

I . Pengenalan Database

1.1. Pengantar Database

Database atau basis data dapat dibayangkan ibarat sebuah lemari arsip. Jika anda bekerja diperkantoran tentu sangat banyak arsip-arsip atau dokumen kantor yang disimpan kedalam lemari tersebut. Sering anda melihat bahwa penempatan arsip didalam lemari tidak dilakukan secara sembarangan/acak, tapi diberi pembeda antara satu arsip dengan arsip yang lainnya. Pembedanya dapat berupa nomor urut, warna sampul, urutan letak atau berdasarkan tanggal dan lain sebagainya. Begitu juga halnya dengan buku telpon/alamat yang anda punyai, biasanya diatur menurut abjad nama atau kategori tertentu, seperti teman, keluarga, klien dan lain sebagainya. Semuanya disimpan dan diatur menurut aturan tertentu sesuai dengan yang diinginkan. Upaya anda untuk menyusun dan melakukan hal diatas baru akan terasa kalau data arsip dan daftar alamat yang dimiliki sudah banyak. Mengapa hal ini anda lakukan ? tidak lain tujuannya adalah untuk memudahkan dalam mencari atau mengambil data tertentu dalam arsip maupun buku alamat tadi secara Cepat dan mudah. Bayangkan kalau semua arsip atau daftar buku alamat tersebut tidak diatur penyusunannya, tentu akan sangat sulit untuk mencari arsip atau nomor telpon tertentu. Tidak saja sulit tapi akan sangat membutuhkan waktu yang lama.

1.2. Pengertian Database

Banyak sekali definisi tentang database yang diberikan oleh para pakar dibidang ini. Database terdiri dari dua penggalan kata yaitu data dan base, yang artinya berbasiskan pada data, tetapi secara konseptual, database diartikan sebuah koleksi atau kumpulan data-data yang saling berhubungan (relation), disusun menurut aturan tertentu secara logis, sehingga menghasilkan informasi. Sebuah informasi yang berdiri sendiri tidaklah dikatakan database.

Contoh : Nomor telpon seorang pelanggan, disimpan dalam banyak tempat apakah itu difile pelanggan, difile alamat dan dilokasi yang lain. Antara file yang satu dengan file yang lainnya tidak saling berhubungan, sehingga apabila salah seorang pelanggan berganti nomor telpon dan anda hanya mengganti difile pelanggan saja, maka akibatnya akan terjadi ketidakcocokan data, karena di lokasi yang lain masih tersimpan data nomor telpon yang lama. Dalam sistem database hal ini tidak boleh dan tidak bisa terjadi, karena antara file yang satu dengan file yang lain saling berhubungan, jika suatu data yang sama anda ubah, maka data tersebut difile yang lain akan otomatis berubah juga. Sehingga tingkat keakuratan/kebenaran data sangat tinggi. Secara prinsip, dalam suatu database tercakup dua komponen penting, yaitu Data dan Informasi. Jadi tujuan akhir anda adalah bagaimana mengelola data sehingga mampu menjadi informasi yang diinginkan dan dapat dilakukan proses pengambilan, penghapusan, pengeditan terhadap data secara mudah dan cepat (efektif, efisien dan akurat) Data sedangkan informasi adalah data yang telah diolah sehingga bernilai guna dan dapat dijadikan bahan dalam pengambilan keputusan.

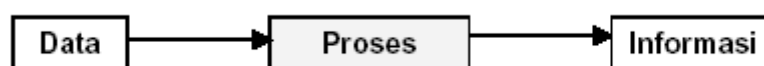
Data

adalah fakta, baik berupa sebuah obyek, orang dan lain-lain yang dapat dinyatakan dengan suatu nilai tertentu (angka, symbol, karakter tertentu, dll).

Informasi

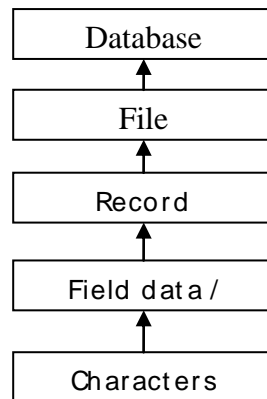
Merupakan hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (event) yang nyata (fact) yang digunakan untuk pengambilan keputusan.

Hubungan data dan informasi dapat digambarkan sebagai berikut :



Gambar 1. Data dan Informasi

Jenjang Data



Gambar 2. Jenjang data

1. Karakter (*Characters*)

Merupakan bagian terkecil dalam database, dapat berupa karakter numerik (angka 0 s.d 9), huruf (A - Z, a - z) ataupun karakter-karakter khusus, seperti *, &. %, # dan lain-lain.

2. Field atau *Attribute*

Merupakan bagian dari record yang menunjukkan suatu item data yang sejenis, Misalnya : field nama, file NIM dan lain sebagainya. Setiap field harus mempunyai nama dan tipe data tertentu. Isi dari field di sebut **Data Value**. Dalam table database, field ini disebut juga **kolom**.

3. Record atau *Tupple*

Tupple/Record adalah kumpulan data value dari attribute yang berkaitan sehingga dapat menjelaskan sebuah entity secara lengkap. Misal : Record entity pegawai adalah kumpulan data value dari field ID_Peg, nama, alamat, kota, kd_pos dan no_telp per-barisnya. Dalam tabel database, Record disebut juga **baris**.

4. Table/Entity

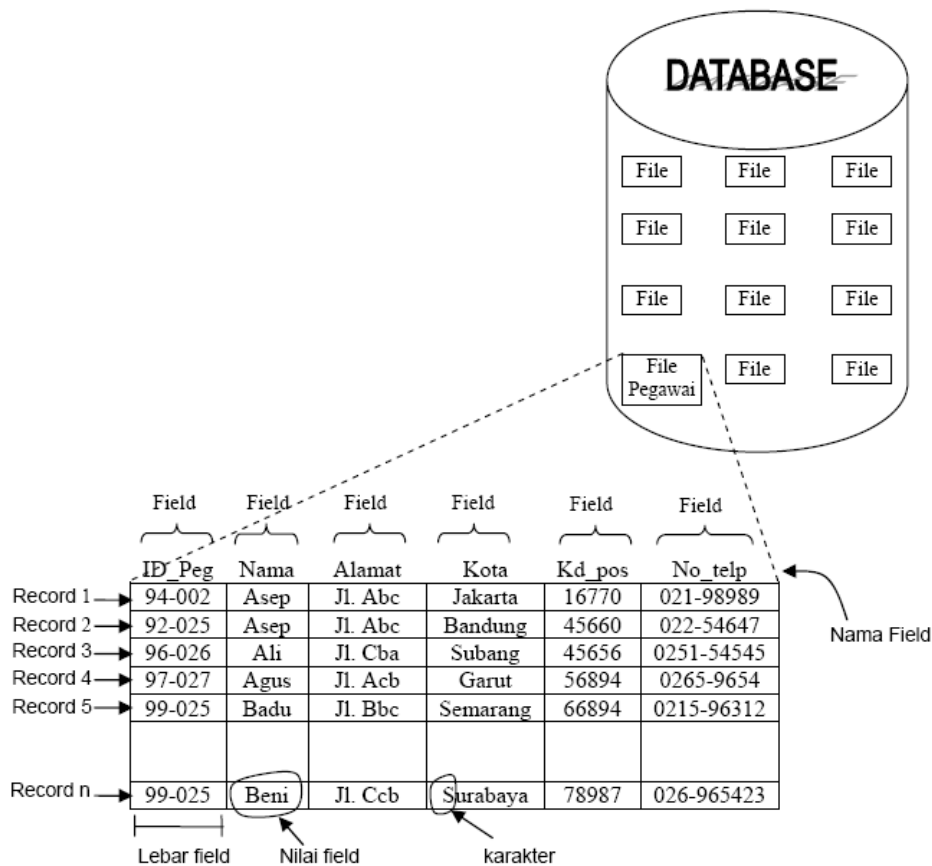
Entity merupakan sesuatu yang dapat diidentifikasi dari suatu sistem database, bisa berupa objek, orang, tempat, kejadian atau konsep yang informasinya akan disimpan dalam database. Misal. Pada sistem database akademik, yang menjadi entity adalah, mahasiswa, dosen, matakuliah dan lain-lain. Dalam aplikasi, penggunaan istilah Entity sering di samakan dengan istilah Tabel. (Entity = table). Disebut tabel, karena dalam merepresentasikan datanya di atur dalam bentuk baris dan kolom. Baris mewakili 1 record dan kolom mewakili 1 field. Dalam system database tradisional, entity/table ini disebut juga dengan file.

5. Database

Kumpulan dari tabel-tabel yang saling berelasi, disusun secara logis, sehingga menghasilkan informasi yang bernilai guna dalam proses pengambilan keputusan.

Database

Kumpulan dari item data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di hardware komputer dan dengan software untuk melakukan manipulasi untuk kegunaan tertentu.



Gambar 3. Database

1.3. Database Management Sistem (DBMS)

Database Management Sistem atau disingkat DBMS adalah perangkat lunak (*Software*) yang berfungsi untuk mengelola database, mulai dari membuat database itu sendiri, sampai dengan proses-proses yang berlaku dalam database tersebut, baik berupa entry, edit, hapus, query terhadap data, membuat laporan dan lain sebagainya secara efektif dan efisien. Salah satu jenis DBMS yang sangat terkenal saat ini adalah *Relational DBMS* (RDBMS), yang merepresentasikan data dalam bentuk tabel-tabel yang saling berhubungan. Sebuah tabel disusun dalam bentuk baris (record) dan kolom (field)

Pengelolaan Data Manual vs Database

Secara umum pengelolaan data secara manual hanya cocok untuk mengelola data dalam jumlah kecil dan informasi yang diharapkan terhadap data tersebut bersifat monoton dan tidak banyak berulang. Kelemahan yang terdapat dalam pengelolaan data secara manual adalah :

a. Duplikasi data

Duplikasi data terjadi karena masing-masing bagian mengelola data secara sendiri-sendiri. Sehingga data yang sama tersimpan pada berbagai tempat. Misalnya : Bagian kemahasiswaan telah menyimpan dan mengelola data mahasiswa untuk kepentingannya, tapi di bagian jurusan juga menyimpan dan mengelola data mahasiswa sesuai dengan kepentingannya juga.

b. Terbatasnya berbagi data

Hal inilah yang menyebabkan terjadi duplikasi data, karena antara satu bagian dengan bagian lainnya tidak saling berhubungan atau berdiri sendiri.

c. Ketidakkonsistennya data

Ketidakkonsistennya data terjadi karena terjadipenyimpanan dan pengelolaan data yang sama di berbagai tempat. Misalnya : Si Dodi adalah mahasiswa jurusan manajemen, pada semester 3 Dodipindah ke jurusan akuntansi. Bagian salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi Menentukan kualitas informasi : akurat, tepat pada waktunya dan relevan. Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Mengurangi duplikasi data (data redundancy) Mengurangi pemborosan tempat simpanan luar kemahasiswaan telah mencatat dan menyimpan data Dodi sebagai mahasiswa jurusan akuntansi. Tapi di bagian jurusan manajemen, karena tidak adanya informasi, maka si Dodi tetap tercatat sebagai mahasiswa jurusan manajemen.

Tentu hal seperti ini akan berakibat fatal, bukan !

d. Kurangnya integritas data

Karena adanya ketidakkonsistenan data mengakibatkan kurangnya Integritas terhadap data. Integritas menyangkut dalam hal kevalidan data.

e. Kesulitan dalam mendapatkan informasi

Misalnya pada suatu saat, kepala akademik menginginkan data mahasiswa dengan IPK diatas 3.00. Maka tentu hal ini akan menghabiskan waktu yang lama untuk memprosesnya, apalagi kalau jumlah data yang diolah sudah mencapai lebih dari ribuan record.

f. Ketidakluwesannya

Kurangnya respon dalam hal menghadapi perubahan dan pengembangan atas informasi yang diinginkan. Misalnya, kalau terjadiperubahan terhadap data yang diinginkan, maka haruslah diulang dari awal lagi. Begitu juga halnya dengan tingkat kompatibilitas dengan perkembangan perangkat lunak di masa depan.

Semua kelemahan-kelemahan dalam pengelolaan data secara manual diatas dapat diatasi dengan pengelolaan data dalam suatu sistem database. Berikut keuntungan menggunakan database dalam mengelola suatu sistem informasi ;

a. Duplikasi data dapat diminimalkan

Duplikasi data dapat diminimalkan, dan biasanya data yang duplikat tersebut merupakan field kunci. Hal ini memang tidak bisa di hindari, karena field kunci ini digunakan nantinya sebagai key untuk hubungan antar tabel dan menyangkut integritas serta independensi data.

b. Integritas data tinggi

Tingkat kevalidan data tinggi, karena data yang sama saling berelasi, sehingga apabila ada perubahan pada suatu data, maka data yang sama difile yang lain otomatis juga berubah.

c. Independensi data

Tingkat ketergantungan data sangat tinggi, dimana anda tidak bisa melakukan perubahan terhadap suatu data, jika data tersebut sedang dipakai oleh file lain. Misalnya ; Anda tidak dapat menghapus data matakuliah tertentu pada file matakuliah, kalau matakuliah tersebut sedang diambil oleh mahasiswa pada file KRS misalnya.

d. Konsistensi data tinggi

Ini berhubungan dengan independensi data, sehingga tingkat konsistensi data jadi tinggi.

e. Dapat berbagi (sharing) data

Ini merupakan salah satu keunggulan dari pengelolaan data dengan sistem database, dimana anda dapat saling berbagi dalam penggunaan file, baik bersifat jaringan maupun client server.

f. Tingkat keamanan tinggi

Adanya pemberian password dan hak akses pada suatu file mengakibatkan hanya orang yang berhak saja yang bisa mengakses file tersebut. Sehingga lebih aman.

g. Mudahnya mendapatkan data

Proses mendapatkan data dan informasi pada database dapat dilakukan dengan mudah, terutama menggunakan Structure Query Language (SQL), yang merupakan bahasa standar dalam DBMS.

II. PENGOLAHAN DATABASE DENGAN MICROSOFT EXCEL

Microsoft Excel adalah salah satu program aplikasi spreadsheet canggih yang populer dan banyak digunakan untuk membantu menghitung, memproyeksikan, menganalisa dan mempresentasi data.

Microsoft Excel lebih pada pemanfaatan pengentrian data-data dan informasi tentang berbagai kepentingan data yang kemudian dianalisa sesuai dengan kebutuhan lembaga yang bersangkutan. Microsoft Excel memudahkan Pengguna dalam pengelompokan data dalam bentuk kotak/bagian sesuai kebutuhan informasi.

2. 1 Pivot Table

Pivot table dapat anda gunakan untuk menganalisa, meringkas, dan memainkan data dalam daftar yang besar. Pivot table menawarkan analisa data yang mudah disesuaikan dan intuitif. Walaupun data yang tersaji dalam pivot tables nampak seperti worksheet data lainnya, anda tidak dapat langsung memasukkan atau mengubah data di area pivot table. Pivot table dihubungkan dengan sumber data, dan apa yang anda lihat di cell table merupakan data read-only (hanya bias dibaca). Namun disini anda masih dapat mengubah format (Number, Alignment, Font, Patterns) dan memilih dari berbagai pilihan komputasi seperti SUM, AVERAGE, MIN, dan MAX.

2.1.1 Persiapan Menggunakan Pivot Table

Misalkan anda mempunyai data Bencana seperti tabel 1. Hal yang pertama harus dilakukan adalah menentukan karakteristik data. Berdasarkan data tersebut terdapat dua nilai dari setiap bencana yaitu **ada** dan **tidak**. Kita tentukan **ada** bernilai **1** dan **tidak** bernilai **0**, kemudian kita transformasikan data pada tabel 1 menjadi seperti pada tabel 2. Selain cara ini bias juga dilakukan dengan membuat flat table seperti pada table 3.

Tabel 1. Contoh Dokumen Bencana

Desa	Bencana							
	banjir		tanah longsor		gempa bumi		abrasi pantai	
	ada	tidak	ada	tidak	ada	tidak	ada	tidak
KLAISU		x	x			x		x
NEMBU GRESI		x		x		x		x
IBUB		x		x	x	x		x
HATIB	x		x			x		x
BRING								x
PUPEHABU		x	x		x			x
DAMOI KATI		x	x			x		x
DEMETIN		x		x		x		x
YANBRA		x		x	x			x
BRASO	x		x			x		x
BAGAI		x		x		x		x
IWON		x		x	x			x
JAGRANG		x						x
OMON		x	x		x			x
SWENTAB	x		x			x		x

Tabel 2. Transformasi dari Tabel 1

desa	banjir	tanah_longsor	gempa_bumi	abrasi_pantai
KLAISU	0	1	0	0
NEMBU GRESI	0	0	0	0
IBUB	0	0	1	0
HATIB	1	1	0	0
BRING	0	0	0	0
PUPEHABU	0	1	1	0
DAMOI KATI	0	1	0	0
DEMETIN	0	0	0	0
YANBRA	0	0	1	0
BRASO	1	1	0	0
BAGAI	0	0	0	0
IWON	0	0	1	0
JAGRANG	0	0	0	0
OMON	0	1	1	0
SWENTAB	1	1	0	0

Tabel 3. Contoh Flat tabel/ bentuk lain transformasi dari tabel 1

desa	banjir	tanah_longsor	gempa_bumi	abrasi_pantai
KLAISU	tidak	ya	tidak	tidak
NEMBU GRESI	tidak	tidak	tidak	tidak
IBUB	tidak	tidak	Ya	tidak
HATIB	ya	ya	tidak	tidak
BRING	tidak	tidak	tidak	tidak
PUPEHABU	tidak	ya	Ya	tidak
DAMOI KATI	tidak	ya	tidak	tidak
DEMETIN	tidak	tidak	tidak	tidak
YANBRA	tidak	tidak	Ya	tidak
BRASO	ya	ya	tidak	tidak
BAGAI	tidak	tidak	tidak	tidak
IWON	tidak	tidak	Ya	tidak
JAGRANG	tidak	tidak	tidak	tidak
OMON	tidak	ya	Ya	tidak
SWENTAB	ya	ya	tidak	tidak

Latihan 1.

Buatlah Flat Tabel dari data yang dipunyai

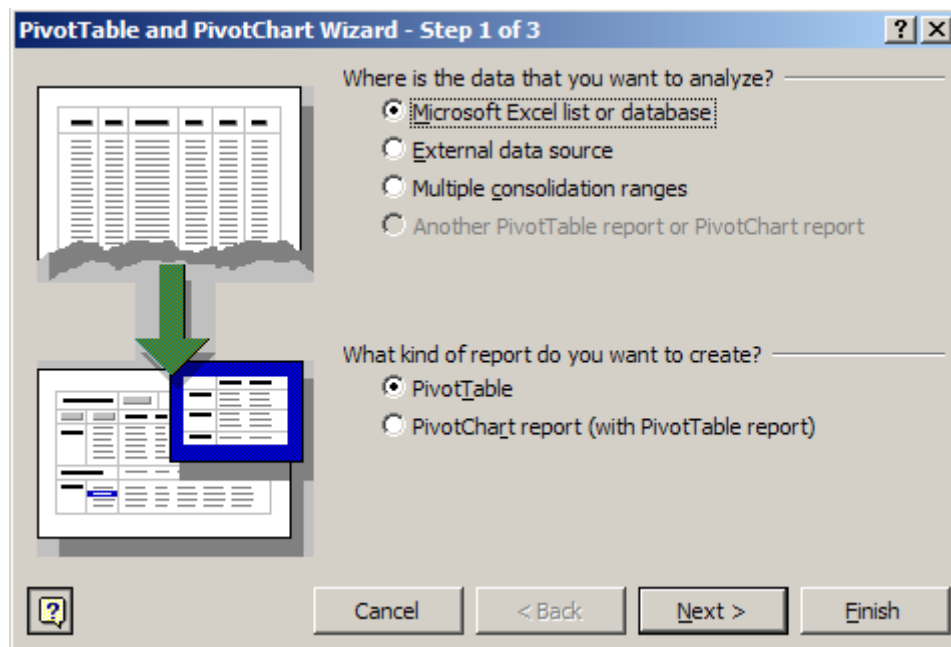
2.1.2 Bekerja dengan Pivot Table

Aktifkan Sel A1

	A	B	C	D	E
1	desa	banjir	tanah longsor	gempa bumi	abrasi pantai
2	KLAISU	0	1	0	0
3	NEMBU GRESI	0	0	0	0
4	IBUB	0	0	1	0
5	HATIB	1	1	0	0
6	BRING	0	0	0	0
7	PUPEHABU	0	1	1	0
8	DAMOI KATI	0	1	0	0
9	DEMETIN	0	0	0	0
10	YANBRA	0	0	1	0
11	BRASO	1	1	0	0
12	BAGAI	0	0	0	0
13	IWON	0	0	1	0
14	JAGRANG	0	0	0	0
15	OMON	0	1	1	0
16	SWENTAB	1	1	0	0

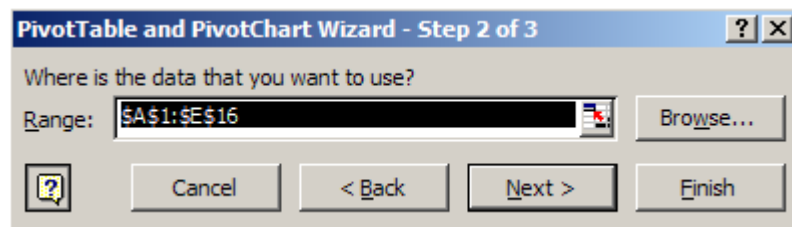
Gambar 4. Tabel Bencana di Excel

Pilih menu Data → PivotTable and PivotChart Report



Gambar 5. Kotak dialog PivotTable and PivotChart Step 1

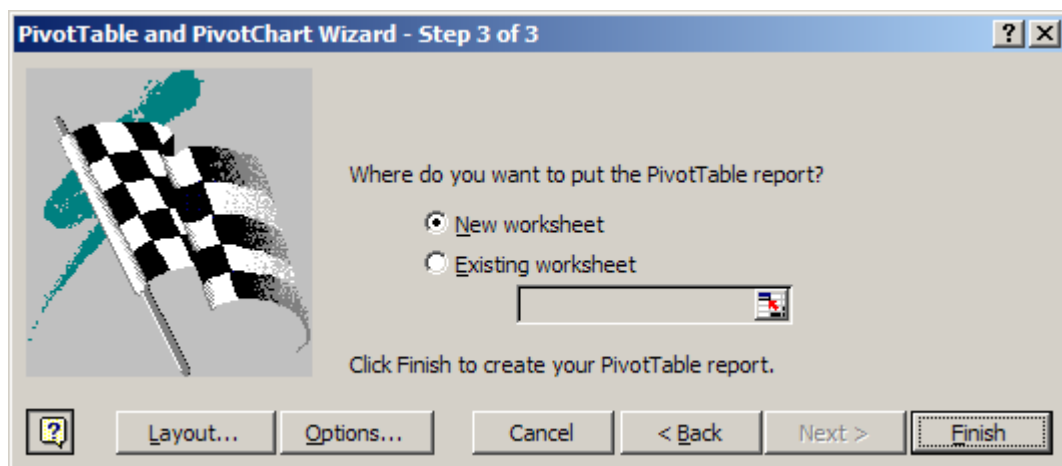
Pilih Microsoft Office Excel list or database dan pilih juga PivotTable. Kemudian klik tombol next. Selanjutnya akan muncul kotak dialog



Gambar 6. Kotak dialog PivotTable and PivotChart Step 2

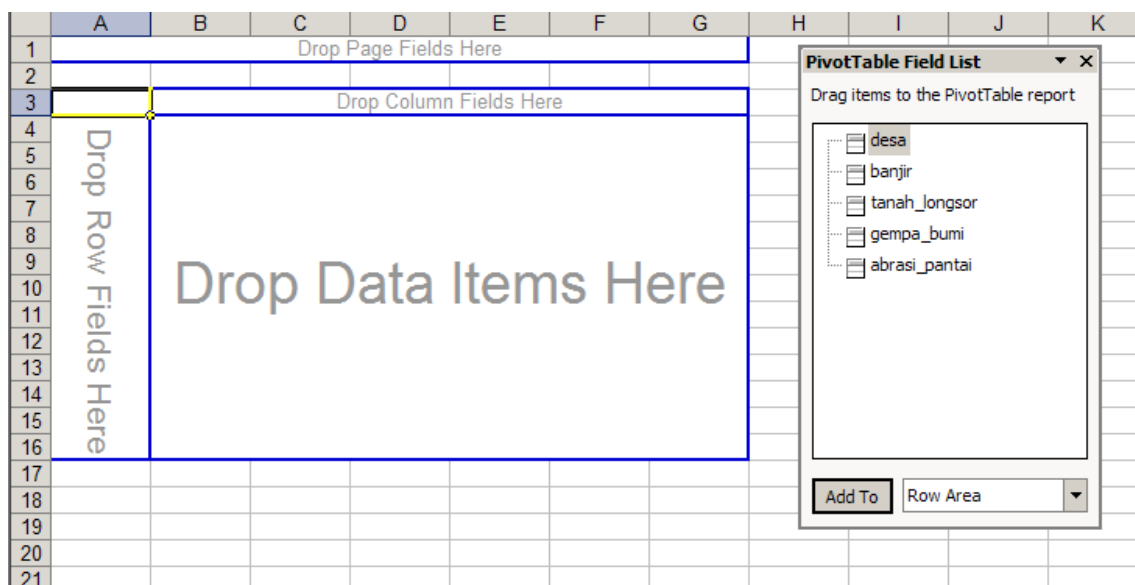
pada langkah ini anda akan menentukan jangkauan data yang akan anda masukkan dalam analisa PivotTable.

Kemudian klik tombol Next, dan anda akan masuk ke langkah terakhir PivotTable and Pivot Chart Wizard. Pilih New Worksheet dan klik tombol Finish



Gambar 7. Kotak dialog PivotTable and PivotChart Step 3

Hasil yang anda peroleh dari langkah-langkah diatas dapataa di lihat pada gambar 6 di bawah ini



Gambar 8. Tampilan Awal PivotTable

	A	B	C	D	E	F
1	Drop Page Fields Here					
2						
3	Sum of banjir					
4	desa	Total				
5	BAGAI	0				
6	BRASO	1				
7	BRING	0				
8	DAMOI KATI	0				
9	DEMETIN	0				
10	HATIB	1				
11	IBUB	0				
12	IWON	0				
13	JAGRANG	0				
14	KLAISU	0				
15	NEMBU GRESI	0				
16	OMON	0				
17	PUPEHABU	0				
18	SWENTAB	1				
19	YANBRA	0				
20	Grand Total	3				
21						

PivotTable Field List ▾ ×

Drag items to the PivotTable report

- desa
- banjir
- tanah_longsor
- gempa_bumi
- abrasi_pantai

Add To Row Area ▾

Gambar 9. Hasil Analisis PivotTable

III. PERANCANGAN DATABASE

Merancang database merupakan hal yang sangat penting, karena disini anda akan menentukan entity, attribut, relasinya dan konsep lainnya dalam suatu sistem database, sehingga hasil rancangan tersebut memenuhi kebutuhan anda akan informasi untuk saat ini dan masa yang akan datang.

Ada tiga langkah dalam merancang database, yaitu :

1. Perancangan Database Konseptual (Conceptual Database Design)
2. Perancangan Database Logik (Logical Database Design)
3. Perancangan database Fisik (Physical Database Design)

Sebelum merancang suatu database anda harus mengetahui, apakah akan merancang database baru atau merancang pengembangan database dari sistem yang sedang berjalan. Jika yang dirancang adalah pengembangan dari database yang telah berjalan, maka anda harus memahami konsep yang berlaku pada sistem yang lama dan berusaha merancang database baru agar tetap mengakomodasi sistem yang lama. Proses pengembangan merancang database lebih sulit daripada merancang database yang benar-benar baru.

3.1 Perancangan Database Konseptual

Perancangan secara konsep merupakan langkah pertama dalam merancang database. Sesuai dengan namanya, pada tahap ini anda hanya menentukan konsep-konsep yang berlaku dalam sistem database yang akan di bangun. Dalam tahap ini, setidaknya anda harus mengetahui :

1. Prosedur kerja secara keseluruhan yang berlaku pada sistem yang sedang berjalan.
2. Informasi (output) apa yang diinginkan dari database ?
3. Apa saja kelemahan-kelemahan dari sistem yang sedang berjalan ?
4. Pengembangan sistem di masa yang akan datang.
5. Bagaimana tingkat keamanan data saat ini ?
6. Siapa saja yang terlibat dalam sistem yang sedang berjalan.
7. Apa saja input yang di perlukan ?

Seorang perancang database harus paham benar terhadap sistem yang sedang berjalan dan harus mengetahui sistem yang bekerja pada database yang akan di bangun serta output apa yang diharapkan. Jika anda akan merancang **Sistem Informasi Komunitas Masyarakat Adat**, maka anda harus mengetahui bagaimana prosedur kerja dalam Komunitas secara keseluruhan, siapa-siapa saja yang terlibat didalamnya, apa saja peraturan-peraturan yang berlaku, apakah semua orang boleh mengakses database tersebut ?, dan lain sebagainya. Pemahaman seorang perancang database terhadap sistem yang akan di bangun sangat menentukan baik atau tidaknya hasil rancangan databasenya.

3.2 Perancangan Database Logik

Perancangan database logik merupakan tahapan untuk memetakan proses perancangan konseptual kedalam model database yang akan digunakan, apakah model data hirarki, jaringan atau relasi. Perancangan database secara logik ini tidak tergantung pada DBMS yang digunakan, sehingga tahap perancangan ini disebut juga pemetaan model data. Berikut langkah-langkah dalam merancang database logik :

3.2.1 Mendefinisikan Entity Yang Dibutuhkan

Entiti adalah sesuatu yang mudah diidentifikasi dengan mudah dari suatu sistem database, bisa berupa objek, orang, tempat, kejadian atau konsep yang informasinya akan disimpan. Hal-hal yang terlibat dalam suatu sistem database dapat dijadikan entity. Dari sekian banyak kemungkinan entity yang ada, anda harus memilah-milah entity mana saja yang sesuai dan mampu mengakomodasi kebutuhan sistem yang akan dirancang. Misalnya dalam proses merancang Sistem Informasi Akademik, ada banyak kemungkinan yang bisa di jadikan entity, Misalnya entity mahasiswa, matakuliah, dosen, fakultas, jurusan, lokal dan lain sebagainya.

Maka secara sederhana, anda dapat menentukan tiga entity utama yang terlibat dalam proses kegiatan akademik, yaitu :

1. Entity Mahasiswa, berfungsi untuk menyimpan data mahasiswa.
2. Entity Dosen, berfungsi untuk menyimpan data dosen, dan
3. Entity Matakuliah, untuk menyimpan data matakuliah.



Gambar 10. Entity pada sistem informasi akademik

Proses menentukan entity ini memang agak sulit tapi akan menjadi mudah apabila sering latihan terhadap berbagai macam kasus database.

3.2.2 Menentukan Attribut Setiap Entity Beserta Kuncinya

Setelah menentukan entity-entity yang terlibat pada sistem database yang dirancang, langkah berikutnya adalah menentukan attribut yang melekat pada entity tersebut. Attribut adalah ciri khas yang melekat pada suatu entity dan menunjukkan item sejenis. Sama halnya dalam menentukan entity, dalam menentukan attribut ini juga banyak kemungkinan, maka anda harus memilah-memilah attribut apa saja yang diperlukan oleh sistem database yang dirancang. Berikut beberapa attribut yang mungkin pada entity mahasiswa :

- Nobp
- Nama
- Tempat Lahir
- Tanggal Lahir
- Fakultas
- Jurusan
- Agama
- Jenis Kelamin
- Tanggal Masuk
- Nama Pembimbing Akademik
- Asal SMU
- Alamat
- Nama Orang Tua
- Pendidikan Orang Tua
- Pekerjaan Orang Tua
- Alamat Orang Tua
- Dan seterusnya.

3.2 Perancangan Model Konseptual

Perancangan ini menunjukkan entitas dan relasinya berdasarkan proses yang diinginkan organisasi. Entitas sendiri adalah sebuah objek yang berbeda dari objek lainnya. Dalam perancangan konseptual ini, dibutuhkan analisis data tentang informasi yang ada dalam spesifikasi di masa yang akan datang. Tugas ini diemban oleh database administrator. Model Konseptual tidak bergantung pada developer tools, DBMS, hardware, maupun model fisik.

Ada 2 teknik dari model konseptual ini :

Normalisasi

Entity-Relationship

Teknik Normalisasi biasanya digunakan untuk merancang database dari sistem yang sudah ada sebelumnya, atau perbaikan/penyempurnaan sistem. Teknik E/R tepat untuk perancangan database dari awal, dari kondisi tidak ada. Pada modul ini akan lebih di bahas tentang Normalisasi

Normalisasi

Proses normalisasi merupakan proses pengelompokan data elemen menjadi tabel-tabel yang menunjukkan entity dan relasi sehingga membentuk struktur relasi yang baik. ('sedapat mungkin' menghilangkan redundansi,--pada kondisi tertentu redundansi bisa dibutuhkan--). Sebelum kita terlalu jauh, sebaiknya kita tahu dulu pengertian istilah-istilah berikut ini :

Jenis-Jenis Kunci (Key)

Key atau kunci adalah suatu field yang dapat mewakili dari suatu record. Misal : nobp merupakan field kunci dari entity mahasiswa, sehingga setiap melakukan pencarian atas entity mahasiswa cukup menyebutkan nobp saja, maka field nama, jurusan dan alamat dapat diketahui. Syarat utama pemilihan suatu field kunci dari entity adalah field tersebut harus unik dan tidak boleh bernilai NULL.

Ada 4 jenis dari key ini :

a. **Candidate Key** (Kunci Calon)

Sebuah attribute atau lebih yang secara unit mengidentifikasi sebuah record, disebut candidate key. Attribute ini mempunyai nilai yang unik pada hampir setiap recordnya. Fungsi dari candidate key ini adalah sebagai calon primary key. **Misal** : Entity dosen dalam database akademik :

KodeDosen	NIP	Nama_Dosen	Alamat	Pendidikan
01	131656765	Johan, SE	Padang	S.1
02	130876543	Rafdinal, Drs, M.Si	Padang	S.2
03	132098675	Ratna, SE	Padang	S.1
04	130987567	Effendi Bakar, SE	Padang	S.1

Gambar 11. Tabel Dosen

Dalam kasus ini, field KodeDosen dan NIP merupakan Candidate Key karena masing-masing bersifat unik dan tidak boleh ada yang sama atau bernilai Null.

Contoh 2 : File Karyawan : noinduk, nama, tempatlahir, tanggallahir, alamat, kota

Kunci Kandidat :

noinduk → pasti unik

nama → sering dipakai sebagai kunci pencarian tetapi tidak cocok untuk key bisa ada nama yang sama.

nama + tanggallahir → jarang orang mempunyai nama dan tanggal lahir yang sama, walaupun bisa saja untuk orang kembar (kembar pun biasanya namanya berbeda)

nama + tanggallahir + tempatlahir → jarang orang mempunyai nama, tanggal lahir, dan tempat lahir yang sama, walaupun bisa saja untuk orang kembar, kembar pun biasanya namanya berbeda).

alamat + kota → tidak cocok untuk kunci.

b. **Primary Key** (Kunci Utama)

Merupakan candidate key yang telah dipilih untuk mengidentifikasi setiap record secara unik. Primary key harus merupakan field yang benar-benar unik dan tidak boleh ada nilai NULL.

Misal : Perhatikan kembali entity dosen di atas, yaitu mempunyai dua Candidate Key, maka anda dapat mengambil field KodeDosen sebagai Primary Key, Karena lebih sederhana dalam jumlah digit dan unik. Boleh juga menggunakan NIP sebagai Primary Key-nya, hal ini tergantung kepada si perancang database itu sendiri.

c. **Alternate Key** (Kunci Alternatif)

Adalah candidate key yang tidak terpilih. Misal : dalam suatu entity terdapat dua field yang bisa dijadikan sebagai kunci. Sementara yang boleh dijadikan kunci hanya satu, maka anda harus

memilih salah satu. Field yang anda pilih, disebut primary key, sedangkan field yang tidak dipilih disebut dengan alternate key. **Misal** : Kembali ke kasus entity dosen diatas, jika anda pilih field KodeDosen sebagai Primary Key, maka otomatis field NIP menjadi Alternate Key-nya, begitu juga sebaliknya.

d. **Foreign Key** (kunci Tamu)

Jika sebuah primary key terhubungan ke table/entity lain, maka keberadaan primary key pada entity tersebut di sebut sebagai foreign key. Misal : Primary Key KodeDosen dari entity Dosen digunakan juga pada field entity KRS, maka keberadaan field KodeDosen pada entity KRS disebut sebagai foreign key.

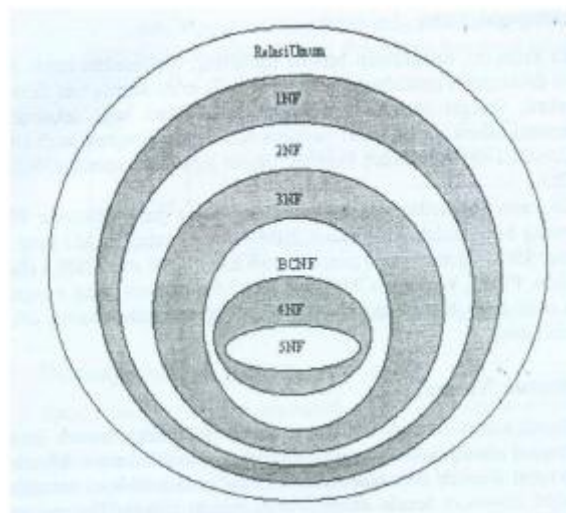
Bentuk Normalisasi

Aturan-aturan normalisasi dinyatakan dalam istilah **bentuk normal**. Bentuk normal adalah suatu aturan yang dikenakan pada entity-entity dalam database dan harus dipenuhi oleh entity-entity tersebut sehingga tercapai normalisasi. Suatu entity dikatakan dalam bentuk normal apabila entity tersebut memenuhi aturan pada bentuk normal tersebut. Proses normalisasi dilakukan secara bertingkat. Pada tingkat ketiga (*Third Normal Form*, 3NF) sebenarnya telah dapat menghasilkan suatu rancangan database yang baik.

Berikut tingkatan bentuk normal dalam proses normalisasi :

- Bentuk Normal Pertama (1NF)
- Bentuk Normal Kedua (2NF)
- Bentuk Normal Ketiga (3NF)
- Bentuk Normal Boyce-Codd (BCNF)
- Bentuk Normal Keempat (4NF)
- Bentuk Normal Kelima (5NF)

Perhatikan gambar berikut :



Gambar 12. Bentuk-Bentuk Normalisasi

Terlihat pada gambar diatas, bahwa setiap level normalisasi bergantung pada level sebelumnya. Misalnya, bentuk normal kedua pasti telah memenuhi bentuk normal pertama, bentuk normal ketiga pasti telah memenuhi bentuk normal kedua, dan seterusnya.

Normal Pertama (1st Normal Form)

Aturan :

Suatu entity dikatakan dalam bentuk normal pertama jika setiap atributnya bernilai tunggal untuk setiap barisnya. "suatu tabel dapat dikatakan 1NF jika setiap cell mempunyai single value (Flat Tabel)"

Normalisasi Kedua (2nd Normal Form)

Aturan :

Suatu entity dikatakan dalam bentuk normal kedua jika :

1. Berada pada bentuk normal pertama.
2. Semua atribut bukan kunci memiliki ketergantungan fungsional (*Depedensi Fungsional*) dengan kunci utama (*primary key*)

Ketergantungan fungsional adalah suatu atribut X mempunyai ketergantungan fungsional terhadap atribut Y apabila setiap nilai X berhubungan dengan sebuah nilai Y. Misalnya Atribut **Nama** pada entity Mahasiswa, mempunyai ketergantungan fungsional terhadap atribut **NoBP**, karena setiap **nama** mahasiswa harus mempunyai **NoBP**.

Pada tahap ini anda harus memilah-memilah dan membagi entity tersebut menjadi beberapa entity lainnya yang mempunyai kunci utama. Sehingga masingmasing atribut yang bukan kunci mempunyai ketergantungan fungsional dengan kunci utama tersebut.

Normalisasi Ketiga (3rd Normal Form)

Aturan :

Suatu entity dikatakan dalam bentuk normal pertama jika :

1. Berada pada bentuk normal kedua.
2. Semua atribut bukan kunci tidak memiliki ketergantungan transitif (*Depedensi transitif*) dengan kunci utama (*primary key*)

Ketergantungan Transitif terjadi pada entity yang menggunakan attribute gabungan sebagai kunci utama. Seperti pada entity nilai pada bentuk normal kedua diatas, yang menjadi kunci utama adalah **NoBP** dan **Kode Matakuliah**. Ketergantungan transitif terjadi bila :

- a. Atribut X memiliki ketergantungan fungsional dengan atribut Y.
- b. Atribut Z memiliki ketergantungan fungsional dengan atribut X.

Bentuk Normal Boyce-Codd (BCNF)

Aturan :

Bentuk Normal Boyce-Codd (BCNF) :

Suatu entity dikatakan dalam bentuk BCNF jika : Semua kunci utama adalah kunci kandidat yang bersifat unik

BCNF memiliki ketentuan yaitu masing-masing atribut utama bergantung fungsional penuh pada masing-masing kunci dimana kunci tersebut bukan bagiannya. Relasi adalah BCNF (optimal) jika setiap determinan atribut-atribut relasi adalah kunci relasi. Relasi adalah BCNF (optimal) jika kapanpun fakta-fakta disimpan mengenai beberapa atribut, maka atribut-atribut ini merupakan satu kunci relasi. BCNF dapat memiliki lebih dari satu kunci. Properti penting BCNF adalah relasi tidak memiliki informasi yang redundan.

Bentuk Normal Keempat (4NF)

Aturan :

Bentuk normal keempat berhubungan dengan sifat ketergantungan banyak nilai (*Multivalued Dependency*) pada suatu tabel yang merupakan pengembangan dari ketergantungan fungsional.

Bentuk Normal Kelima (5NF)

Aturan :

Bentuk normal kelima berkenaan dengan ketergantungan relasi antar tabel (*Join Depedency*).

Kita tidak membahas tahap normal BCNF, karena biasanya tahap Normalisasi sudah cukup sampai normal ke 3

Contoh Kasus

Lakukan normalisasi pada kasus berikut

Nama MHS	Matakuliah	SKS	Nilai
Elzar	S I M	3	A
	S I A	3	C
Fikri	S I M	3	C
	S I A	3	D
	Database Management	2	B
Helga	Peny. Prog. Komp.	2	B
Helga	Peny. Prog. Komp.	2	A

Gambar 13. Bentuk tabel tidak normal

Ada dua kelemahan utama pada bentuk tidak normal diatas :

1. Terdapat attribut yang berulang (duplikat), yaitu attribut matakuliah. Mahasiswa dengan nama Elzar mengambil 2 matakuliah, sementara Si Fikri mengambil 3 matakuliah dimana matakuliah yang mereka ambil ada yang sama.
2. Terdapat informasi yang meragukan, dimana ada dua baris memiliki matakuliah yang sama, tapi berbeda nilainya. Sebenarnya kedua baris tersebut menunjukkan dua orang yang sama namanya tapi berbeda nilai.

Normalisasi Pertama :

Nama MHS	Matakuliah	SKS	Nilai
Elzar	S I M	3	A
Elzar	S I A	3	C
Fikri	S I M	3	C
Fikri	S I A	3	D
Fikri	Database Management	2	B
Helga	Peny. Prog. Komp.	2	B
Helga	Peny. Prog. Komp.	2	A

Gambar 14. Bentuk Normal Pertama

Terlihat pada entity diatas bahwa setiap attribut telah bernilai tunggal untuk setiap barisnya. Tapi redudansi dan adanya informasi yang meragukan masih belum teratasi.

Normalisasi Kedua :

<u>NoBP</u>	Nama Mahasiswa
05955001	Elzar
05955002	Fikri
05955003	Helga
05955004	Helga

Gambar 15. Bentuk Normal Ke dua

<u>NoBP</u>	<u>Kode Matakuliah</u>	Matakuliah	SKS	Nilai
05955001	EKP414	S I M	3	A
05955001	EKP415	S I A	3	C
05955002	EKP414	S I M	3	C
05955002	EKP415	S I A	3	D
05955002	EKP355	Database Management	2	B
05955003	EKP344	Peny. Prog. Komp.	2	B
05955003	EKP344	Peny. Prog. Komp.	2	A

Gambar 16 Bentuk Normal Ke dua

Pada gambar diatas terlihat, ada dua entity yang memiliki **kunci utama** (atribut yang bergaris bawah), sehingga attribut-attribut yang lainnya mempunyai ketergantungan fungsional terhadapnya.

- Atribut Nama mahasiswa mempunyai ketergantungan fungsional terhadap atribut **NOBP**
Atribut SKS dan Nilai mempunyai ketergantungan fungsional terhadap attribute **NoBP** dan **Kode Matakuliah**.

Ternyata rancangan entity baru diatas masih belum benar, khususnya pada entity nilai, karena : **adanya data yang berulang**, yaitu nama matakuliah dan sks, Kalau seandainya salah mengentrikan data matakuliah dan sks, dapat mengakibatkan **data tidak konsisten** lagi.

Normalisasi Ketiga :

Pada atribut **Kode Matakuliah** pada entiti nilai diatas, mempunyai ketergantungan fungsional dengan atribut **NoBP**, Atribut **Nama Matakuliah** mempunyai ketergantungan fungsional dengan atribut **Kode Matakuliah**. Entity Nilai berikut merupakan contoh entity yang memenuhi normal kedua tapi tidak memenuhi bentuk normal ketiga, karena adanya ketergantungan transitif.

<u>NoBP</u>	<u>Kode Matakuliah</u>	Matakuliah	SKS	Nilai
05955001	EKP414	S I M	3	A
05955001	EKP415	S I A	3	C
05955002	EKP414	S I M	3	C
05955002	EKP415	S I A	3	D
05955002	EKP355	Database Management	2	B
05955003	EKP344	Peny. Prog. Komp.	2	B
05955003	EKP344	Peny. Prog. Komp.	2	A

Gambar 17. Ketergantungan Transitif

Pada contoh diatas, kunci utama merupakan gabungan antara **NoBP** dan **Kode Matakuliah**. Atribut matakuliah, sks dan nilai mempunyai ketergantungan fungsional terhadap kunci utama tersebut. Namun perlu di perhatikan, bahwa jika **Kode Matakuliah** bernilai sama, **Nama Matakuliah** juga bernilai sama. Hal ini menandakan adanya suatu ketergantungan antara kedua atribut tersebut. Lalu manakah yang menjadi penentu ? Apakah **Kode Matakuliah** bergantung pada **Nama matakuliah** ? atau sebaliknya. Yang jadi penentu tentulah **Kode Matakuliah**, karena kode bersifat unik dan akan berbeda untuk setiap nama matakuliah. Oleh karena itu entity Nilai harus dibagi lagi agar memenuhi aturan bentuk normal ketiga.

Perhatikan tabel berikut setelah di ubah kedalam bentuk normal ketiga :

Entity Mahasiswa

<u>NoBP</u>	Nama Mahasiswa
05955001	Elzar
05955002	Fikri
05955003	Helga
05955004	Helga

<u>NoBP</u>	<u>Kode Matakuliah</u>	Nilai
05955001	EKP414	A
05955001	EKP415	C
05955002	EKP414	C
05955002	EKP415	D
05955002	EKP355	B
05955003	EKP344	B
05955003	EKP344	A

<u>Kode Matakuliah</u>	Matakuliah	SKS
EKP414	S I M	3
EKP415	S I A	3
EKP414	S I M	3
EKP415	S I A	3
EKP355	Database Management	2
EKP344	Peny. Prog. Komp.	2
EKP344	Peny. Prog. Komp.	2

Gambar 18 Bentuk Normal Ke tiga

Coba anda amati, apakah pada bentuk normal ketiga ini, masih ada redundansi dan ketidakkonsistenan data ? Penerapan aturan normalisasi sampai dengan bentuk ketiga ini, sebenarnya telah memenuhi dalam menghasilkan desain database yang berkualitas baik. Namun demikian dari sejumlah literatur dapat pula dijumpai pembahasan tentang bentuk **normal keempat** (4NF) dan bentuk **normal kelima** (5NF) dan adapula bentuk normal **Boyce-Codd** sebagai perbaikan dari bentuk normal ketiga.

Latihan 2

Normalisasi data profil masyarakat adat

IV. Implementasi dengan Microsoft Access

Database atau biasa disebut basis data adalah kumpulan data yang berhubungan dengan suatu obyek, topik atau tujuan khusus tertentu. Sebagai contoh katalog buku di perpustakaan, data profil perusahaan, data pelanggaran, konflik, dll (sesuai dengan kebutuhan lembaga). Dengan microsoft access anda dapat mengelola seluruh data yang anda miliki ke dalam sebuah file database. Database pada access dapat terdiri atas satu atau beberapa tabel, query, form, report, page, makro, dan modul yang semuanya berhubungan atau saling keterkaitan.

Tables, berupa tabel kumpulan data yang merupakan komponen utama dari sebuah database.

Query, digunakan untuk mencari dan menampilkan data yang memenuhi syarat tertentu dari satu tabel atau lebih. Query dapat juga digunakan untuk mengupdate atau menghapus beberapa record data pada satu saat yang sama. Selain itu query dapat digunakan untuk

menjalankan perhitungan terhadap sekelompok data. Sebuah query dapat digunakan untuk menjalankan perhitungan terhadap sekelompok data, sebuah query dapat memiliki sumber data sampai dengan 16 tabel, dapat memiliki 255 field yang berbeda.

Form, dipergunakan untuk menampilkan data, mengisi data dan mengubah data yang ada di dalam tabel. Ketika anda membuka Form, microsoft access mengambil data dari satu tabel atau lebih dan menampilkannya kelayar monitor menggunakan layout yang anda buat melalui form wizard atau dari layout yang anda rancang sendiri.

Reports, dipergunakan untuk menampilkan laporan hasil analisa data. Anda dapat mencetak sebuah laporan yang telah dikelompokkan, dihitung subtotal dan total datanya berdasarkan kriteria tertentu. Anda juga dapat membuat laporan yang berisi grafik atau label data.

Pages, digunakan untuk membuat halaman web (Page) berupa data access page yang dapat anda tempatkan diserver sistem jaringan intranet atau internet.

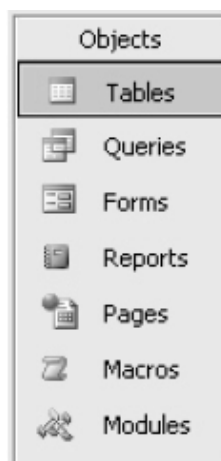
Macros, untuk mengotomatisasi perintah-perintah yang sering anda gunakan dalam mengolah data.

Modules, digunakan untuk perancangan berbagai modul aplikasi pengolahan database tingkat lanjut sesuai dengan kebutuhan anda. Module ini berisi kode visual basic for applications yang anda tulis untuk menangani even dalam access.

Mengapa Harus Access??

Dengan Microsoft Access pengguna mendapat kemudahan dalam mengelola database, karena semua pengguna yang menggunakan Sistem Operasi Microsoft Windows telah menyediakan program sistem operasi microsoft Access dalam microsoft Office. Selain menyediakan fasilitas Wizard dan contoh-contoh database yang mudah untuk dimanipulasi memungkinkan pengguna mengelola database dengan relatif mudah, cepat dan profesional. Dalam Microsoft Access ini pertanyaan kita terjawab karena berisi tentang beberapa hal yang umum dan sering digunakan dalam mengelola suatu database baik untuk perusahaan, instansi pemerintah maupun lembaga.

Sebagai sebuah software yang berfungsi untuk mengelola database, tentu MS. Access mempunyai tool atau komponen utama untuk mendukung fungsinya tersebut.. Berikut komponen utama dari MS. Access 2003 :



Gambar 19. Komponen Utama MS. Access 2003

Komponen	Fungsi
 Tables	Tabel , merupakan komponen inti yang berfungsi untuk menyimpan data dengan topic tertentu, Misalnya. Data mahasiswa disimpan pada tabel mahasiswa. Data disimpan dan disusun dalam bentuk baris(record) dan kolom (Field)
 Queries	Queries , berfungsi untuk menampilkan atau meminta data tertentu dari database, sesuai dengan yang diinginkan. Sebuah query dapat digunakan mengakses beberapa tabel secara bersamaan. Queries memegang peranan penting dalam mengelola sebuah database.
 Forms	Form , berfungsi untuk menampilkan, input, edit, hapus data dan lain sebagainya dengan tampilan antar muka (interface) yang lebih baik.
 Reports	Reports , sesuai dengan namanya, bagian ini berfungsi untuk menyajikan laporan, sehingga dapat dicetak pada layar monitor atau langsung ke printer.
 Pages	Pages , berfungsi untuk menyediakan database yang dibuat dengan access sehingga dapat juga di akses melalui internet explorer.
 Macros	Macros , merupakan rangkaian instruksi yang dapat dikenakan pada object yang ada pada form. Sebuah Macro dapat disimpan dan dijalankan ulang secara otomatis. Misal : Instruksi untuk membuka atau menutup sebuah form, instruksi untuk mencetak report pada form dan lain sebagainya.
 Modules	Modules , adalah sebuah code yang anda buat sesuai dengan aturan MS. Access, sebagai pendukung dari aplikasi yang dibuat.

Memulai Microsoft Access

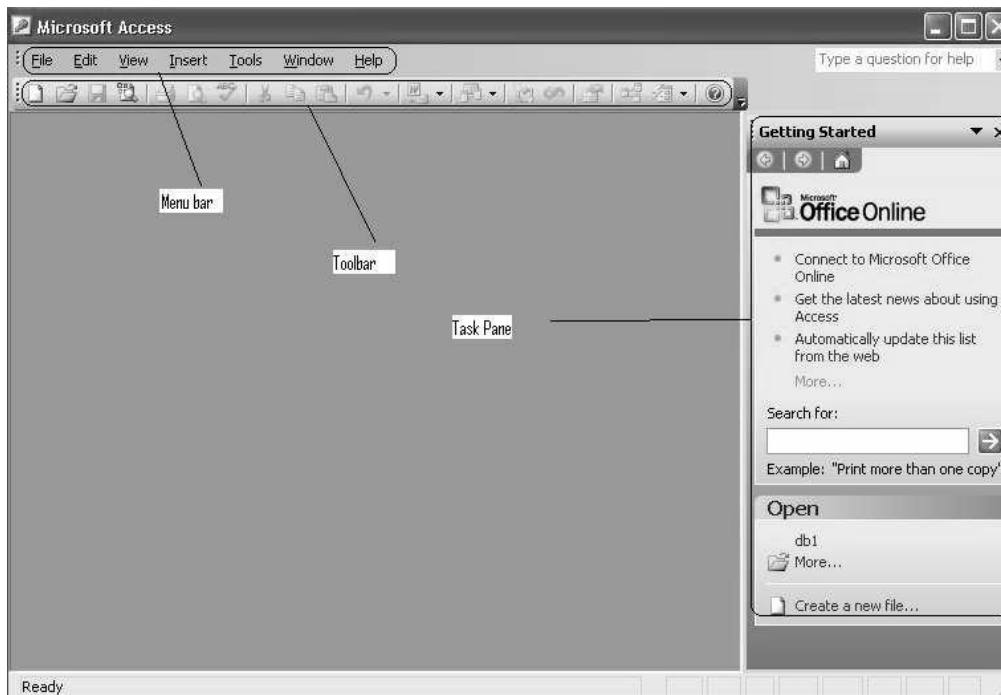
MS. Access baru dapat dijalankan apabila system operasi windows telah diaktifkan. Berikut langkah-langkah memulai bekerja dengan MS. Access 2003 :

- Dari Dekstop windows, klik **Start**, yang ada pada taskbar.
- Muncul sejumlah menu, lalu klik **All Program->Microsoft Office**
- Lalu klik **Microsoft Office Access 2003**, perhatikan gambar berikut :



Gambar 20. Menjalankan MS. Access

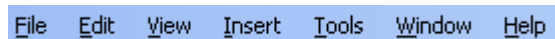
Muncul tampilan awal dari Microsoft Access. Pada tampilan ini kita dapat membuka file database (pilih nama file database, atau klik **more ...**, pilih nama file database). Akhiri dengan tekan tombol **OK**.



Gambar 21. Tampilan Awal Microsoft Access 2003

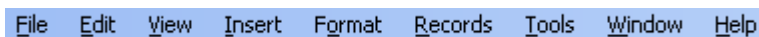
1. Menu Bar

berisi item-item menu yang dapat digunakan untuk mengelola database, tampilan item menu dapat berubah sesuai dengan object yang sedang di buka. Sewaktu pertama kali membuka MS. Access, maka item menu yang muncul adalah :



Gambar 22. Menu Bar

Bandingkan jika anda telah membuka sebuah tabel, maka tampilan item menu sebagai berikut :

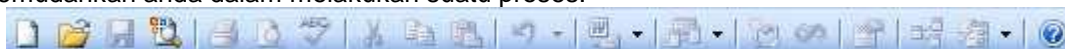


Gambar 23. Menu Bar Tabel

Terlihat adanya penambahan item menu, yaitu menu format dan Records. Setiap menu mempunyai submenu-submenu tersendiri, sesuai dengan fungsinya masing-masing.

2. Baris Toolbar

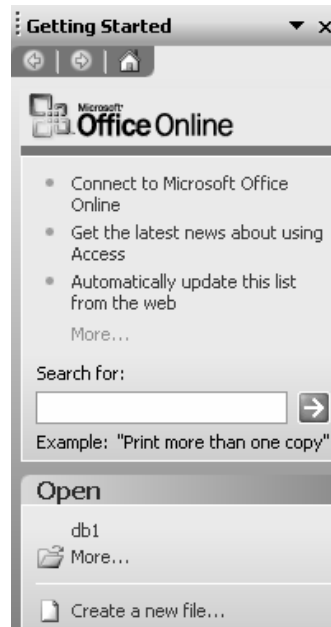
merupakan kumpulan icon-icon yang sering digunakan dalam mengelola database. Icon yang tampil pada toolbar juga ditentukan oleh object apa yang sedang dibuka. Icon-icon yang ada pada toolbar sebenarnya sudah ada pada item menu, tapi dengan adanya toolbar ini lebih memudahkan anda dalam melakukan suatu proses.



Gambar 24. Toolbar

3. Task Pane

merupakan fasilitas yang berfungsi untuk mempermudah anda bekerja dengan MS. Access, baik untuk membuat database baru, membuka database yang lama, atau menggunakan fasilitas template.



Gambar 25. Task Pane

Melalui Task Pane, anda bisa langsung terhubung ke Microsoft Office Online untuk mendapat informasi terbaru seputar Microsoft Office, disamping itu, anda juga bisa :

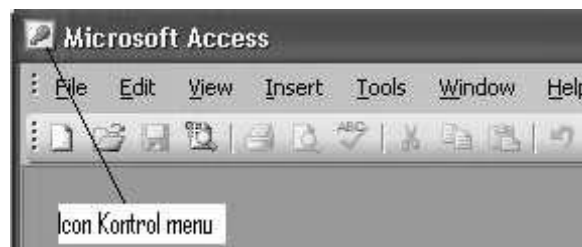
- **Membuka File Microsoft Access** yang telah anda buat, dengan mengklik link **More**.
- **Membuat File Database** Baru, dengan cara mengklik link **Create a new file..**

Untuk mengaktifkan atau menonaktifkan fasilitas Task Pane ini, dapat dilakukan dengan cara : **klik Menu View -> Toolbars -> Lalu Klik Task Pane**.

Mengakhiri MS. Access 2003

Jika anda telah selesai bekerja dengan MS. Access 2003, dapat diakhiri dengan beberapa cara yaitu :

- Pilih dan klik menu **File -> Exit**, atau
- Klik Tombol **Close ()** yang berada pada pojok kanan atas dari jendela MS. Access 2003, atau
- Klik ganda icon kontrol menu () yang berada pada pojok kiri atas dari jendela MS. Access 2003,



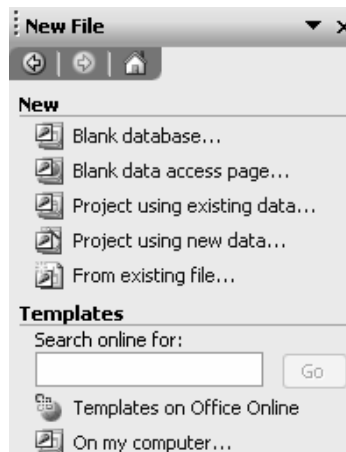
Gambar 26. Icon Kontrol Menu

atau

- Tekan kombinasi tombol **Alt-F4**.

Membuat Database

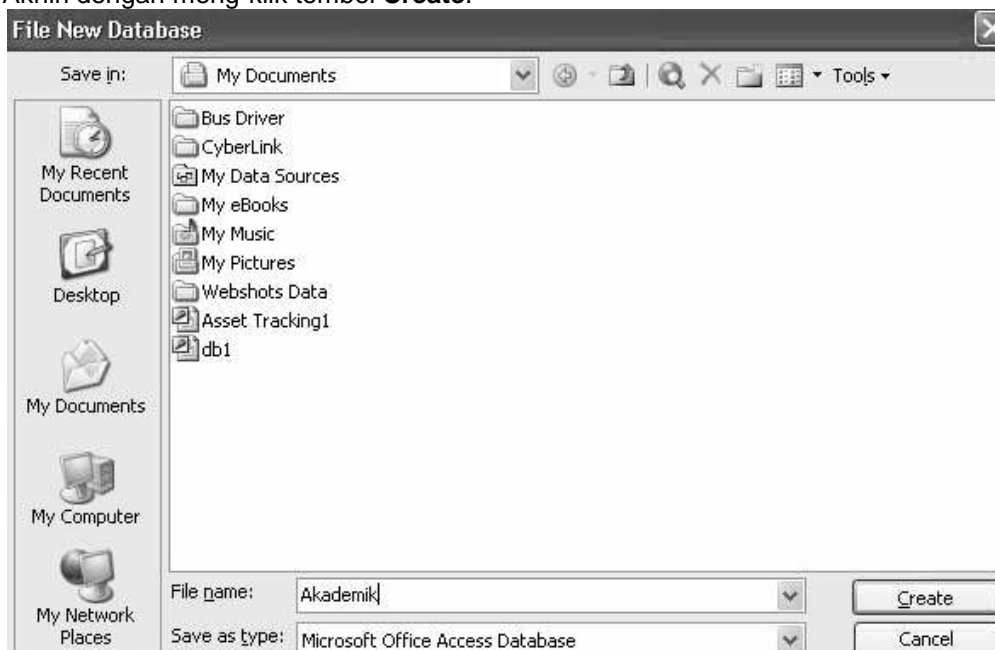
Untuk membuat Database baru dengan Microsoft Access 2003 cukup mudah, yaitu dengan cara meng-klik menu **File** kemudian diiringi dengan meng-klik menu **New** atau bisa juga dengan cara menekan tombol kombinasi **CTRL+N** secara bersamaan dari jendela utama Microsoft Access 2003. Secara otomatis, jendela Task Pane akan Muncul di sebelah kanan jendela Microsoft Access 2003. Seperti gambar berikut :

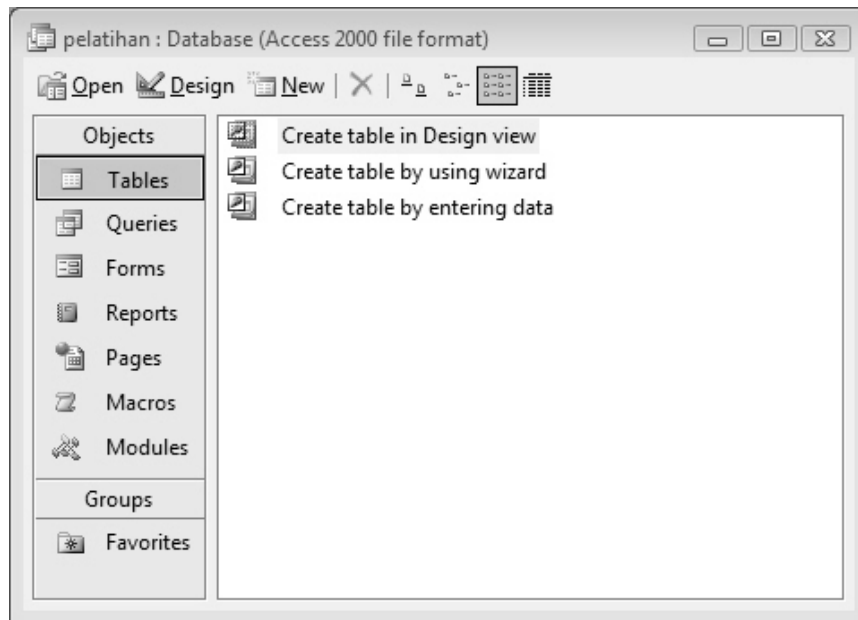
**Gambar 27. Blank Database**

Silahkan anda klik link **Blank database...** untuk membuat database baru yang betul-betul kosong, dengan cara ini akan menampilkan sebuah jendela baru untuk memasukkan nama dari database yang akan anda buat. Berikut ketentuan pemberian nama database dalam MS. Access 2003 :

- Nama database maksimal panjangnya 64 karakter.
- Boleh terdiri dari kombinasi huruf, angka, spasi dan spesial karakter seperti #, %, ^, dll, kecuali karakter titik (.), tanda seru (!), karakter ` dan tanda kurung siku ([]).
- Nama sebaiknya menggambarkan isi dari tabel.

Sesuai dengan kasus yang sedang anda selesaikan, yaitu Sistem Informasi Komunitas Adat, maka nama database yang akan anda buat nantinya diberi nama Database Komunitas Adat saja. Akhiri dengan meng-klik tombol **Create**.

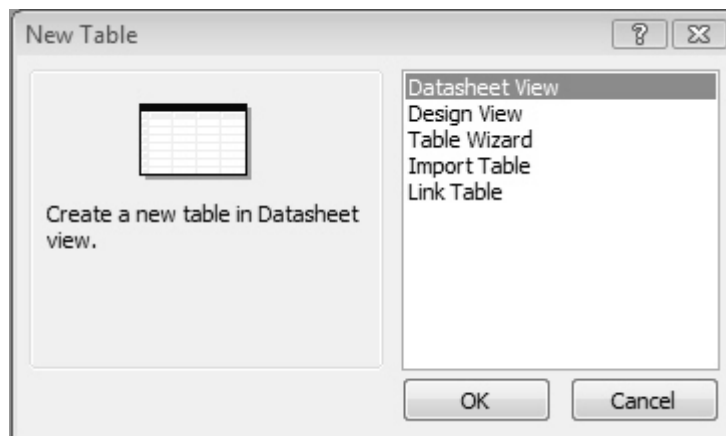
**Gambar 28. Penamaan Database**



Gambar 29. Database Windows

Tabel

Setelah membuat database baru maka akan terlihat database window, Klik **Tables**, pada bagian **Object**, Klik tombol **New**, muncul pilihan :



Gambar 30. New Table

Datasheet View – Membuat table baru dalam bentuk Datasheet View

Design View - Membuat table baru dalam bentuk Design View

Table Wizard – Langkah-langkah membuat table baru untuk menyimpan data.

Import Table – Mengimport table dari data external misal Excel, Foxpro dll

Link Table – Membuat table baru yang merupakan link dari table lain.

Atau bisa dengan cara :

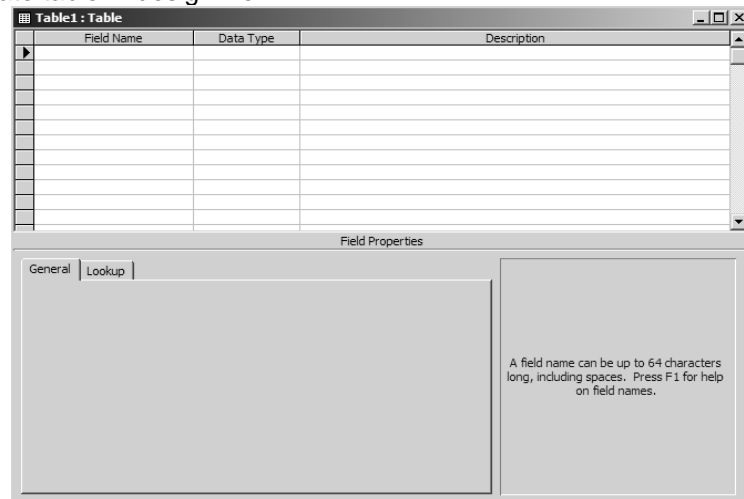
Double klik **Create table in Design view**, kita bisa membuat nama field, memilih type data dan formatnya untuk setiap field

Create table by using wizard, ikuti petunjuk yang ada di kotak dialog

Create table by entering data, dipilihan ini kita bisa langsung memasukan data dan formatnya sudah di atur

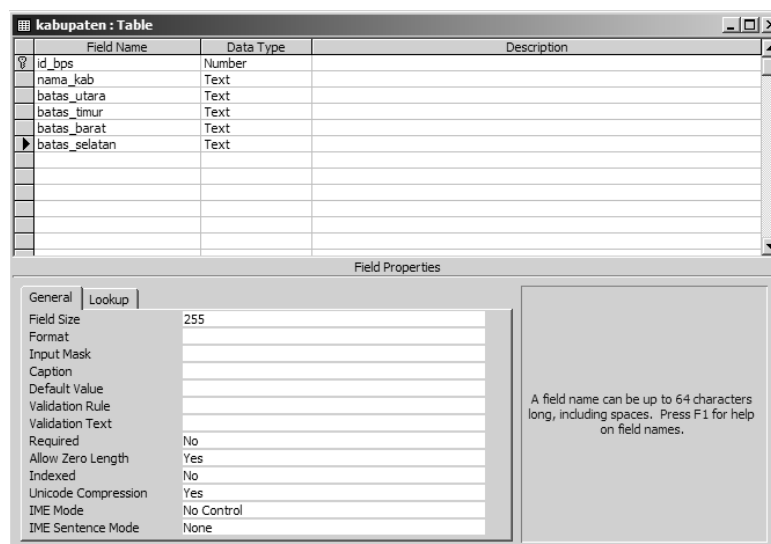
Create table in Design view

Pada jendela database klik Table.
klik dua kali create table in design view



Gambar 31. Table in Design View

Pada jendela table ketikkan field-field table kabupaten beserta tipe datanya, jangan lupa tentukan primary key. seperti gambar dibawah ini



Gambar 32. Isian Field

Format Tabel

Design tabel



Gambar 33. Menu Design

Primary key

Untuk memberikan primary key pada field klik gambar kunci yang ada di menu



Gambar 34. Primary Key

Formatting tipe data

	Field Name	Data Type
	id_bps	Number
	nama_kab	Text
	batas_utara	Text
	batas_timur	Memo
	batas_barat	Number
	batas_selatan	Date/Time
		Currency
		AutoNumber
		Yes/No
		OLE Object
		Hyperlink
		Lookup Wizard...

Gambar 35. Tipe Data**Tabel 4. Tipe Data**

NO	TIPE DATA	KETERANGAN	UKURAN
1	Text	Karakter alfanumeris (untuk data yang berisi rangkaian huruf, angka dan simbol)	0-255 karakter
2	Memo	Karakter alfanumeris.	0-65536 karakter
3	Number	Nilai numeris (angka)	<ol style="list-style-type: none"> 1. Byte, mampu menampung nilai data, dalam rentang 0 – 255. 2. Integer, mampu menampung data angka (bilangan bulat), dalam rentang –32,768 sampai dengan 32,767. 3. Long Integer, mampu menampung data angka (bilangan bulat), dalam rentang – 2,147,483,648 sampai dengan 2,147,483,647. 4. Single, mampu menampung data angka, dalam rentang – 3.402823E38 5. sampai dengan –1.401298E–45 untuk nilai negatif dan 1.401298E–45 sampai dengan 3.402823E38 untuk nilai positif, baik bilangan pecahan maupun bilangan bulat. 5. Double, mampu menampung data angka, dalam rentang – 1.79769313486231E308 sampai dengan –4.94065645841247E–324 untuk nilai negatif dan 1.79769313486231E308 sampai dengan 4.94065645841247E–324 untuk nilai positif, baik bilangan pecahan maupun bilangan bulat. 6. Decimal, mampu menampung data angka dari -10^{28} -1 sampai dengan 10^{28} –1.

4	Date/Time	Data tanggal dan waktu	sesuai dengan setting tanggal dan waktu pada komputer.
5	Currency	Data berupa mata uang	
6	AutoNumber	berisi data angka yang unik dan terurut serta otomatis. Type data ini tidak bisa diupdate.	
7	Yes/No	Nilai logika (untuk data yang berisi dua kemungkinan)	
8	OLE Objects	Untuk data yang berisi dari data eksternal misal gambar, suara, video dan lain-lain.	
9	Hyperlink	Link ke lokasi internet tertentu.	0-64000 karakter.
10	Lookup Wizard	Untuk data dimana sumber data tersebut tertentu misal dari field dari table lain atau pilihan dan hasil akhirnya berupa Combo Box.	

Latihan 3.

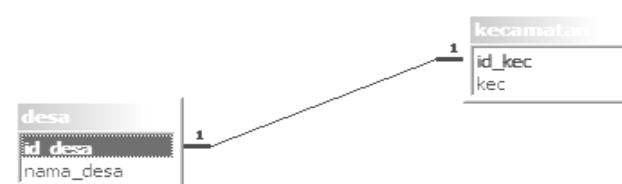
buatlah tabel-tabel selanjutnya dari data profil masyarakat adat

Relationship

Relationship adalah hubungan antara 2 atau lebih entitas/tabel. Hubungan antar entitas/tabel terdiri dari 3 macam, yaitu :

One to One (1-1)

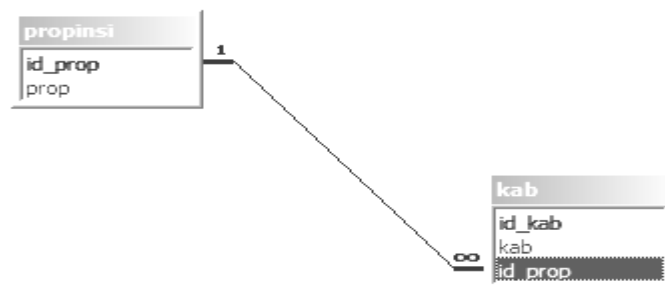
Contoh : 1 desa memiliki satu kecamatan



Gambar 36. Relasi One to One

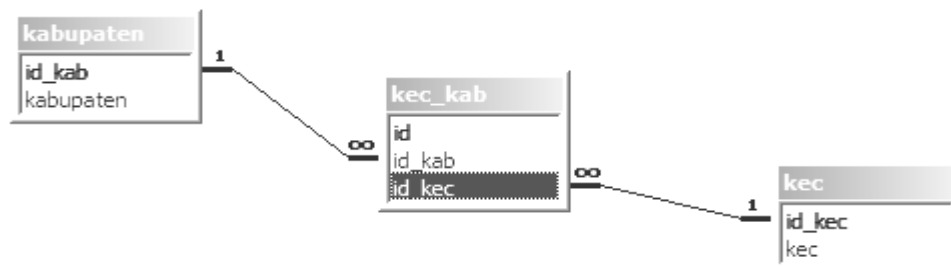
One to Many

Contoh : 1 provinsi memiliki banyak kabupaten

**Gambar 37. Relasi One to Many**

Many to Many

Bila relasi many to many maka dibuat “file konektor” yang berisi 2 foreign key yang berasal dari kedua entitas

**Gambar 38. Relasi Many to Many**

Cara Membuat Relationship

di dalam database window klik table

klik tombol relationship bisa di lihat pada gambar di bawah ini

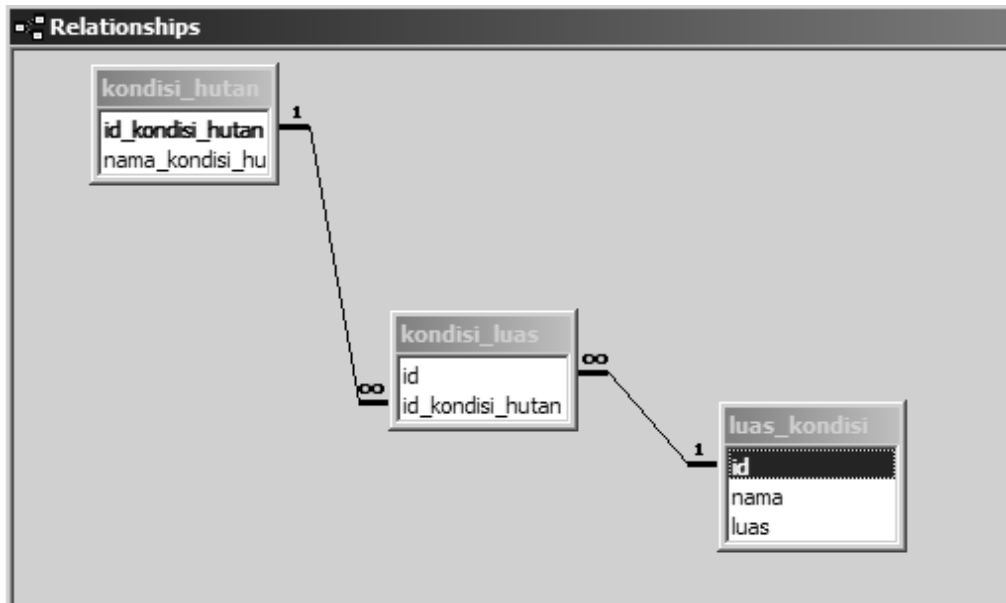
**Gambar 39. Tombol Relationship**

klik show table

**Gambar 40. menu show table**

pilih tabel yang akan direlasikan

klik dan drag field dari tabel satu ke tabel yang lain

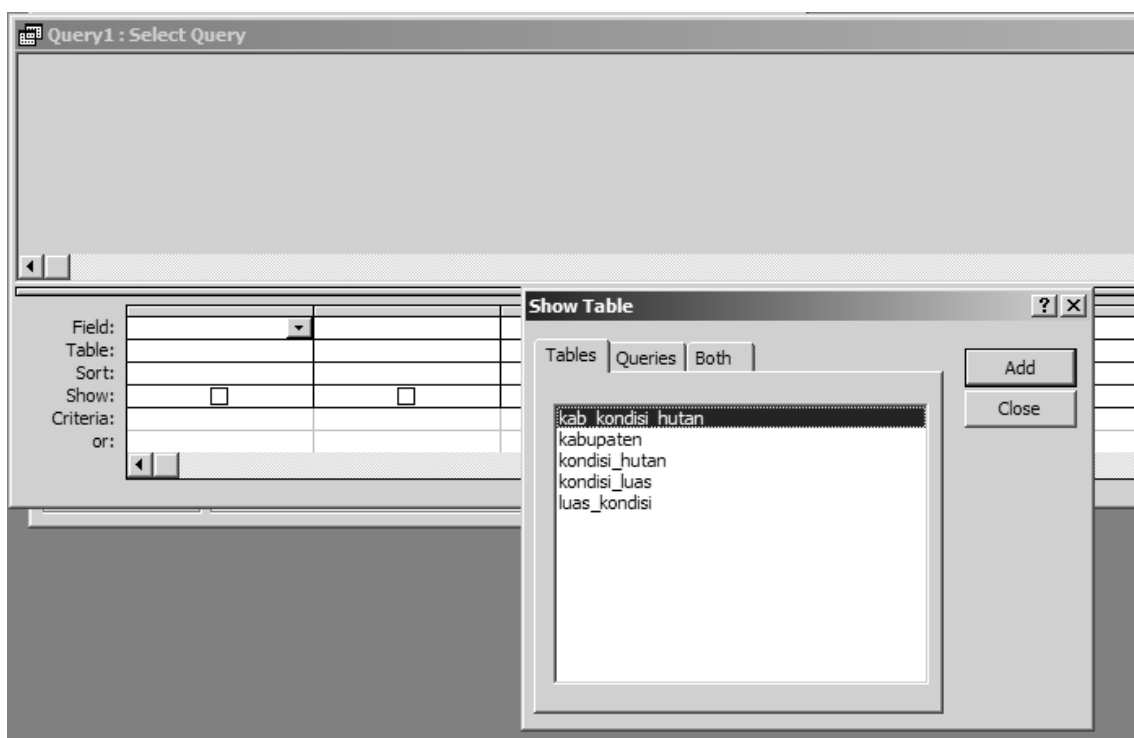


Gambar 41. Relationship

Query

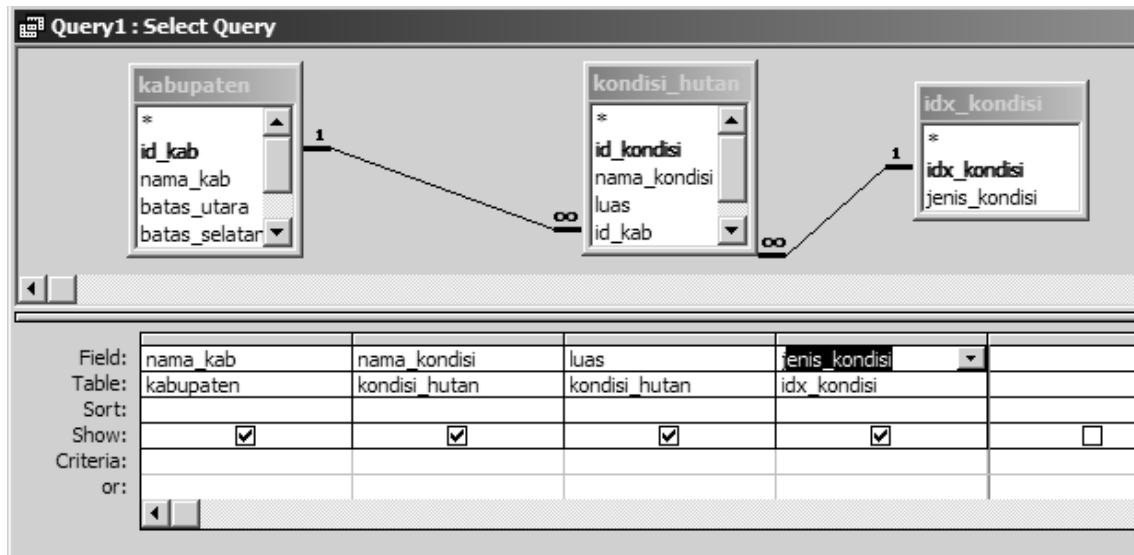
Design Query

Untuk membuat query, Ikuti langkah-langkah berikut
Klik jendela database Query
klik dua kali Create Query in Design View



Gambar 42. Show Table Query

Pilih Tabel yang akan di buat query nya



Gambar 43. Query Design

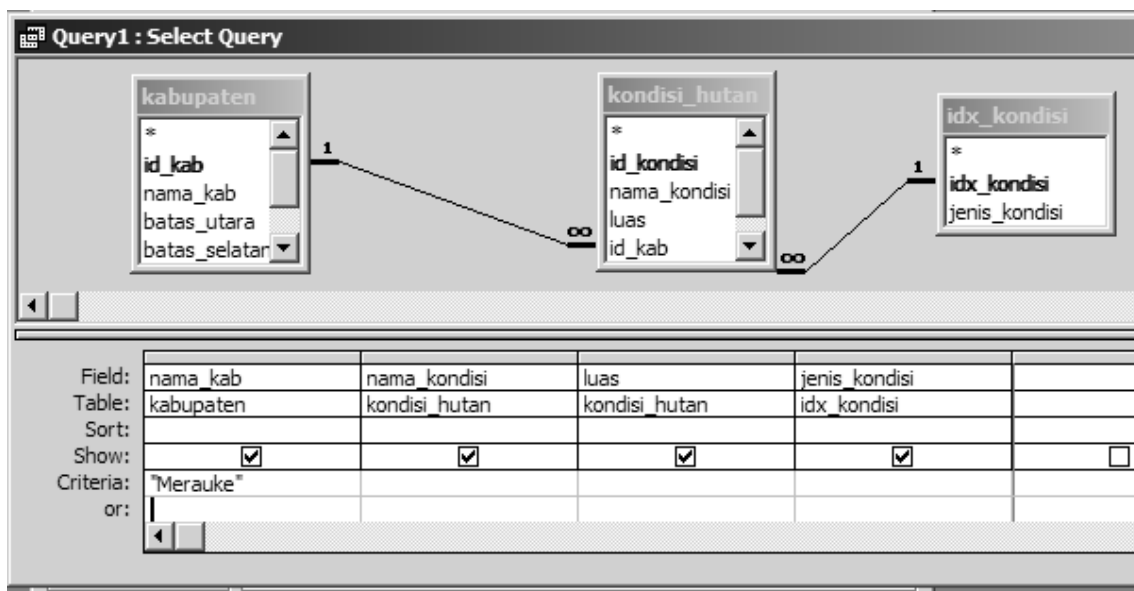
dari design query diatas maka setelah di eksekusi akan menghasilkan tabel sebagai berikut

Query1 : Select Query

	nama_kab	nama_kondisi	luas	jenis_kondisi
▶	MERAUKE	CAGAR ALAM MERAUKE	345030	CG
	JAYAWIJAYA	CAGAR ALAM JAYAWIJAYA	456030	CG
	MERAUKE	TAMAN NASIONAL MERAUKE	230900	TN
	MERAUKE	TAHURA MERAUKE	203000	TAHURA
	ASMAT	SUAKA MARGASATWA	234566	SM
*				

Gambar 44. Hasil Eksekusi dari Query

Jika kita hanya ingin menampilkan hanya di satu kabupaten saja maka kita bisa menggunakan kriteria diquery tersebut, misalnya kita masukan kriteria di nama kabupaten



Gambar 45. Kriteria di Query

hasilnya adalah hanya data di kabupaten Merauke saja yang akan tampil

nama_kab	nama_kondisi	luas	jenis_kondisi
MERAUKE	CAGAR ALAM MERAUKE	345030	CG
MERAUKE	TAMAN NASIONAL MERAUKE	230900	TN
MERAUKE	TAHURA MERAUKE	203000	TAHURA

Gambar 46. Hasil Eksekusi Query Berkriteria

Latihan 4.

1. Buatlah Query Untuk menampilkan komunitas di Pulau Sulawesi

Membuat Form

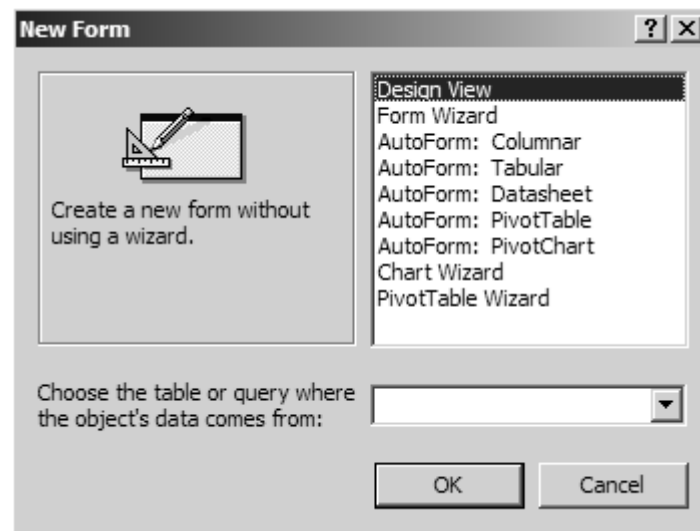
Untuk membuat template form ikuti langkah-langkah berikut ini :

Pilih Forms pada jendela database

Klik dua kali pada Create Form in Design View → membuat design template sesuai keinginan

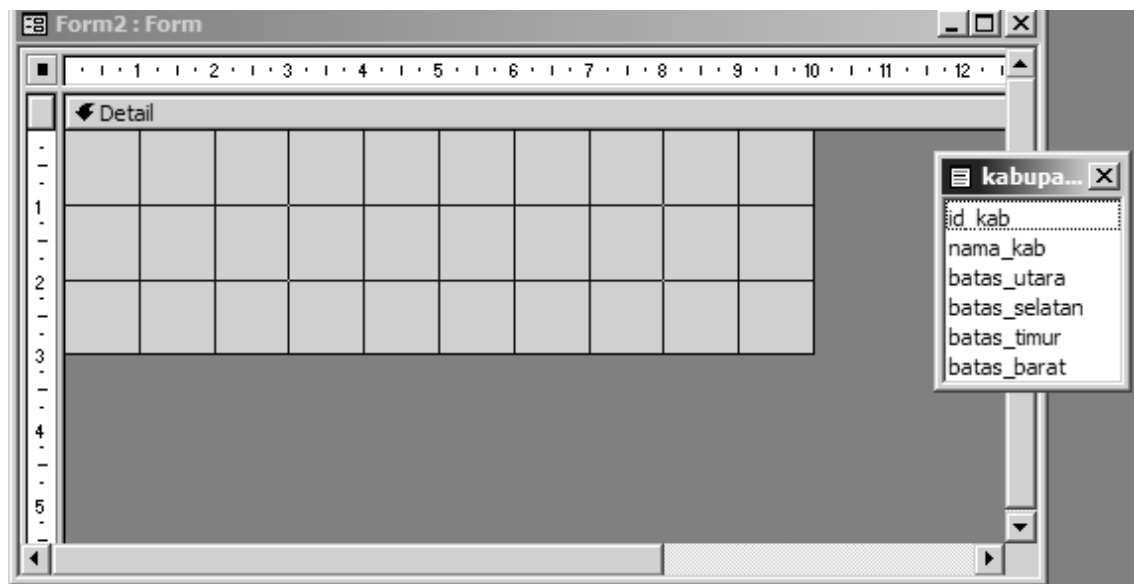
Gambar 47. Tampilan Form Design View

Klik New pada Jendela database → untuk membuat design form sesuai keinginan dan datanya diambil dari tabel atau query yang sudah dibuat



Gambar 48. Dialog Box New Form

Pada dropdown menu pilih tabel kabupaten kemudian klik OK maka akan muncul tampilan form seperti ini :

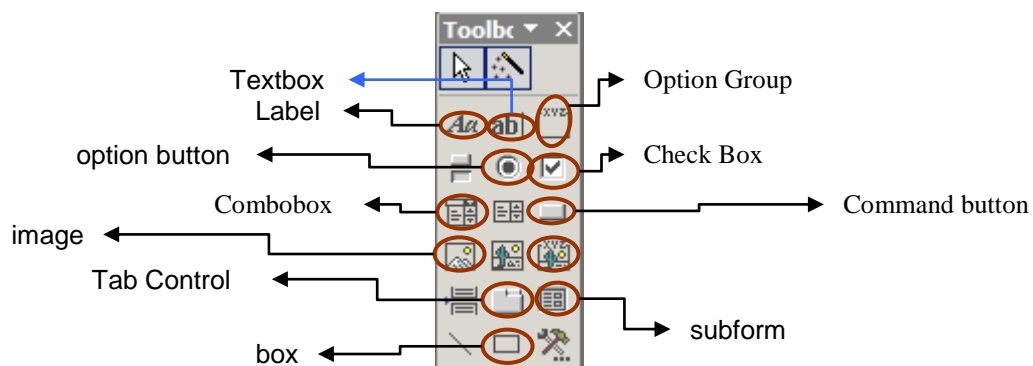


Gambar 49. Desain Form dengan data dari Tabel

kemudian Drag field-field yang ada di dalam tabel sehingga menghasilkan form di bawah ini

Gambar 50. Isian Form

Toolbox Form



Gambar 51. Toolbox pada Form

Mempercantik Tampilan Form

Untuk mempercantik tampilan form dapat dipercantik dengan mengubah properties formnya.

Gambar 52. Properties Form

Latihan 5

Buatlah Form dari tabel-tabel yang sudah di buat

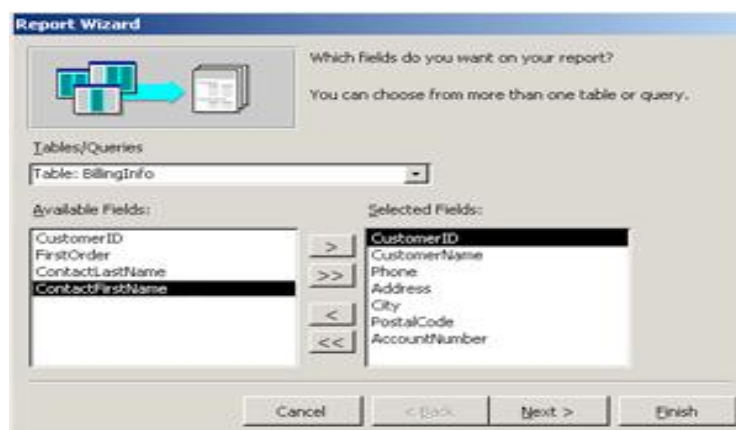
Report**Design Report Wizard**

Membuat Report dengan Report Wizard, ikuti langkah-langkah berikut:

klik dua kali pada jendela database, dan kotak dialog Report dialog akan muncul

di kotak dialog ini pilih tabel/query

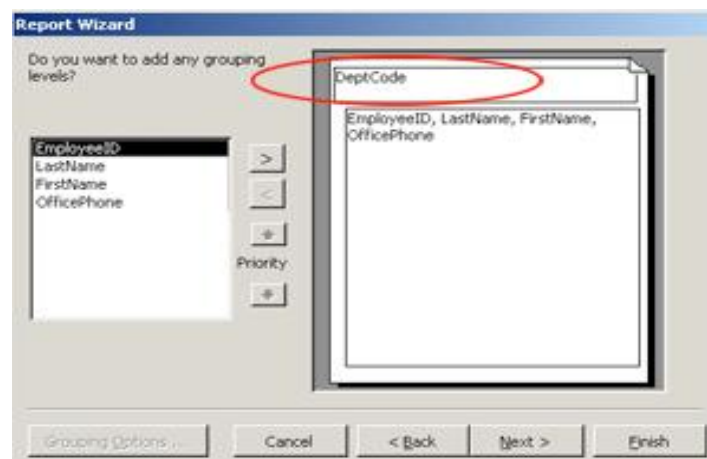
klik tanda > untuk memilih salah satu field yang akan di tampilkan di Report atau klik >> jika ingin semua field ditampilkan



Gambar 53. Report Wizard

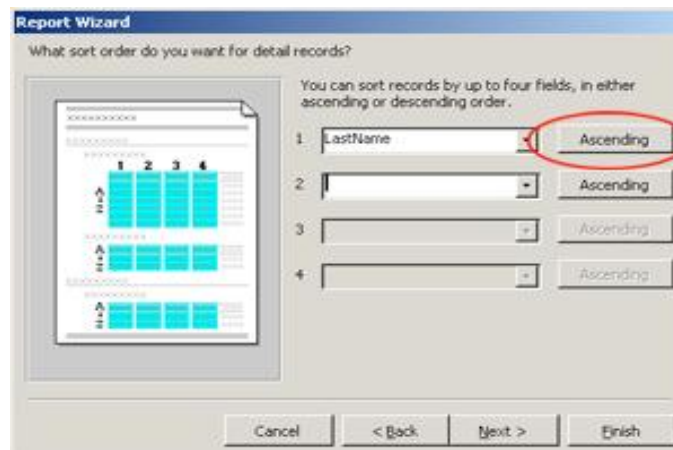
klik next

pada tahap selanjutnya kamu dapat membuat group penempatan field report dengan memilih salah satu atau lebih field kemudian klik next



Gambar 55. Report Wizard Grouping

sorting data anda berdasarkan Ascending atau Descending untuk report yang ingin di tampilkan



Gambar 56. Report Wizard Sorting

Klik Next, klik radio button pilih layout and orientation → klik next
Pilih Style yang diinginkan, Klik next
Ikuti Instruksi selanjutnya kemudian Klik Finish

Employees' Phones by Department

DeptCode	LastName	EmployeeID	Phone	OfficePhone
100	Deletions	0010	Cell	0000
	Deletions	0011	Any	0000
	Deletions	0012	Mobile	0000
	Deletions	0013	Mobile	0000
	Deletions	0014	Mobile	0000
	Deletions	0015	Mobile	0000
	Deletions	0016	Mobile	0000
	Deletions	0017	Mobile	0000
	Deletions	0018	Mobile	0000
	Deletions	0019	Mobile	0000
200	Deletions	0020	Mobile	0000
	Deletions	0021	Mobile	0000
	Deletions	0022	Mobile	0000
	Deletions	0023	Mobile	0000
	Deletions	0024	Mobile	0000
	Deletions	0025	Mobile	0000
	Deletions	0026	Mobile	0000
	Deletions	0027	Mobile	0000
	Deletions	0028	Mobile	0000
	Deletions	0029	Mobile	0000
	Deletions	0030	Mobile	0000
	Deletions	0031	Mobile	0000
	Deletions	0032	Mobile	0000
	Deletions	0033	Mobile	0000
	Deletions	0034	Mobile	0000
	Deletions	0035	Mobile	0000
	Deletions	0036	Mobile	0000
	Deletions	0037	Mobile	0000
	Deletions	0038	Mobile	0000
	Deletions	0039	Mobile	0000
	Deletions	0040	Mobile	0000
	Deletions	0041	Mobile	0000
	Deletions	0042	Mobile	0000
	Deletions	0043	Mobile	0000
	Deletions	0044	Mobile	0000
	Deletions	0045	Mobile	0000
	Deletions	0046	Mobile	0000
	Deletions	0047	Mobile	0000
	Deletions	0048	Mobile	0000
	Deletions	0049	Mobile	0000
	Deletions	0050	Mobile	0000
	Deletions	0051	Mobile	0000
	Deletions	0052	Mobile	0000
	Deletions	0053	Mobile	0000
	Deletions	0054	Mobile	0000
	Deletions	0055	Mobile	0000
	Deletions	0056	Mobile	0000
	Deletions	0057	Mobile	0000
	Deletions	0058	Mobile	0000
	Deletions	0059	Mobile	0000
	Deletions	0060	Mobile	0000
	Deletions	0061	Mobile	0000
	Deletions	0062	Mobile	0000
	Deletions	0063	Mobile	0000
	Deletions	0064	Mobile	0000
	Deletions	0065	Mobile	0000
	Deletions	0066	Mobile	0000
	Deletions	0067	Mobile	0000
	Deletions	0068	Mobile	0000
	Deletions	0069	Mobile	0000
	Deletions	0070	Mobile	0000
	Deletions	0071	Mobile	0000
	Deletions	0072	Mobile	0000
	Deletions	0073	Mobile	0000
	Deletions	0074	Mobile	0000
	Deletions	0075	Mobile	0000
	Deletions	0076	Mobile	0000
	Deletions	0077	Mobile	0000
	Deletions	0078	Mobile	0000
	Deletions	0079	Mobile	0000
	Deletions	0080	Mobile	0000
	Deletions	0081	Mobile	0000
	Deletions	0082	Mobile	0000
	Deletions	0083	Mobile	0000
	Deletions	0084	Mobile	0000
	Deletions	0085	Mobile	0000
	Deletions	0086	Mobile	0000
	Deletions	0087	Mobile	0000
	Deletions	0088	Mobile	0000
	Deletions	0089	Mobile	0000
	Deletions	0090	Mobile	0000
	Deletions	0091	Mobile	0000
	Deletions	0092	Mobile	0000
	Deletions	0093	Mobile	0000
	Deletions	0094	Mobile	0000
	Deletions	0095	Mobile	0000
	Deletions	0096	Mobile	0000
	Deletions	0097	Mobile	0000
	Deletions	0098	Mobile	0000
	Deletions	0099	Mobile	0000
	Deletions	0100	Mobile	0000

Monday, July 26, 2010 Page 1 of 2

Gambar 57. Contoh Hasil Report Wizard

Kamu dapat mengubah tampilan report kamu dengan mengklik design view, dan dapat menambahkan control lain misalnya textbox, label, listbox, combobox dll.

Latihan 6

Buat Report dari tabel/query database Profil Masyarakat adat

Daftar Pustaka

www.acrobatplanet.com

www.office.microsoft.com

Access Cookbook 2nd Edition, 2004

Pengantar Perancangan Sistem, Erlangga

Database Management