



Bab 4

Relasi Table dan Query

1. Membuat hubungan Beberapa Tabel

Tujuannya membuat hubungan antar table agar ada relasi antar table yang satu dengan yang lain. Sebagai contoh sistem informasi perpustakaan, yang terdiri table sebagai berikut: table buku, table peminjaman, table anggota.

Langkah-langkah :

- Buat table Buku, seperti dibawah ini :

	Field Name	Data Type
PK	KodeBuku	Text
	Judul	Text
	TahunTerbit	Number
	Eksemplar	Number
	Pengarang	Text

Gambar 4.1: Struktur table Buku

Dimana KodeBuku buat sebagai Primary Key.

- Buat Table Anggota sebgai berikut :

	Field Name	Data Type
PK	KodeAnggota	Text
	Nama	Text
	Alamat	Text
	Telpn	Text

Gambar 4.2: Struktur table Anggota

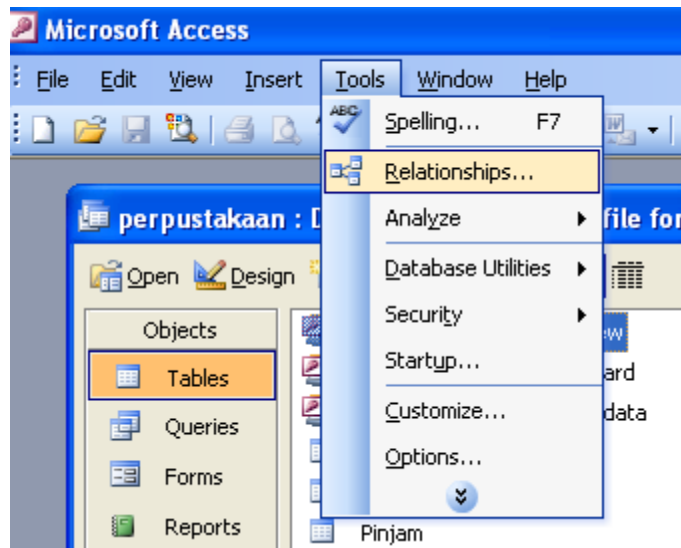
Dimana KodeAnggota buat sebagai Primary Key

- Buat table Peminjaman seperti dibawah ini :

Pinjam : Table	
Field Name	Data Type
NoPinjam	Text
Tanggal	Date/Time
KodeBuku	Text
KodeAnggota	Text

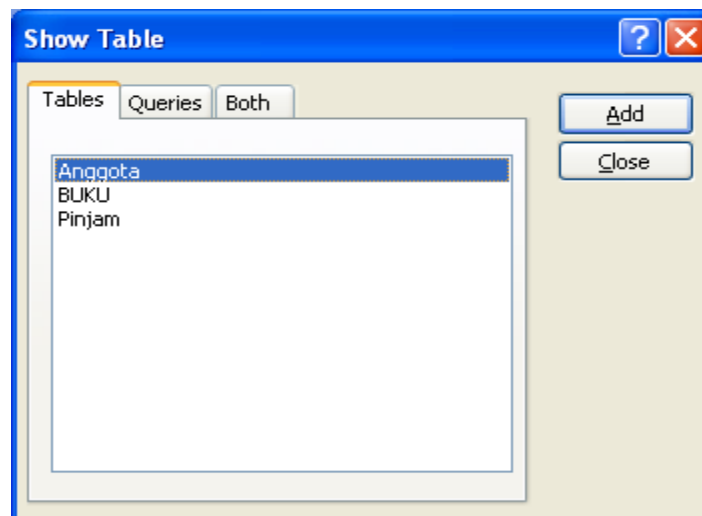
Gambar 4.3 : Struktur table Pinjam

- Relasikan dengan memilih menu Relationships berikut :



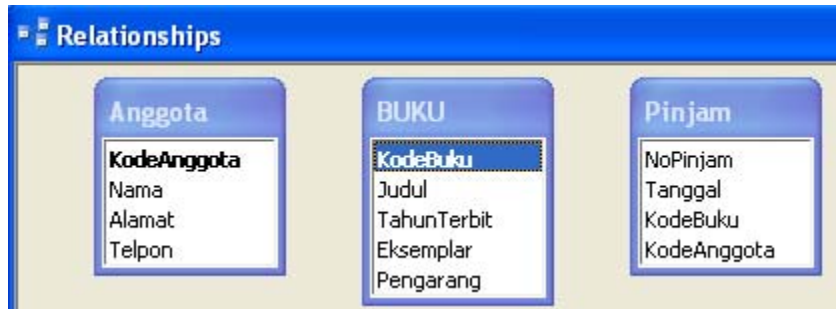
Gambar 4.4: Menu Relationships

Maka akan muncul table-table yang akan direlasikan, pilih table table tersebut :



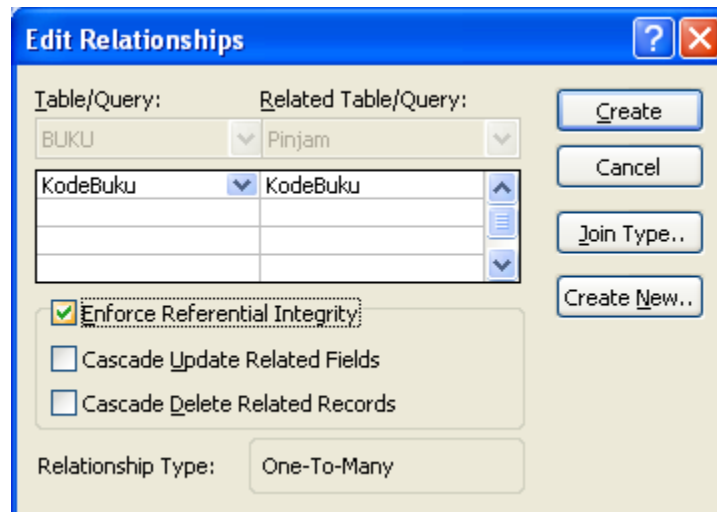
Gambar 4.5: Menambah Data Relasi

Sehingga table-table yang ditambahkan akan siap direlasikan seperti dalam gambar berikut :

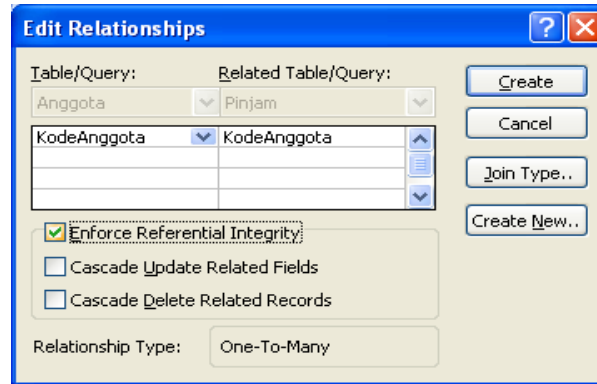


Gambar 4.6: Table-table yang akan direlasikan

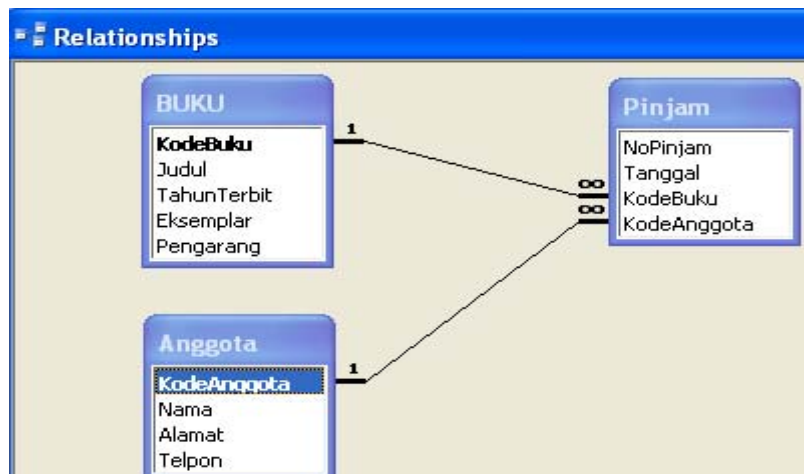
Tarik field dari tabel satu ketable lain yang memiliki field yang sama, maka tarik field KodeAnggota dari tabel Anggota ke Field KodeAnggota ke tabel Pinjam, tarik pula field KodeBuku dari tabel Buku ke field Kode Buku pada tabel Pinjam, lalu hubungkan antara field tersebut secara terpadu(integrity), maka klik **Enforce Referential Integrity** sebagai berikut:



Gambar 4.7: Menghubungkan relasi KodeBuku



Gambar 4.8: Menghubungkan relasi KodeAnggota



Gambar 4.9: Hasil Relasi tabel

- Lalu simpanlah hasil Relationships tersebut, klik file pilih save.
- Lalu Lakukan beberapa Entri data master untuk Buku dan Anggota seperti berikut ini :

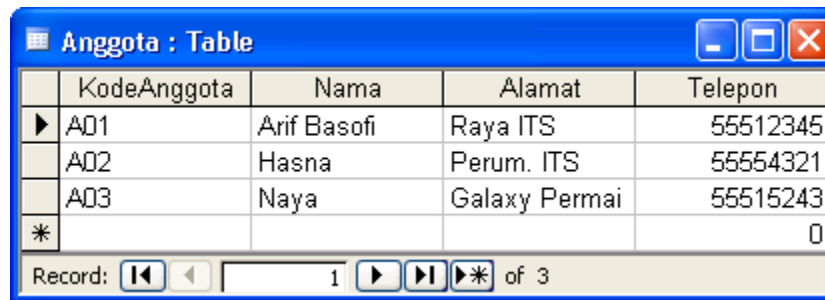
Dan entri data Buku sebagai berikut :

Buku : Table					
	KodeBuku	Judul	TahunTerbit	Eksemplar	Pengarang
▶	B01	Basis Data	2005	40	Arif Basofi
	B02	Administrasi Basis Data	2007	25	Arif Basofi
	B03	Image Processing	2004	20	Nana
	B04	Web Programming	2004	30	Farah
	B05	Pemrograman Visual Java	2003	25	Salsa
*			0	0	

Record: 1 of 5

Gambar 4.10: Entri data Buku

Dan entri data Anggota seperti dibawah ini :

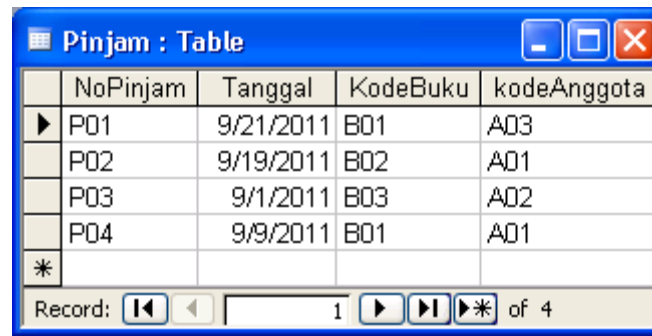


	KodeAnggota	Nama	Alamat	Telepon
▶	A01	Arif Basofi	Raya ITS	55512345
	A02	Hasna	Perum. ITS	55554321
	A03	Naya	Galaxy Permai	55515243
*				0

Record: 1 of 3

Gambar 4.11: Entri data anggota

- Lalu entri data peminjaman sebagai berikut :

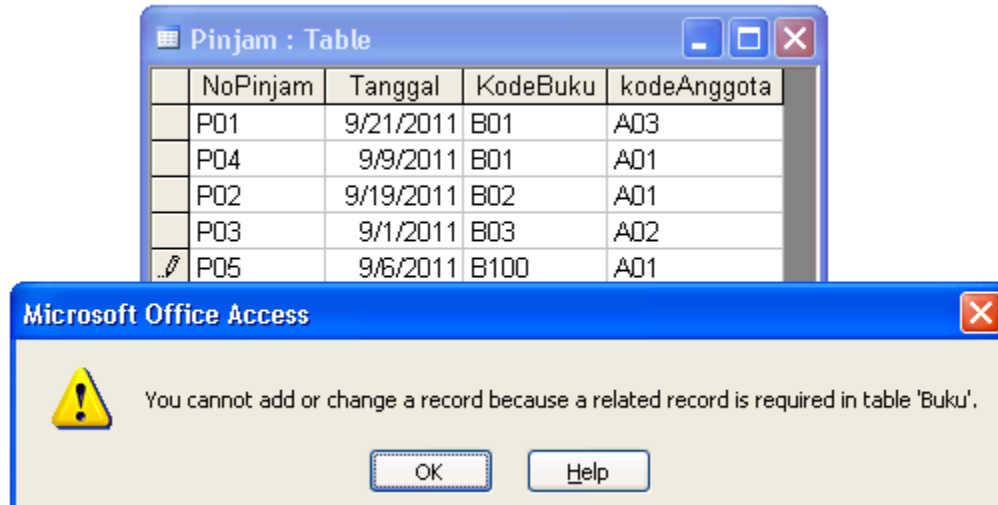


	NoPinjam	Tanggal	KodeBuku	kodeAnggota
▶	P01	9/21/2011	B01	A03
	P02	9/19/2011	B02	A01
	P03	9/1/2011	B03	A02
	P04	9/9/2011	B01	A01
*				

Record: 1 of 4

Gambar 4.12: Entri data Peminjaman

Tapi bila data yang dimasukkan ada **perbedaan** dari data referensi (master buku atau anggota) maka entri data transaksi akan **error** karena secara fisik terintegrasi antara table Pinjam, Buku dan Anggota artinya yang bisa sebagai sumber data hanya data-data yang ada dalam tabel Buku dan Anggota, selain itu tidak bisa dimasukkan. Lihat entri data yang tidak ada dalam data Buku atau Anggota, seperti dibawah ini KodeBuku A100 tidak terdapat dalam table Buku sehingga ada pesan error sebagai berikut :



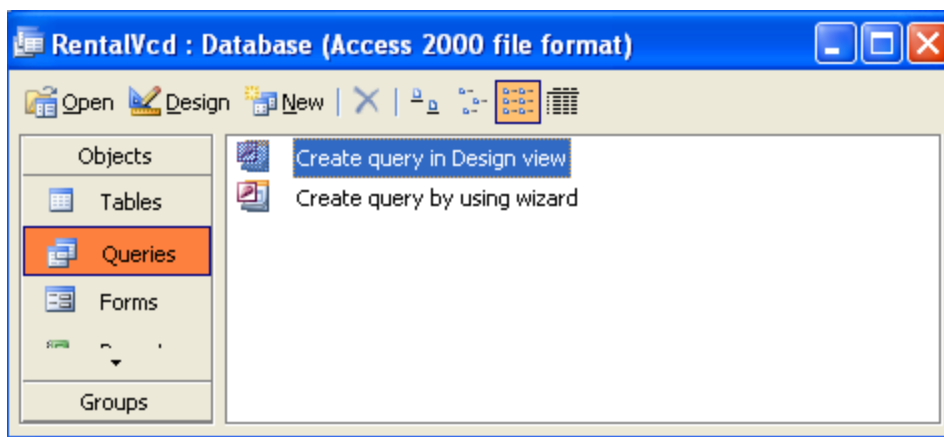
Gambar 4.13: Pesan error entri data tidak sesuai dengan refrensi table

2. Membuat Query

Membuat **Query** pada dasarnya adalah perintah untuk **menyajikan informasi** dengan menggunakan hubungan antara variable yang saling melengkapi informasinya, sehingga informasi bias sesuai dengan kebutuhan yang dicari maupun yang akan dilaporkan.

Langkah-langkah :

- Buka object/menu Query seperti sebagai berikut :



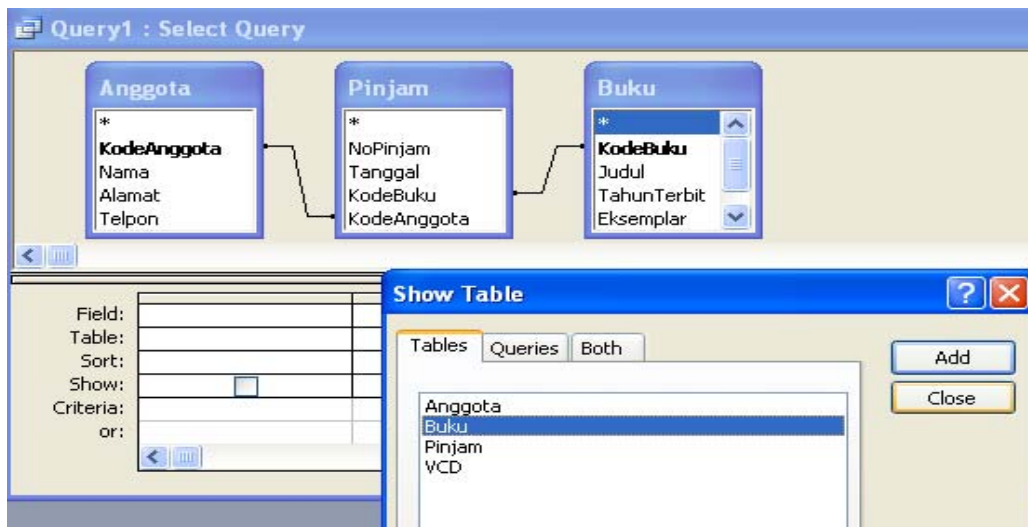
Gambar 4.14: Membuat Query

- Pilih Create Query in design, maka akan muncul dialog memilih query seperti dibawah ini :



Gambar 4.15: Memilih Query

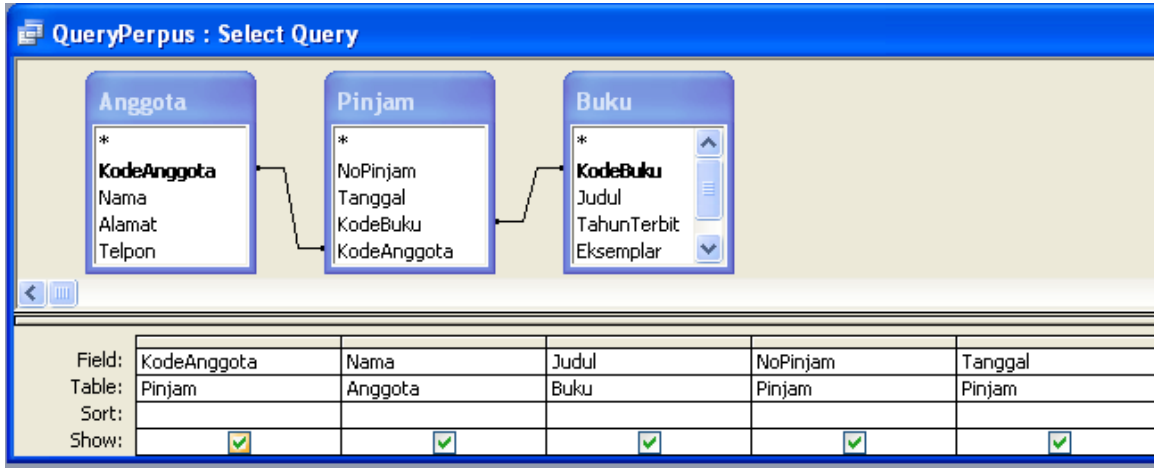
- Buka table Anggota, kemudian table Pinjam, dan table buku, maka akan muncul relasi table-table tersebut sebagai berikut :



Gambar 4.16: Hasil pemilihan query table

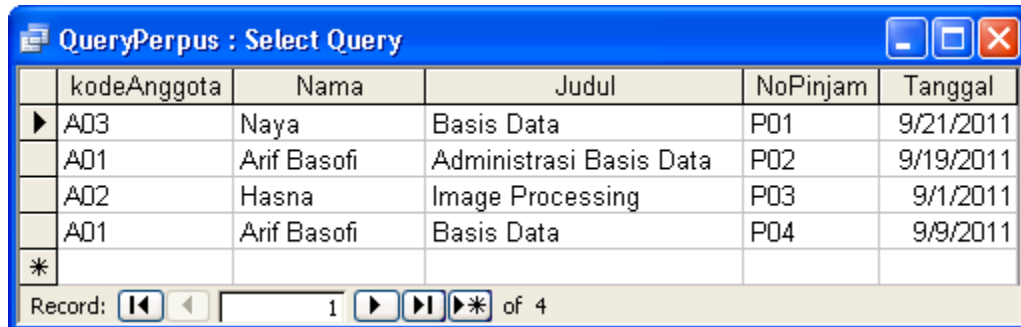
- Bila Antar table tidak terhubung secara langsung maka setelah table dipanggil, pilih field yang menjadi relasi antar table, pilih field KodeAnggota geser ke table Pinjam, dan field KodeBuku geser ke table Pinjam.

- Lalu tentukan informasi yang akan ditampilkan, pilih Field pada table yang berisi informasinya, seperti sebagai berikut :



Gambar 4.17: Menentukan informasi relasi antar table

- **Menyimpan Query** : Pilih File dan klik Save, lalu beri nama **QueryPerpus**
- **Menjalankan hasil Query** , Pilih menu **Query** dan pilih **Run**, maka akan muncul hasil query relasi antar table sebagai berikut :



Gambar 4.18 : Hasil Run Query

- Bila ingin menyajikan hasil query dengan memberikan **criteria**, misalnya yang pinjam anggota dengan KodeAnggota A01 maka pada **Criteria** isi dengan **=”A01”** bila sebagai criteria berupa **data teks**, harus diapit tanda **“...”**, bila **numeric tanpa tanda “...”**.

Field:	kodeAnggota	Nama	Judul	NoPinjam	Tanggal
Table:	Pinjam	Anggota	Buku	Pinjam	Pinjam
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"A01"				
or:					

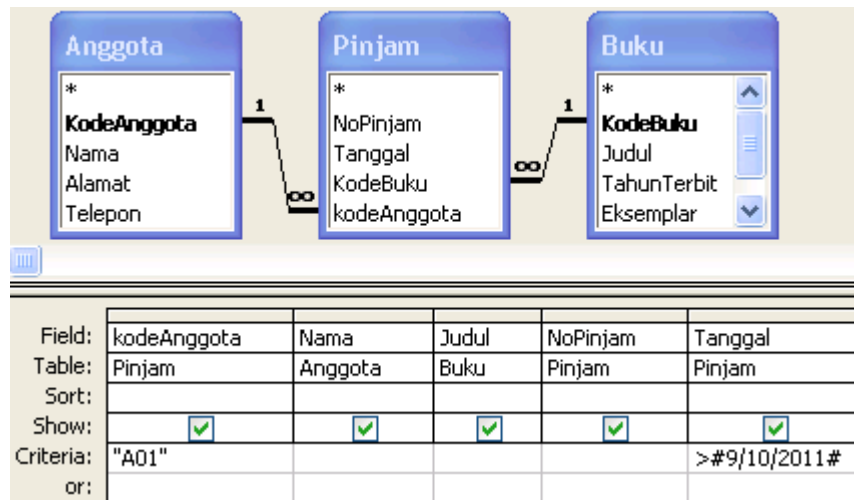
	kodeAnggota	Nama	Judul	NoPinjam	Tanggal
▶	A01	Arif Basofi	Administrasi Basis Data	P02	9/19/2011
	A01	Arif Basofi	Basis Data	P04	9/9/2011
*					

- Bila ingin menyajikan data peminjam yang meminjamnya mulai tanggal **10 September 2011** maka pada **Criteria** tanggal isi dengan **>#9/10/2011#**, nampak seperti dibawah ini :

Field:	kodeAnggota	Nama	Judul	NoPinjam	Tanggal
Table:	Pinjam	Anggota	Buku	Pinjam	Pinjam
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					>#9/10/2011#
or:					

	kodeAnggota	Nama	Judul	NoPinjam	Tanggal
▶	A03	Naya	Basis Data	P01	9/21/2011
	A01	Arif Basofi	Administrasi Basis Data	P02	9/19/2011
*					

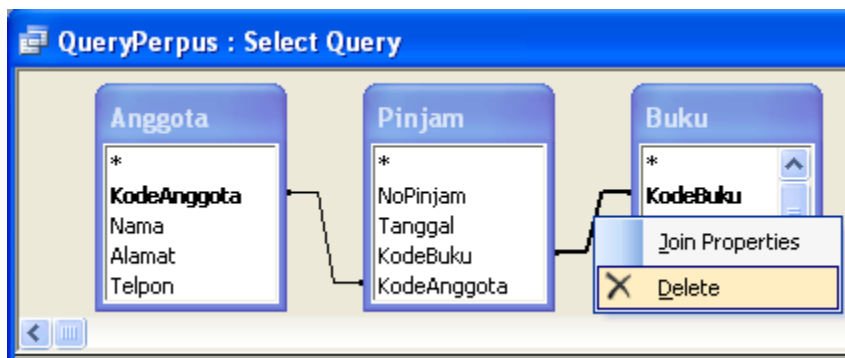
Gambar 4.19 : Query dengan Kriteria



	kodeAnggota	Nama	Judul	NoPinjam	Tanggal
▶	A01	Arif Basofi	Administrasi Basis Data	P02	9/19/2011
*					

3. Menghapus Relationships antar table

Bila ingin menghapus relasi antar table mak, hubungan antar table tersebut, klik garis penghubung antar table atau ditunjuk dengan mouse lalu klik kanan maka akan muncul seperti dibawah ini, lalu pilih Delete, maka tanda [penghubung akan terhapus.



Gamabr 4.20: Menghapus Relasi Query

4. Menutup kotak dialog relationship

Menutup kotak dialog relationships berarti mengakhiri rangkaian perintah relationships.

Langkah-langkah :

- Klik Menu File - Pilih menu Close
- Bila perancangan query tersebut belum disimpan maka akan ada konfirmasi apakah perintah relasi query tersebut disimpan atau tidak, jika ingin menyimpan pilih yes, dan beri nama query.

Latihan Soal :

1. Buat table master Pegawai sebagai berikut :

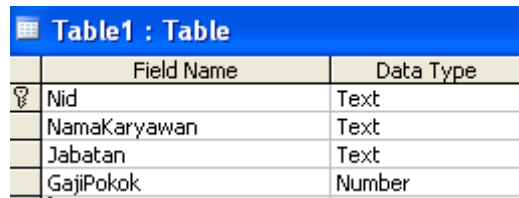


Table1 : Table		
	Field Name	Data Type
PK	Nid	Text
	NamaKaryawan	Text
	Jabatan	Text
	GajiPokok	Number

Gambar 4.21. Table Pegawai

Catatan : Buat Nid menjadi Primary Key

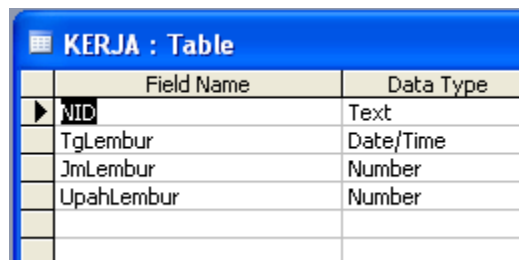
Dan entri data-datanya sebagai berikut :



Pegawai : Table				
	Nid	NamaKaryawan	Jabatan	GajiPokok
	01	Jupri	Kuli	35000
	02	Sunaryo	Kuli	35000
	03	Budi	Tukang	50000
*				0

Gambar 4.22. Data Pegawai

2. Buat table Kerja sebagai berikut :



KERJA : Table		
	Field Name	Data Type
PK	NID	Text
	TgLembur	Date/Time
	JmLembur	Number
	UpahLembur	Number

Gambar 4.23. Table Kerja

Dan entri data-datanya sebagai berikut :

KERJA : Table				
	NID	TgLembur	JmLembur	UpahLembur
	01	2/22/2006	10	10000
	02	2/22/2006	12	10000
	03	2/22/2006	10	20000
▶			0	0

Gambar 4.24. Data Kerja

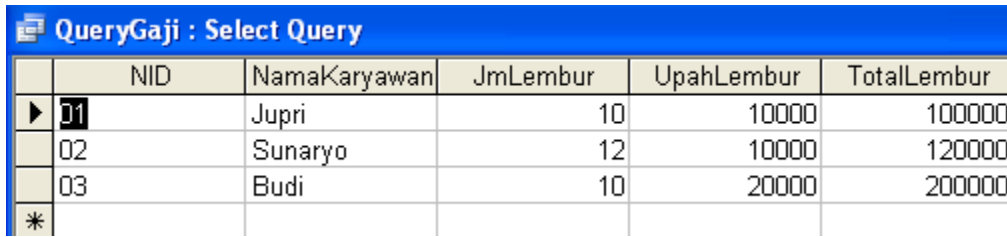
3. Relasikan ke dua tabel diatas yaitu table Pegawai dan table Kerja secara terpadu.
4. Gunakan query untuk mencari siapa saja nama pegawai yang bekerja, gaji pokoknya berapa, jumlah jam lemburnya.
5. Tampilkan hasil query dengan criteria, bagi pegawai yang jam lemburnya lebih dari 10 jam.
6. Tampilkan informasi Total Lembur, tambahkann rumus perhitungan pada field seperti dibawah ini:

TotalLembur:=[jmLembur]*[UpahLembur]

QueryGaji : Select Query				
	Pegawai		KERJA	
	* Nid NamaKaryawan Jabatan GajiPokok	←	* NID TgLembur JmLembur UpahLembur	
Field:	NamaKaryawan	JmLembur	UpahLembur	TotalLembur:=[jmLembur]*[UpahLembur]
Table:	Pegawai	KERJA	KERJA	
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				
or:				

Gambar 25. Relasi Query

Sehingga hasilnya sebagai berikut :



	NID	NamaKaryawan	JmLembur	UpahLembur	TotalLembur
▶	01	Jupri	10	10000	100000
	02	Sunaryo	12	10000	120000
	03	Budi	10	20000	200000
*					

Gambar 25: Informasi data hasil relasi Query

7. Hitung juga Gaji Total dengan rumus :

$$\text{GajiTotal} := [\text{GajiPokok}] + [\text{jmLembur}] * [\text{UpahLembur}]$$