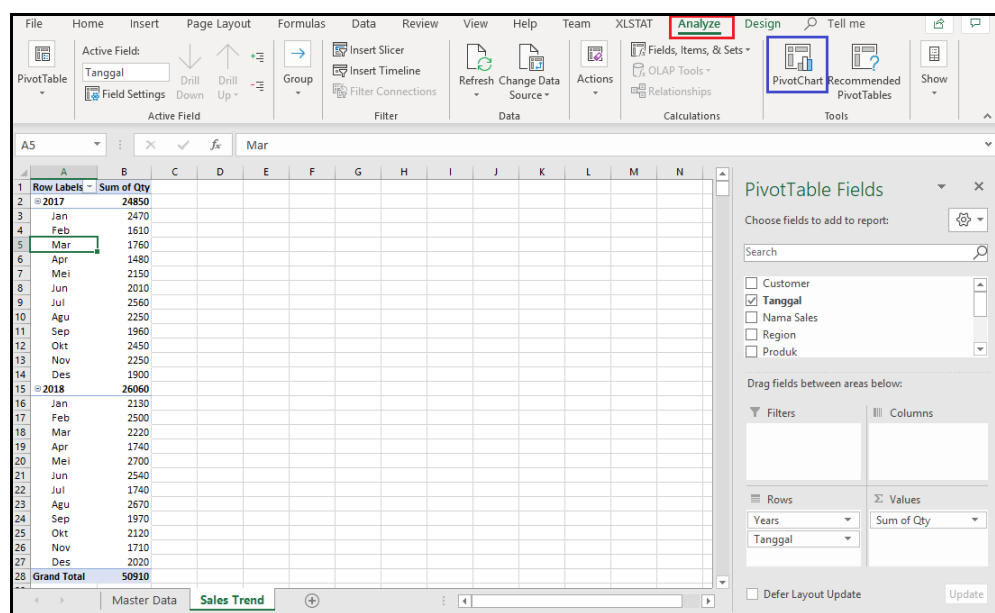
- Klik sheet Master Data, kemudian blok data dari sel A1:F1001, lalu klik OK



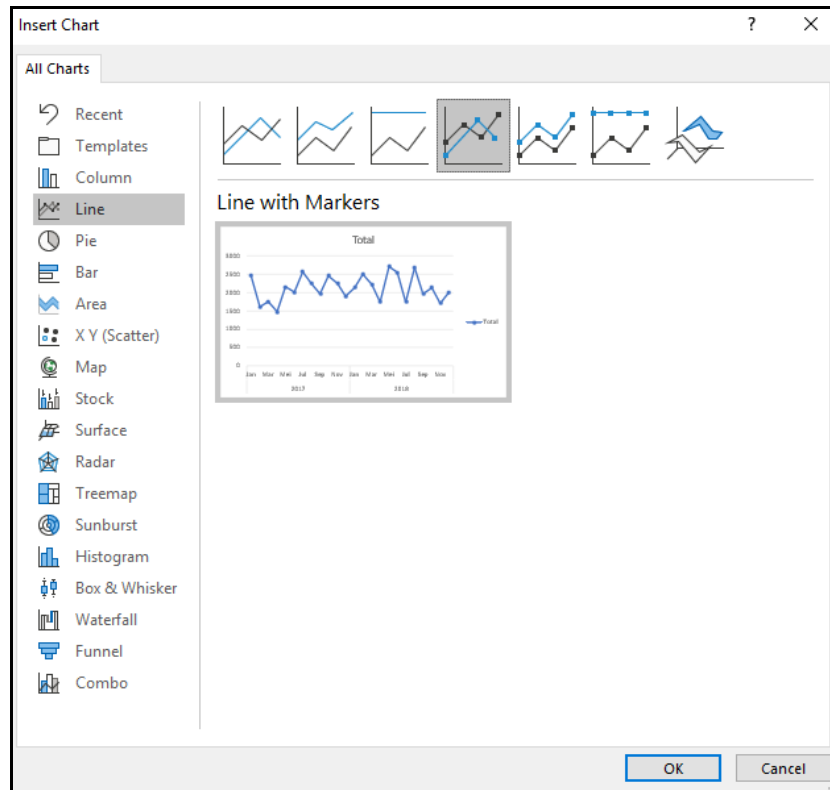
7. Pada bagian PivotTables Fields, pilih Tanggal dan Years lalu drag ke bagian Rows, dan pilih Qty lalu drag ke bagian Values



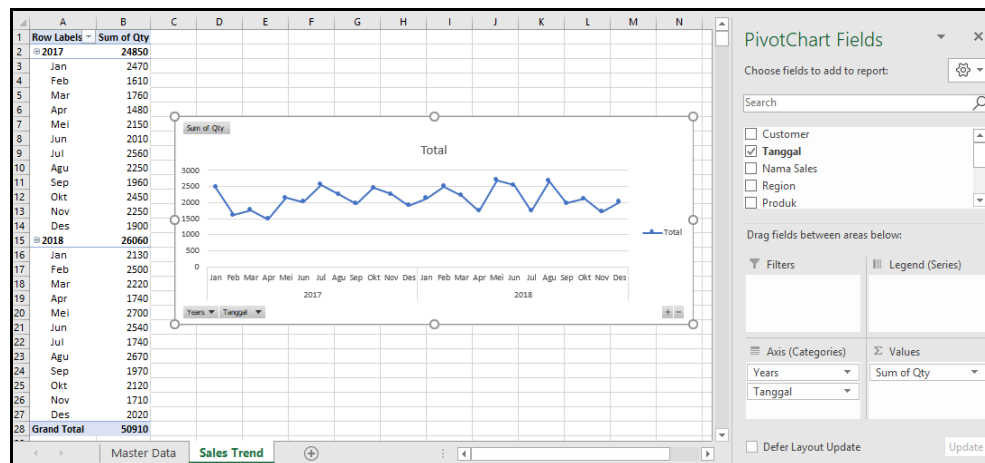
8. Letakkan kursor di sel A3, lalu pilih menu Analyze, klik PivotChart



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



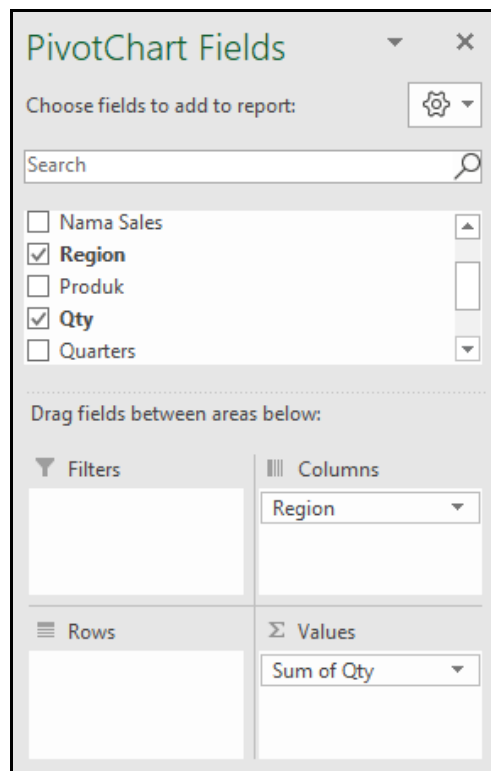
10. Maka akan terlihat hasil pivot chart (dalam bentuk grafik line) seperti gambar di bawah ini



### 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	

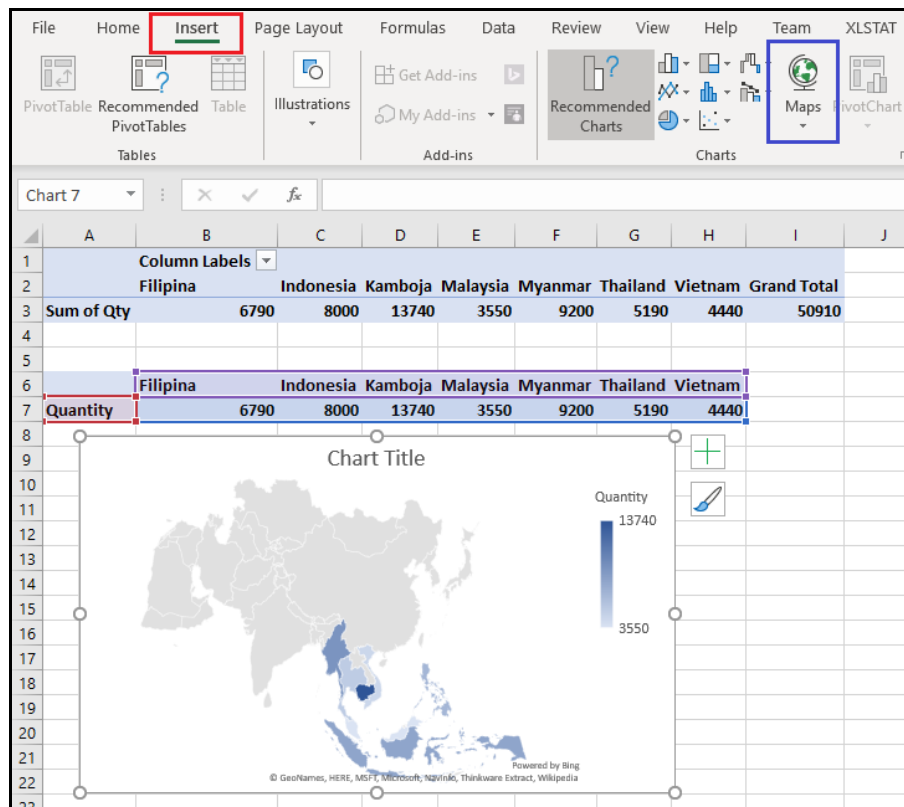
5. Klik kursos pada sel B7, lalu ketikkan formula =B3, lalu tekan tombol Enter

	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

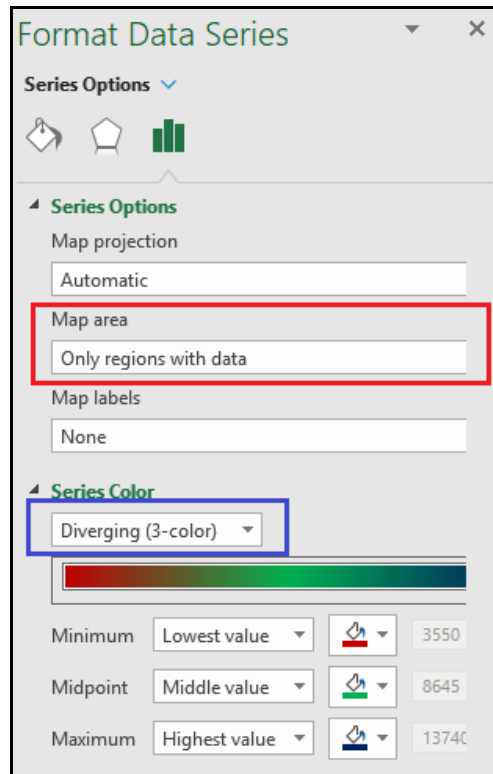
1	Column Labels								
2	Filipina	Indonesia	Kamboja	Malaysia	Myanmar	Thailand	Vietnam	Grand Total	
3	Sum of Qty	6790	8000	13740	3550	9200	5190	4440	
4									
5									
6	Filipina	Indonesia	Kamboja	Malaysia	Myanmar	Thailand	Vietnam		
7	Quantity	6790	8000	13740	3550	9200	5190	4440	

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

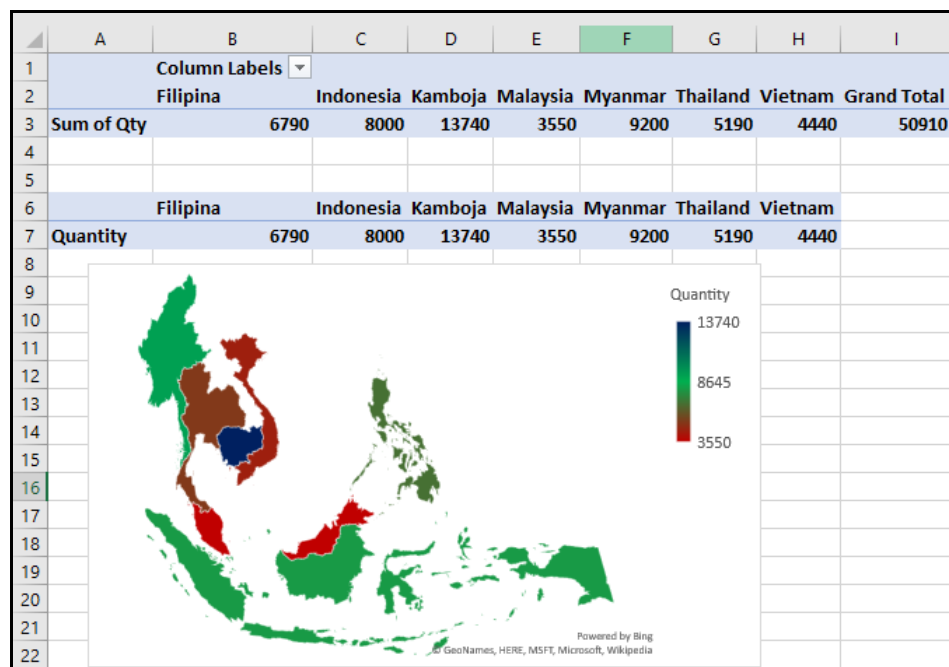


- Double klik, salah satu negara di dalam map, sehingga akan muncul box Format Data Series. Pada bagian Map area pilih Series Option - Only regions with data. Kemudian pilih Series Color – Diverging (3-color), lalu pada bagian Minimum, Midpoint, dan Maximum, pilih warna sesuai selera



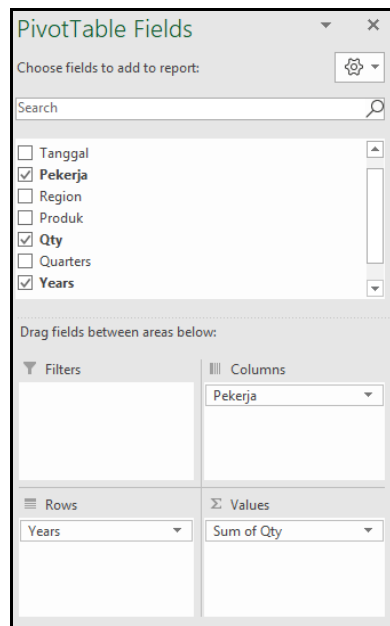


9. Hasil akhir pivot table dan map chart Sales per Region terlihat seperti gambar di bawah ini

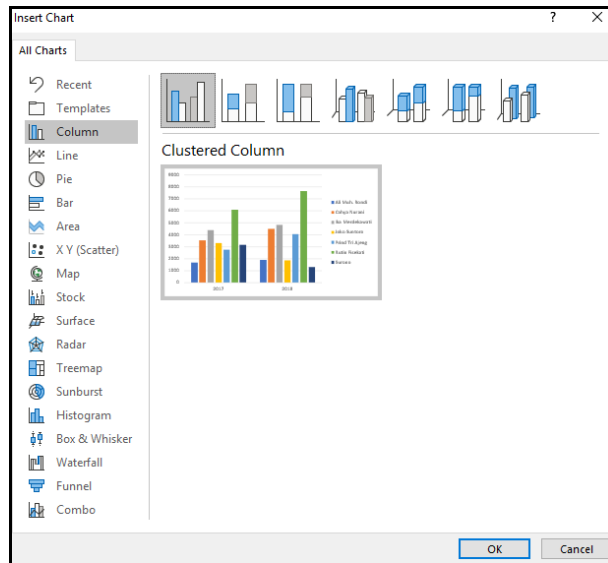


### 3.3 Pivot Table dan Pivot Chart Sales per Pekerja

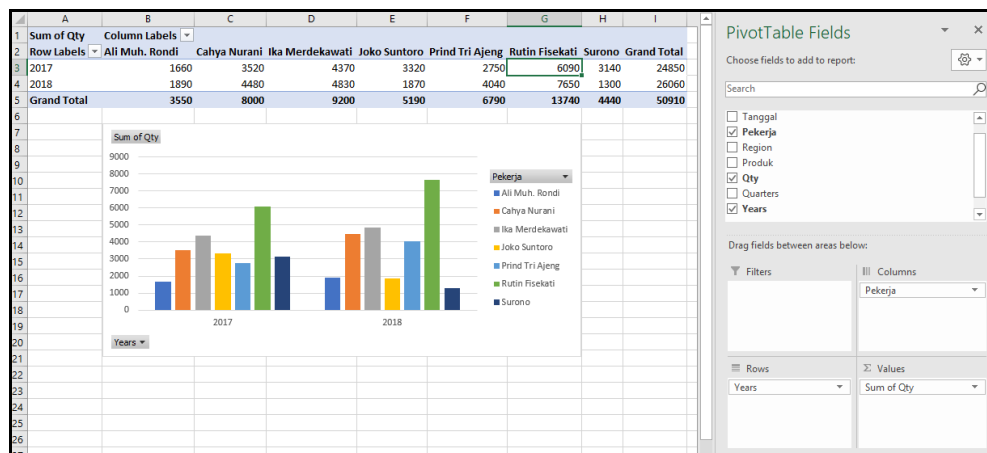
1. Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Sales per Region, lalu letakkan kursor pada sel A1
2. Tambahkan pivot table dengan mengambil semua data dari sheet Master Data
3. Pada bagian PivotTables Fields, pilih Pekerja lalu drag ke bagian Columns, pilih Years lalu drag ke Rows dan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

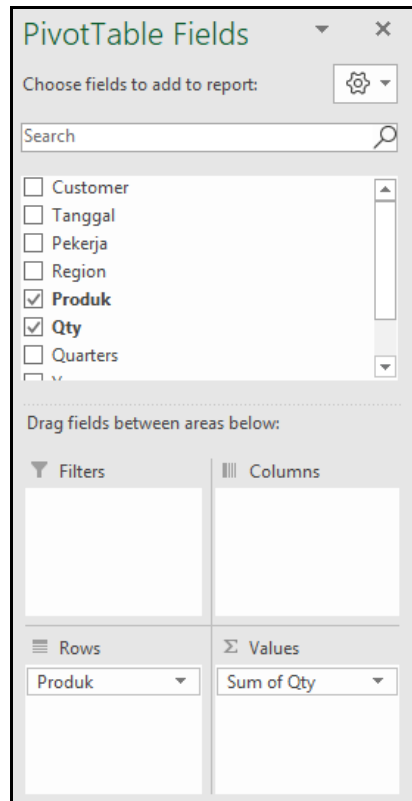


5. Hasil akhir pivot table dan pivot chart Sales per Pekerja terlihat seperti gambar di bawah ini

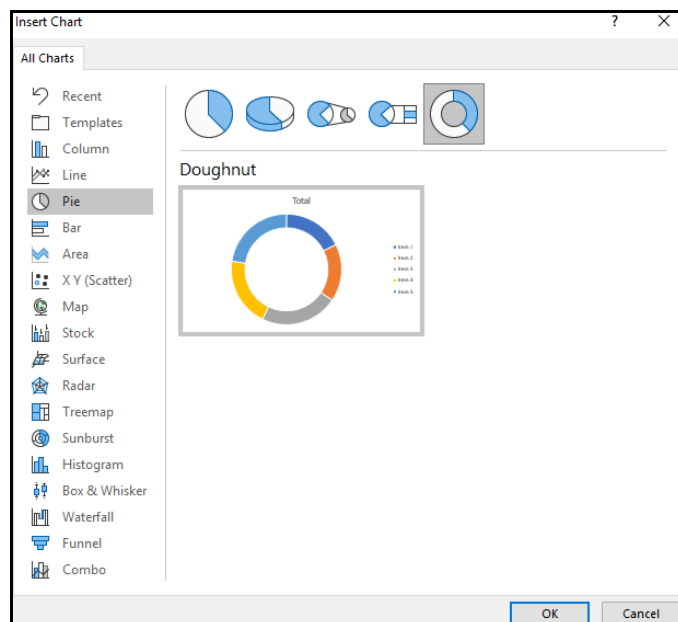


### 3.4 Pivot Table dan Pivot Chart Sales per Produk

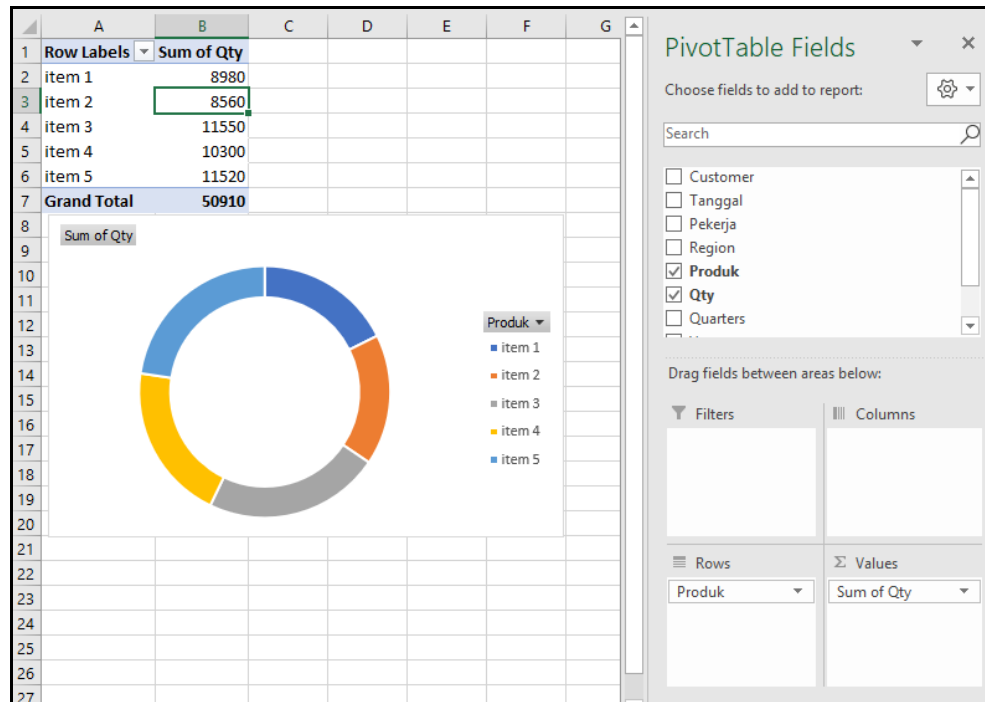
1. Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Sales per Produk, lalu letakkan kursor pada sel A1
2. Tambahkan pivot table dengan mengambil semua data dari sheet Master Data
3. Pada bagian PivotTables Fields, pilih Produk lalu drag ke Rows dan pilih Qty lalu drag ke bagian Values



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

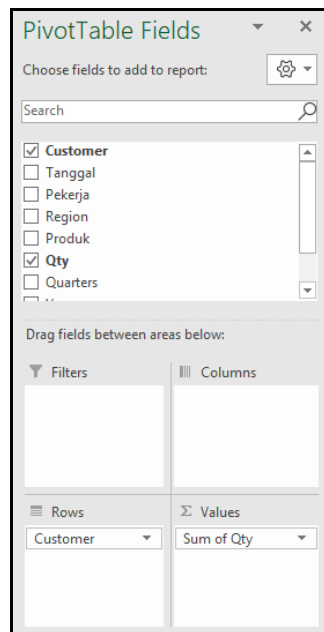


5. Hasil akhir pivot table dan pivot chart Sales per Produk terlihat seperti gambar di bawah ini

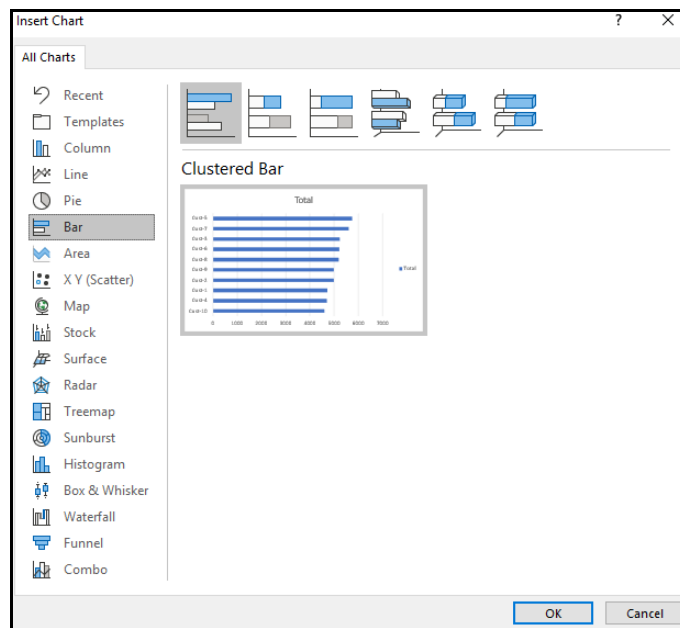


### 3.5 Pivot Table dan Pivot Chart Sales per Customer

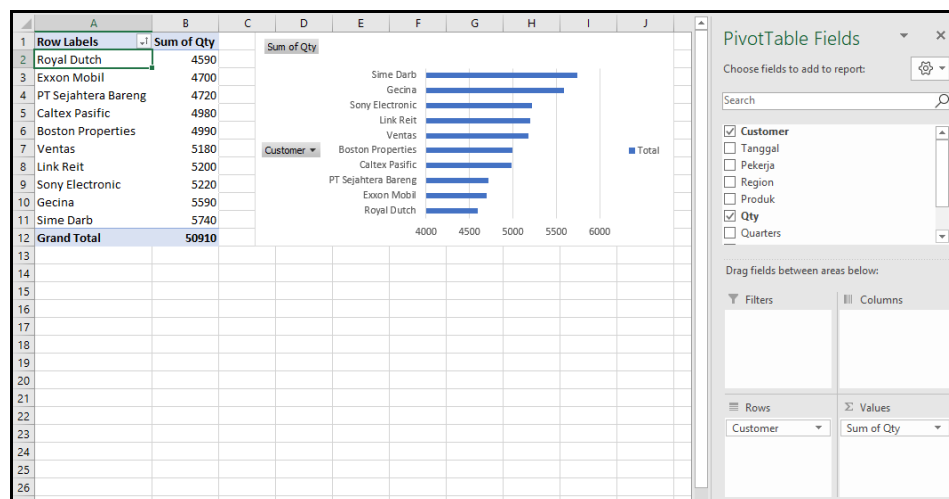
1. Tambahkan sheet baru pada file sales.xlsx, ubah nama dengan Sales per Customer, lalu letakkan kursor pada sel A1
2. Tambahkan pivot table dengan mengambil semua data dari sheet Master Data
3. Pada bagian PivotTables Fields, pilih Customer lalu drag ke Rows dan pilih Qty lalu drag ke bagian Values



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



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



**BAB 4**  
**PENGENALAN RAPIDMINER**

**BAB 5**  
**DATA WAREHOUSE FOR BUSINESS INTELLIGENCE I**  
**(Studi Kasus: Prediksi Kelulusan Mahasiswa)**

*[ON PROGRESS]*



**BAB 6**  
**DATA WAREHOUSE FOR BUSINESS INTELLIGENCE II**  
**(Studi Kasus: Klustering Data Pekerja)**

*[ON PROGRESS]*