

## MICROSOFT OFFICE ACCESS 2013

### 1. PENDAHULUAN

Microsoft Access 2013 atau disingkat Access 2013 merupakan salah satu perangkat lunak yang tergolong Relational Database Management System (RDBMS) yang banyak digunakan saat ini. Perangkat lunak ini sudah termasuk dalam aplikasi paket Microsoft Office 2013.

Access 2013 menyediakan banyak fasilitas yang berkaitan dengan pengelolaan database. Dengan fasilitas pada Access 2013 yang tersedia, kita dapat melakukan proses penyortiran, pengaturan data, pembuatan tabel, query, form, report, pages, macros, dan modules yang sangat berguna dalam mengelola database.

### 2. KONSEP DASAR PEMBUATAN DATABASE

Database merupakan kumpulan data yang saling berhubungan yang disimpan sedemikian rupa tanpa adanya *redundancy* (perulangan) yang tidak perlu. Database terbentuk dari sekelompok data-data yang memiliki jenis atau sifat yang sama. Contohnya: data mahasiswa, data dosen, data mata kuliah, data karyawan, dll. Untuk dapat membuat suatu database yang baik, diperlukan suatu rancangan yang tepat dan sesuai kebutuhan. Konsep dasar dalam pembuatan database antara lain:

#### a. Entitas dan Relationship

Hal yang sangat mendasar dan harus dipahami adalah pemodelan entitas dan relationship. Entitas adalah berbagai hal dalam dunia nyata yang informasinya disimpan dalam database. Sebagai contoh, kita dapat menyimpan informasi tentang dosen yang mengajar mahasiswa. Dalam hal ini, dosen dan mahasiswa merupakan entitas.

Relationship hubungan antara entitas. Sebagai contoh, dosen mengajar mahasiswa. Mengajar merupakan relationship antara entitas dosen dan entitas mahasiswa.

Relationship terdiri dari 3 derajat yang berbeda, yaitu:

1. *One-to-one* menghubungkan secara tepat dua entitas dengan satu kunci (*key*). Misalnya seorang mahasiswa memiliki satu dosen wali.
2. *One-to-many (many-to-one)* merupakan hubungan antar entitas dimana kunci (*key*) pada satu tabel muncul berkali-kali pada tabel lainnya. Misalnya banyak mahasiswa memiliki satu dosen wali.

3. *Many-to-many* merupakan hubungan antar entitas dimana kunci utama pada tabel pertama dapat muncul beberapa kali pada tabel kedua, dan sebaliknya. Misalnya seorang mahasiswa dapat mengambil banyak matakuliah, dan satu matakuliah bias dimabil oleh banyak mahasiswa.

#### **b. Relasi atau Tabel**

Relasi merupakan tabel yang mewakili entitas, dimana didalamnya terdapat kolom-kolom yang merupakan attribute dari entitas. Penamaan tabel juga disesuaikan dengan nama dari entitas agar mudah dipahami. Contoh tabel pegawai merupakan tabel yang mewakili entitas pegawai.

#### **c. Kolom atau Attribute**

Kolom atau attribute merupakan bagian dari tabel yang mewakili ciri dari suatu entitas. Contoh entitas mahasiswa. Maka attribute yang ada antara lain NIM, nama, dan kelas.

#### **d. Kunci (key)**

Kunci atau key atau primary key merupakan suatu nilai dalam sebuah tabel yang dapat digunakan untuk mengidentifikasi suatu baris dalam tabel. Contoh dalam tabel mahasiswa, didalamnya terdapat kolom NIM, nama, dan kelas. Untuk mengidentifikasi suatu baris dalam tabel, maka cukup menggunakan NIM.

### **3. RELASI ANTAR TABEL**

#### **A. KUNCI (KEY)**

Sebelum membahas mengenai relasi, faktor penting yang menyangkut relasi antar table adalah kunci (*key*). Kunci (*key*) merupakan satu atau gabungan dari beberapa atribut yang dapat membedakan semua baris data (*record*) dalam tabel secara unik.

Macam-macam kunci dibedakan menjadi:

##### **1) KUNCI PRIMER**

Kunci Primer merupakan sebuah field pada tabel yang merupakan identitas bagi field-field lainnya. Kunci Primer ini tidak boleh memiliki record kembar. Kunci Primer biasanya berada pada tabel induk.

## 2) KUNCI SEKUNDER

Kunci Sekunder merupakan kebalikan dari kunci primer. Kunci Sekunder biasanya berada pada tabel anak. Kedua kunci ini saling berhubungan, dan karena saling berhubungan maka tipe data yang dipakai harus sama.

## B. JENIS-JENIS RELASI

Secara garis besar, jenis-jenis relasi dibagi menjadi empat, yaitu:

### 1) ONE – TO ONE (SATU KE SATU)

Relasi ini adalah relasi yang menghubungkan sebuah record pada table induk ke tepat sebuah record pada table anak, akan tetapi relasi ini relative jarang dipergunakan karena jika record pada table anak hanya cocok dengan satu record pada table induk, mengapa record pada table anak tersebut tidak dijadikan satu pada table induk dengan menambahkan field-field table anak pada table induk?.

Contoh. Daftar pelanggan sebuah toko online dengan alamat pengiriman berbeda dengan alamat penagihan tetapi ada beberapa kelemahan yaitu:

- a. Pelanggan yang membedakan alamat pengiriman dengan alamat penagihan tidak banyak.
- b. Kalau alamat pengiriman dengan alamat penagihan dijadikan dalam satu tabel, maka field penagihan akan banyak yang kosong karena hanya terisi sedikit saja.
- c. Sebaiknya dibuat tabel sendirisendiri yaitu tabel alamat pengiriman dan table alamat penagihan untuk menyimpan informasi dan kemudian masingmasing tabel tersebut direlasikan.
- d. Satu record alamat pengiriman hanya akan terhubung dengan satu record nama pelanggan.

Contoh:

Tabel alamat pengiriman		Tabel alamat penagihan	
Nama	Alamat Pengiriman	Nama	Alamat Penagihan
A	Jl. Tlogomas	-	-
B	Jl. Dinoyo	B	Jl. Mergojoyo

### 2) ONE – TO – MANY (SATU KE BANYAK)

Relasi yang menghubungkan satu record pada satu tabel dengan beberapa record pada tabel lainnya.

Contoh. Database Perpustakaan dengan beberapa kriteria yaitu:

- a. Nama satu anggota perpustakaan hanya akan tercatat satu kali dalam table anggota.
- b. Akan tercatat berkali-kali dalam tabel pinjam.
- c. Satu orang bisa meminjam buku sampai berkali-kali.

Contoh:

Tabel Anggota

Nama :
NIM :

Tabel Pinjam

Nama	NIM	Tanggal Pinjam	Kode Buku
A	0012001	25 Februari 2013	

### 3) MANY – TO – ONE (BANYAK KE SATU)

- a. Relasi kebalikan dari One To Many (satu ke banyak) atau biasa disebut dengan look up table relationship.
- b. Relasi ini tidak saling terhubung ke kunci primer pada kedua tabel.

Contoh. Database Nama Propinsi dengan kriteria yaitu:

- a) Tabel informasi propinsi di Indonesia dihubungkan ke tabel alamat pelanggan.
- b) Record kota yang ada pada alamat pelanggan berelasi dengan record propinsi.
- c) Banyak kota yang bisa masuk ke dalam sebuah propinsi.

Contoh:

Tabel Propinsi

Kota	Provinsi
Mlg, Sby, Kediri, Blitar dll	Jawa Timur

Tabel alamat pelanggan

Nama	Alamat
Andi	Malang
Asri	Surabaya

#### 4) MANY – TO – MANY (BANYAK KE BANYAK)

- a. Relasi many to many ini hampir tidak mungkin ada.
- b. Jika ada biasanya terjadi karena kesalahan dalam perancangan tabel.

#### C. KEAKURATAN RELASI (REFERENTIAL INTEGRITY)

Dalam sebuah database, data yang sudah tersimpan biasanya akan mengalami perbaikan – perbaikan dalam jangka waktu tertentu, hal ini berpengaruh pada relasi sehingga setiap jangka waktu tertentu perlu adanya perbaikan supaya relasi yang ada bisa akurat, keakuratan relasi ini dibagi menjadi dua yaitu:

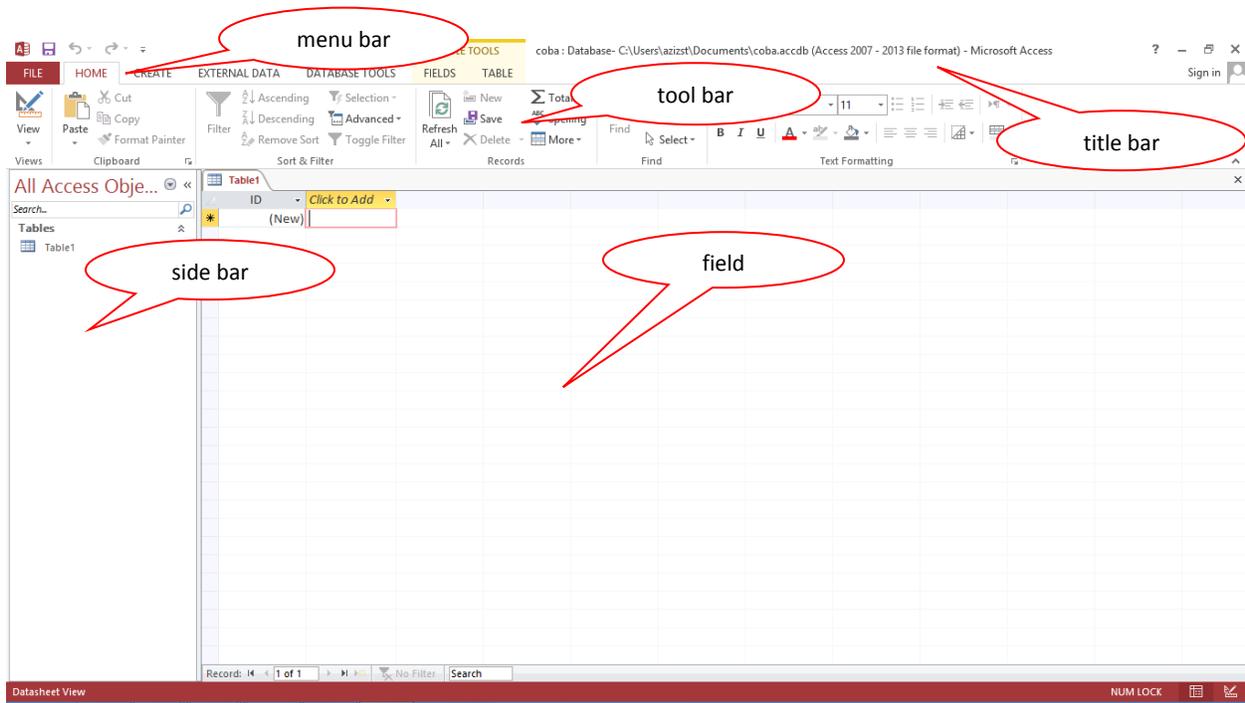
- a. Cascade Delete Related Record : yaitu Penghapusan semua record.

Contoh: Jika ada mahasiswa yang mengundurkan diri maka data mahasiswa tersebut dihapus semua recordnya bukan mengganti relasinya.

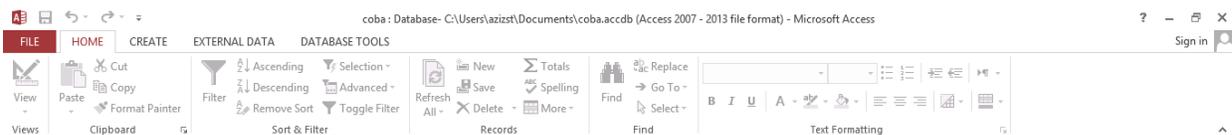
- b. Cascade Update Related Record : yaitu Perbaharuan record.

#### 4. MENU PADA MICROSOFT OFFICE ACCESS 2013

Berikut adalah tampilan awal dari halaman baru microsoft office access 2013. Didalamnya terdapat beberapa bagian yang mendasar. Ada title bar, menu bar, toolbar, sidebar dan field.

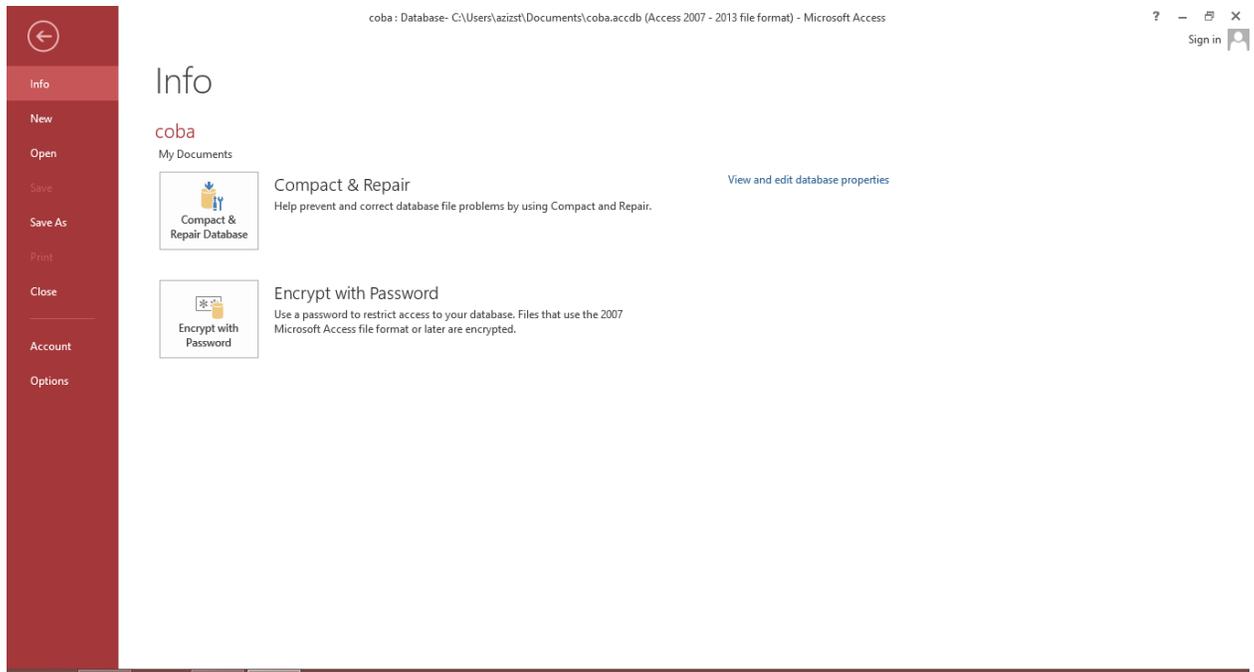


Pada title bar berisi tentang judul atau nama dari file. Pada menu bar terdapat beberapa menu yang dapat digunakan oleh user untuk memanajemen basis data.

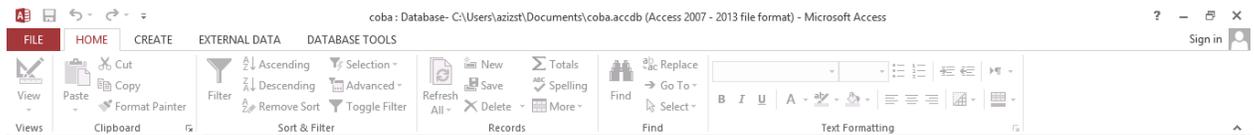


Yaitu diantaranya ialah:

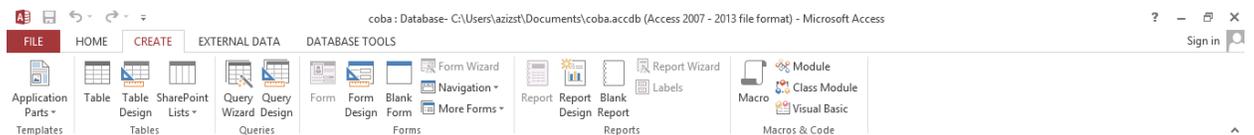
1. Menu “FILE” yang mana didalamnya terdapat beberapa pilihan menu atau perintah info untuk *repair* dan memproteksi *password*, *new* untuk membuat file yang baru, *open* untuk membuka file yang telah tersimpan dalam suatu direktori harddisk, *save* untuk melakukan penyimpanan, *save as* untuk melakukan penyimpanan dengan nama berbeda. *Print* untuk melakukan pengeprinan. *Close* untuk melakukan penutupan halaman yang sedang terbuka. *Account* untuk melihat keterangan akun yang sedang menggunakan dan juga keterangan produk. Dan *option* berisi tentang pengaturan yang opsional yang ada dalam *ms office access 2013*. Ketika menu file diklik maka akan muncul seperti dibawah ini.



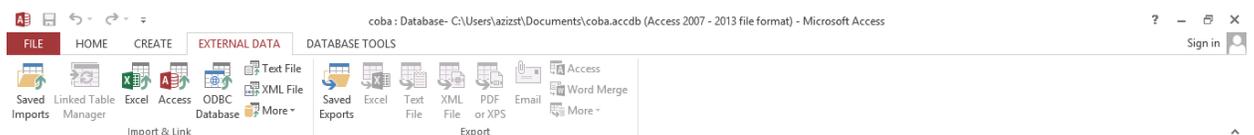
2. Menu “HOME” yang berisi tool-tool yang dapat digunakan oleh user untuk mengedit format dari isi / data file basis data. Seperti dibawah ini.



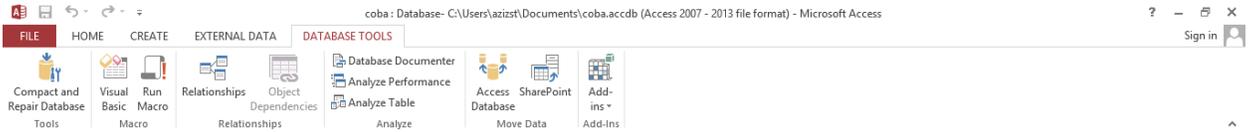
3. Menu “CREATE” yang berisi tool-tool yang dapat digunakan oleh user untuk melakukan penambahan tabel, query, form baru dll. Seperti dibawah ini.



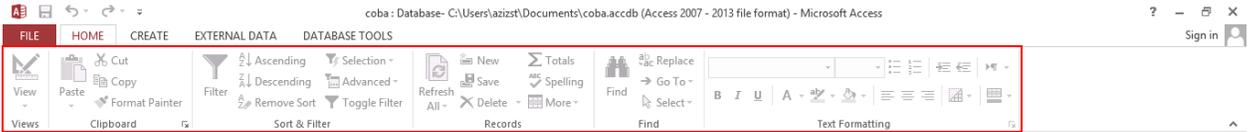
4. Menu “EXTERNAL DATA” yang berisi tool-tol yang dapat digunakan oleh user untuk melakukan penambahan data dari file lain.



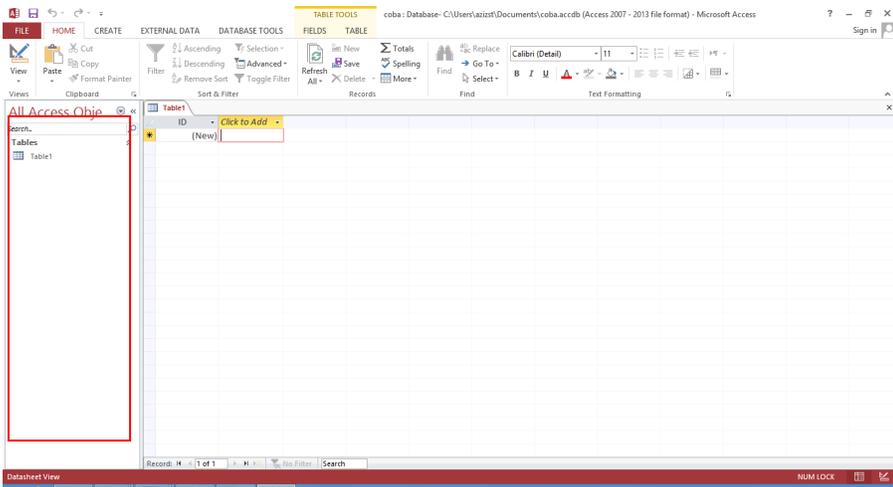
5. Menu “DATABASE TOOLS” yang berisi tool-tool yang dapat digunakan oleh user untuk melakukan pembuatan atau menghubungkan *database-database* atau tabel-tabel yang telah dibuat.



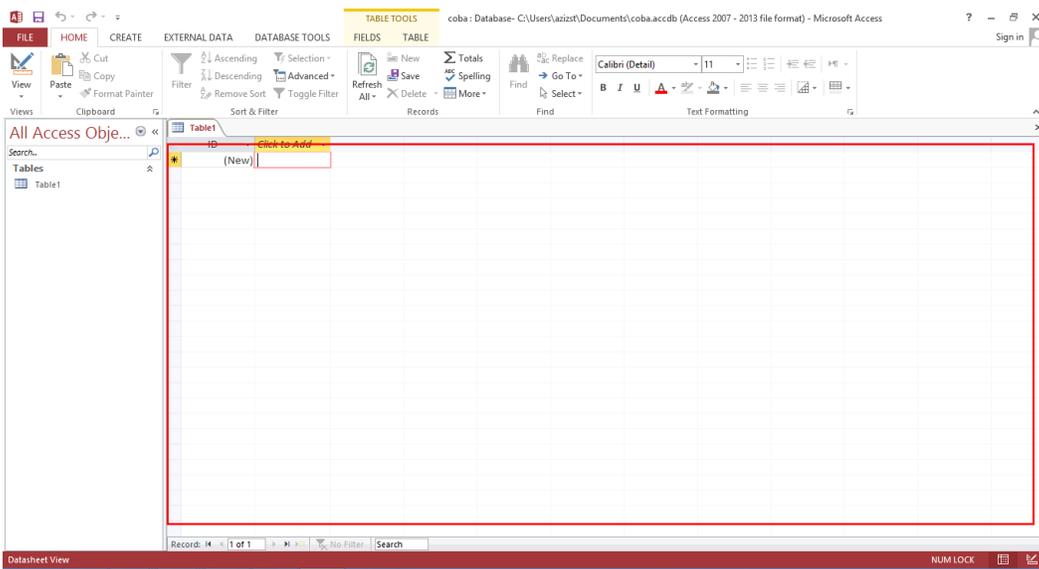
Bagian dari halaman ms access 2013 yang berikutnya ialah *toolbar* yang mana merupakan rincian tool yang ada dalam menubar.



Kemudian ada *sidebar* yang berisi objek-objek yang sedang aktif dalam halaman file seperti tabel, form, query dll.



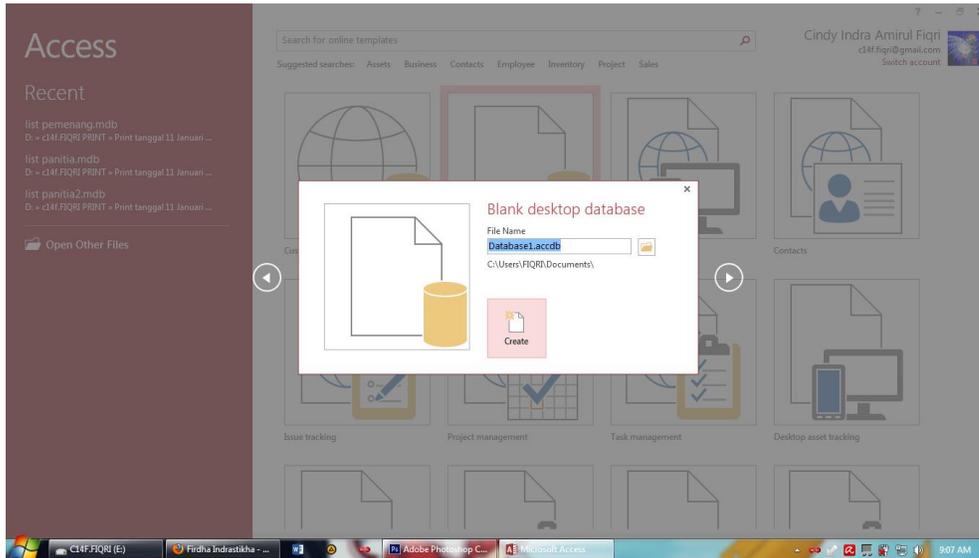
Dan yang terakhir ada *field* yang merupakan tempat mengedit data dalam ms access 2013.



## 5. MEMBUAT FILE BARU, MENYIMPAN FILE, DAN MEMBUKA FILE

1. Untuk membuat file baru pada Microsoft Office Access 2013 yaitu :

- Buka Microsoft Access 2013 maka akan muncul tampilan awal
- Pilih Blank Desktop Database dan akan muncul tampilan dibawah ini

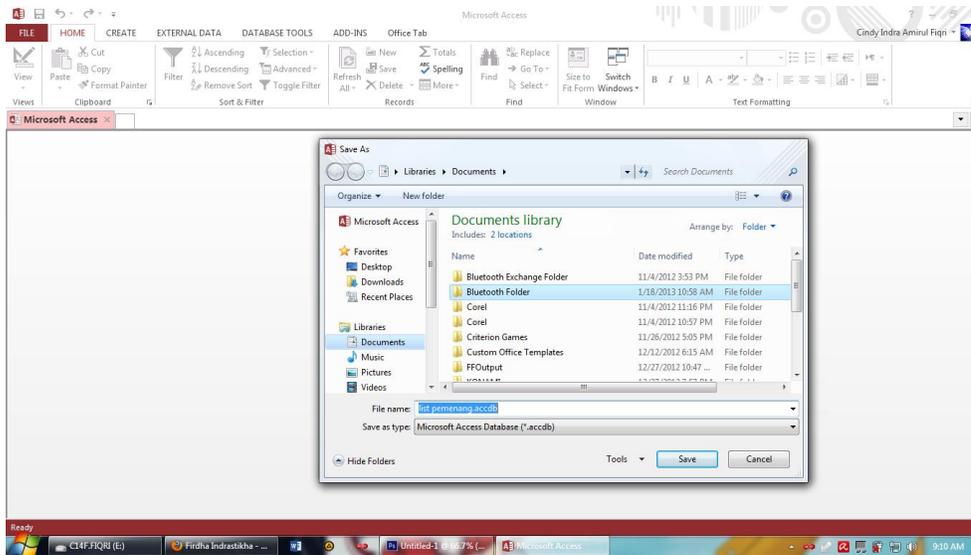


- Beri nama file pada file name lalu klik create.

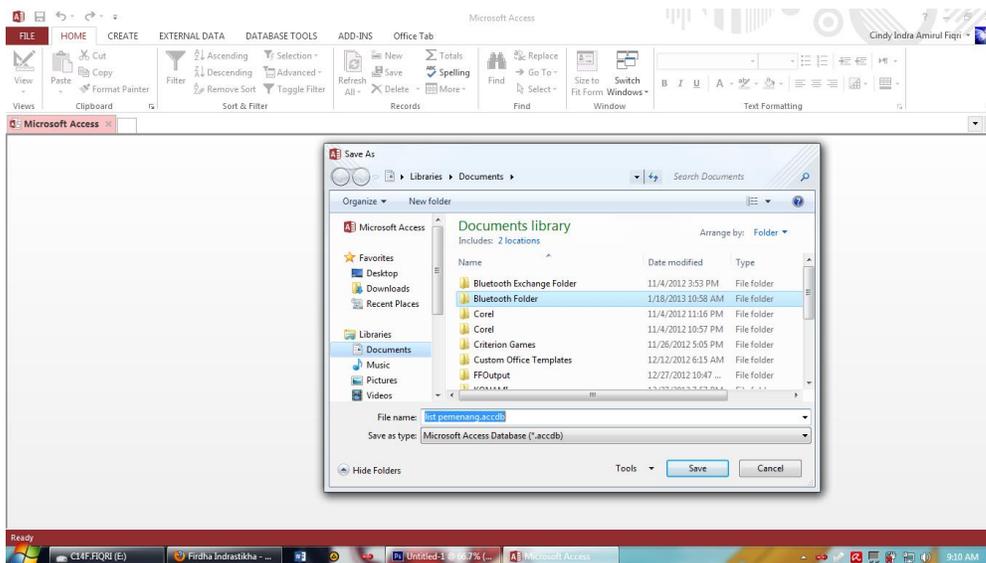
Atau bisa dengan Pilih Menu File – New dan untuk langkah berikutnya sama dengan langkah-langkah diatas.

2. Untuk menyimpan file pada Microsoft Access 2013 dapat dilakukan dengan :

- Pilih menu File – Save As
  - Beri nama file pada kolom File name, pilih drive tempat menyimpan file
  - Kemudian klik tombol Save

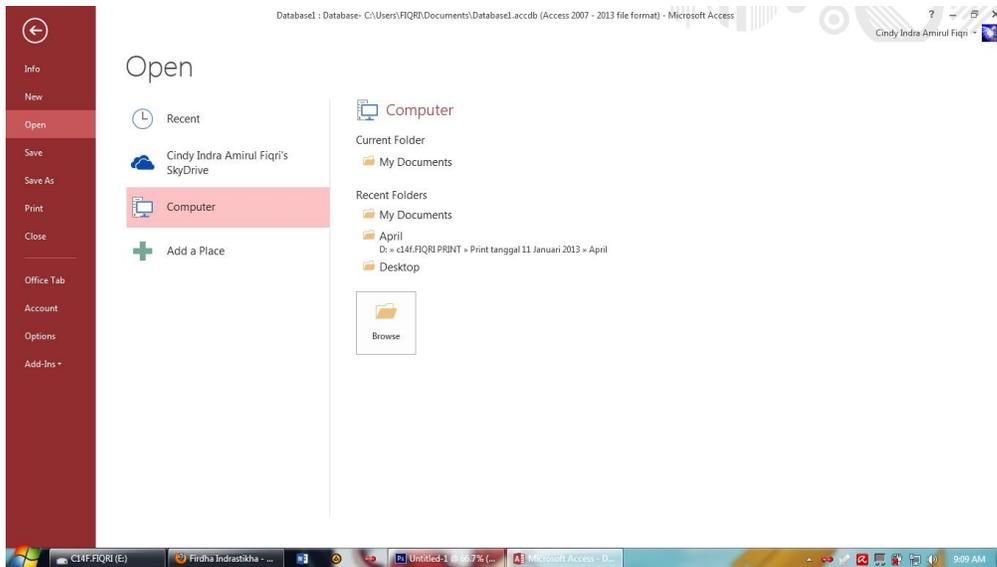


- b. - Tekan tombol CTRL+N pada keyboard
- Beri nama file pada kolom File name, pilih drive tempat menyimpan file
- Kemudian klik tombol Save

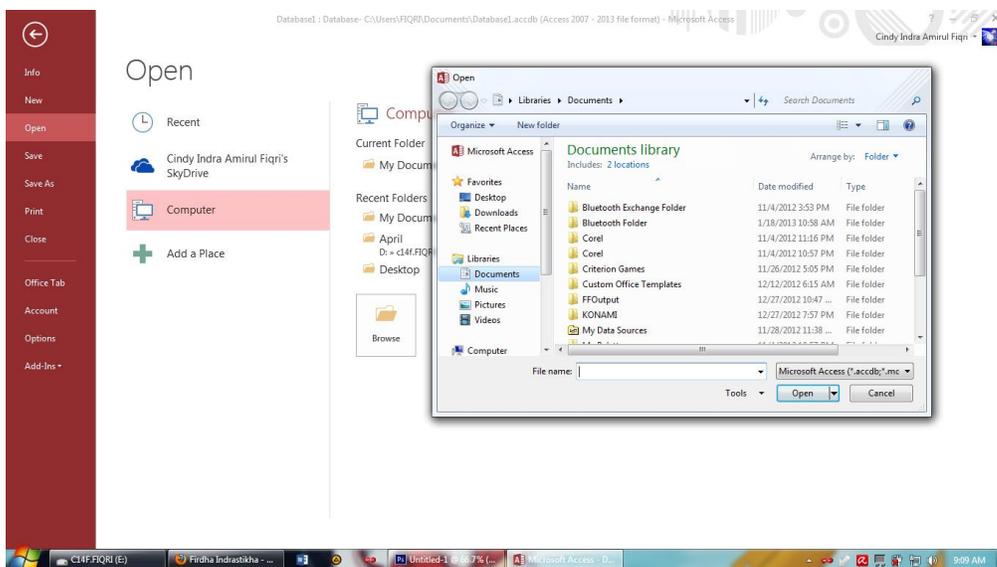


3. Untuk membuka file baru pada Microsoft Access yaitu :

- a. - Pilih menu File – Open



- Pilih tempat penyimpanan file yang akan dibuka
- Pilih Browse



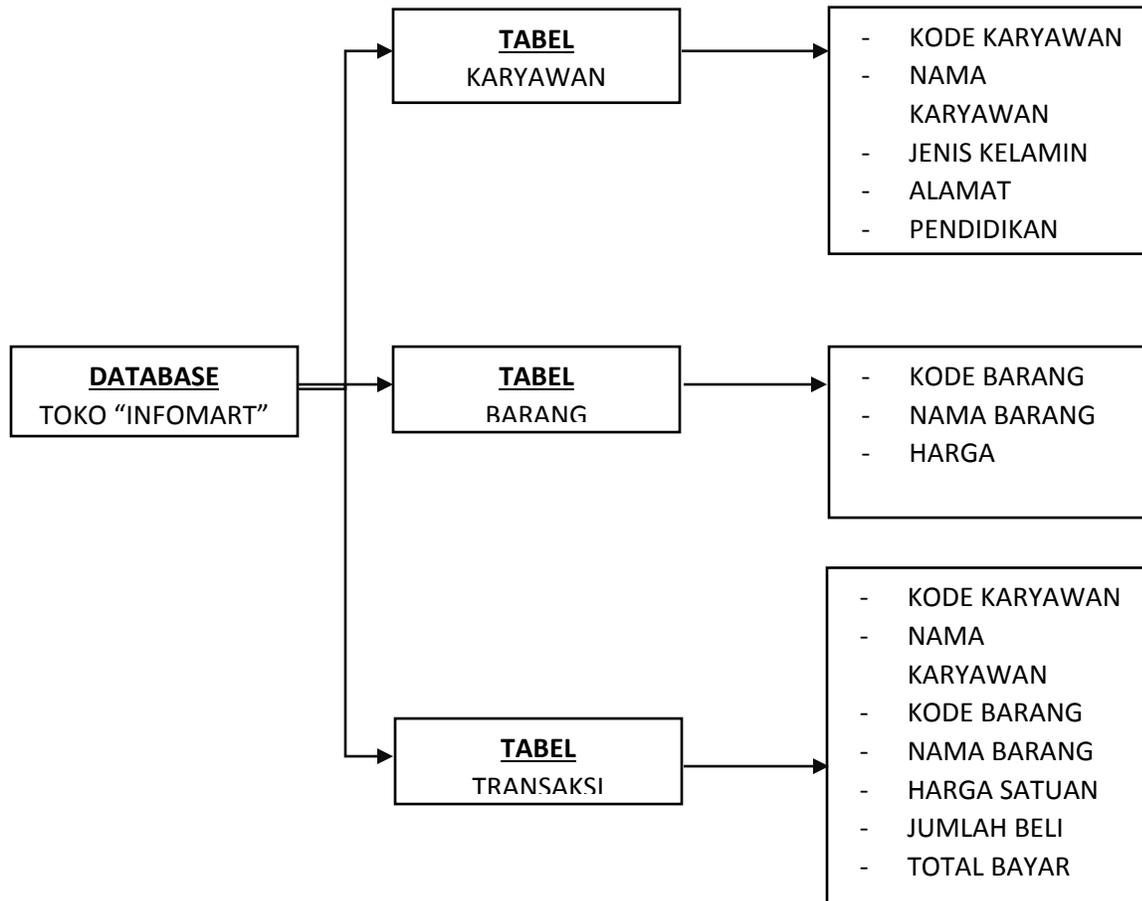
- Kemudian pilih file yang akan dibuka
- Dan klik Open

## 6. TABEL

Dalam pembuatan database, data yang pertama dibuat adalah tabel. Tabel merupakan kumpulan data yang tersusun menurut aturan tertentu dan merupakan komponen utama pada database. Table disusun dalam bentuk baris (*record*) dan kolom (*field*). Baris menunjukkan kumpulan data satu subjek (*record*)

*data*) sedangkan kolom menunjukkan kumpulan satu jenis atau kelompok data dari beberapa subjek (*field data*).

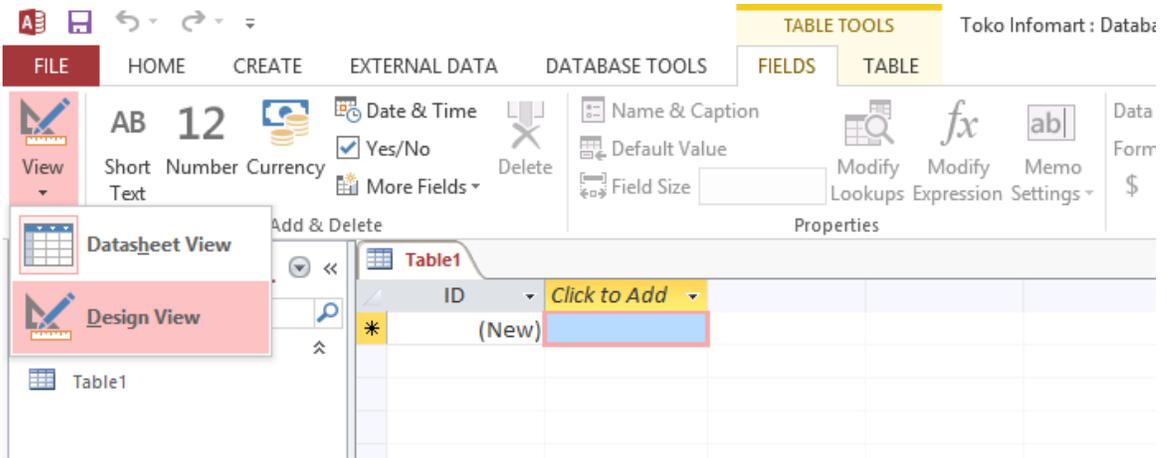
Misalnya seorang programmer diminta manager Toko “Infomart” untuk membuat database toko tersebut. Maka langkah awal pembuatannya adalah menentukan dahulu tabel-tabel yang dibutuhkan.



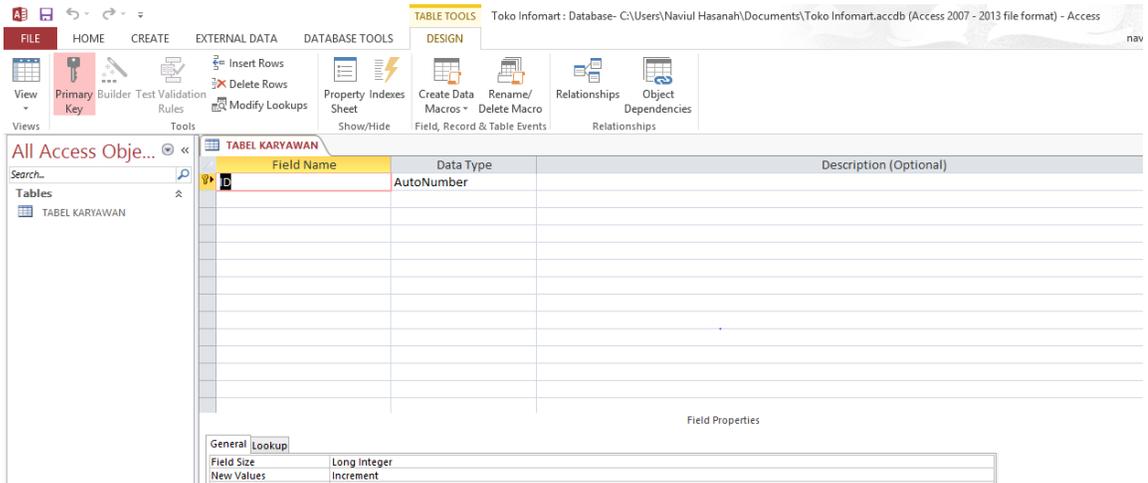
- **MEMBUAT TABEL**

Setelah konsep yang dibutuhkan sudah terbentuk, selanjutnya membuat tabel dengan langkah-langkah sebagai berikut :

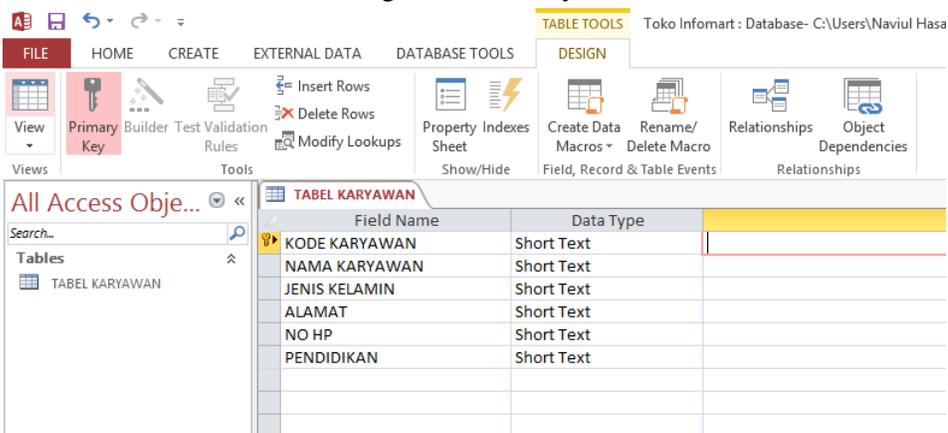
1. Jika membuka halaman baru pada Microsoft Access maka secara otomatis muncul tab *Table1*. Untuk merubah *Table1* menjadi tabel karyawan, langkah selanjutnya klik pada *menu home* dan pilih *View* kemudian *Design View*.



akan muncul kotak dialog *Save As*, kemudian ubah *Table1* menjadi Tabel Karyawan, klik OK maka akan muncul QBE (*Query By Example*) seperti pada tampilan berikut :



2. Isikan *Field Name* sesuai rancangan di tabel karyawan

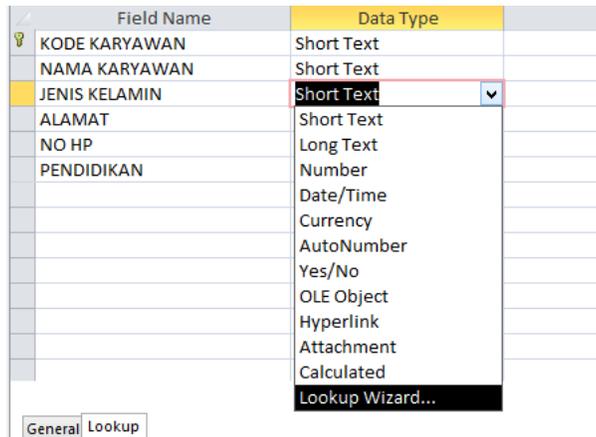


Data Type access 2013

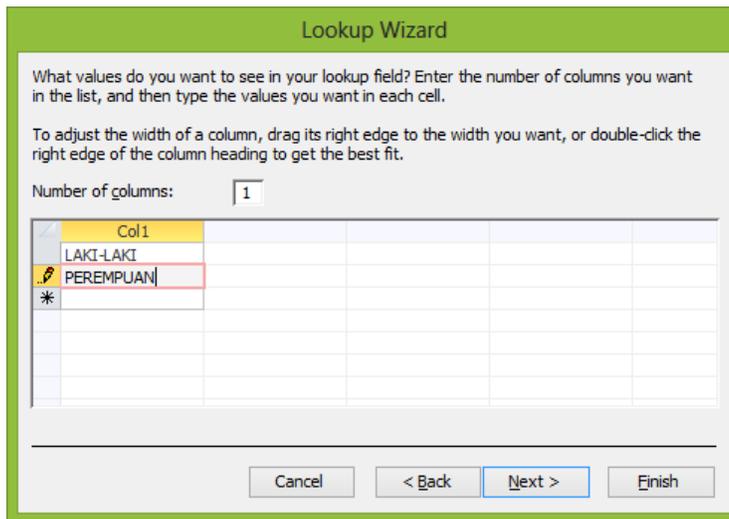
<b>Data Type</b>	<b>Keterangan</b>
Text	data yang bisa diisi dengan nilai kombinasi antara text dan number, dengan maximum karakter sebanyak 255 karakter.
Number	data angka yang dapat digunakan untuk perhitungan matematika dan dapat menampung sebanyak sampai 255
Date/Time	Adalah data tanggal atau waktu seperti hari, tanggal, jam, menit dll
Currency	data untuk menampung bilangan bilangan tanpa proses pembulatan pada saat perhitungan. Biasanya berupa mata uang dll
Autoumber	bilangan yang secara otomatis dihasilkan oleh <i>Access 2013</i> saat kita tambahkan <i>record</i> baru.
Yes/No	Untuk menampung dua macam keadaan, seperti ya atau tidak, benar atau salah
OLE Object	data yang diambil dari system <i>OLE</i> seperti <i>Microsoft Excel</i> spreadsheet, <i>Microsoft Word</i> document, graphics, sounds, atau data-data biner lainnya baik yang dilink ataupun dimasukkan secara permanen (embedded) kedalam table <i>Microsoft Access</i> .
Hyperlink	<i>type</i> data yang digunakan untuk menyimpan alamat internet atau file yang ditunjukkan melalui alamat URL.
Attachment	data <i>type</i> yang digunakan untuk menyimpan attachment file yang berformat apa saja (bebas, bisa file gambar, file suara, dll).
Calculated	fasilitas yang berguna untuk menghitung operasi matematika antara <i>field</i> yang satu dengan field yang lainnya. Misalnya, kita bisa menjumlahkan <i>field A</i> dengan <i>field B</i> , dll.
Lookup Wizards	fasilitas <i>combo box (list)</i> yang dibuat secara <i>wizard</i> sehingga kita dapat memilih ( <i>lookup</i> ) suatu data dari daftar pada table lainnya.

3. Untuk field Jenis Kelamin dan Pendidikan agar pada waktu mengisi data akan muncul pilihan secara otomatis, maka digunakan perintah *Lookup Wizard*. Adapun langkah-langkahnya sebagai berikut :

- Pada *Data type* pilih *Lookup Wizard*

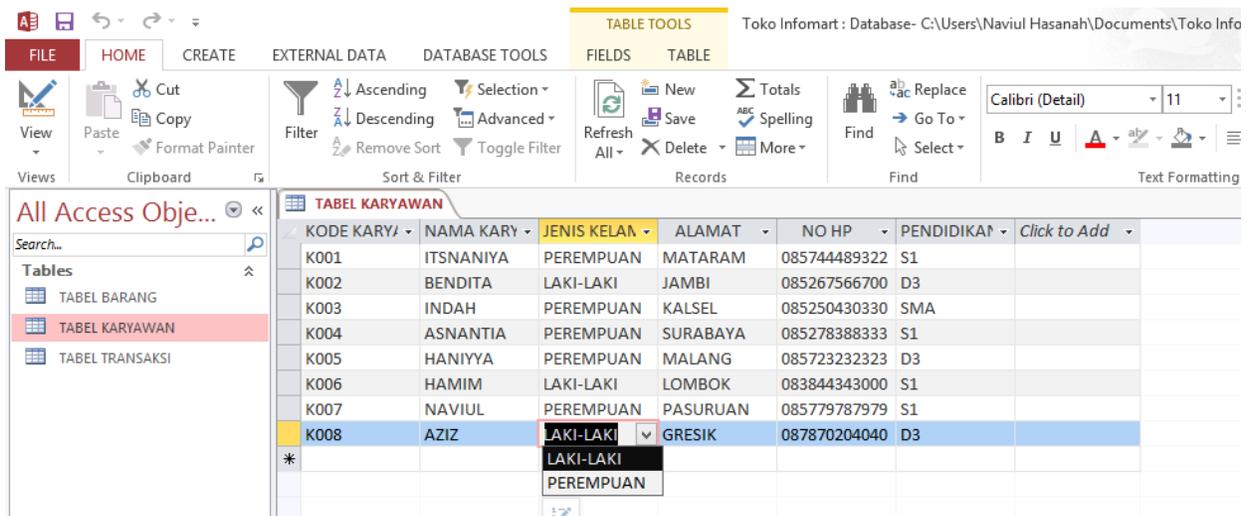


- Akan muncul kotak dialog *Lookup Wizard*, pilih *I will type in values that I want*. Kemudian klik *next*, tuliskan pilihan pada Col1 seperti pada tampilan berikut ini :



- Klik *Next* kemudian *Finish*
4. Membuat tabel baru (tabel barang dan transaksi) langkahnya klik pada menu *Create – Table*, kemudian lakukan langkah yang sama dengan langkah 1 sampai 3.
- **MENGENAL DATA**

Untuk mengisi data ke dalam tabel, terlebih dahulu bukalah tabel data yang akan dimasukkan dalam tampilan lembar data (*datasheet*). *Datasheet* akan menampilkan data secara keseluruhan dalam bentuk tabular yaitu susunan baris dan kolom. Setelah itu, kita dapat memasukkan data langsung pada lokasinya masing-masing.

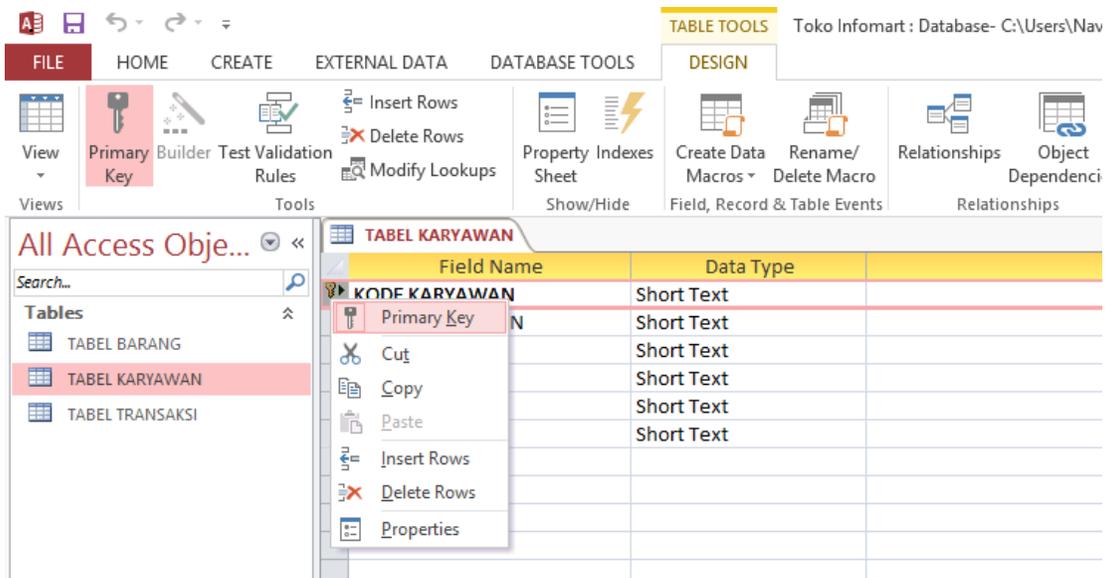


- **PRIMARY KEY**

*Primary Key* berguna sebagai wakil dari table yang menunjukkan nilai unik suatu *field*. Fungsi dari *Primary Key* adalah untuk menghindari adanya nilai yang sama (kembar) dalam suatu *field*.

Langkah-langkah membuat *primary key* :

1. *Primary key* dibuat pada *Design View* suatu tabel
2. Untuk memberikan *primary key* pada *field* yang diinginkan, caranya sorot *field* yang diinginkan kemudian klik tombol *Primary key* (bergambar kunci) pada menu *Design*. Atau dapat juga dilakukan dengan cara klik kanan pada *field* yang akan dijadikan *Primary key*, kemudian pilih *Primary key*

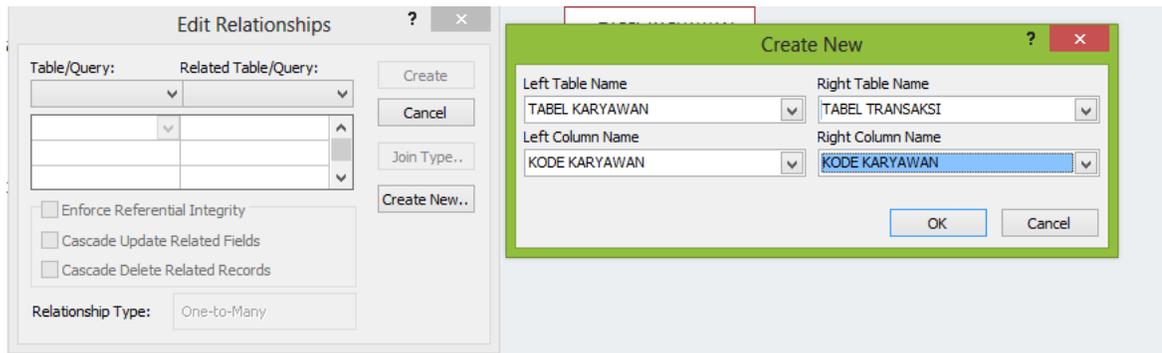


- **Relationships**

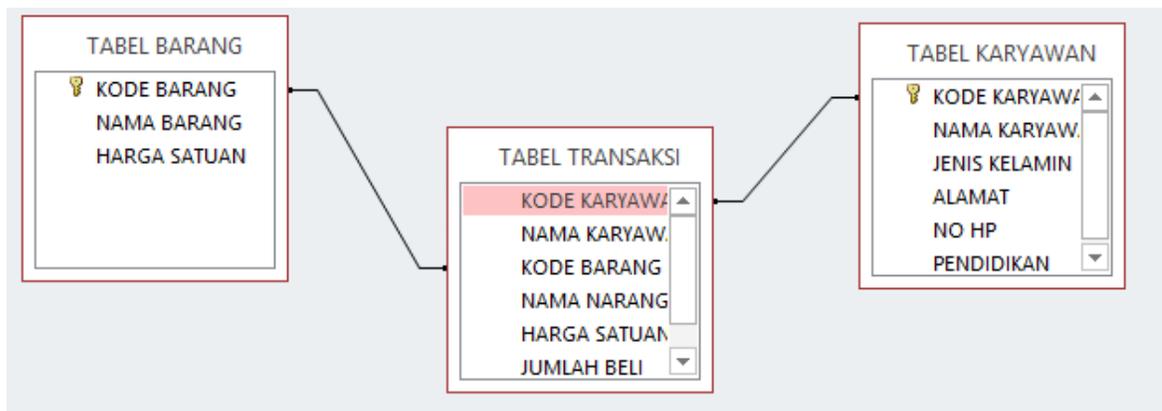
Database dengan kapasitas besar menggunakan lebih dari satu table. Dan seluruh table itu harus saling dihubungkan, untuk itu mutlak diperlukan kehadiran suatu *relationship* (relasi/hubungan).

Langkah-langkah untuk membuat Relationship adalah sebagai berikut :

1. Klik tombol *Relationship* pada menu *Database Tools*.
2. Kemudian akan muncul kotak dialog *Show Table*, tampilkan tabel yang akan direlasikan dengan cara sorot nama tabel lalu klik *add*.
3. Untuk membuat relasinya, klik *Edit Relationship* , maka akan muncul kotak dialog *Edit Relationship* kemudian klik tombol *Create New*.
4. Tentukan tabel dan field yang direlasikan pada *Right* dan *Left Table Name* serta *Right* dan *Left Column Name*, klik *OK* kemudian klik *Create*.



5. Lakukan langkah 3 dan 4 untuk membuat relasi pada tabel barang dan tabel transaksi, maka bentuk relasinya adalah sebagai berikut :

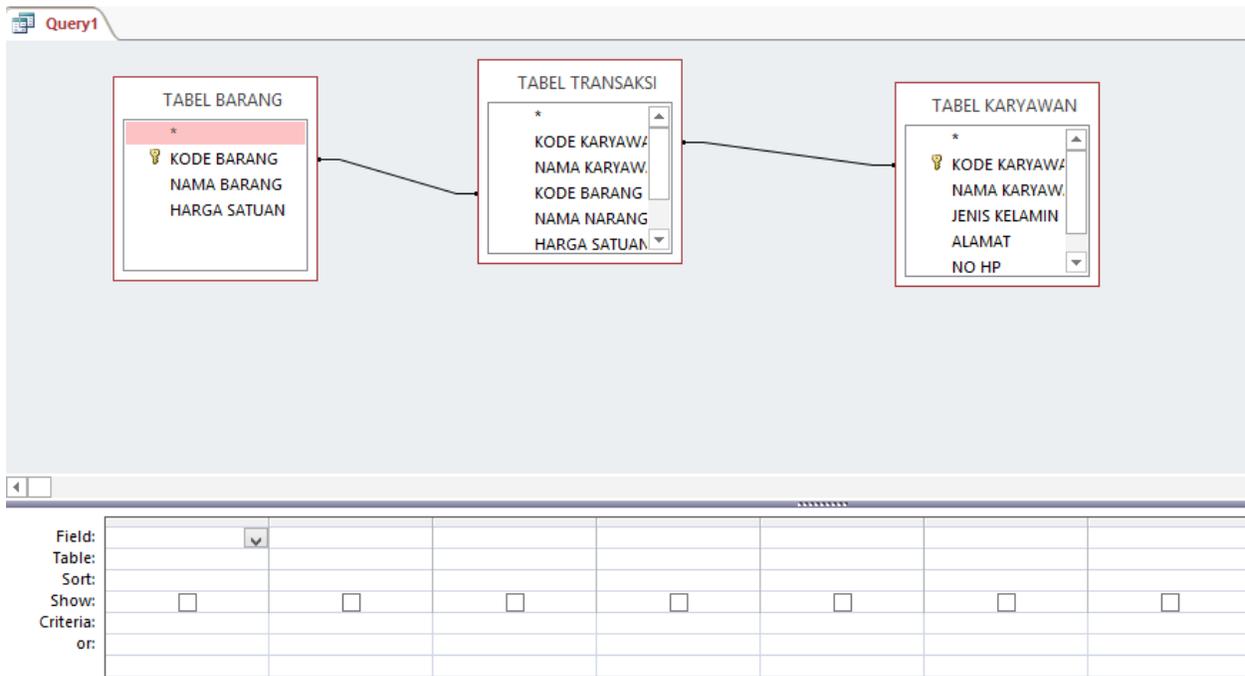


## 7. QUERY

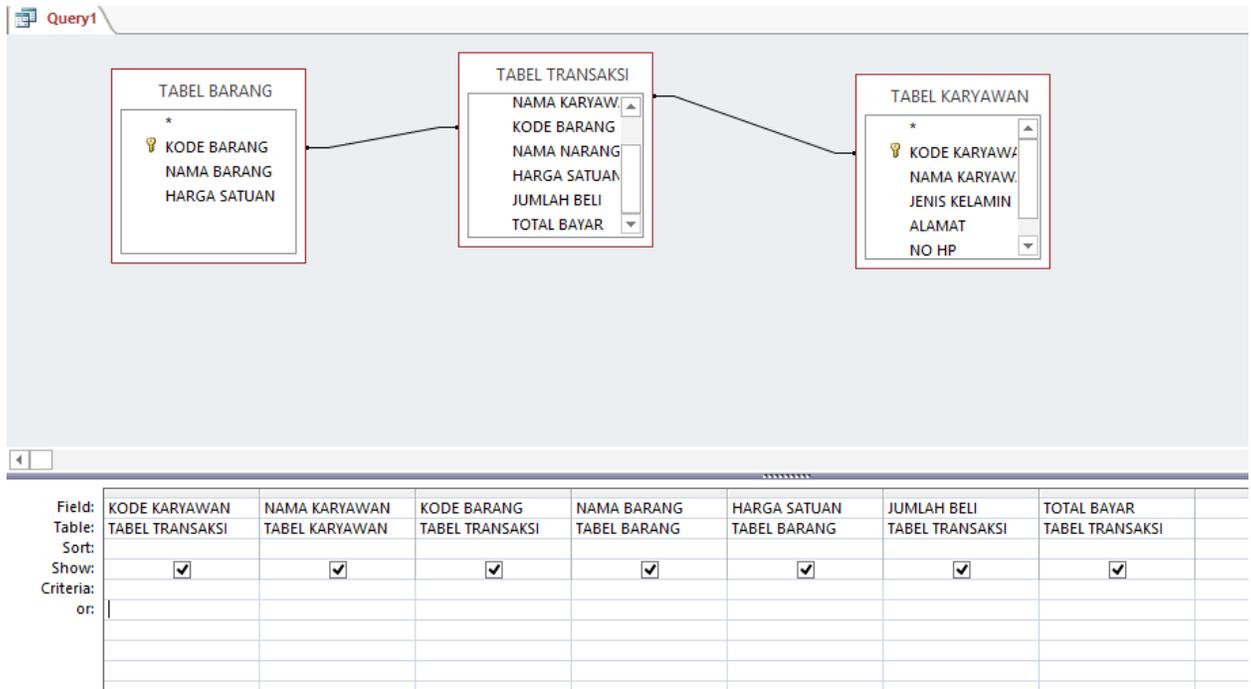
*Query* adalah sarana untuk mengatur data yang disimpan dalam table, sehingga hanya data tertentu yang akan ditampilkan dalam tabel.

Langkah-langkah membuat Query secara manual adalah sebagai berikut :

1. Pilih menu *Create*, kemudian klik *Query Design*, maka akan tampil dialog *Add table* untuk pemilihan table/query yang akan dijadikan acuan.
2. *Add table* satu persatu kemudian tekan tombol *Close* Sehingga muncul seperti gambar di bawah ini :



3. Setelah itu munculkan Field-field dari tabel yang diinginkan. Tentukan terlebih dahulu tabelnya kemudian Fieldnya dan jangan sampai terbalik.

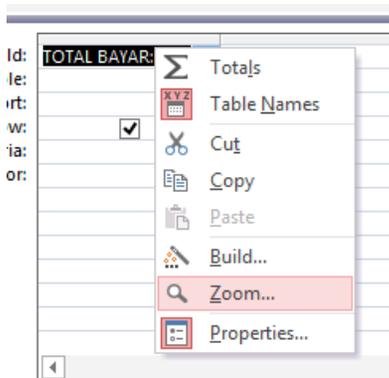


4. klik icon *Run* yang bergambar tanda seru  , maka akan tampil hasil *Query* yang sudah dibuat. Simpan dan beri nama “Query\_TRANSAKSI”

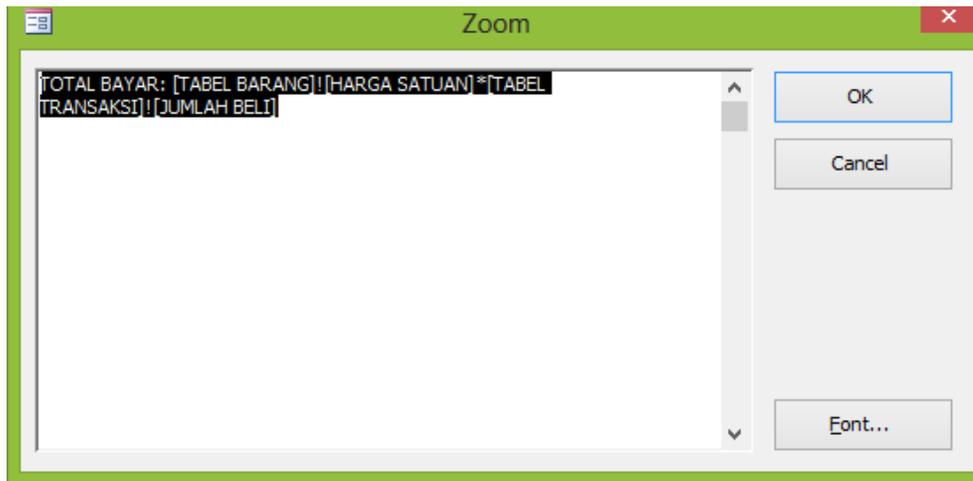
### Memberikan Formula pada Field

Seperti *cell* pada *Microsoft Excel*, field pada *Microsoft Access* juga bisa diberikan formula, baik Formula *Arithmetic*, *Comparison*, *Logical* maupun *string*. Contoh pada database di atas, akan diberikan formula pada *Field* Total Bayar yang ada di *Query\_TRANSAKSI*. Langkahnya adalah sebagai berikut :

1. Buka terlebih dahulu *Design View* *Query\_TRANSAKSI*. Kemudian klik pada *field* Total bayar, pilih *zoom*.



2. Kemudian akan muncul kotak dialog *Zoom*. Tuliskan formula yang diinginkan seperti pada contoh berikut ini :

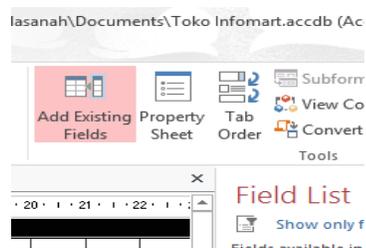


3. Klik *OK* kemudian cobalah hasilnya.

## 8. FORM

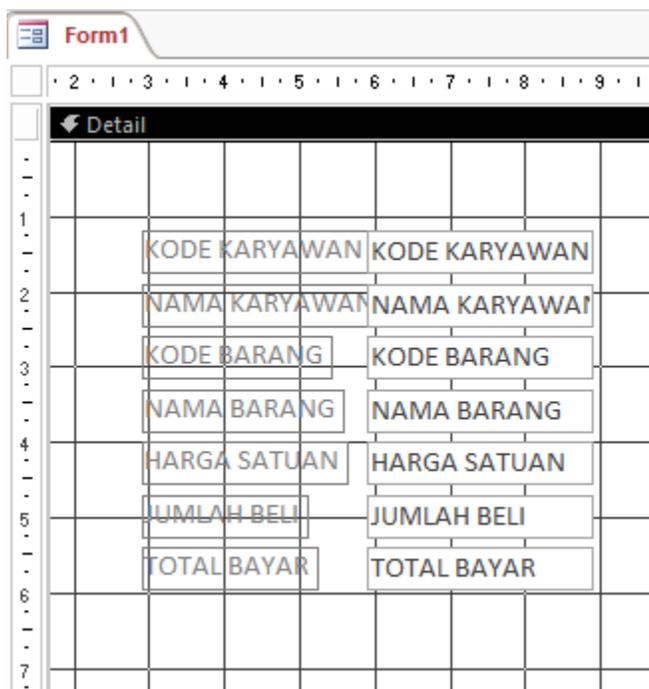
Untuk menghasilkan tampilan yang lebih komunikatif, maka dibutuhkan suatu *interface* yang dapat membantu pengguna atau operator dalam menjalankan suatu program aplikasi. Salah satu *interface* yang disediakan oleh *Microsoft Access* adalah *Form*. *Form* merupakan *window* untuk menampilkan *record* dalam bentuk lain, didalamnya kita bisa menambahkan unsur lain seperti tombol, gambar dan lainnya. Ada dua cara dalam pembuatan *form* diantaranya 1) *Form Design*, membuat form secara manual dan sesuai dengan keinginan user, 2) *Form Wizard*, membuat form secara otomatis. Berikut akan disajikan langkah - langkah membuat *form* menggunakan *Form Design*.

1. Pilih menu *Create* kemudian *Form Design*.
2. Pilih *Add Existing Fields* di menu *Design*. Maka akan tampil *field list* di sebelah kanan. Gunakan button  untuk menampilkan beberapa *field* di tiap tabel.



3. *Fields* yang ditampilkan harus sesuai dengan *Query* yang telah dibuat. Untuk menambah *field* pada *Form*, *double click* pada *fields* yang diinginkan. Ulangi langkah ini sampai *fields* terakhir..

KODE KARYAWAN diambil dari TABEL TRANSAKSI  
 NAMA KARYAWAN diambil dari TABEL KARYAWAN  
 KODE BARANG diambil dari TABEL TRANSAKSI  
 NAMA BARANG diambil dari TABEL BARANG  
 HARGA SATUAN, diambil dari TABEL BARANG  
 JUMLAH BELI diambil dari TABEL TRANSAKSI  
 TOTAL BAYAR diambil TABEL TRANSAKSI



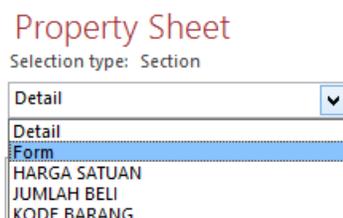
Untuk melihat hasil form yang telah dibuat, klik *View* pada menu *Home*. Untuk kembali ke pengaturan *form design*, pilih *Design View*.

Terdapat satu field yang belum terisi dengan benar yaitu field TOTAL BAYAR.

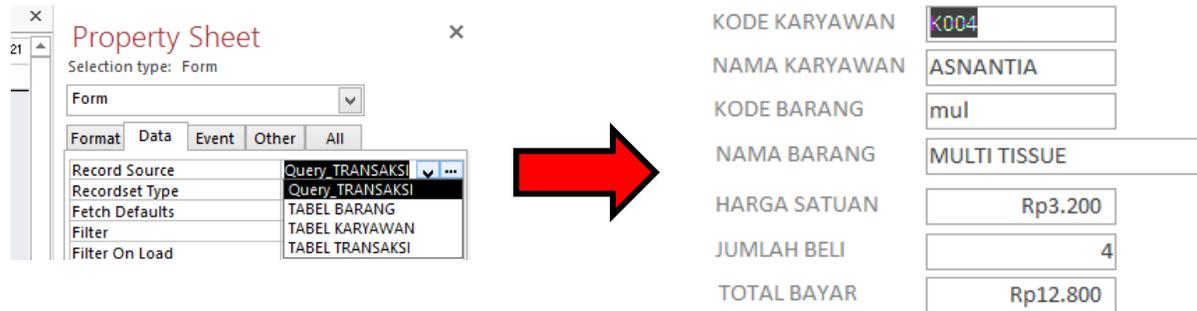
Mengapa ?

Karena field ini berasal dari tabel. Sedangkan formula TOTAL BAYAR telah disetting di Query\_TRANSAKSI. Jadi, *source* / sumber yang diambil harus dari Query\_TRANSAKSI.

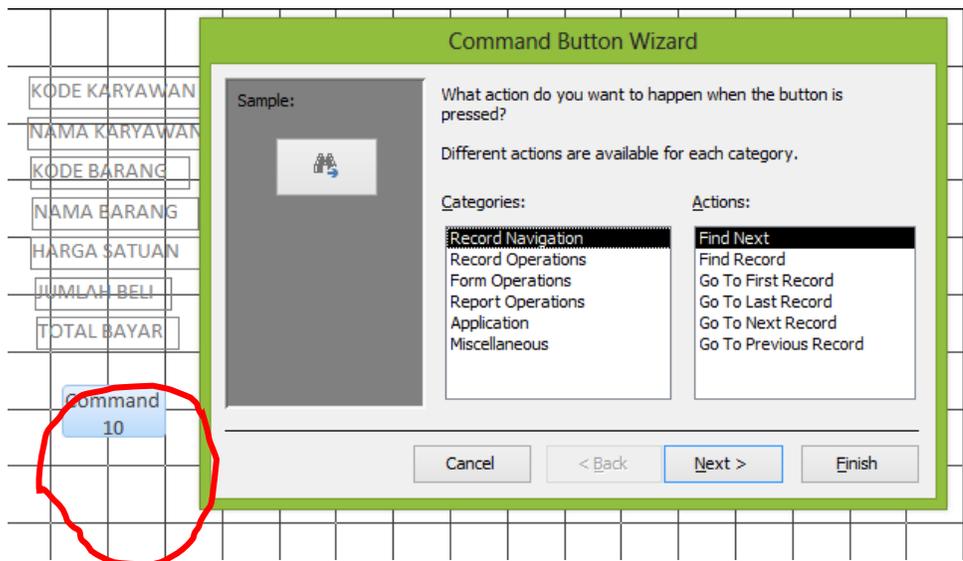
4. Pilih *property sheet* pada menu *Design*. Klik pada area form yang kosong, kemudian di bagian *Selection type*, pilih *Form*.



5. Di bagian *Data*, untuk *Record Source* pilih *Query\_TRANSAKSI*. Lihat hasilnya di *View*.

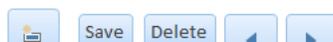


6. Agar transaksi pada form dapat beroperasi dengan baik, maka perlu untuk menambahkan beberapa tombol seperti *add*, *save*, *delete*, *previous*, *next*.
7. Untuk membuat tombol *add*, pilih *Command Button* pada menu *Design* dan buat tombol di area form yang kosong. Secara otomatis akan muncul kotak dialog *Command Button Wizard*. Dalam kasus ini, tombol berada di bawah field *TOTAL BAYAR*.



8. Pilih *Categories - Record Operations* dan *Actions - Add New Record*, kemudian klik *Next*. Akan ada dua pilihan untuk menggambarkan tombol yang dibuat yaitu berupa *Text* atau *Picture*. Dalam kasus ini, pilih *picture*, kemudian *next* dan *finish*. Ulangi langkah ini sampai tombol terakhir *previous*.

KODE KARYAWAN	K004
NAMA KARYAWAN	ASNANTIA
KODE BARANG	mul
NAMA BARANG	MULTI TISSUE
HARGA SATUAN	Rp3.200
JUMLAH BELI	4
TOTAL BAYAR	Rp12.800



9. Aplikasikan tombol-tombol yang sudah dibuat dengan menambah data baru dan yang lainnya.
10. Simpan form ini dengan nama Form Transaksi. Form ini merupakan model form yang sangat sederhana. Ada banyak model form yang cukup menarik yang bisa Anda buat. Lanjutkan !

## 9. REPORT

*Report* merupakan laporan tertulis dari hasil transaksi, baik tertulis secara report (pada layer monitor) maupun hard report (pada kertas). Report juga berarti beberapa informasi yang diorganisir sedemikian rupa untuk memenuhi berbagai kebutuhan seperti label surat-menyurat, summary penjualan, dll.

Langkah-langkah membuat *Report* :

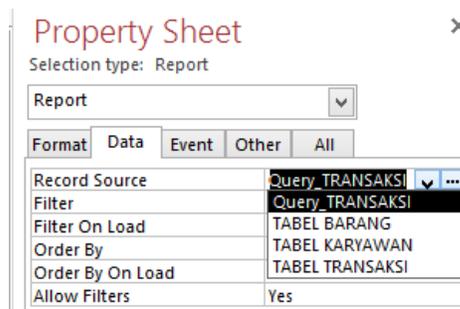
1. Pilih menu *Create* lalu *Report design*.
2. Pilih beberapa *field* yang akan dijadikan laporan. Caranya sama dengan menambah field pada Form.

NAMA KARYAWAN, NAMA BARANG, JUMLAH BELI, TOTAL BAYAR diambil dari TABEL TRANSAKSI.

HARGA SATUAN diambil dari TABEL BARANG.

NAMA KARYAWAN	NAMA KARYAWAN
NAMA BARANG	NAMA BARANG
HARGA SATUAN	HARGA SATUAN
JUMLAH BELI	JUMLAH BELI
TOTAL BAYAR	TOTAL BAYAR

3. Agar setiap field dapat terhubung dengan Form dan Query, maka pilih *Property Sheet* pada menu *Design*. Kemudian di bagian *selection type* pilih report. Untuk bagian *Record source* pilih Query\_TRANSAKSI.



4. Lihat hasilnya di View.

NAMA KARYAWAN	ITSNANIYA
NAMA BARANG	AQUA GELAS
HARGA SATUAN	Rp500
JUMLAH BELI	20
TOTAL BAYAR	Rp10.000

NAMA KARYAWAN	HAMIM
NAMA BARANG	BERAS CAP TAWON
HARGA SATUAN	Rp45.500
JUMLAH BELI	3
TOTAL BAYAR	Rp136.500

Tampilan pada report agak sedikit berbeda dengan form. Pada report, hasil yang ditampilkan akan sangat banyak sesuai dengan banyaknya transaksi. report yang telah dibuat ini merupakan report yang sangat sederhana. Ada banyak model report yang simple dan menarik yang bisa anda buat. Seperti contoh di bawah ini.

 Laporan Transaksi Toko Infomart 28 Februari 2013  
17:14:32

NAMA KARYAWAN	NAMA BARANG	JUMLAH BELI	TOTAL BAYAR	HARGA SATUAN
ITSNANIYA	AQUA GELAS	20	Rp10.000	Rp500
HAMIM	BERAS CAP TAWON	3	Rp136.500	Rp45.500
NAVIUL	BERAS CAP TAWON	2	Rp91.000	Rp45.500
HANIYYA	BIORE BODY WASH	4	Rp39.200	Rp9.800
AZIZ	KECAP ABC	12	Rp30.000	Rp2.500
ASNANTIA	MULTI TISSUE	4	Rp12.800	Rp3.200
INDAH	NICE TISSUE	1	Rp3.500	Rp3.500
NAVIUL	SAUS TOMAT ABC	3	Rp16.800	Rp5.600
ITSNANIYA	SUN KARA SANTAN	4	Rp9.600	Rp2.400
ASNANTIA	SUN KARA SANTAN	3	Rp7.200	Rp2.400
INDAH	SUPERMIE AYAM BAWANG	3	Rp4.500	Rp1.500

## 10. MACRO

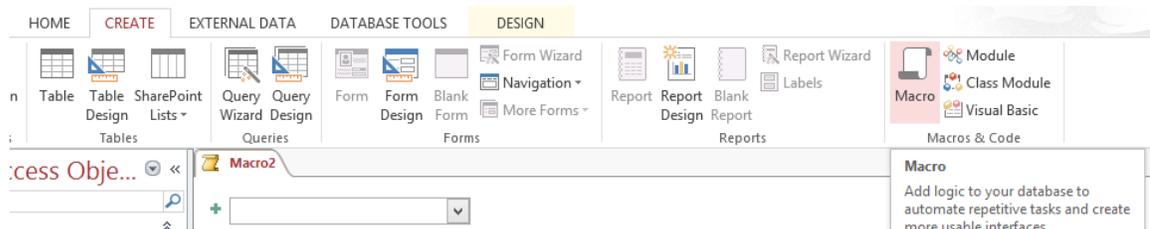
MACRO merupakan salah satu fasilitas dalam Access untuk melakukan otomatisasi sekumpulan pekerjaan-pekerjaan tertentu dengan satu perintah. Misalnya dengan menekan tombol klik, macro akan menjalankan perintah membuka form, menampilkan data tertentu dan lain sebagainya.

Makro adalah suatu otomatisasi sekumpulan pekerjaan-pekerjaan tertentu yang sering dilakukan pada Access dengan adanya makro dapat menjalankan sekelompok tugas sekaligus hanya dengan satu klik pada tombol form yang sudah dirancang.

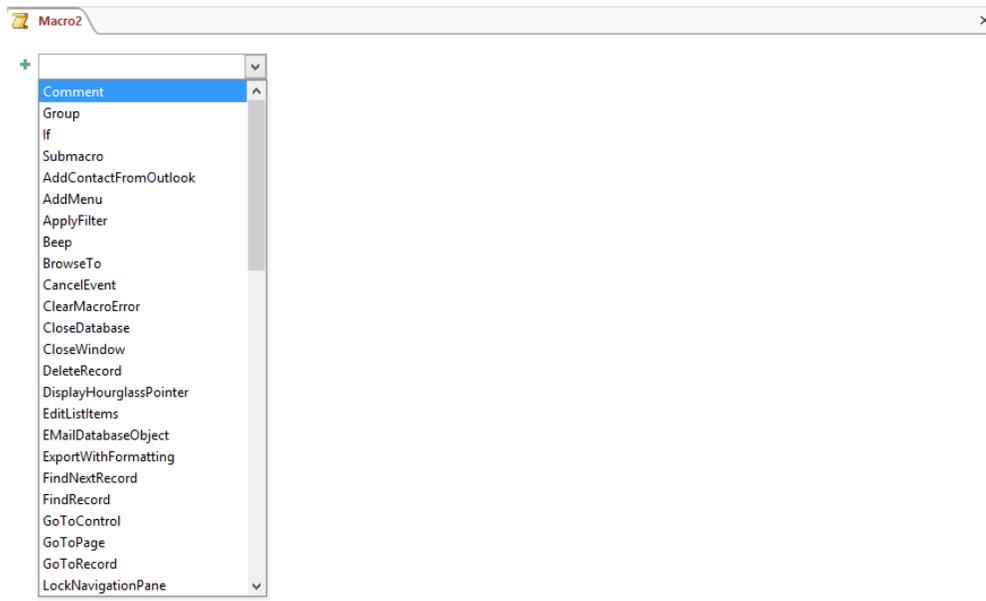
Membuat makro disini penekannya lebih pada penyusunan makro aksi yang telah disediakan Access 2013 bukan membuat makro sendiri dengan pemrograman VB.

Berikut langkah-langkah dari contoh pembuatan Macro Access 2013,

1. Klik Menu *Create* → *Macro*



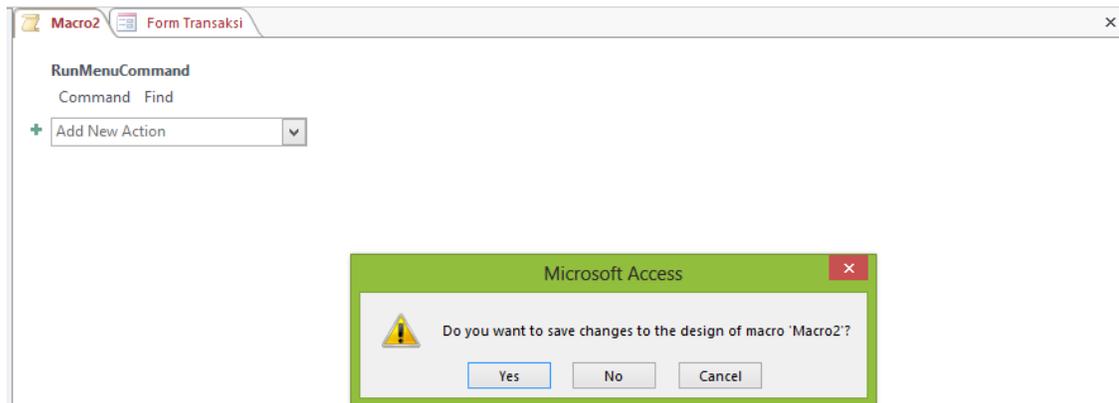
2. Pilih *New Action* sesuai dengan yang dibutuhkan, misal *Run Menu Command*



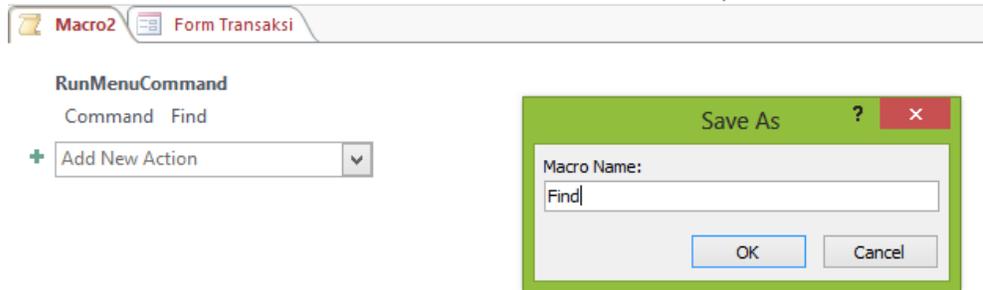
3. Dalam *Action* tersebut, misalnya dimasukkan command ”*Find*”



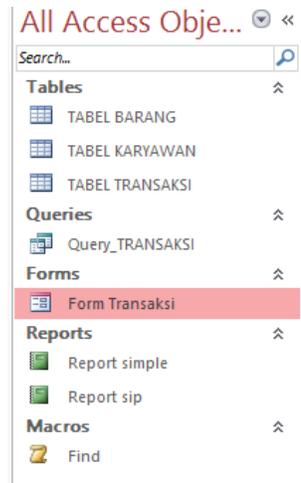
4. Setelah Selesai menulis *action* dan command yang diinginkan, *close Macro*, dan jangan lupa untuk menyimpannya.



5. Beri nama yang sesuai dengan *action macro*



6. Setelah tersimpan, *macro* juga akan muncul di sebelah kanan *sidebar* dalam database anda.



Untuk Penggunaannya, *macro* dipadukan dengan *button* yang ada di *form*.

Daftar *Macro Action* yang disediakan oleh *Access 2013*

Nama Macro	Aksi
<i>AddMenu</i>	Membuat menu bar sendiri.
<i>ApplyFilter</i>	Mengaplikasikan suatu penyaringan data.
<i>Beep</i>	Menimbulkan suara bip dari komputer.
<i>CancelEvent</i>	Membatalkan event yang menyebabkan Access 2013 menjalankan suatu aksi.
<i>Close Database</i>	Menutup suatu database
<i>Close Window</i>	Menutup suatu jendela
<i>Delete Record</i>	Menghapus Record
<i>FindRecord</i>	Mencari suatu record.
<i>GoToControl</i>	Memindahkan fokus ke kontrol tertentu.
<i>GoToPage</i>	Memindahkan fokus ke kontrol pertama di suatu halaman.

<i>GoToRecord</i>	Menuju ke record tertentu.
<i>If</i>	Mengkodisikan suatu perintah
<i>Maximize Window</i>	Memaksimalkan ukuran suatu jendela.
<i>Minimize Window</i>	Meminimalkan ukuran suatu jendela.
<i>MoveSize</i>	Mengubah ukuran suatu jendela.
<i>MsgBox</i>	Menampilkan suatu kotak dialog berisi pesan tertentu.
<i>On Error</i>	Apabila action menunjukkan kesalahan
<i>OpenDataAccessPage</i>	Membuka suatu data access page.
<i>OpenDiagram</i>	Membuka suatu diagram.
<i>OpenForm</i>	Membuka suatu form pada mode form view, form design view, print preview, atau datasheet view.
<i>OpenFunction</i>	Membuka sebuah fungsi buatan sendiri (User Defined Function/UDF). UDF merupakan query parameter yang memberikan hasil seperti prosedur tersimpan (stored procedure).
<i>OpenModule</i>	Membuka suatu modul Visual Basic.
<i>OpenQuery</i>	Membuka select query atau crosstab query. Macro aksi ini menjalankan action query.
<i>OpenReport</i>	Membuka suatu report pada mode design view, print preview, atau langsung mencetaknya.
<i>OpenStoredProcedure</i>	Membuka suatu prosedur tersimpan yang ada pada Access 2003 project.
<i>OpenTable</i>	Membuka suatu table pada mode datasheet view, design view, atau print preview.
<i>OpenView</i>	Membuka suatu view yang ada pada Access 2003 project.
<i>OutputTo</i>	Menampilkan output suatu obyek ke dalam format tertentu, seperti HTML, Excel, RTF, XML, dan lain-lain.
<i>PrintOut</i>	Mencetak suatu obyek database.
<i>Quit</i>	Keluar dari Access 2003.
<i>Rename</i>	Mengganti nama suatu obyek.
<i>RepaintObject</i>	Memperbarui suatu tampilan obyek pada layer monitor computer.

<i>Requery</i>	Memperbarui data pada suatu control dengan melakukan query ulang.
<i>Restore</i>	Mengembalikan ukuran jendela yang dimaksimalkan atau diminimalkan ke ukuran semula.
<i>RunApp</i>	Menjalankan sebuah program aplikasi eksternal.
<i>RunCode</i>	Menjalankan suatu kode VBA.
<i>RunCommand</i>	Menjalankan perintah built-in Access 2003.
<i>RunMacro</i>	Menjalankan sebuah macro.
<i>RunSQL</i>	Menjalankan sebuah perintah SQL.
<i>Set Filter</i>	Untuk menyaring data dalam tabel, report atau form
<i>SubMacro</i>	Dalam submacro, dapat diisi macro baru lagi
<i>Save</i>	Menyimpan suatu obyek.
<i>SelectObject</i>	Memilih suatu obyek.
<i>SendKeys</i>	Mengirimkan event penekanan tombol pada keyboard.
<i>SendObject</i>	Mengirimkan obyek tertentu via email.
<i>SetMenuItem</i>	Menentukan status suatu item menu.
<i>SetValue</i>	Menentukan nilai suatu obyek.
<i>SetWarnings</i>	Mengaktifkan atau mematikan pesan peringatan yang berasal dari sistem.
<i>ShowAllRecords</i>	Menampilkan seluruh record dengan mengabaikan query atau filter.
<i>ShowToolbar</i>	Menampilkan atau menyembunyikan suatu toolbar.
<i>StopAllMacros</i>	Menghentikan seluruh makro yang sedang berjalan.
<i>StopMacro</i>	Menghentikan suatu makro yang sedang berjalan.
<i>Set Property</i>	Menentukan property parameter dari action dari tabel, form,query,dsb
<i>Undo Record</i>	Mengulangi Record (mengcopy)

## 11. MODULE

Modul merupakan kumpulan perintah yang sedikit sama seperti *VBA (Visual Basic for Application)*. Perintah yang akan anda buat akan jauh lebih mudah dan memiliki banyak nilai untuk membuat kondisi tertentu, dikarenakan penulisan modul hamper sama seperti penulisan listing program pada *Visual Basic* .

Sebuah objek dalam *module* merujuk pada sebuah *module* standar atau *module* kelas. Dalam *Ms. Acces 2013* terdiri dari *module* kelas yang tidak terhubung dengan objek manapun. Sedangkan modul *form* dan modul *report* terhubung dengan *form* dan *report*.

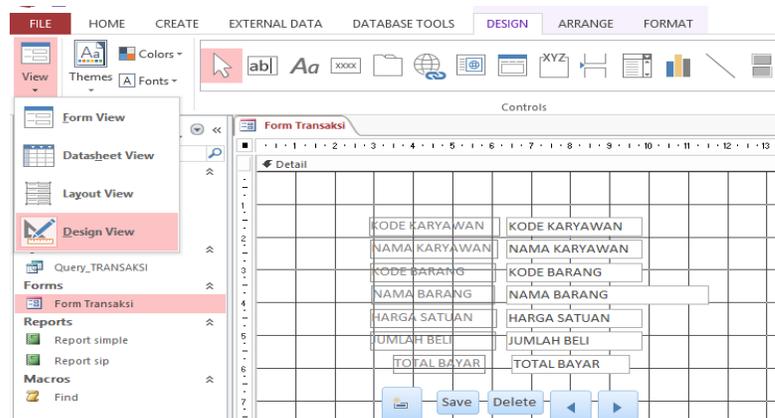
Untuk menentukan jenis *module* yang digunakan, *check the Module object's Type Property*. *Module Collection* terdiri dari *Open Module Objects*, *Module Collection* dapat dibentuk atau tidak tergantung dari tipe *module* dalam *Module Collection*.

Berikut beberapa *syntax* dalam penggunaan *Module*,

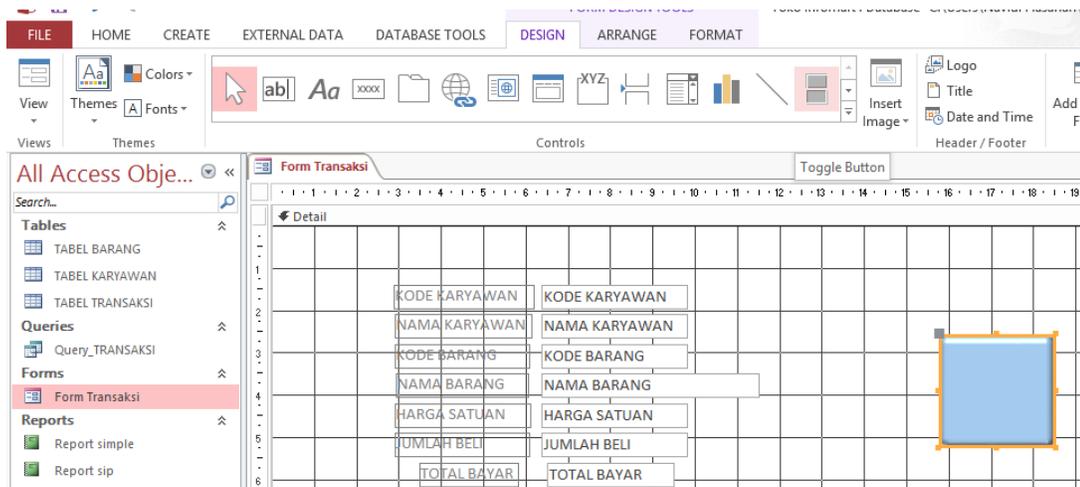
Syntax	Description
<i>Modules!modulname</i>	Penamaan Module
<i>Modules("modulename")</i>	Penamaan module
<i>Form.Module.Property</i>	Untuk menyusun module yang berasal dari properties form
<i>ReplaceLine Methods</i>	Mengganti baris tertentu dalam modules standar atau modules kelas

Langkah – langkahnya sebagai berikut:

1. Pada Database, setelah jendela database dibuka, aktifkan *object form* dan klik *Form Transaksi* dan klik tombol *Design*.

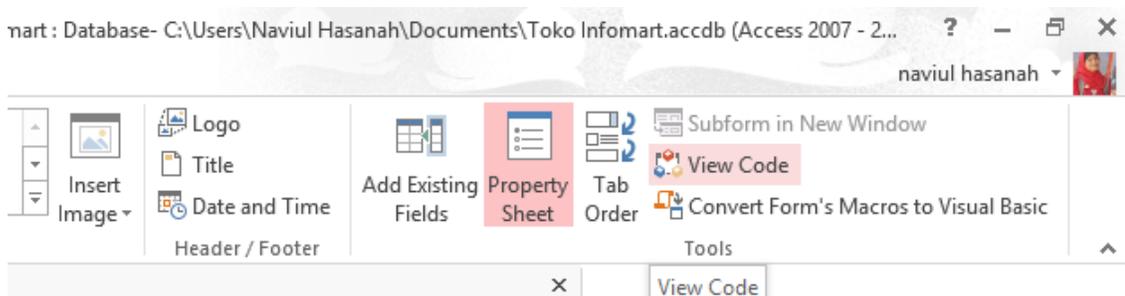


2. Setelah *form* desain tampil, masukkan sebuah komponen *Toggle Button* dengan penempatan seperti gambar di bawah ini
3. Klik ganda *Toggle button* dan klik hari seperti gambar dibawah ini.



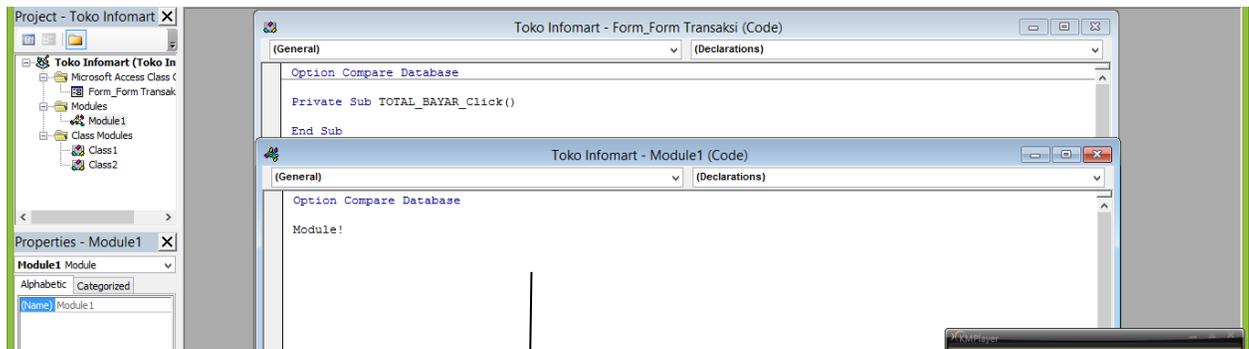
Memasukkan tombol Toggle Button

4. Pada menu *toolbar* atas, klik *View Code*.



Membuka Jendela Access

5. Setelah jendela code tampil, klik ganda *module1* pada project eksplorer dan ketikkan perintah di bawah ini.



Memasukkan listing perintah

6. Setelah selesai, klik ganda *form* pada jendela *project explorer*.
7. Ubah kode general menjadi *Command15* dan masukkan listingnya sesuai gambar.
8. Setelah seluruh *listing* selesai, tutup seluruh jendela VBA dan simpan file dengan nama *Modul1*.
9. Masih pada jendela *form desain*, ubah nama *Command15* menjadi hari dan tutup *form*.
10. Jalankan kembali *Form desain*, ubah nama tombol hari untuk menampilkan sebuah pesan.