



PRAKTIKUM PERANCANGAN BASIS DATA

[Modul Praktikum Diploma III]

Akademi Manajemen Informatika dan Komputer Bina Sarana Informatika

Tim Penyusun Modul BSI 2014

Basis Data dan Normalisasi

☑ Konsep Dasar Basis Data

Basis Data (*Database*) dapat dibayangkan sebagai sebuah lemari arsip. Jika kita memiliki sebuah lemari arsip dan berwenang untuk mengelolanya. Atau kumpulan informasi yang terorganisasi dan disajikan untuk tujuan khusus. Prinsip utama basis data adalah pengaturan data atau arsip. Sedangkan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan data atau arsip. Sedangkan sistem basis data merupakan perpaduan antara basis data dan sistem manajemen basis data (DBMS). DBMS (*Database Management System*) adalah *software* yang menangani semua akses ke basis data. Contoh dari DBMS yaitu Microsoft Access, MySQL, Oracle, Server 2000, Interbase, Paradox, dan Lain-Lain.

Komponen-komponen basis data antara lain :

1. Entitas

Entitas digunakan untuk menerapkan integritas pada tingkat Entity (Tabel), agar setiap Instances (Record/Baris) pada suatu Entity bersifat unique yang disebut sebagai Primary Key sehingga dapat dibedakan antara yang satu dengan yang lain.

Contoh:

- a. Semua pelanggan, atau pelanggan saja dengan entitas Adi, Ryan, Endah dan seterusnya.
- b. Semua Mobil atau mobil apa saja dengan entitas mobil Kijang, Starlett dan lain-lain.

2. Atribut

Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik dari entitas tersebut. Penentuan atau pemilihan atribut-atribut yang relevan bagi sebuah entitas merupakan hal penting lainnya dalam pembentukan model data. Penetapan atribut bagi sebuah entitas umumnya memang didasarkan pada fakta yang ada. Tetapi tidak selalu seperti itu. Istilah atribut sebenarnya identik dengan pemakaian kolom data.

Contoh :

a. Entitas pelanggan

Atributnya kd_pelanggan, nm_pelanggan, alamat, notelpon.

☑ Normalisasi

Dalam perancangan sebuah basis data perlu dilakukan secara cermat agar dihasilkan data yang kompak dan efisien dalam penggunaan ruang penyimpanan, cepat dalam pengaksesan dan mudah dalam manipulasi data. Salah satu cara yang dapat dilakukan dalam merancang basis data seperti ini adalah dengan melakukan normalisasi.

Normalisasi adalah proses penyusunan tabel-tabel yang tidak redundan (double), yang dapat menyebabkan anomali pada saat terjadi operasi manipulasi data seperti tambah, ubah, dan hapus. Anomali yaitu proses basis data yang memberikan efek samping yang tidak diharapkan (misalnya menyebabkan ketidakkonsistenan data atau membuat sesuatu data menjadi hilang ketika data lain dihapus). Adapun bentuk-bentuk dalam normalisasi:

1. Bentuk Tidak Normal

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti format tertentu, dapat saja tidak lengkap dan terduplikasi, data dikumpulkan apa adanya sesuai keadaannya. Data didapat dari bentuk dokumen yang ada.

2. Bentuk Normal Pertama (1st Normal Form)

Sebuah tabel disebut 1NF jika :

- a. Tidak ada baris yang duplikat dalam tabel tersebut.
- b. Masing-masing cell bernilai tunggal.

3. Bentuk Normal Kedua (2nd Normal Form)

Bentuk Normal Kedua(2NF) terpenuhi jika pada sebuah tabl semua atribut yang tidak termasuk dalam primary key memiliki ketergantungan fungsional pada primary key secara utuh.

4. Bentuk Normal Ketiga (3rd Normal Form)

Menghilangkan anomali-anomali hasil dari ketergantungan fungsional.

5. Bentuk Normal Keempat (4th Normal Form)

6. BCNF (Boyce Code Normal Form)

Berikut ini adalah contoh kasus mengenai pembelian barang yang digambarkan dalam normalisasi:

1. Dokumen Dasar

Faktur Pembelian Barang				
PT. Belia				
Jl. Bekasi Timur No. 24				
Tanggal : 5 Juli 2013		Kode Supplier : S001		
Nomor Faktur : F001		Nama Supplier : PT. Suplia		
Kode Barang	Nama Barang	Jumlah Beli	Harga	Subtotal
T001	TV 20"	10	Rp. 1.200.000,-	Rp. 12.000.000,-
T002	TV 29"	5	Rp. 2.400.000,-	Rp. 120.000.000,-
Total Bayar				Rp. 132.000.000,-

2. Bentuk Tidak Normal (UNF)

kode_fak
kd_sup
nm_sup
kd_brg
nm_brg
tgl
jumbel
harga
subtotal
total

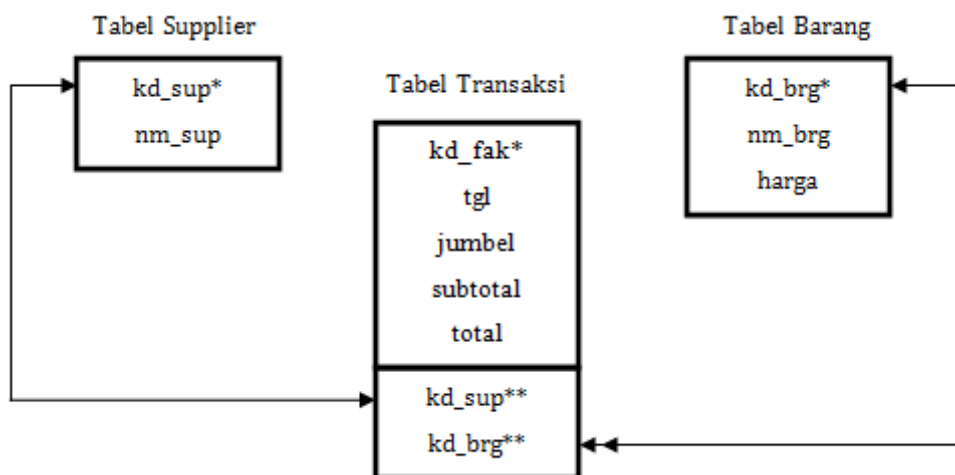
3. Bentuk Normal Pertama (1NF)

kode_fak*
kd_sup*
nm_sup
kd_brg*
nm_brg
tgl
jumbel
harga
subtotal
total

Keterangan :

* = *Candidate Key*

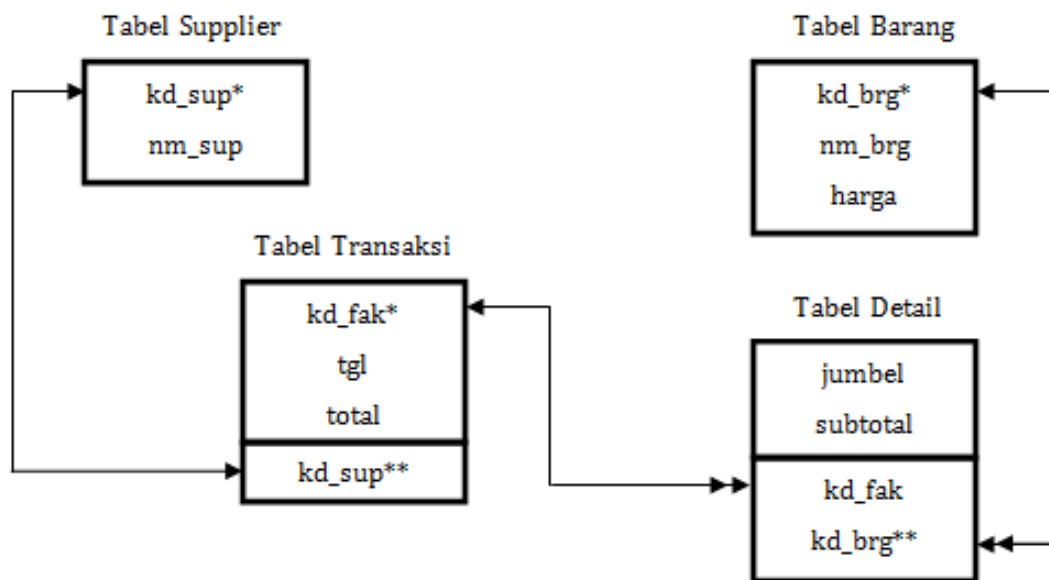
4. Bentuk Normal Kedua (2NF)



Keterangan:

- * : Primary Key ↔ : One to One Relationships
- ** : Foreign Key ↔ : One to Many Relationships

5. Bentuk Normal Ketiga (3NF)



Keterangan:

- * : Primary Key ↔ : One to One Relationships
- ** : Foreign Key ↔ : One to Many Relationships



Pengenalan MySql dan Lingkungan Kerja

MySql adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user serta menggunakan perintah standar SQL (*Structure Query Language*). MySql merupakan sebuah database server yang *free* artinya kita bebas menggunakan database ini untuk keperluan pribadi atau usaha tanpa membeli atau membayar lisensiya.

MySQL merupakan hasil ciptaan dari **Michael "Monty" Widenius, David Axmark,** dan **Allan Larson**. Pada tahun 1995 mereka lalu mendirikan perusahaan bernama MySQL AB di Swedia. Tujuan awal diciptakannya MySQL yaitu untuk mengembangkan aplikasi web yang akan digunakan oleh salah satu klien MySQL AB. Database MySQL, biasa kita baca *mai-es-ki-el* merupakan perangkat lunak database yang berbentuk database relasional atau dalam bahasa basis data sering disebut dengan *Relation Database Management System* (RDBMS) yang menggunakan suatu bahasa permintaan bernama SQL. Perbedaan antara MySQL dan SQL, MySQL adalah program database server sedangkan SQL adalah bahasa yang digunakan didalamnya.

Kelebihan MySQL

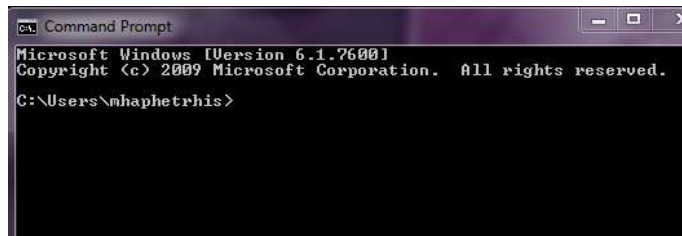
MySQL adalah sebuah database *server* dapat juga berperan sebagai *client* sehingga disebut database *client / server* yang *open source* dapat berjalan pada OS manapun dengan *platform* Windows maupun Linux.

Penggunaan MySQL dengan Command Prompt

Dalam pembahasan ini kita akan menggunakan paket program **XAMPP versi 1.7.3** dengan sistem operasi Windows XP. Paket XAMPP biasanya di-instal pada direktori atau folder C:\ XAMPP. Sedangkan untuk databasenya akan tersimpan di direktori **C:\xampp\mysql\data**.

Untuk dapat menggunakan mysql terlebih dahulu aktifkan dengan cara sebagai berikut :

1. Pilih menu **Start** → **All Programs** → **Accessories** Kemudian Pilih **Command Prompt**. Selanjutnya akan tampil jendela sebagai berikut:



```
Command Prompt
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\mhaphetris>
```

2. Kemudian ketikkan perintah seperti dibawah ini pada command prompt tersebut untuk menjalankan program mysql:

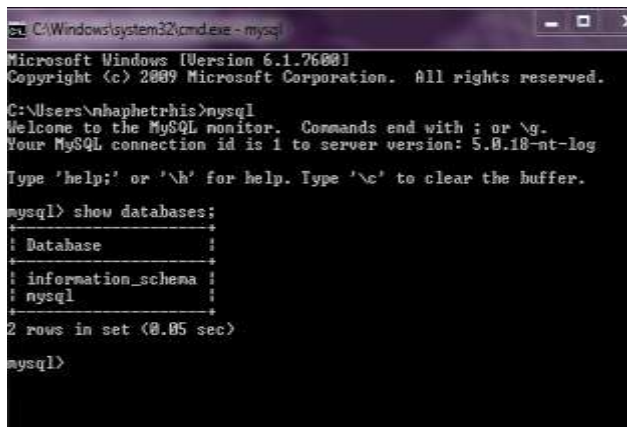
3. Pada saat anda diminta untuk memasukkan password, tekan **enter** pada keyboard. Bila perintah tersebut berhasil, maka pada layar akan tampil ucapan selamat datang seperti tampilan dibawah ini:

☑ Pengantar Sintak Dasar

MySQL merupakan bahasa pemrograman database dimana penulisan sintaknya tidak serumit bahasa pemrograman lainnya seperti Java, C++, dan sebagainya. Satu hal yang perlu diingat bahwa setiap penulisan script MySQL di Dos-Prompt harus selalu diakhiri dengan tanda titik koma (;). Didalam *source* MySQL yang terinstal secara *default* telah terisi sebuah database yang bernama *mysql* dan *information_schema*.

1. Sintak untuk menampilkan semua nama database yang telah ada adalah:

```
Mysql>show databases;
```



```
C:\Windows\system32\cmd.exe - mysql
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\nhaphetrhis>mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 5.0.10-nt-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

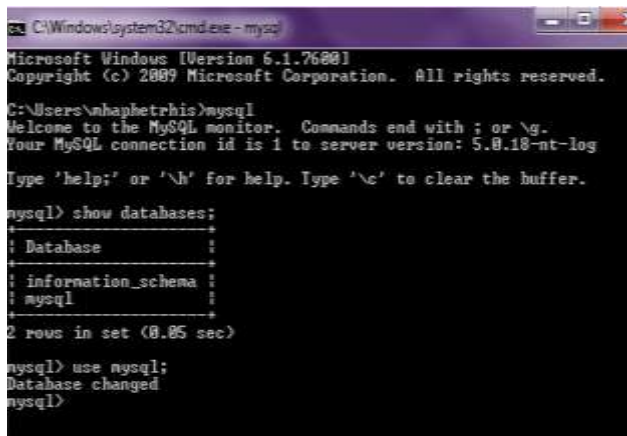
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
+-----+
2 rows in set (0.05 sec)

mysql>
```

Keterangan: Contoh diatas menunjukkan bahwa dengan menggunakan sintak `show databases;` berarti kita dapat menampilkan seluruh nama database yang ada.

2. Sintak untuk bisa masuk kedalam salah satu sistem database adalah:

```
Mysql>use nama_database;
```



```
C:\Windows\system32\cmd.exe - mysql
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\nhaphetrhis>mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 5.0.10-nt-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
+-----+
2 rows in set (0.05 sec)

mysql> use mysql;
Database changed

mysql>
```

Keterangan: dengan mengetikkan sintak `use mysql;` akan tampil kalimat database change, berarti kita telah masuk ke dalam database `mysql` yang telah ada.

Dari gambar di atas akan terlihat kolom host, user dan password yang telah terisi. Isi dalam kolom tersebut seperti gambar dibawah ini:

Host	User	Password
localhost	Root	1fb744464c9bba54

localhost merupakan nama komputer kita sendiri apabila komputer tersebut kita jadikan server. Sedangkan **root** adalah nama user yang berhak penuh untuk mengatur database yang telah ada seperti menambah user, memberi fasilitas kepada user lain agar user lain dapat akses ke database tersebut, bahkan juga root bisa membuat database baru yang dapat digunakan kapanpun. Kolom password yang terisi oleh kombinasi angka dan huruf diatas yaitu **1fb744464c9bba54** merupakan hasil password yang telah terenkripsi pada sintak terdahulu yaitu dengan menambahkan sintak password itu sendiri.

Data Definition Language

☑ *Data Definition Language*

Skema basis data dispesifikasikan oleh sekumpulan definisi dengan sebuah bahasa khusus yang disebut *Data Definiton Language* (DDL). Hasil kompilasi DDL berupa tabel-tabel yang disimpan dalam sebuah file, disebut data *dictionary* (Kamus Data) atau data *directory*. Kamus data adalah sebuah *file* yang berisi metadata. *File* ini yang dikonsultasi sebelum data yang sebenarnya dibaca atau dimodifikasi oleh sistem basis data.

☑ **Penggunaan sintak DDL dalam membuat database**

Database adalah sebuah media utama yang harus dibuat dalam membangun sebuah basis data agar nantinya dapat diletakkan beberapa tabel dengan field-fieldnya. Bila kita lihat dalam mysql database hanyalah semacam direktori saja, berbeda jika dibandingkan dengan program database yang berbentuk visual seperti access ataupun paradox. Pada mysql database akan disimpan dan dikumpulkan pada sebuah direktori khusus yang bernama data. Dalam direktori **C:\Apache2triad\mysql\data**, semua sub direktori yang ada didalamnya disebut database.

1. Sintak untuk menciptakan database baru

```
Mysql>create database nama_database;
```

```
Mysql>create db_penjualan;
```

☑ **Penggunaan sintak DDL dalam membuat tabel**

Tabel adalah obyek utama yang harus ada pada sebuah basis data karena didalamnya semua data akan disimpan. Tabel terletak pada sebuah database, sehingga pembuatan table dilakukan setelah database dibuat. Untuk membuat tabel kita harus terlebih dahulu **mengaktifkan** database, perintah yang digunakan adalah *use*. Syntak penulisannya adalah : **Mysql>use namadatabase;** Setelah masuk kedalam database, anda telah diijinkan melakukan operasi-operasi menyangkut tabel dan isinya.

2. Sintak untuk membuat sebuah tabel

```
Mysql>create table nama_tabel(field-1 type(lenght), field-2 type(lenght),..... field-  
n type(lenght),(.....));
```

```
Mysql>create table tb_barang (kd_brg char(5), nm_brg varchar(20), harga int);
```

3. Sintak untuk melihat suatu struktur/field tabel

```
Mysql>desc nama_tabel;
```

```
Mysql>desc tb_barang;
```

4. Sintak untuk membuat *primary key*

Dalam membuat sebuah database, kita pasti akan menemui sebuah *record* yang datanya tidak boleh sama dengan *record* yang lain. Agar data tidak sama kita dapat membuat sebuah kolom data dengan isi yang berbeda dengan kolom lain. Kita dapat mendeklarasikan sebuah kolom sebagai kunci primer atau *primary key*.

```
Mysql >alter table nama_table add primary key (nama_field_primary);
```

```
Mysql >alter table tb_barang add primary key (kd_brg);
```

5. Sintak untuk menghapus *Primary Key*

```
Mysql>alter table nama_table drop primary key;
```

```
Mysql>alter table tb_barang drop primary key;
```

6. Sintak untuk menghapus tabel

```
Mysql>drop table nama_tabel;
```

```
Mysql>drop table tb_barang;
```

7. Sintak untuk menghapus database.

```
Mysql>drop database nama_database;
```

```
Mysql>drop database db_penjualan;
```

☑ **Perintah-Perintah DDL yaitu:**

No.	Keterangan	Perintah	Contoh
1.	Perubahan field table	Alter table nama_table alter spesifikasi kondisi;	Alter table tb_barang add ket varchar(30); Alter table tb_barang add ket varchar(30) after nm_brg;
2.	Menambah field unik	Alter table nama_table add index (nama_field);	Alter table tb_barang add index (nm_brg);
3.	Mengubah nama field	Alter table nama_table change field_lama field_baru type(length);	Alter table tb_barang change nm_brg nabar varchar(20);
4.	Membuat Validasi	Create table nama_tabel(field-1 type(length), field-2 type(length), ... field-n enum ("pilihan1","pilihan2"));	Create table tb_barang (kd_brg char(5), nm_brg varchar(20), ket varchar(50), jenis enum ("new","old"));
5.	Menghapus field	Alter table nama table drop nama_field;	Alter table tb_barang drop ket;
6.	Menghapus index	Alter table nama_table drop index nama_field;	Alter table tb_barang drop index nm_brg;
7.	Mengganti nama tabel	Rename table tabel_lama to tabel_baru;	Rename table tb_barang to tb_barang_baru;

Data Manipulation Language

☑ *Data manipulation Language (DML)*

DML atau *Data Manipulation Language* merupakan perintah-perintah yang berfungsi untuk melakukan manipulasi data ataupun objek-objek yang ada didalam tabel. Definisi lainnya tentang DML adalah perintah-perintah untuk memanipulasi data pada basis data, misalnya perintah untuk memilih data (*query*), menyisipkan, mengubah dan menghapus data dalam basis data. Bentuk manipulasi yang dapat dilakukan oleh DML diantaranya adalah untuk keperluan pencarian kembali data lama, penyisipan data baru, penghapusan data, pengubahan data dll. Sedangkan DML sendiri menurut jenisnya dapat dibagi menjadi 2 Jenis yaitu *procedural* dan *non procedural*. Beberapa manfaat atau kegunaan dari DLL diantaranya adalah sebagai berikut :

1. Pengambilan informasi yang disimpan dalam basis data.
2. Penyisipan informasi baru ke basis data.
3. Penghapusan informasi dari basis data.
4. Modifikasi informasi yang disimpan dalam basis data.
5. *Query* adalah perintah yang ditulis untuk mengambil informasi. Bagian dari DML yang menangani pengambilan informasi ini disebut bahasa *query*.

☑ *Penggunaan Data Manipulation Language (DML)*

1. Perintah Insert

Perintah INSERT digunakan untuk menambah data baru kedalam tabel. Insert mempunyai 2 format penulisan. Berikut ini contoh-contoh penulisan perintah INSERT dalam proses penambahan data kedalam tabel pelanggan:

```
mysql>insert into nama_tabel (daftar_field) values (daftar_nilai);
```

```
Mysql>insert into nama_tabel set nama_field=nilai_field;
```

```
mysql>insert into tb_barang (kd_brg, nm_brg, harga) values ("BRG01","Pulpen",2500);
```

atau

2. Perintah Select

Berfungsi untuk menampilkan data yang terdapat didalam database atau tabel.

```
Mysql>select * from nama_tabel;
```

```
Mysql>select * from tb_barang;
```

3. Perintah Update

Berfungsi untuk melakukan perubahan terhadap sejumlah data yang ada didalam tabel yang telah dibuat.

```
Mysql>update nama_tabel set field-2=nilai_field-2, field-n=nilai_field-n where field-1=nilai_field-1;
```

```
Mysql>update tb_barang set nm_brg="Penggaris", harga=1500 where kd_brg="BRG01";
```

4. Perintah Delete

Berfungsi untuk menghapus sejumlah data yang ada didalam table yang telah dibuat.

```
Mysql> delete from nama_tabel where field=nilai_field;
```

```
Mysql> delete from tb_barang where kd_brg="BRG01";
```


Latihan Command Prompt

1. Buat tabel dengan nama **tb_supplier** didalam database **db_penjualan** dengan struktur sebagai berikut:

Field	Type	Size	Keterangan
kd_sup	Char	5	Primary key
nama	Varchar	25	

Isilah record pada tabel tb_supplier seperti dibawah ini:

kd_sup	nama
SUP01	Kayla Prastina
SUP02	Fara Putri
SUP03	Fahri Putra

2. Tampilkanlah isi *record* dari tabel **supplier**.
3. Gantilah **nama** pada kd_sup=**SUP02** menjadi **Atilah Surya Pribadi**.
4. Hapuslah **record** untuk ko_sup=**SU03**.
5. Ganti nama *field* **nama** menjadi **nm_sup**.
6. Tambahkan field **alamat** setelah **nama** dengan tipe data **varchar 10**.
7. Ganti nama tabel **tb_sup** menjadi **tb_supplier**.
8. Tampilkanlah struktur dari **tabel supplier**.
9. Hapus *field* alamat dari **tabel supplier**.
10. Buatlah field unik untuk *field* nm_sup yang ada di **tabel supplier**.

PhpMyadmin

PhpMyadmin adalah alat yang ditulis dalam PHP yang ditujukan untuk menangani administrasi MySQL melalui web. Membuat dan *drop* database. Membuat/merubah table, menghapus/mengedit/menambahkan kolom, jalankan SQL pernyataan apapun, mengatur kunci pada *field*, mengatur hak, ekspor data kedalam berbagai format dan tersedia dalam 50 bahasa.

1. Mengaktifkan PHPMyadmin

- a. Buka *browser* (IE/Mozilla Firefox/Opera)
- b. Kemudian ketikkan **localhost/phpmyadmin** pada **address bar**.



- c. Masukkan username : **root** dan passwordnya : **password**, kemudian akan tampil seperti di bawah ini :



2. IDE *phpmyadmin*



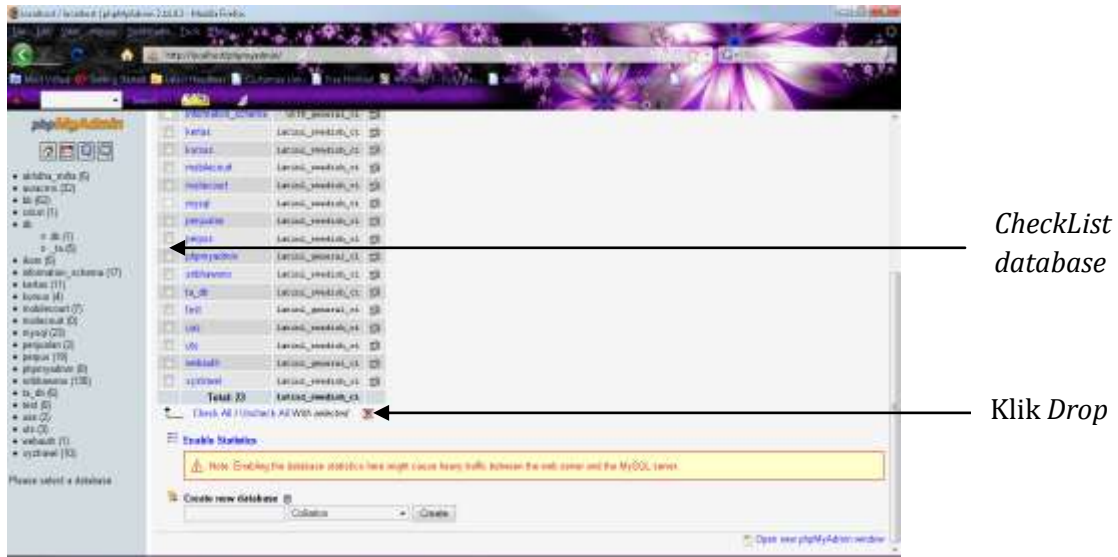
3. Membuat Database

- a. Ketikkan nama database pada textbox ***create new database***
- b. Kemudian klik ***create***



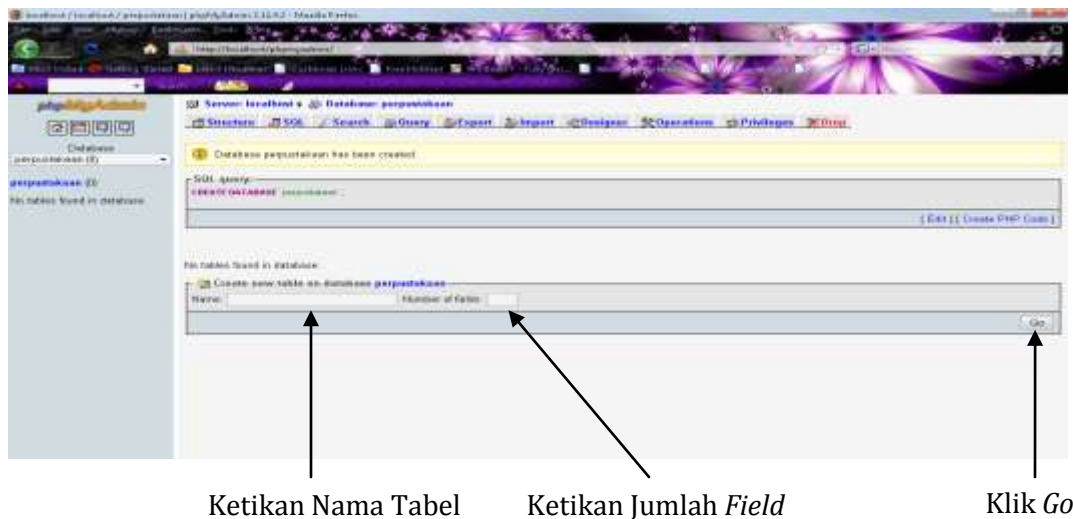
4. Menghapus database

- a. Pilih menu database pada tampilan *Home* Phpmyadmin.
- b. *Checklist* pada database yang akan dihapus kemudia pilih ***drop***.



5. Pembuatan Tabel

- a. Ketikkan nama tabel pada **textbox name**, kemudian ketikkan jumlah **field** yang akan dibuat pada **number of field** tersebut setelah itu klik **Go** untuk memulai pengetikan struktur datanya.



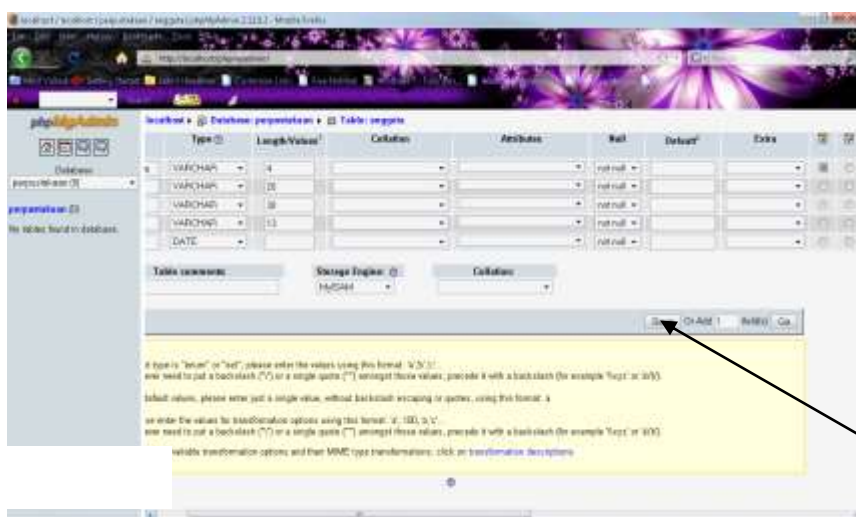
b. Maka tampil layar seperti gambar di bawah ini.



c. Kemudian buatlah struktur tabel seperti contoh dibawah ini

Nama Field	Type	Size	Keterangan
id_anggota	Varchar	4	Primary Key
nama	Varchar	20	
alamat	Varchar	30	
notelp	Varchar	13	
tgl_lahir	Date		

d. Setelah kita masukkan *field-field* yang dibutuhkan kemudian klik **save**

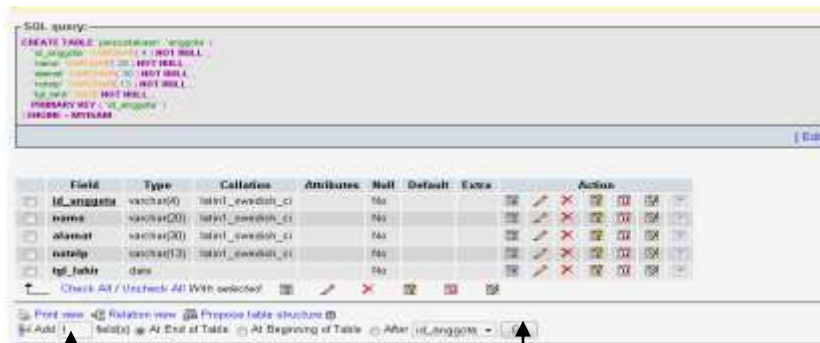


Klik Save

6. Mengelola Tabel di PHPMyadmin

a. Menambah *Field*

- 1) Buka tabel yang ingin diubah fieldnya.
- 2) Klik menu tab **structure**.
- 3) Masukkan jumlah *field* yang ingin ditambah.
- 4) Pilih posisi *field* **at end of table/at beginning of table/after dari field** yang sudah ada.
- 5) Setelah itu klik **Go**



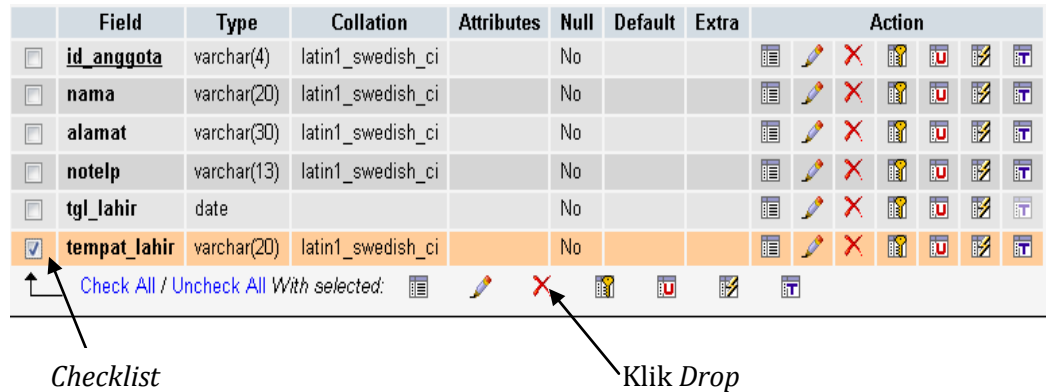
Jumlah *field* yang baru Posisi *field* yang ditambah Klik GO

- 6) Masukan nama *field* yang akan ditambahkan → kemudian klik **save**



b. Menghapus *Field*

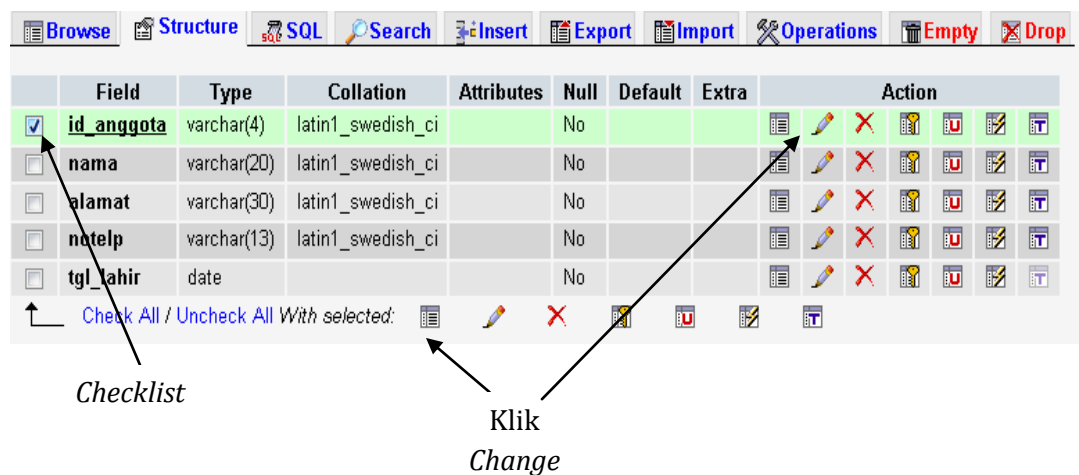
1) *Checklist field* yang ingin dihapus kemudian klik **drop**



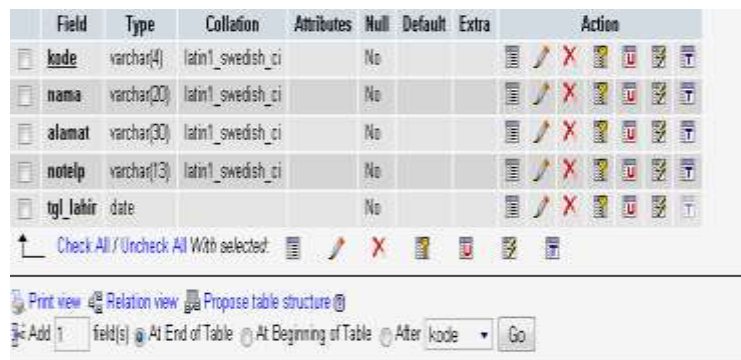
2) Setelah klik **drop** maka akan tampil pertanyaan, pilih **yes**. Kemudian akan tampil pesan sukses.

c. Merubah *Field*

1) *Checklist field* yang ingin dirubah, kemudian klik **change**

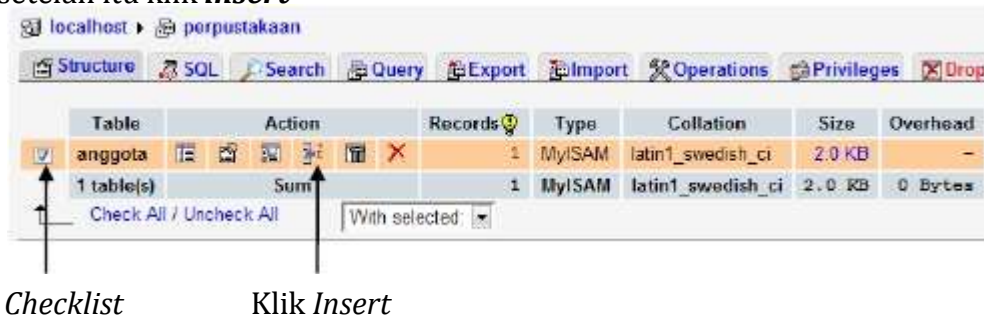


2) Ubahlah nama *field id_anggota* menjadi *field kode*, setelah itu klik **save**



d. Menambah Isi Record Tabel

- 1) Klik database perpustakaan, kemudian *checklist* tabel yang ingin kita isikan data setelah itu klik *insert*



- 2) Maka akan tampil layar, kemudian isikan 2 record, setelah itu klik *Go*.

e. Mencari Data

- 1) Klik *Search* pada menu tab, kemudian masukkan kriteria yang ingin dicari di *value*, kemudian klik *Go*.

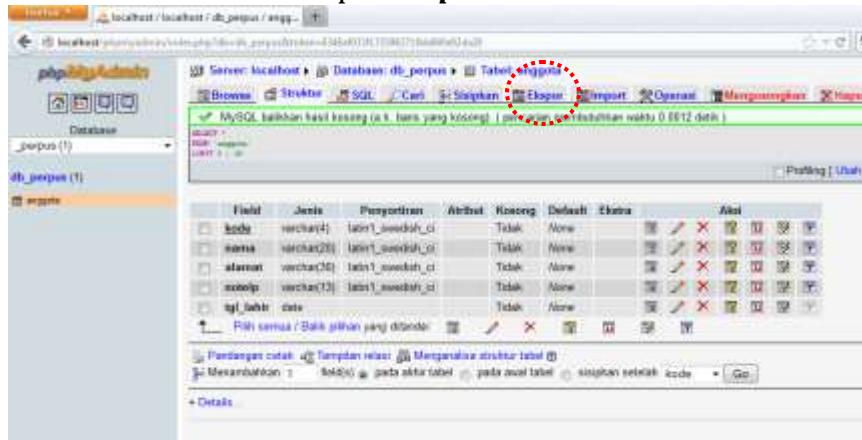


7. Eksport dan Import Database

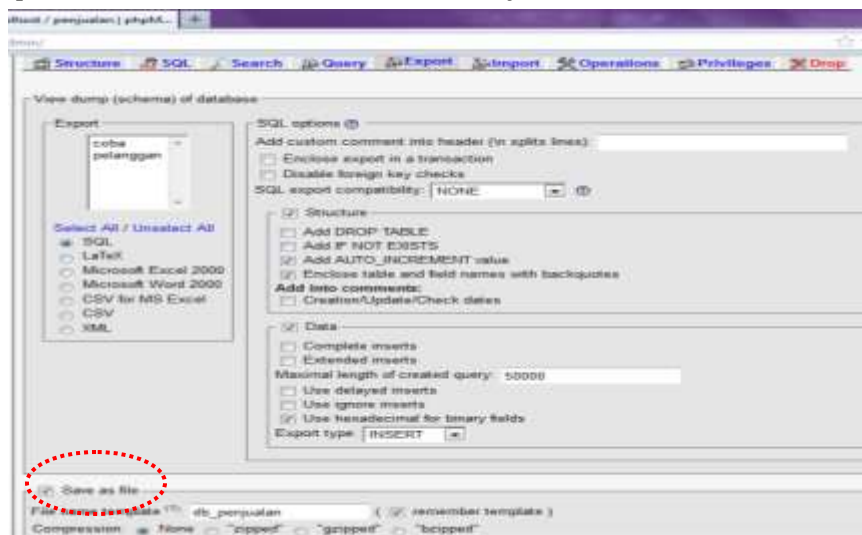
PhpMyadmin memiliki fungsi-fungsi untuk proses pengelolaan database, dimana dengan PhpMyadmin kita bisa menghapus tabel, import database maupun export database.

a. **Eksport Database**

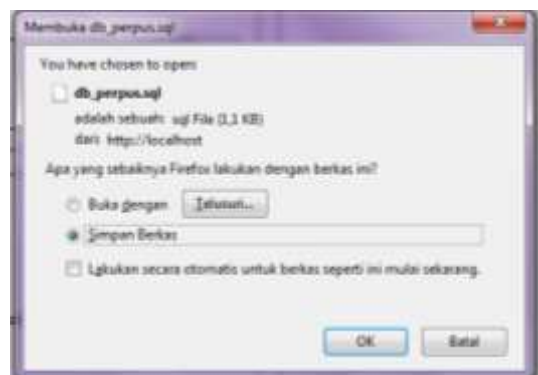
- 1) Masuk kembali ke halaman PhpMyadmin.
- 2) Pilih database yang hendak di-**Export**.
- 3) Pada tab menu database pilih **Export**.



- 4) Setelah itu akan tampil halaman **Export**, lalu pada opsi **Export** pilih SQL dan pada opsi **Save as File** ketikkan nama filenya.

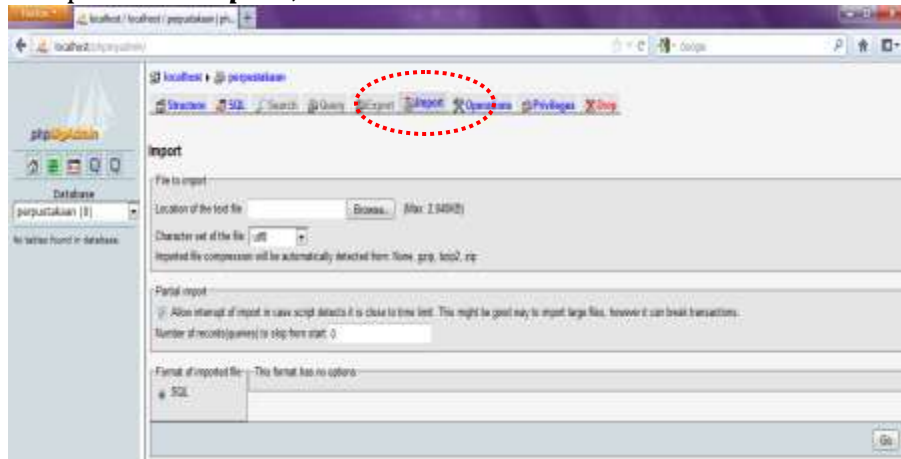


- 5) Lalu klik tombol **Go**.
- 6) Simpan dan download database tersebut. Lama proses download database tergantung besar atau tidaknya database yang tersimpan.



b. **Import Database**

- 1) Buat database baru.
- 2) Masuk kembali ke PhpMyadmin.
- 3) Pilih database yang hendak di **import** (dengan asumsi belum ada tabel didalamnya).
- 4) Klik pada tab **Import**, lalu klik tombol **Browse**.



- 5) Pilih database yang hendak di **import**, lalu klik **Open**.
- 6) Lalu klik tombol **Go**.
- 7) Tunggulah beberapa saat, proses **import** database sedang dilakukan.
- 8) Jika **import** database sudah dilakukan, maka akan ditampilkan tabel-tabelnya.

Latihan PPBD UTS

1. Kerjakalah soal dibawah ini dengan menggunakan perintah SQL di *Command Prompt*.
 - a. Buatlah sebuah database dengan nama “**db_latihan_uts**”.
 - b. Buatlah sebuah tabel dengan nama “**tb_petugas**” dengan struktur dibawah ini:

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Keterangan</i>
kd_petugas	<i>Char</i>	5	<i>Primary Key</i>
nm_petugas	<i>Varchar</i>	25	
alamat	<i>Varchar</i>	50	
telp	<i>Varchar</i>	15	

- c. Isilah tabel “**tb_petugas**” sebanyak 5 *record*.
 - d. Ubahlah nama *field telp* menjadi **telepon**.
 - e. Tambahkan *field jenis_kelamin* setelah nama **nm_petugas**.
 - f. Gantilah nama “**tb_petugas**” menjadi “**tb_user**”.
2. Kerjakalah soal dibawah secara wizard ini di *PhpMyadmin*.
 - a. Buatlah sebuah tabel dengan nama “**tb_buku**” didalam database “**db_latihan_uts**” dengan struktur sebagai berikut:

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Keterangan</i>
kd_buku	<i>Char</i>	5	<i>Primary Key</i>
judul	<i>Varchar</i>	30	
pengarang	<i>Varchar</i>	20	
penerbit	<i>Varchar</i>	30	

- b. Isilah tabel “**tb_buku**” sebanyak 10 *record*.
 - c. Export database “**db_latihan_uts**” dan simpan di C:/LatihanUTS/NIM/db_latihan_uts.sql.

Normalisasi

Pada pembahasan pertama telah dibahas mengenai normalisasi dalam penjualan yang telah diterapkan dengan menggunakan sintak SQL di Command Prompt dan PhpMyadmin. Berikut ini adalah contoh kasus mengenai perpustakaan yang digambarkan dalam normalisasi:

1. Dokumen Dasar

a. Dokumen Masukan

KARTU ANGGOTA PERPUSTAKAAN "ISTANA BUKU"		
No. Anggota :	<input type="text" value="PIB001"/>	Pas Foto 2 x 3
Nama :	<input type="text" value="KHANSA"/>	
Alamat :	<input type="text" value="JL. KEBON JERUK
JAKARTA BARAT"/>	
Telp :	<input type="text" value="021-5657821"/>	

DATA USER PERPUSTAKAAN "ISTANA BUKU"	
ID User :	<input type="text" value="UIB001"/>
Nama :	<input type="text" value="DEARAZ"/>
Alamat :	<input type="text" value="JL. KEBAYORAN LAMA
JAKARTA SELATAN"/>
Telp :	<input type="text" value="021-7326788"/>
Level User :	<input type="text" value="ADMIN"/>

DATA BUKU PERPUSTAKAAN "ISTANA BUKU"	
Kode Buku :	<input type="text" value="BIBKOM001"/>
Judul :	<input type="text" value="PERANCANGAN BASIS DATA"/>
Kategori :	<input type="text" value="KOMPUTER"/>
Penerbit :	<input type="text" value="GRAMEDIA"/>
Pengarang :	<input type="text" value="BINA SARANA INFORMATIKA"/>
Tahun :	<input type="text" value="2013"/>

b. Dokumen keluaran

**STRUK PEMINJAMAN
PERPUSTAKAAN "ISTANA BUKU"**

No. Struk : Tanggal :

No. Anggota : Nama User :

Nama anggota :

Kode Buku	Judul Buku	Jumlah Pinjam
BIBKOM001	PERANCANGAN BASISDATA	1
BIBEK0001	MELIHAT EKONOMI DUNIA	1
BIBKTA001	TUGAS AKHIR "PERANCANGAN WEB"	1

Keterangan:

- Jumlah buku yang dipinjam max 3 pcs.
- Lama peminjaman max selama 3 hari.
- Denda Rp. 2.500 /buku * lama keterlambatan.

**STRUK PENGEMBALIAN
PERPUSTAKAAN "ISTANA BUKU"**

NO. Struk: SIBPGL30704001
08 Juli 2013

No. Struk Pinjam : Tanggal Pinjam :

No. Anggota : Nama User :

Nama anggota :

Kode Buku	Judul Buku	Jumlah Pinjam	Jumlah Kembali
BIBKOM001	PERANCANGAN BASISDATA	1	1
BIBEK0001	MELIHAT EKONOMI DUNIA	1	1
BIBKTA001	TUGAS AKHIR "PERANCANGAN WEB"	1	1

Lama Keterlambatan : Hari

Jumlah Denda yang harus dibayar :

2. Normalisasi

a. Bentuk Tidak Normal (UNF)

No_anggota
 Nama_anggota
 Alamat_anggota
 Telp_anggota
 Id_user
 Nama_user
 Alamat_user
 Telp_user
 Level_user
 Kode_buku
 Judul
 Kategori
 Penerbit
 Pengarang
 Tahun
 No_pinjam
 Tanggal_pinjam
 Nama_user
 No_anggota
 Nama_anggota
 Kode_buku
 Judul
 Jumlah_pinjam
 No_kembali
 Tanggal_kembali
 No_pinjam
 Tanggal_pinjam
 No_anggota
 Nama_anggota
 Nama_user
 Kode_buku
 Judul
 Jumlah_pinjam
 Jumlah_kembali
 keterlambatan
 Jumlah_denda

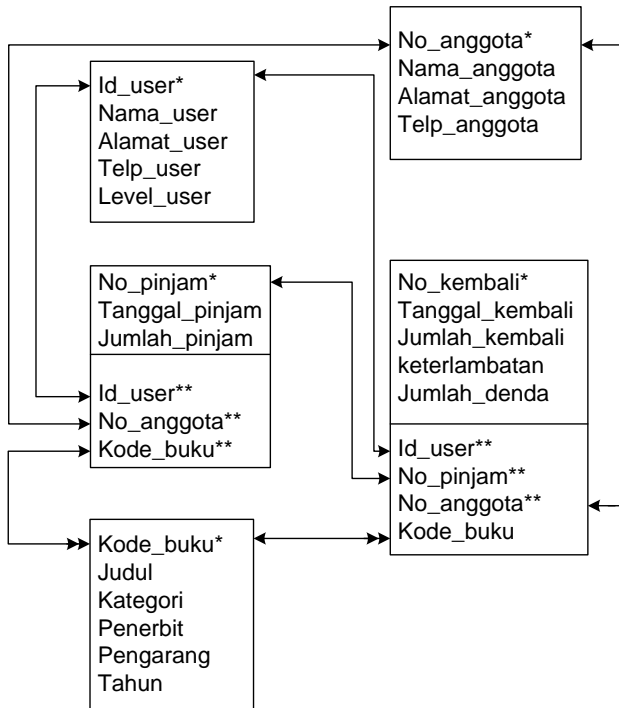
b. Bentuk Normal Pertama (1NF)

No_anggota*
 Nama_anggota
 Alamat_anggota
 Telp_anggota
 Id_user*
 Nama_user
 Alamat_user
 Telp_user
 Level_user
 Kode_buku*
 Judul
 Kategori
 Penerbit
 Pengarang
 Tahun
 No_pinjam*
 Tanggal_pinjam
 Jumlah_pinjam
 No_kembali*
 Tanggal_kembali
 keterlambatan
 Jumlah_denda
 Jumlah_kembali

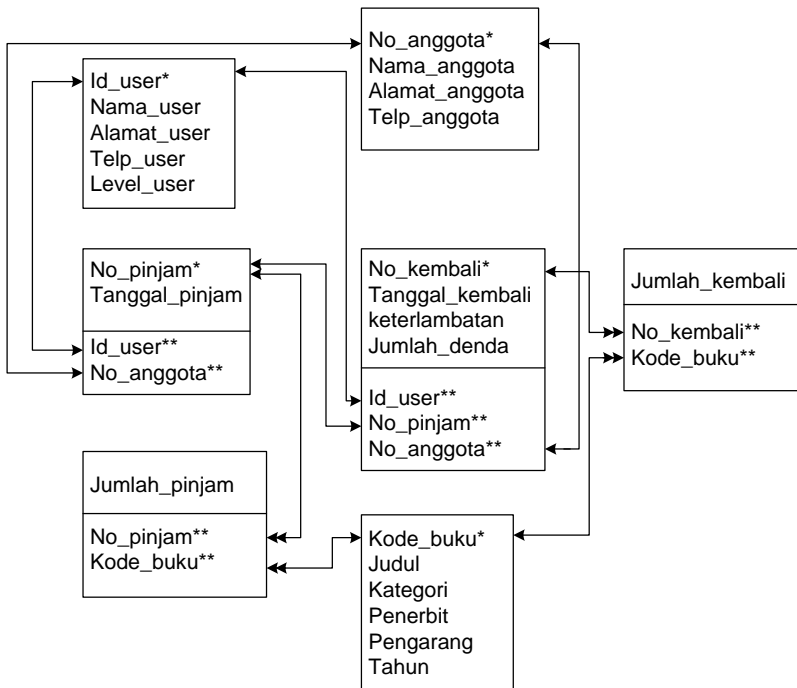
Keterangan :

* = Candidate Key

c. Bentuk Normal Kedua (2NF)



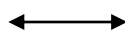
d. Bentuk Normal Ketiga (3NF)



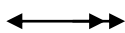
Keterangan:

* : Primary Key

** : Foreign Key



: One to One Relationships



: One to Many Relationships

Database, Tabel dan Relasi

☑ Mengenal Lingkungan Kerja Microsoft Access 2010

Microsoft Access adalah sebuah program aplikasi basis data dengan model relasional. Aplikasi ini menggunakan mesin basis data Microsoft Jet Database Engine, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna. Versi yang akan kita gunakan adalah Microsoft Office Access 2010.

Microsoft Access dapat menggunakan data yang disimpan di dalam format Microsoft Access, Microsoft Jet Database Engine, Microsoft SQL Server, Oracle Database, atau semua container basis data yang mendukung standar ODBC.

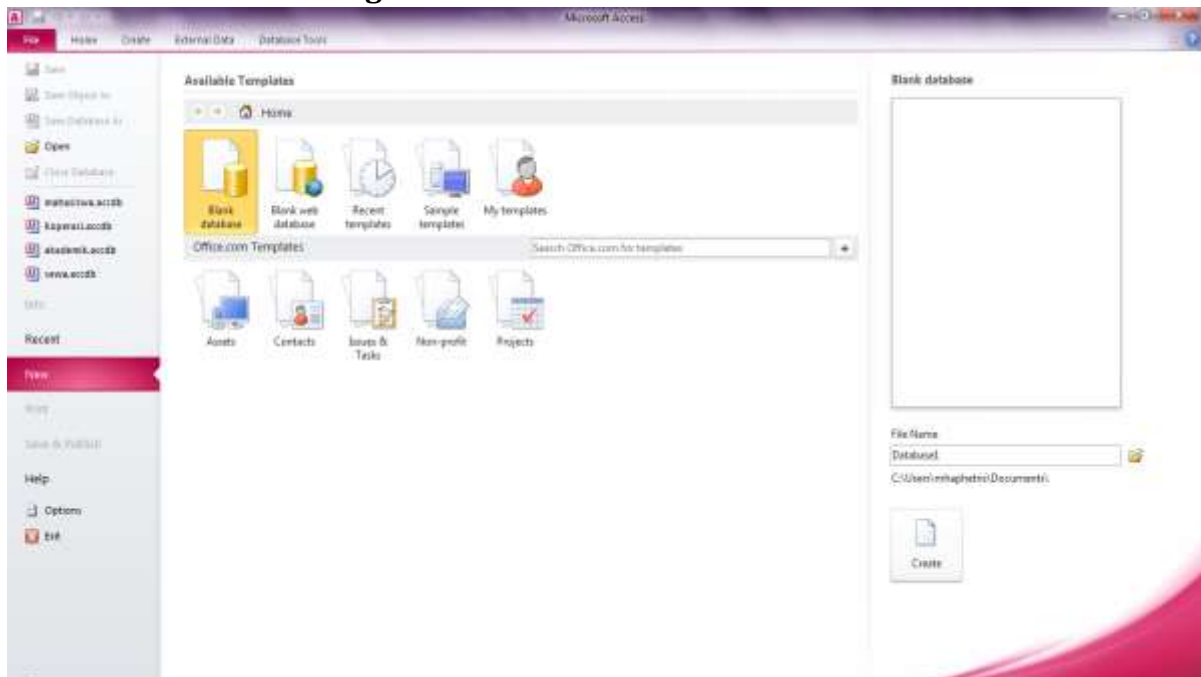
1. Susunan atau Hierarki pada Database

Character → Field → Record → File → Database

☑ Membuat Basis Data di Microsoft Access 2010

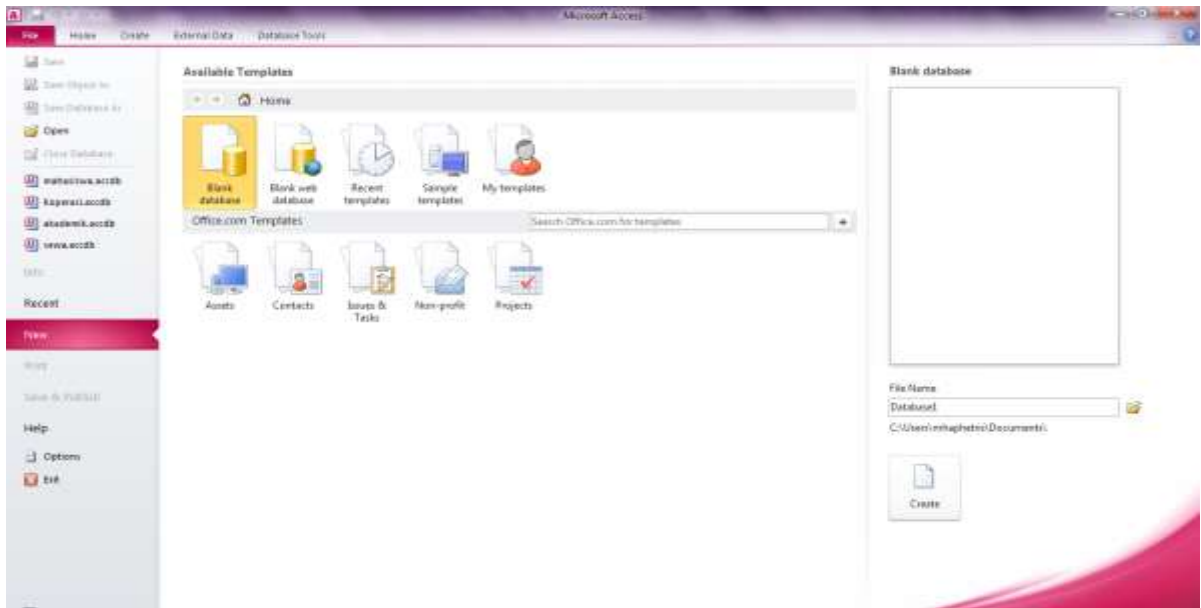
1. Mengaktifkan Ms. Access 2010

Klik menu **Start** → **All Program** → **Microsoft Office** → **Microsoft Access 2010**



2. Membuat Database

Klik **Blank Database** maka akan tampil jendela Blank Database →



→ Isi *File Name* sesuai dengan nama Database → Klik **icon folder** untuk memilih atau mengubah lokasi penyimpanan → lalu Klik **Create** untuk membuat database.

3. Objek yang ada pada Microsoft Acces 2010



a. Tables

Merupakan objek utama dalam sebuah file database sekaligus objek pertama yang harus dibuat karena didalam tabel inilah seluruh data-data yang dibutuhkan disimpan.

b. Query

Merupakan objek yang berfungsi untuk menampilkan, mengubah dan menganalisa data.

c. Form

Merupakan objek yang digunakan untuk menampilkan data yang tersimpan dalam tabel (untuk pembuatan *user interface*).

d. *Reports*

Merupakan objek yang digunakan untuk merangkum dan mencetak data atau suatu laporan secara efektif.

e. *Macros*

Merupakan objek yang digunakan untuk mengoptimasi sekaligus menghemat waktu yang diperlukan dalam pembuatan aplikasi database. Dengan kata lain Macro adalah kumpulan beberapa perintah, yang tersusun dalam sebuah daftar.

f. *Modules*

Merupakan objek yang digunakan untuk menuliskan kode program atau prosedur yang dapat digunakan untuk program aplikasi.

Latihan membuat database:

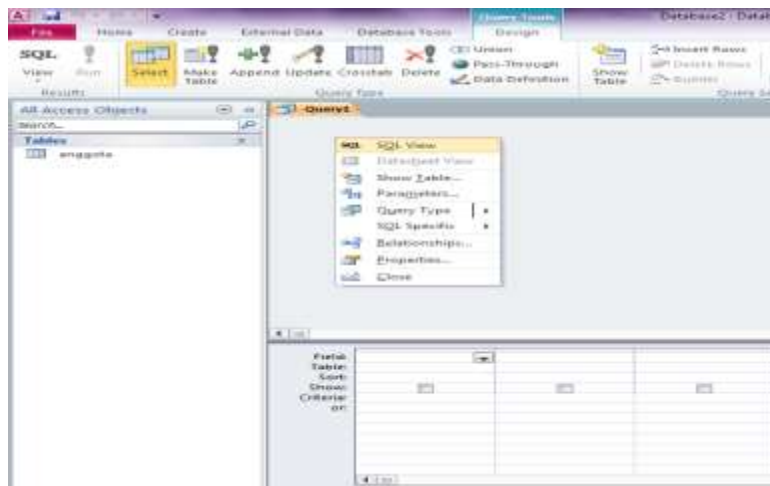
Buatlah sebuah database dengan nama **perpustakaan.accdb**

4. Pembuatan Tabel

Sebelum membuat tabel, kita diharuskan membuat database terlebih dahulu seperti pembahasan sebelumnya. Bukalah database perpustakaan.accdb pada pembahasan sebelumnya.

a. Pembuatan tabel menggunakan sintak SQL (SQL View)

- 1) Klik Tab **Create**.
- 2) Pilih Icon **Query Design**, maka akan muncul jendela **Show Tables** → Klik **Close**.
- 3) Klik kanan lalu pilih **SQL View**.



- 4) Ketikkan sintak untuk membuat tabel dengan melihat struktur dibawah ini.

Tabel Anggota

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Keterangan</i>
no_anggota	<i>Text</i>	6	Primary Key
nama_anggota	<i>Text</i>	30	
alamat_anggota	<i>Text</i>	50	
telp_anggota	<i>Text</i>	12	

Sintak SQL:

```
create table anggota (no_anggota text(6) primary key, nama_anggota text(30),  
alamat_anggota text(50), telp_anggota text(12));
```

- 5) Lalu isi tabel anggota sebanyak 5 record dengan menggunakan SQL View.

Sintak SQL:

```
insert into anggota values ("PIB001","Khansa","Jl. Kebon Jeruk Jakarta  
Barat","021-5657821");
```

Latihan membuat tabel dengan SQL View

Buka kembali database perpustakaan.accdb dan tambahkan 2 buat tabel berdasarkan struktur dibawah ini:

1. **Tabel User** (diisikan sebanyak 3 record)

Field Name	Type	Size	Keterangan
id_user	<i>Text</i>	6	Primary Key
nama_user	<i>Text</i>	30	
alamat_user	<i>Text</i>	50	
telp_user	<i>Text</i>	12	
level_user	<i>Text</i>	5	

2. **Tabel Buku** (diisikan sebanyak 3 record)

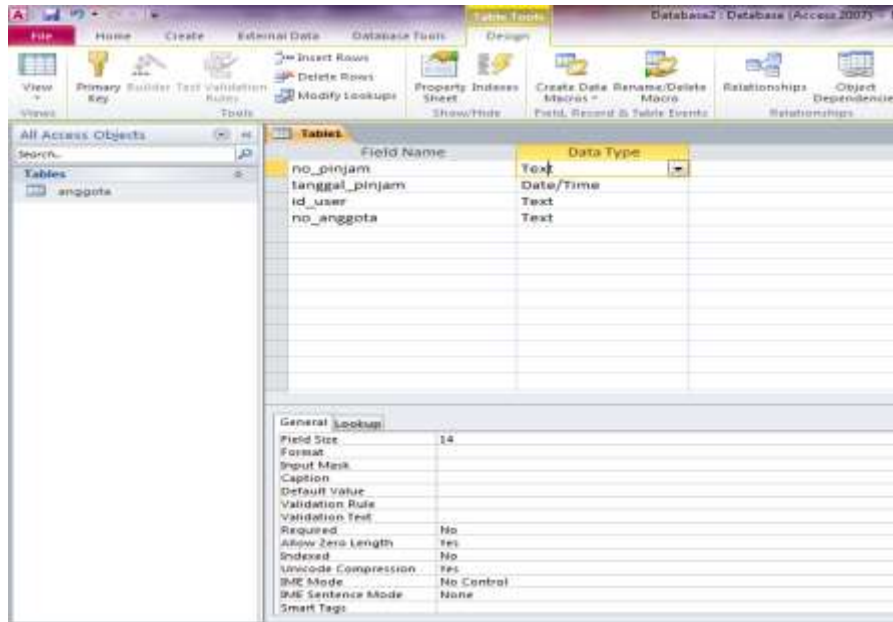
Field Name	Type	Size	Keterangan
kode_buku	<i>Text</i>	9	Primary Key
judul	<i>Text</i>	50	
kategori	<i>Text</i>	20	
penerbit	<i>Text</i>	30	
pengarang	<i>Text</i>	30	
tahun	<i>Text</i>	4	

b. Pembuatan tabel menggunakan **Table Design**

- 1) Klik Menu **Create** Pilih **Table Design**
- 2) Buat tabel sesuai dengan strukturnya.

Tabel Pinjam

Field Name	Type	Size	Keterangan
No_pinjam	<i>Text</i>	14	Primary Key
Tanggal_pinjam	<i>Date/Time</i>		
Id_user	<i>Text</i>	6	<i>Combo Box: tabel user</i>
No_anggota	<i>Text</i>	6	<i>Combo Box: tabel anggota</i>



3) Pembuatan *Combo Box*

Klik Field yang ingin dibuatkan *combo box* → Klik tab **Lookup** pada **Field Properties** → Lalu ubah setingan menjadi seperti dibawah ini:

Display Control : *Combo Box*

Row Source Type : *Table/Query*

Row Source : *anggota*

General	Lookup
Display Control	Combo Box
Row Source Type	Table/Query
Row Source	anggota
Bound Column	1
Column Count	1
Column Heads	No
Column Widths	
List Rows	16
List Width	Auto
Limit To List	No
Allow Multiple Values	No
Allow Value List Edits	Yes
List Items Edit Form	
Show Only Row Source	No

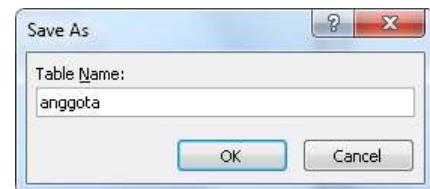
4) Membuat *Primary Key*

Klik Field yang ingin dijadikan primary key → Klik kanan → Pilih **Primary Key** atau klik icon



5) Menyimpan Tabel.

Klik Menu **File** → Klik **Save** → Ketikkan nama tabel yang akan disimpan → **Ok**



6) Menyisipkan record.

Double klik pada tabel yang akan disisipkan *record*, atau klik icon



Datasheet View

Kemudian isikan *record* pada semua tabel yang sudah dibuat.

Latihan membuat tabel dengan *Table Design*

Buka kembali database perpustakaan.accdb dan tambahkan 3 buat tabel berdasarkan struktur dibawah ini:

1. Tabel Detail_pinjam

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Keterangan</i>
jumlah_pinjam	<i>Number</i>		
no_pinjam	<i>Text</i>	14	<i>Combo Box: tabel pinjam</i>
kode_buku	<i>Text</i>	9	<i>Combo Box: tabel buku</i>

2. Tabel Detail_kembali

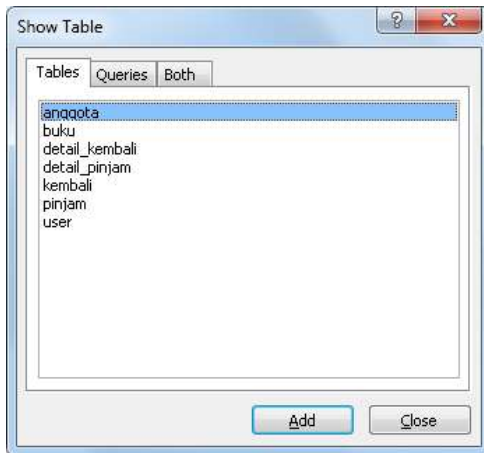
<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Keterangan</i>
jumlah_kembali	<i>Number</i>	1	
no_kembali	<i>Text</i>	14	<i>Combo Box: tabel kembali</i>
kode_buku	<i>Text</i>	9	<i>Combo Box: tabel buku</i>

3. Tabel Kembali

<i>Field Name</i>	<i>Type</i>	<i>Size</i>	<i>Keterangan</i>
no_kembali	<i>Text</i>	14	<i>Primary Key</i>
tanggal_kembali	<i>Date/Time</i>		
keterlambatan	<i>Number</i>		
jumlah_denda	<i>Currency</i>		
id_user	<i>Text</i>	6	<i>Combo Box: tabel user</i>
no_pinjam	<i>Text</i>	14	<i>Combo Box: tabel pinjam</i>
no_anggota	<i>Text</i>	6	<i>Combo Box: tabel anggota</i>

5. Membuat *Relationship*

Untuk membuat *relationship*, buka salah satu tabel dalam keadaan *design* → lalu klik **icon Relationship**, maka akan tampil jendela **Show Table** →

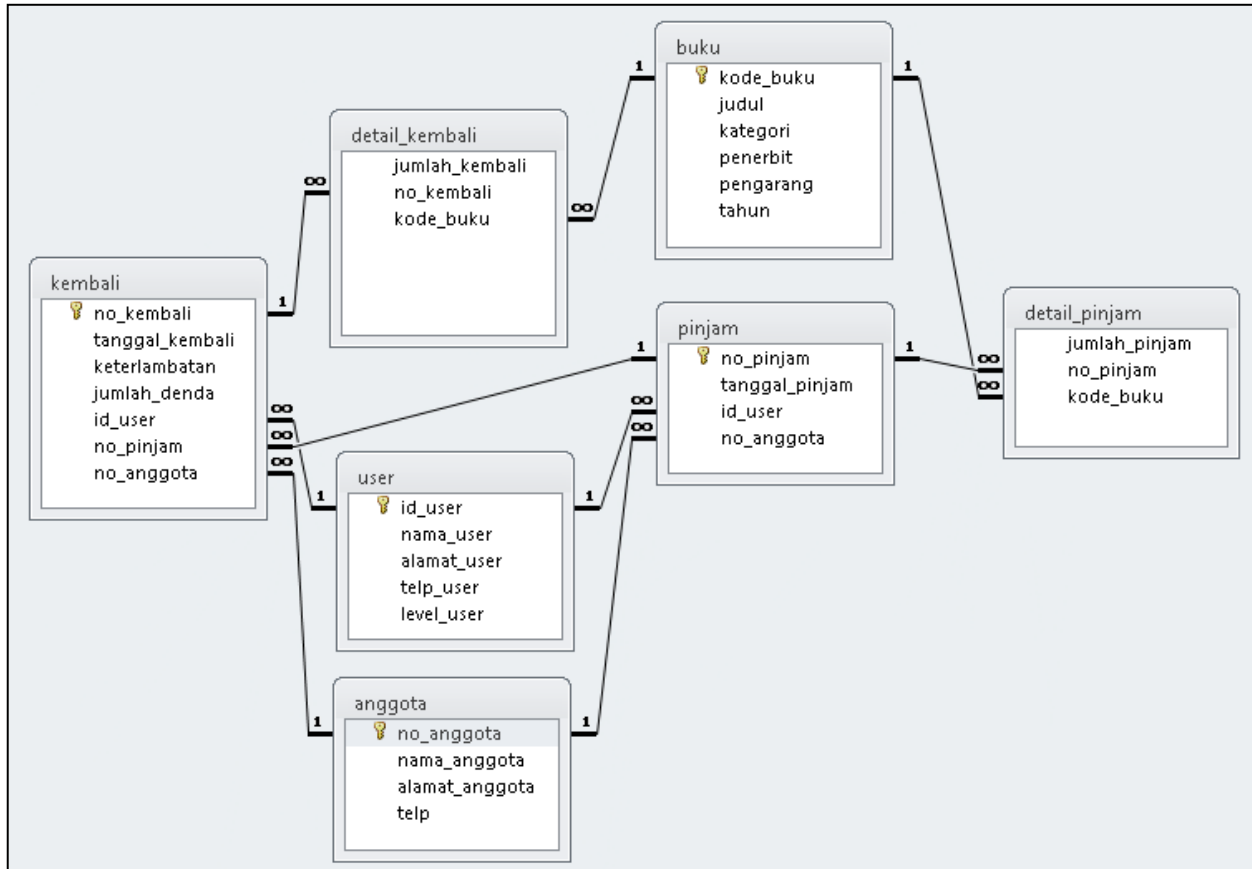


→ Pilih tabel satu persatu

Untuk membuat relasi antar tabel, tarik garis diagonal dari *field* *no_anggota* (*primary key*) dari tabel **anggota** ke *field* *anggota* di tabel **pinjam**, hingga tampil jendela **Edit Relationship** seperti berikut ini:



Lakukan hal yang sama untuk *field* lainnya hingga menjadi sebuah relasi seperti berikut:

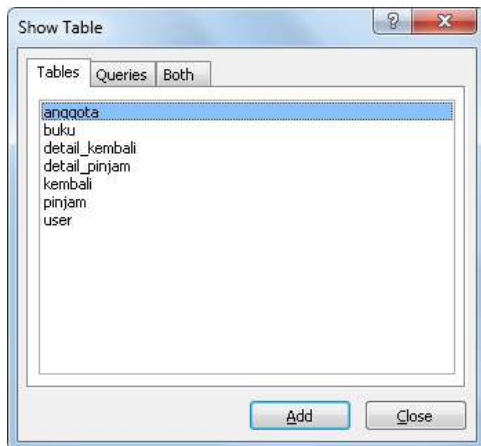


Data Definition Query

☑ Query

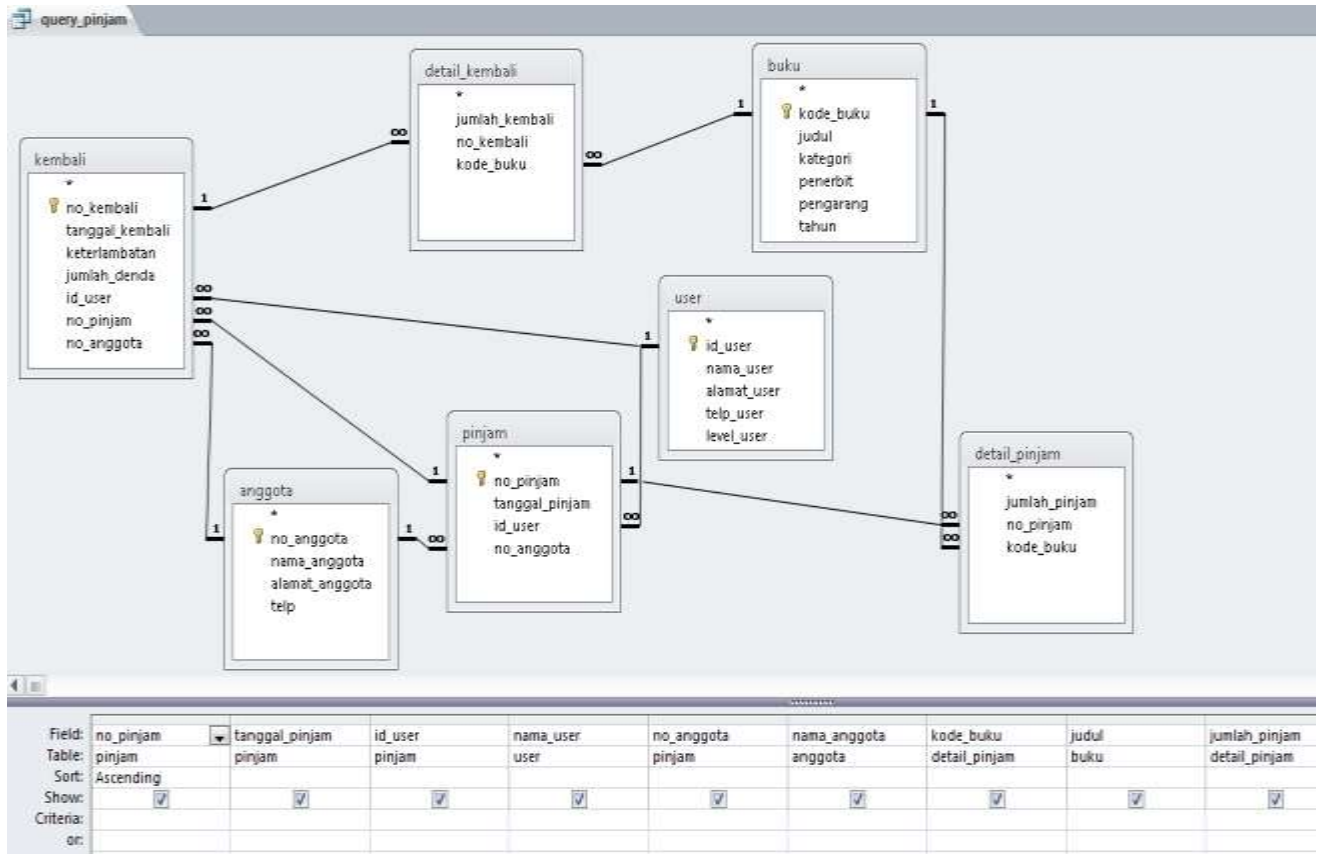
Query adalah suatu objek database yang dapat digunakan untuk menampilkan, menyunting atau menganalisa suatu data dengan memberikan baris-baris perintah tertentu. Pada aplikasi ini sudah dilengkapi dengan *Structured Query Language (SQL)* yang menjadi standard bahasa dalam mengolah database. Dengan *SQL* kita dapat mempersingkat penulisan listing program yang panjang karena pada setiap perintahnya dapat mencakup banyak perintah sekaligus. Langkah-langkah membuat *query*:

1. Klik Tab **Create**
2. Pilih icon **Query Design**, muncul jendela **Show Tables**.
3. Pilih nama tabel kemudian Klik **Add**, contohnya kita pilih tabel **Barang**.



3. *Double* klik *field* yang ingin dijadikan *query* seperti gambar dibawah ini:

a. Membuat **query_pinjam**



Keterangan

- Field** : Nama *field* yang ingin ditampilkan
- Table** : Nama tabel dari *field* yang ditampilkan
- Sort** : Mengurutkan data hasil *query*
- Show** : Mengatur *field* ditampilkan atau tidak
- Criteria** : Syarat dari data yang ingin ditampilkan

Jika ingin menggunakan semua *field* pada tabel maka kita hanya klik dan drop tanda * pada tabel barang ke baris *field*. Jika tidak kita dapat memilih salah satu *field* dengan cara mendouble klik *field* yang diinginkan.

Terdapat 3 pilihan pada *toolbars View*, yaitu :

1. **Design View** : untuk menampilkan desain *tabel query*.
2. **Datasheet View** : untuk menampilkan data hasil *query*.
3. **SQL View** : untuk membuat *query* dengan *script SQL*.

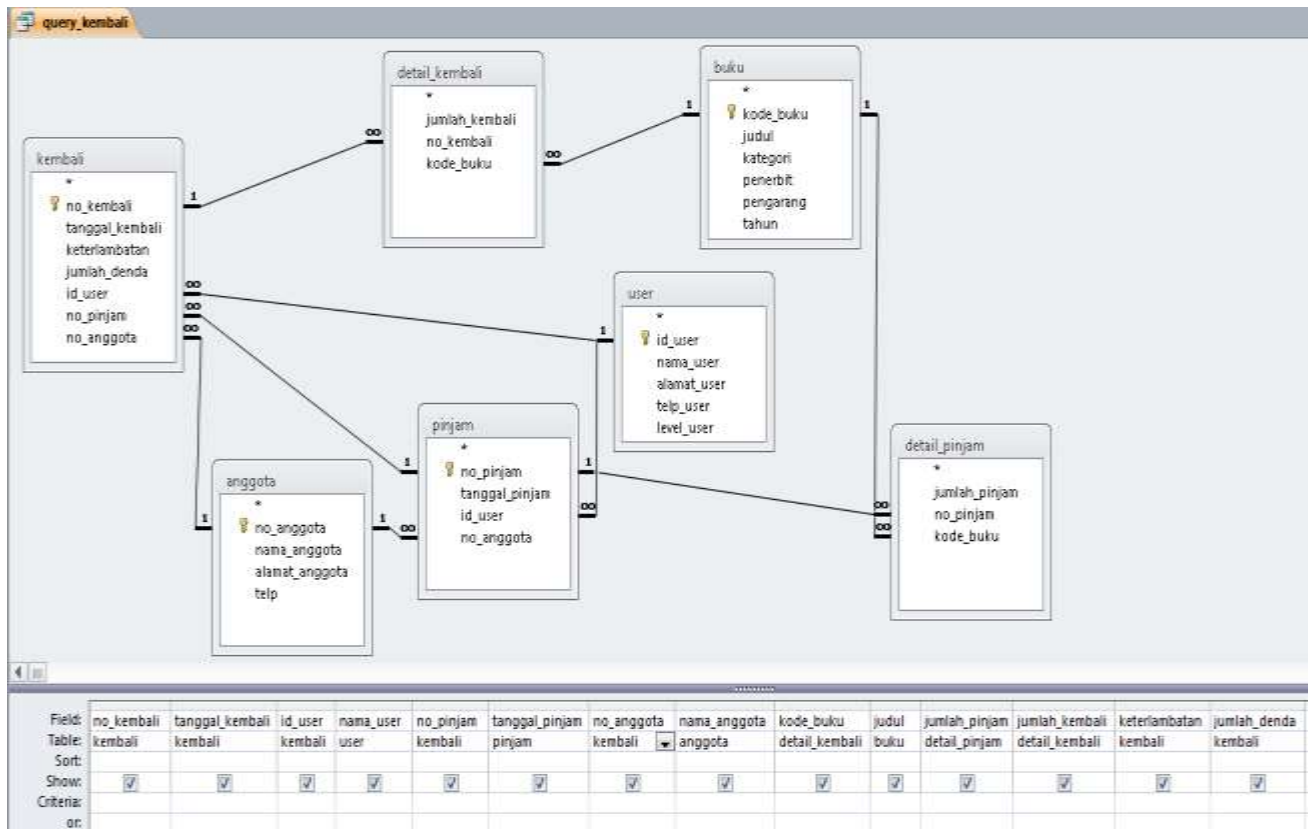
Jika kita sudah membuat desain *query* menggunakan *design view*, kita bisa melihat langsung script SQLnya pada *SQL View* dengan cara memilih icon **SQL SQL View**.

```
SELECT pinjam.no_pinjam, pinjam.tanggal_pinjam, pinjam.id_user, user.nama_user,
pinjam.no_anggota, anggota.nama_anggota, detail_pinjam.kode_buku, buku.judul,
detail_pinjam.jumlah_pinjam FROM [user] INNER JOIN (((anggota INNER JOIN kembali
ON anggota.no_anggota = kembali.no_anggota) INNER JOIN pinjam ON
(pinjam.no_pinjam = kembali.no_pinjam) AND (anggota.no_anggota =
pinjam.no_anggota)) INNER JOIN ((buku INNER JOIN detail_kembali ON buku.kode_buku
= detail_kembali.kode_buku) INNER JOIN detail_pinjam ON buku.kode_buku =
detail_pinjam.kode_buku) ON (pinjam.no_pinjam = detail_pinjam.no_pinjam) AND
(kembali.no_kembali = detail_kembali.no_kembali)) ON (user.id_user = pinjam.id_user)
AND (user.id_user = kembali.id_user) ORDER BY pinjam.no_pinjam;
```

b. Membuat **query_kembali**.

Untuk membuat *query_kembali*, kita gunakan cara sebaliknya yaitu dengan mengetikkan sintak SQL berikut pada *SQL View* dengan cara memilih icon **SQL SQL View**

```
SELECT kembali.no_kembali, kembali.tanggal_kembali, kembali.id_user, user.nama_user,
kembali.no_pinjam, pinjam.tanggal_pinjam, kembali.no_anggota, anggota.nama_anggota,
detail_kembali.kode_buku, buku.judul, detail_pinjam.jumlah_pinjam,
detail_kembali.jumlah_kembali, kembali.keterlambatan, kembali.jumlah_denda FROM
[user] INNER JOIN (((anggota INNER JOIN kembali ON anggota.no_anggota =
kembali.no_anggota) INNER JOIN pinjam ON (pinjam.no_pinjam = kembali.no_pinjam)
AND (anggota.no_anggota = pinjam.no_anggota)) INNER JOIN ((buku INNER JOIN
detail_kembali ON buku.kode_buku = detail_kembali.kode_buku) INNER JOIN
detail_pinjam ON buku.kode_buku = detail_pinjam.kode_buku) ON (pinjam.no_pinjam =
detail_pinjam.no_pinjam) AND (kembali.no_kembali = detail_kembali.no_kembali)) ON
(user.id_user = pinjam.id_user) AND (user.id_user = kembali.id_user);
```



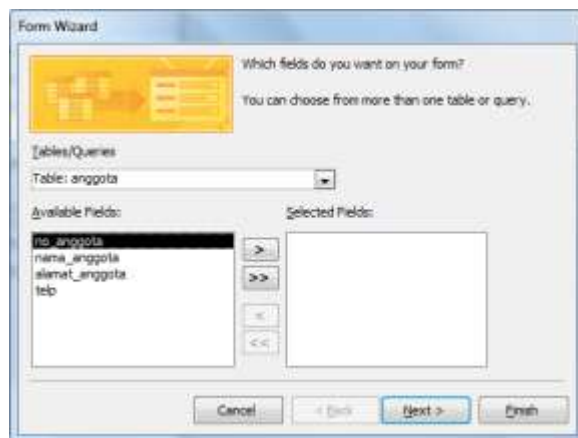
Form dan Navigasi

Merupakan objek yang digunakan untuk menampilkan data yang tersimpan dalam tabel (untuk pembuatan *user interface*) dengan tampilan form.

☑ Membuat Form Data Anggota dengan Form Wizard

Berdasarkan database perpustakaan yang telah dibuat, selanjutnya akan kita membuat Form Data Anggota dengan Form Wizard. Adapun langkah-langkah untuk membuat form yaitu:

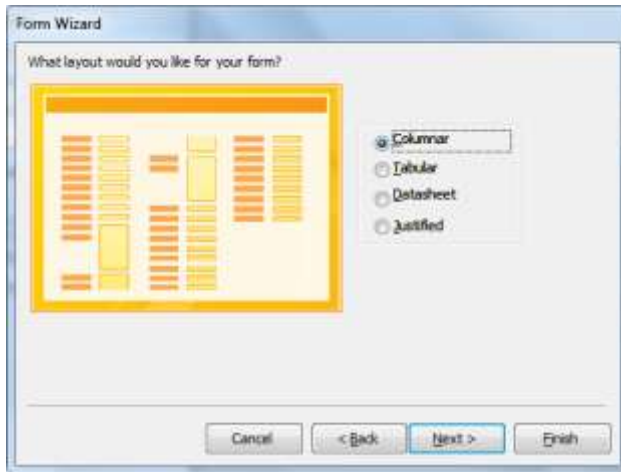
1. Klik Tab **Create** → Pilih **Form Wizard**  Form Wizard



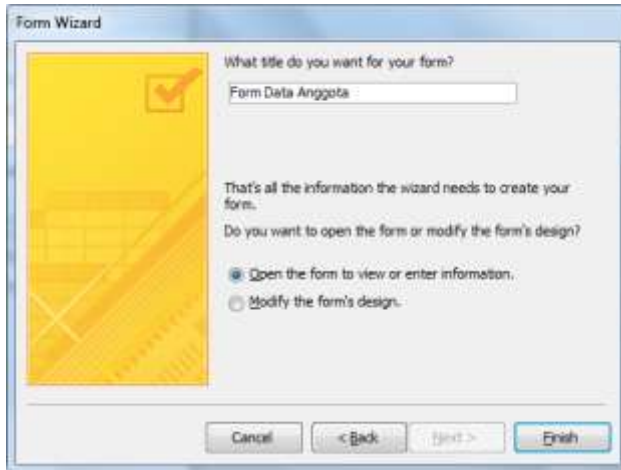
2. Pindahkan *Fields* pada *Available Fields* ke bagian *Selected Fields* → **Next**



3. Pilih *Layout* yang diinginkan untuk form yang akan dibuat → **Next**



4. Ketikkan judul dari form yang akan dibuat, lalu **FINISH**.



5. Maka akan tampil seperti tampilan dibawah ini:

Form Data Anggota

no_anggota	<input type="text" value="PIB001"/>
nama_anggota	<input type="text" value="KHANSA"/>
alamat_anggota	<input type="text" value="JL. KEBON JERUK JAKARTA BARAT"/>
telp	<input type="text" value="021-5657821"/>

6. Klik kanan pada tampilan form → pilih *Design View*, lalu design form menjadi tampilan seperti berikut ini:

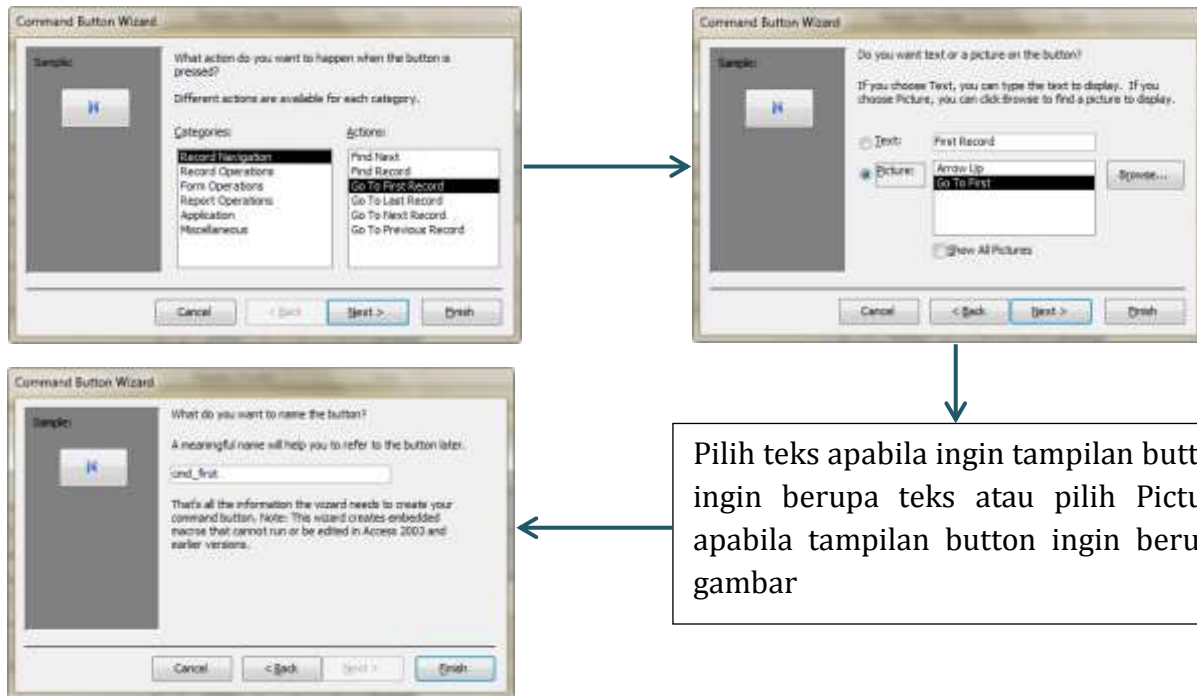
Pembuatan Navigasi

1. Pembuatan *Command Button* menggunakan *ToolBox* dengan mengaktifkan **Control Wizard** pada *menu Design*



<i>Command Button</i>	<i>Categories</i>	<i>Action</i>
<i>First</i>	<i>Record Navigation</i>	<i>Go To First Record</i>
<i>Previous</i>	<i>Record Navigation</i>	<i>Go To Previous Record</i>
<i>Next</i>	<i>Record Navigation</i>	<i>Go To Next Record</i>
<i>Last</i>	<i>Record Navigation</i>	<i>Go To last Record</i>
Tambah	<i>Record Operations</i>	<i>Add New Record</i>
Simpan	<i>Record Operations</i>	<i>Save Record</i>
Hapus	<i>Record Operations</i>	<i>Undo Record</i>
Keluar	<i>Form Operations</i>	<i>Close Form</i>
Cetak Laporan	<i>Report Operations</i>	<i>Preview Report</i>

2. Perbesar area form detail untuk menempatkan *Button*, ambil *Button* dan buatlah *button* tersebut pada area kosong, tampil *command button wizard* Berikut:



☑ **Membuat Tambahan Message Box dan Setfocus pada objek (textfield).**

Klik kanan pada button, pilih **Build Event**, akan tampil **Macro Builder** yang sudah ada pada tombol bersangkutan, untuk menambahkan pesan/messagebox pada

pilihan  pilih **Message**, dan isikan data sbb:

1. Tombol First

GoToRecord

Object Type

Object Name

Record First

Offset

MessageBox

Message Sudah Di Awal Record

Beep Yes

Type Information

Title Informasi

2. Tombol Previous

OnError

Go to Next

Macro Name

GoToRecord

Object Type

Object Name

Record Previous

Offset

If [MacroError]<>0 **Then**

MessageBox

Message Sudah Di Awal Record

Beep Yes

Type Information

Title Informasi

End If

3. Tombol Next

OnError

Go to Next

Macro Name

GoToRecord

Object Type

Object Name

Record Next

Offset

If [MacroError]<>0 **Then**

MessageBox

Message Sudah Di Akhir Record

Beep Yes

Type Information

Title Informasi

End If

4. Tombol Last

GoToRecord

Object Type

Object Name

Record Last

Offset

MessageBox

Message Sudah Di Akhir Record

Beep Yes

Type Information

Title Informasi

5. Tombol Tambah

```

OnError
    Go to Next
    Macro Name
GoToRecord
    Object Type
    Object Name
    Record New
    Offset
GoToControl
    Control Name no_anggota

☐ If [MacroError]<>0 Then
    MessageBox
        Message =[MacroError],[Description]
        Beep Yes
        Type None
        Title
End If
    
```

7. Tombol Simpan

```

OnError
    Go to Next
    Macro Name
☐ RunMenuCommand
    Command SaveRecord
MessageBox
    Message Data Tersimpan
    Beep Yes
    Type Critical
    Title Informasi

☐ If [MacroError]<>0 Then
    MessageBox
        Message =[MacroError],[Description]
        Beep Yes
        Type None
        Title
End If
    
```

6. Tombol Batal

```

OnError
    Go to Next
    Macro Name
RunMenuCommand
    Command UndoRecord
    
```

8. Tombol Hapus

```

OnError
    Go to Next
    Macro Name
GoToControl
    Control Name =[Screen],[PreviousControl],[Name]
ClearMacroError

☐ If Not [Form],[NewRecord] Then
    RunMenuCommand
        Command DeleteRecord
    MessageBox
        Message Data Telah Terhapus
        Beep Yes
        Type Critical
        Title Informasi
End If

☐ If [Form],[NewRecord] And Not [Form],[Dirty] Then
    Beep
End If

☐ If [Form],[NewRecord] And [Form],[Dirty] Then
    RunMenuCommand
        Command UndoRecord
End If
    
```

9. Tombol Keluar

CloseWindow

Object Type

Object Name

Save Prompt

MessageBox

Message Tutup Form

Beep Yes

Type Warning!

Title Informasi

If [MacroError]<>0 **Then**

MessageBox

Message =[MacroError],[Description]

Beep Yes

Type None


Title

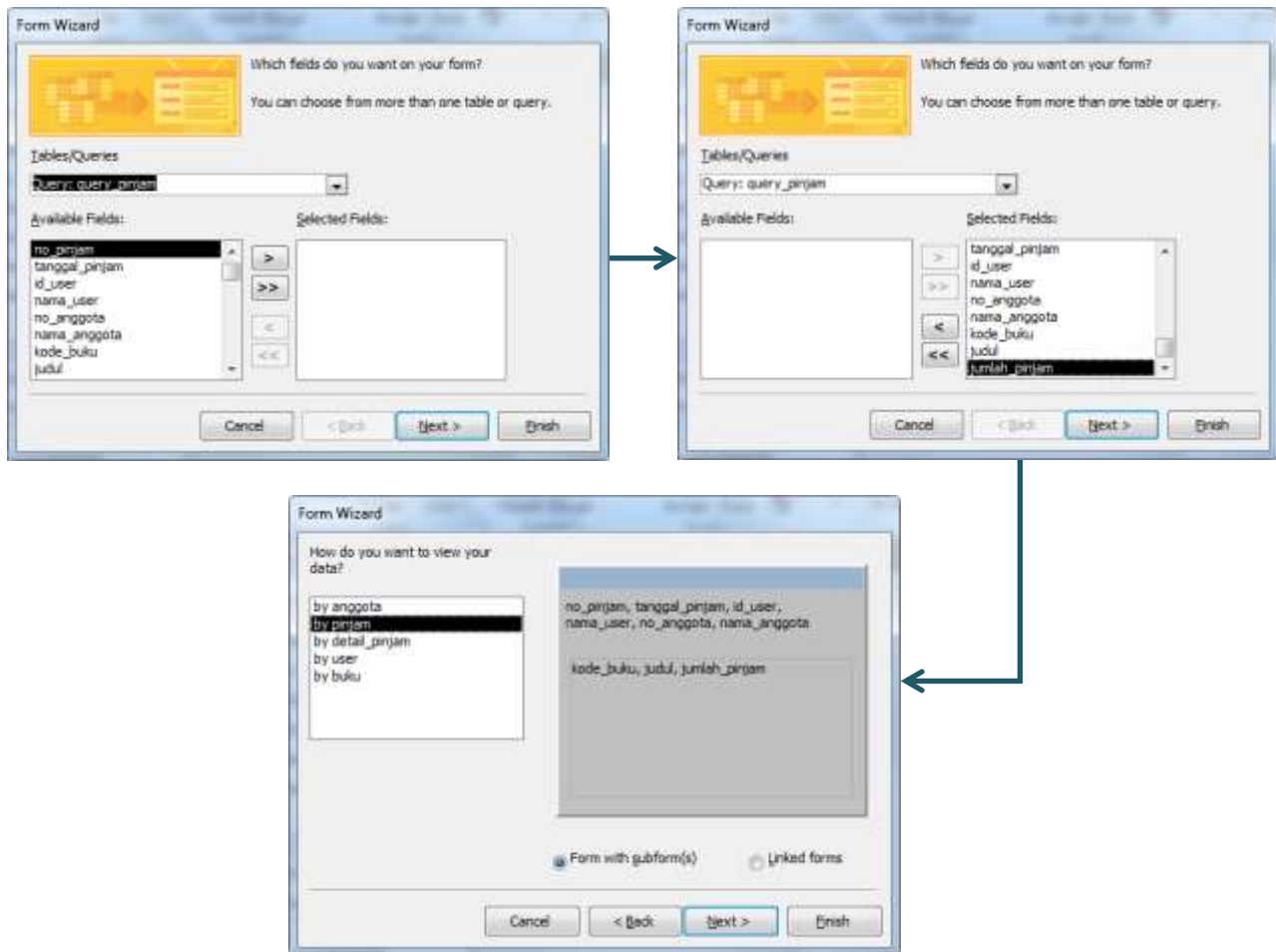
End If

Form dan Subform

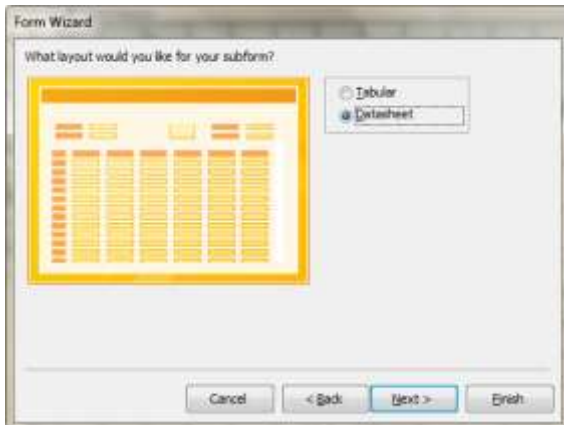
☑ Membuat Form Peminjaman

Adapun langkah-langkah untuk membuat form peminjaman yaitu:

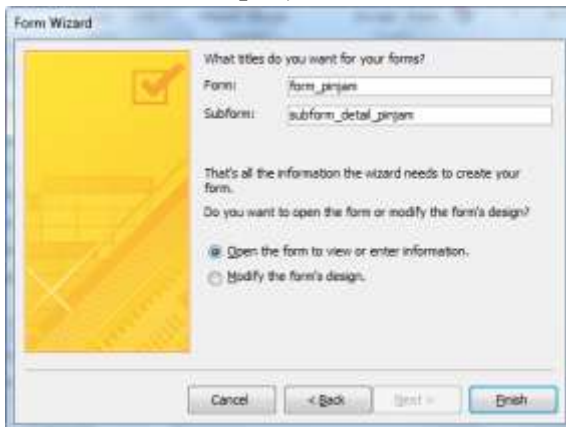
1. Klik Tab **Create** → Pilih **Form Wizard**  Form Wizard
2. Pada combo box Tables/Query pilih **query_pinjam** dan pindahkan semua Fields pada Available Fields ke bagian Selected Fields → **Next**



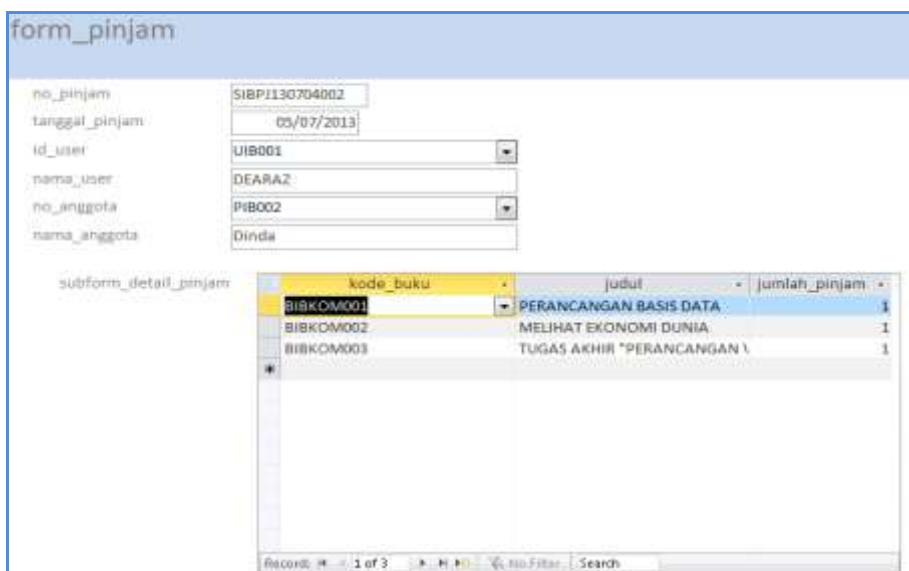
3. Pilih Layout yang diinginkan untuk form yang akan dibuat → **Next**



4. Ketikkan Judul dari form dan subform yang akan dibuat, **form_pinjam** dan **subform_detail_pinjam**, lalu **FINISH**.



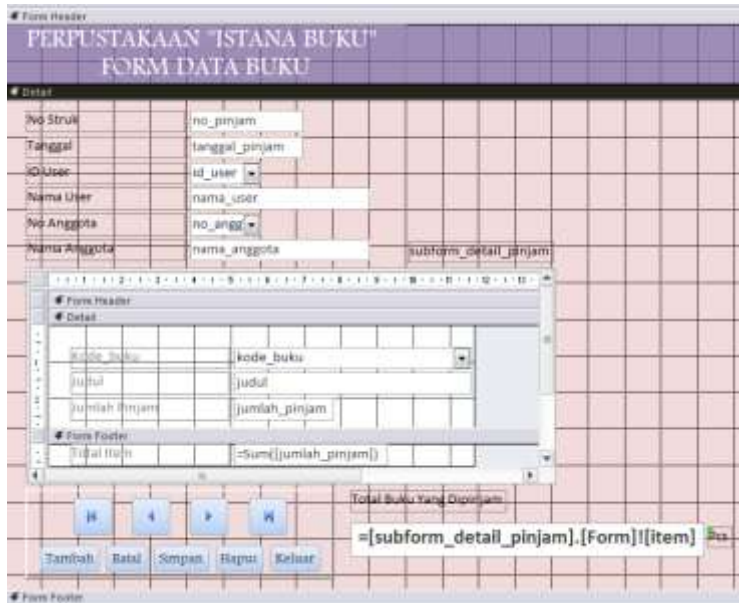
5. Maka akan tampil seperti tampilan dibawah ini:



☑ **Mengatur Tampilan Design Form Peminjaman dan Subformnya.**

Buka form peminjaman yang sudah dibuat, lalu masuk kemenu Design View  Design View

atur design form seperti contoh :



Hasil akhirnya sebagai berikut:



Jangan lupa menambahkan *message* pada *button* dan pengaturan *setfocus* pada objek


Latihan Membuat Form Pengembalian

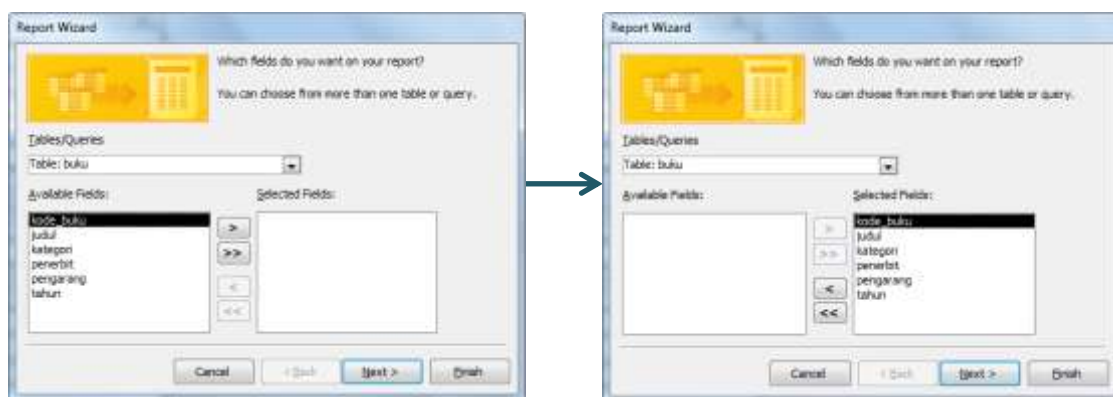
Buatlah form pengembalian buku, untuk tampilan/design form sesuai kan dengan tampilan dokumen keluaran di pertemuan sebelumnya. Langkah-langkah untuk membuat form pengembalian sama dengan langkah yang kita gunakan saat membuat form peminjaman. Form pengembalian ini dibuat menggunakan form wizard yang data-datanya diambil dari **query_kembali**. Simpan form dengan nama : form_kembali, dan subform dengan nama: subform_detail_kembali.

☑ Pembuatan Report

Report dapat diartikan dengan sebuah kata: ‘laporan’. Report merupakan salah satu fungsi pada Microsoft Access yang berguna untuk membantu kita membuat laporan mengenai data – data yang ada dalam database kita. Sebagai contoh, kita dapat menggunakan report untuk membuat laporan transaksi pada sebuah perpustakaan (peminjaman dan pengembalian), laporan data anggota perpustakaan, kartu anggota, laporan data user, ID Card user dan laporan data buku.

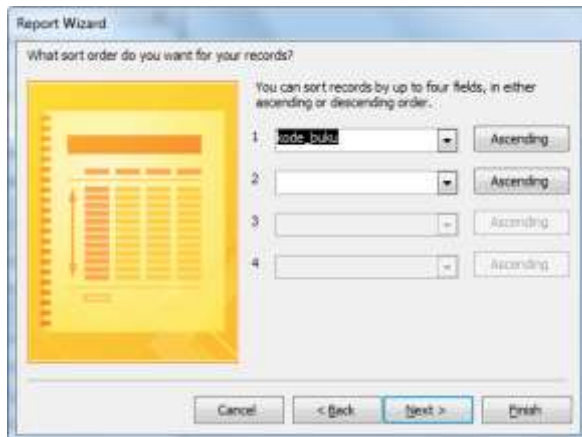
Berikut langkah Pembuatan Report “Laporan Data Buku”:

1. Buka Ms. Access dan database koperasi.accdb yang sebelumnya telah dibuat,
2. Kemudian buka tabel barang,
3. Pilih Menu Create,  Report Wizard
4. Pada report wizard pilih nama tabel dan pindahkan semua available fields ke bagian selected fields

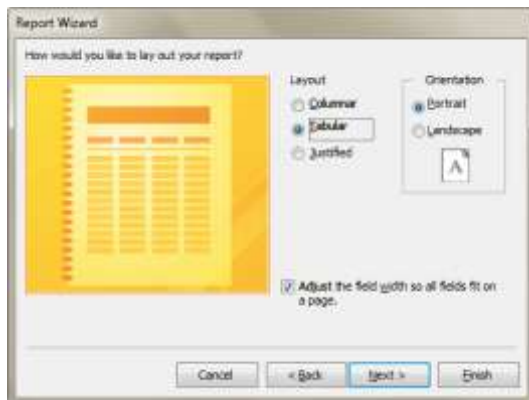


5. Klik next pada pilihan **grouping**

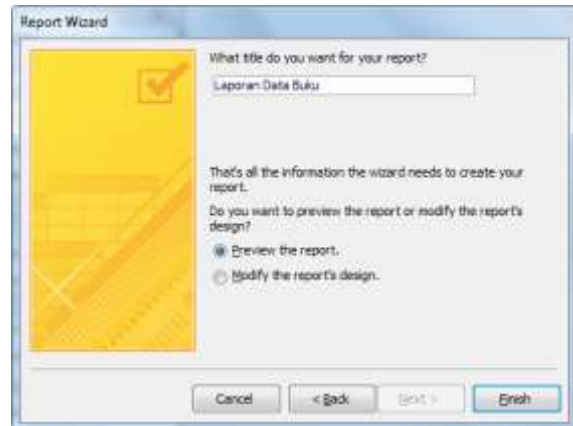
6. Pada bagian sort record pilih **kdbuku**, Next



7. Pilih Layout, Next



8. Ketik Judul Laporan, finish.



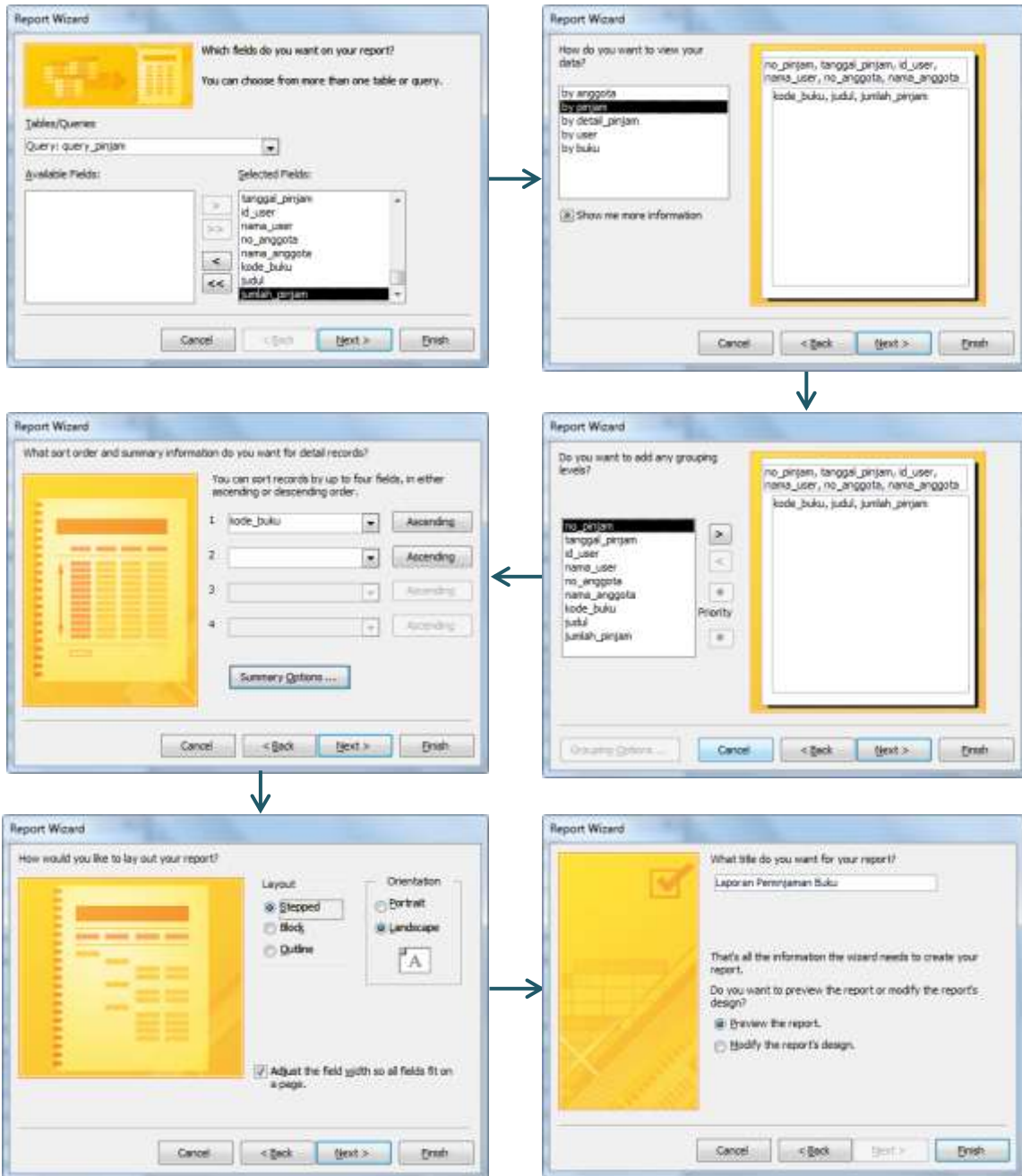
8. Tampilan Hasil Report

PERPUSTAKAAN "ISTANA BUKU"					
LAPORAN DATA BUKU					
Kode Buku	Judul	Kategori	Penerbit	Pengarang	Tahun
BIBKOM001	PERANCANGAN BASIS DATA	KOMPUTER	GRAMEDIA	BSI	2013
BIBKOM002	MELIHAT EKONOMI DUNIA	EKONOMI	ANDI OFFSET	ANDI	2010
BIBKOM003	TUGAS AKHIR "PERANCANGAN WEB"	TUGAS AKHIR	BSI	RAMA	2009

06 Juli 2013 Page 1 of 1

⇒ Buat pula laporan data user, laporan data anggota, kartu anggota dan ID Card user dengan menggunakan langkah yang sama dengan laporan data buku.

⇒Berikut langkah Pembuatan Report “Laporan Peminjaman”: pilih semua field yang ada pada **query_pinjam**.



Berikut adalah tampilan hasil akhirnya:

PERPUSTAKAAN "ISTANA BUKU"							
LAPORAN DATA PEMINJAMAN							
No. Pinjam	Tanggal	ID User	Nama User	No. Anggota	Nama Anggota	Kode Buku	Judul
SIBPJ130704001	04/07/2013	UIB001	DEARAZ	PIB001	KHANSA	BIBKOM001	PERANCANGAN BASIS DATA

06 Juli 2013 Page 1 of 1

Latihan!

Buatkan struk peminjaman, laporan pengembalian buku, dan struk pengembaliannya.