

# Pemrograman dengan C++ Builder

## 14.1. Definisi Basis Data Relational

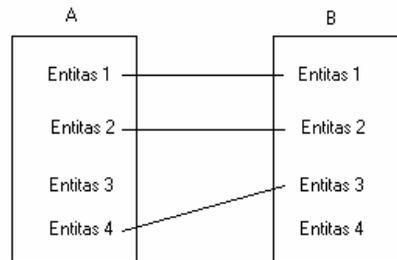
Pada model Relational, basis data akan 'disebar' ke dalam berbagai tabel 2 dimensi. Setiap tabel selalu terdiri atas lajur mendatar yang disebut dengan Baris Data (Row/record) dan lajur vertical yang biasa disebut dengan Kolom (Column/Field). Di setiap pertemuan baris data dan kolom itulah, item-item data ditempatkan.

## 14.2. Relasi Antar Tabel

Ciri utama Relational Database adalah adanya relasi antar tabel-tabelnya. Dikenal ada beberapa jenis relasi:

### a. One to One (satu ke satu)

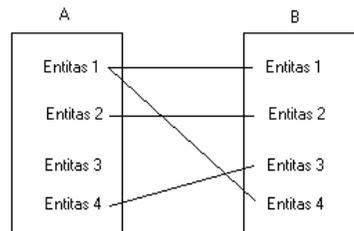
Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan Entitas A.



Gambar 14.1. Relasi One to One

### b. One to Many

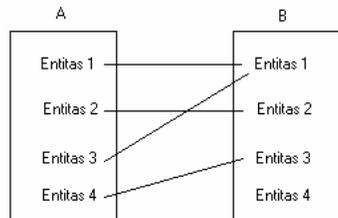
Berarti setiap entitas pada himpunan Entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.



Gambar 14.2. Relasi One to Many

### c. Many to One (banyak ke satu)

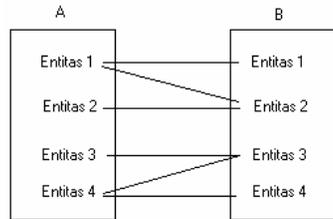
Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan paling banyak satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.



Gambar 14.3. Relasi Many to One

**d. Many to Many (banyak ke banyak)**

Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B dan demikian juga sebaliknya, dimana entitas pada himpunan entitas B dapat berhubungan dengan banyak elemen entitas pada himpunan entitas A.

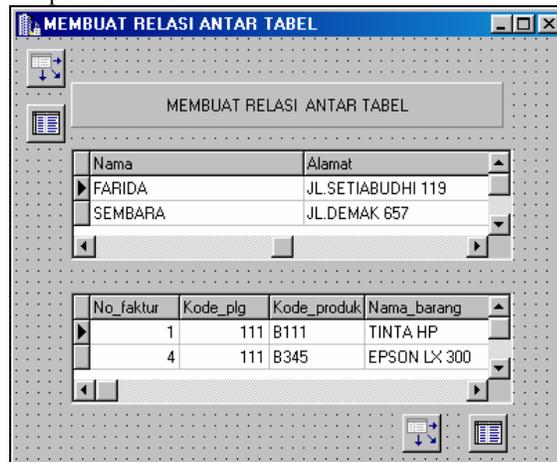


Gambar 14.4. Relasi Many to Many

Kardinalitas Relasi satu ke Banyak dan Banyak ke Satu dapat dianggap sama, karena tujuan kardinalitas relasi selalu dilihat dari dua sisi (dari himpunan entitas A ke himpunan Entitas B dan sebaliknya).

Contoh berikut menunjukkan relasi one-to-one.

1. Buatlah Form seperti berikut:



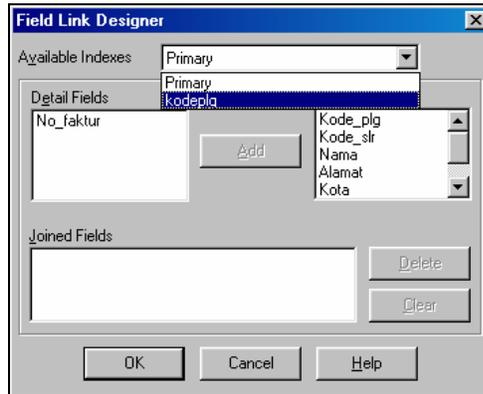
Gambar 14.5 Desain Form Relasi One to One

2. Ganti masing property dari komponen dengan data seperti berikut:

Tabel 14.1. Mengganti Properti Objek

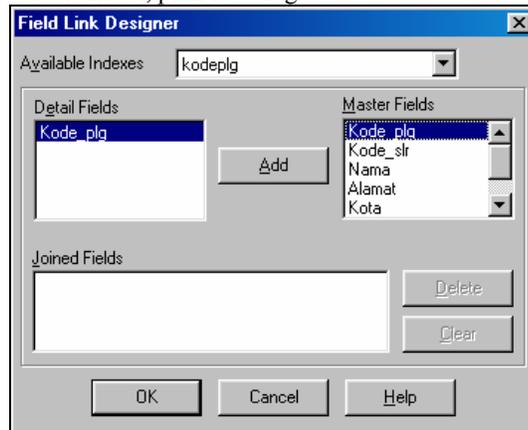
Komponen	Properti
Table1	DataBaseName: BCBDefaultDD TableName: PELANGGAN.DB Active:true
DataSource1	DataSet:Table1
DBGrid1	DataSource: DataSource1
Table2	DataBaseName: BCBDefaultDD TabelName: TRANSAKSI.DB Active:true MasterSource: DataSource1
DataSource2	DataSet:Table2
DBGrid2	DataSource: DataSource2

- Untuk membuat Link, berdasarkan kode pelanggan caranya: Klik komponen Table2. kemudian akan ditampilkan kotak dialog Field Link Designer, seperti berikut:



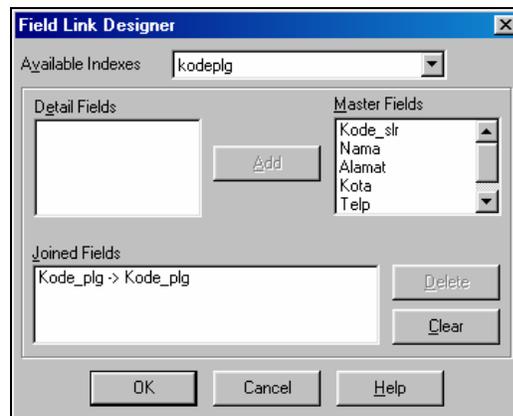
Gambar 14.6. Kotak Dialog Field Link Designer

- Pada kotak Available Index, pilih KodePlg



Gambar 14.7. Pilih Field kodeplg

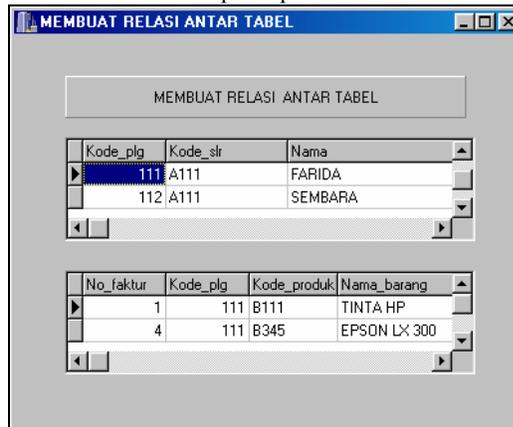
- Pada kotak Detail Field Klik Kode\_plg, begitu juga dengan kotak master field, klik Kode\_plg, kemudian tekan tombol Add, akan ditampilkan jendela seperti berikut:



Gambar 14.8. Pilih field Kode\_plg

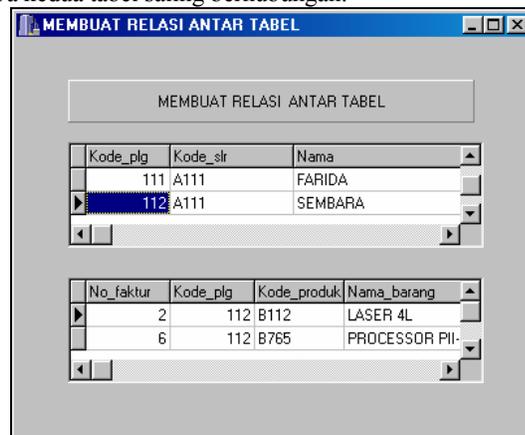
6. Klik tombol Ok

Jika Anda RUN, maka akan dihasilkan output seperti berikut:



Gambar 14.9. Output Relasi One to One

Jika Anda menunjuk record dengan kode pelanggan tertentu maka pada tabel yang ada dibawahnya akan ditampilkan tabel data dengan kode pelanggan yang sama, hal ini menunjukan bahwa kedua tabel saling berhubungan.



Gambar 14.10. Contoh Hasil Relasi One to One

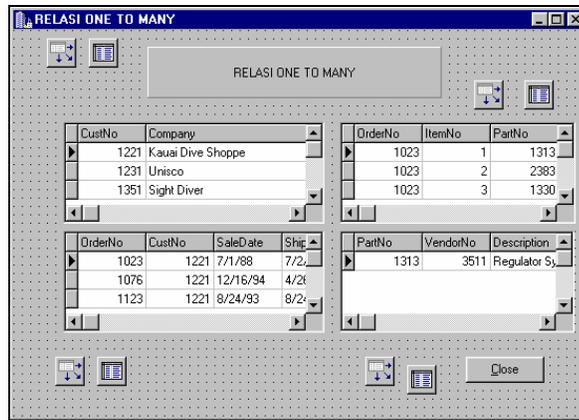
Nb: Buat Secondary Index di Database untuk file transaksi

# Pemrograman dengan C++ Builder

## 14.3. Relasi One to Many

Berikut ini akan kita coba bagaimana cara menyusun sebuah program untuk relasi one-to-many.

1. Buat Form seperti berikut:



Gambar 14.11. Desain Form Relasi One to Many

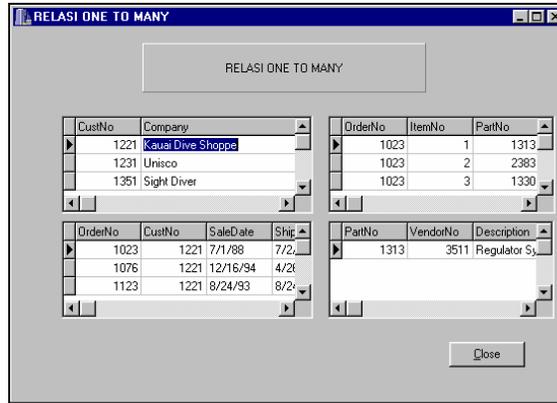
2. Masukkan properti berikut pada masing-masing komponen

Tabel 14.2. Mengganti Properti Objek

Komponen	Properti	Nilai
Table1	DataBaseName	DBDEMOS
	TableName	CUSTOMER.DB
	Active	True
Table2	DataBaseName	DBDEMOS
	TableName	ORDER.DB
	Active	True
	MasterField	CustNo
	MasterSource	DataSource1
Table3	DataBaseName	DBDEMOS
	TableName	ITEM.DB
	Active	True
	MasterField	OrderNo
	MasterSource	DataSource2
Table4	DataBaseName	DBDEMOS
	TableName	PART.DB
	Active	True
	MasterField	PartNo
	MasterSource	DataSource3
DataSource1	DataSet	Table1
DataSource2	DataSet	Table2
DataSource3	DataSet	Table3
DataSource4	DataSet	Table4
DBGrid1	DataSource	DataSource1
DBGrid2	DataSource	DataSource2
DBGrid3	DataSource	DataSource3
DBGrid4	DataSource	DataSource4
Button1	Caption	Close

Jika Program di atas, Anda RUN, maka akan menghasilkan output seperti berikut:

# Pemrograman dengan C++ Builder



Gambar 14.12. Output Relasi One to Many