

*Pertemuan ke 14*

# *SQL (Structure Query Language)*



## ***Pemrograman 2***

Dosen : Eko Budi Setiawan, S.Kom

**Universitas Komputer Indonesia**

## *Apa itu SQL ?*

- ❑ SQL kependekan dari Structured Query Language
- ❑ SQL digunakan sebagai bahasa komunikasi dengan sebuah database (server database)
- ❑ SQL yang akan diterangkan dalam perkuliahan ini adalah SQL yang didukung oleh BDE

## *Jenis - jenis perintah SQL*

### ❑ **DDL (Data Definition Language)**

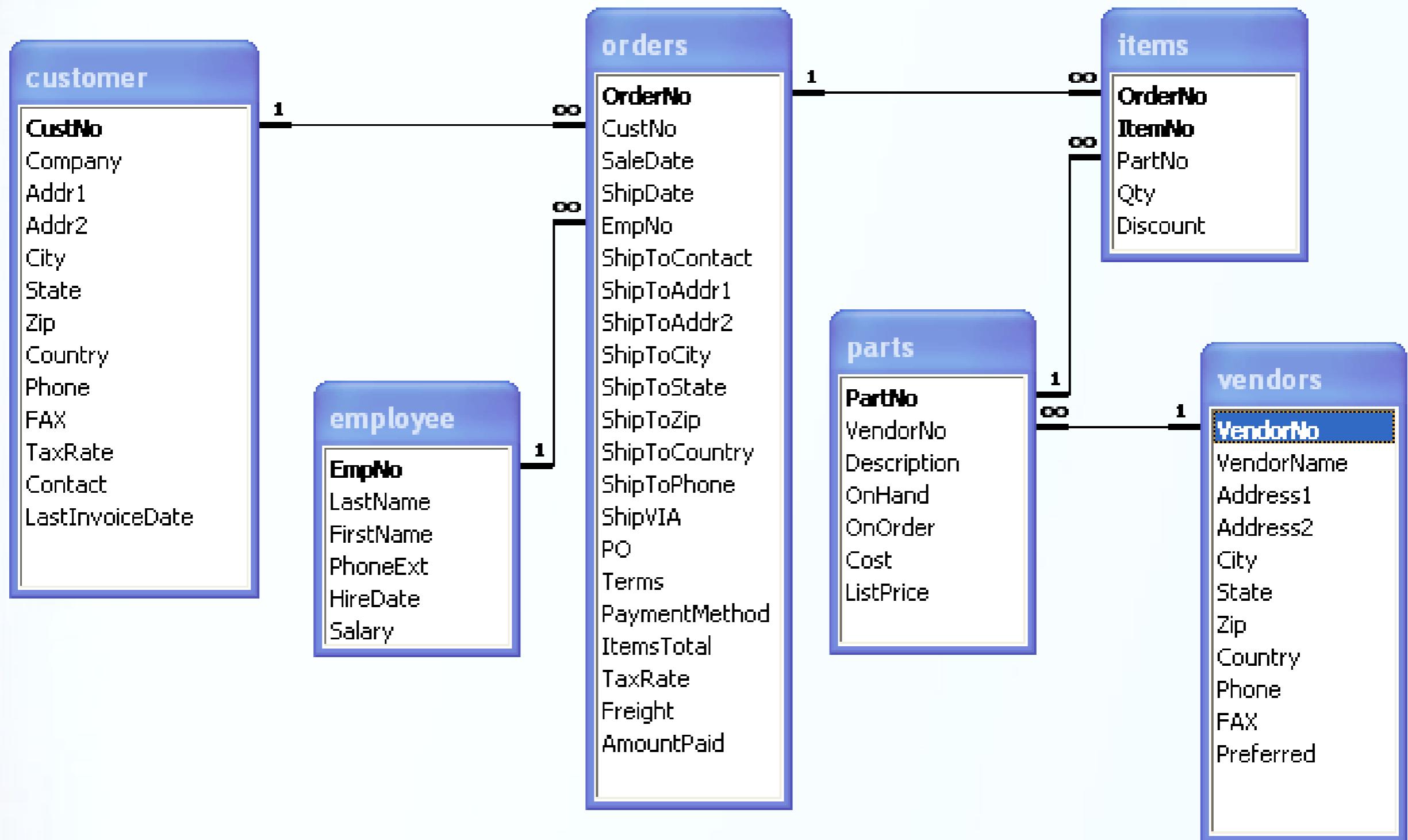
Perintah DDL digunakan untuk membuat definisi data seperti membuat struktur tabel (create table) atau menghapus (drop tabel), membuat index (create index), mengganti struktur tabel (alter table)

### ❑ **DML (Data Manipulation Language)**

DML digunakan untuk melakukan akses terhadap data, seperti pengambilan data (select), penambahan (insert), penghapusan (delete ) dan pembaruan (update)

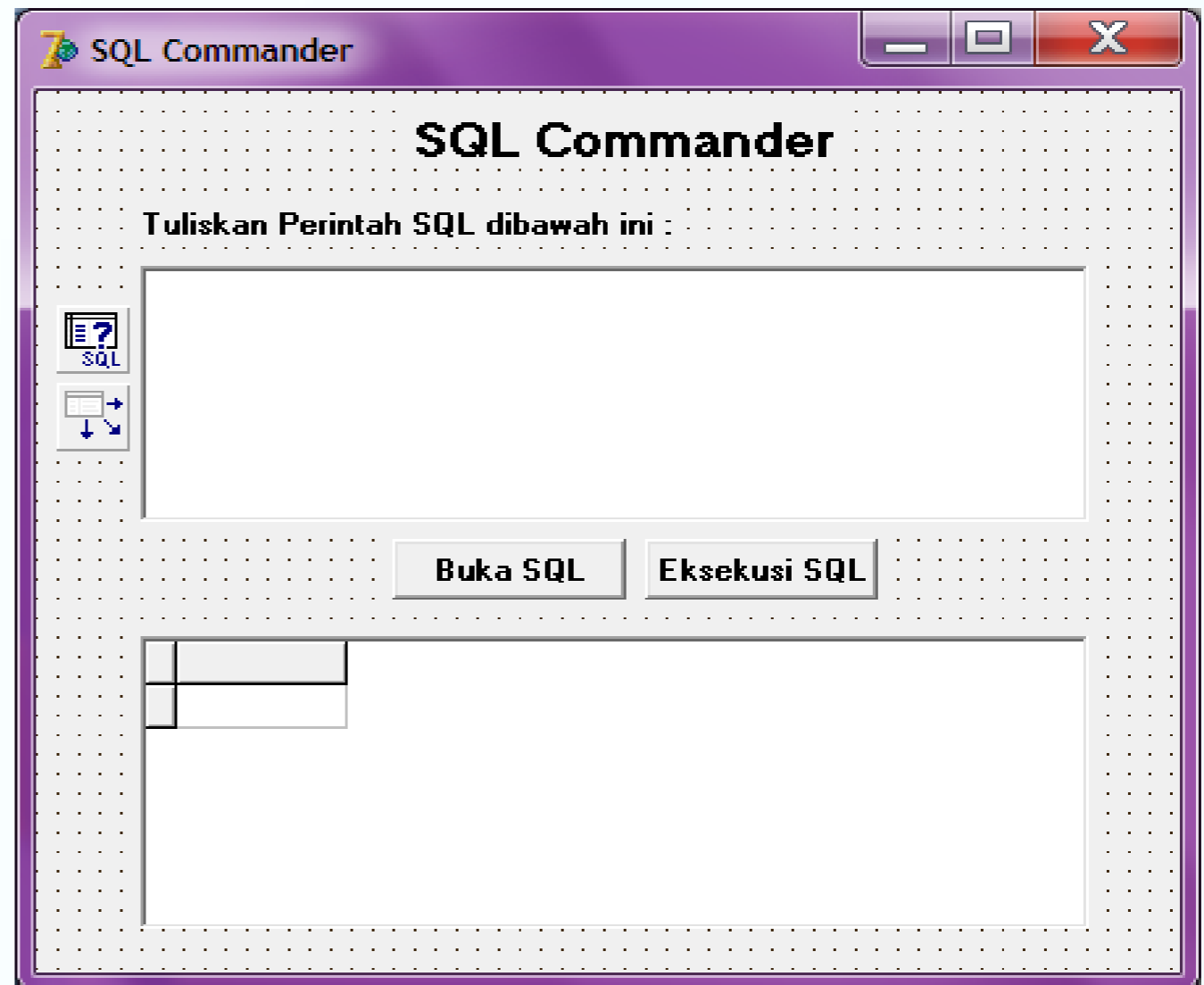


# Skema Relasi Pada Contoh Database



# Membuat Program SQL

- ❑ Buat Aplikasi baru
- ❑ Ganti properti Form dengan Name : FSQL dan Caption : SQL
- ❑ Simpan form dengan nama file U\_FSQL.pas
- ❑ Simpan project dengan nama SQLCommander.dpr
  - Tempatkan sebuah komponen **Query** yang ada di tab BDE ke Form
    - DatabaseName: [Kosongkan] atau isi alias
    - Name : Query1
    - Active : False
  - Tempatkan sebuah komponen **DataSource**
    - DataSet : Query1
    - Name : DataSource1
  - Tempatkan sebuah komponen **DBGrid**
    - DataSource : DataSource1
  - Tempatkan sebuah komponen **Memo**
    - Name : MemoSQL
    - Items : (Kosongkan)
  - Tempatkan sebuah komponen **BitBtn**
    - Name : TBukaSQL
    - Caption : Buka SQL
  - Tempatkan sebuah komponen **BitBtn**
    - Name : TEksekusiSQL
    - Caption : Eksekusi SQL



## Event onClick di TBukaSQL

```
procedure TFSQL.TBukaSQLClick(Sender: TObject);
begin
  if Query1.Active then // Jika Query1 sedang aktif maka
    Query1.Close;      // Tutup Query1. Identik dengan Active:=False;
    Query1.SQL:=MemoSQL.Lines; // Isi SQL di Query1 dari MemoSQL
  try
    Query1.Open;       // Buka Query1. Identik dengan Active:=True;
    // Perintah Open hanya dilakukan jika isi SQL adalah perintah Select
    Showmessage('Ditemukan '+IntToStr(Query1.RecordCount)+' Record');
  except
    on E: Exception do
      MessageDlg('SQL salah, dengan pesan'+#13+E.Message,mtError,[mbOK],0);
  end;
end;
```



## Event onClick di TEksekusiSQL

```
procedure TFSQL.TEksekusiSQLClick(Sender: TObject);
begin
  if Query1.Active then // Jika Query1 sedang aktif maka
    Query1.Close;      // Tutup Query1. Identik dengan Active:=False;
  Query1.SQL:=MemoSQL.Lines; // Isi SQL di Query1 dengan isi dari MemoSQL
  try
    Query1.ExecSQL;      // Buka Query1. Identik dengan Active:=True;
    // Perintah ExecSQL dilakukan jika melakukan operasi Insert, Update, Delete
    if Query1.RowsAffected>0 then
      Showmessage('Data berubah ada'+IntToStr(Query1.RowsAffected)+' Record')
    else
      Showmessage('Tidak ada data yang berubah oleh SQL tadi');
  except
    on E: Exception do
      MessageDlg('SQL salah, dengan pesan'+#13+E.Message,mtError,[mbOK],0);
  end;
end;
```

- Simpan Semua File (File → Save All)
- Compile atau Run program.
- Jika telah berhasil di Run, tutup delphi dan jalankan aplikasi dengan mengklik file SQLCommander.exe di folder anda (Jalankan di luar Delphi).

## Perintah Select

- ❑ Select digunakan untuk mengambil seluruh Field, sebagian field dari satu atau banyak tabel yang berelasi
- ❑ Struktur dasar perintah select adalah :

```
SELECT [ DISTINCT ] * | daftar_field FROM nama_tabel  
[WHERE daftar_kondisi_where]  
[ORDER BY daftar_field_pengurutan]  
[GROUP BY daftar_field_group]  
[HAVING kondisi_having]
```

Catatan :

Perintah yang diapit [ ] boleh tidak dicantumkan (tidak wajib ada)  
Perintah \* | daftar field berarti boleh semua (\*) atau (|) field tertentu saja



## Perintah Select 1 tabel

- **Select \* from employee**
  - Tampilkan semua (\*) field dari tabel employee
- **Select FirstName,Lastname,Salary from Employee order by salary**
  - Tampilkan hanya field FirstName, LastName dan Salary dari tabel Employee, dengan data terurut berdasarkan Salary secara ascending (kalau tidak disebut = ascending)
- **Select FirstName,Lastname,Salary from Employee order by salary desc**
  - Tampilkan hanya field FirstName, LastName dan Salary dari tabel Employee, dengan data terurut berdasarkan Salary secara ascending (kalau tidak disebut = ascending)
- **Select (FirstName||' '|| Lastname) as Nama\_Lengkap,Salary from Employee order by Nama\_Lengkap**
  - Tampilkan Nama\_Lengkap (gabungan FirstName+' '+LastName)) kemudian diurutkan berdasarkan nama lengkapnya. Perintah as digunakan untuk memberi nama lain ke suatu field.





## Perintah Select 1 tabel

- **select min(salary) as Gaji\_Terkecil, max(salary) as Gaji\_Terbesar, avg(salary) as Rata\_Rata\_Gaji, sum(salary) as Total\_Gaji, Count(\*) as Banyak\_Record from employee**
  - Tampilkan Gaji terkecil [min(salary)], gaji terbesar [max(salary)], Rata-rata Gaji [avg(salary)], dan jumlah semua salary [sum(salary)], serta banyak record [count(\*)] dari tabel Employee.
- **select \* from customer where CustNo=1221**
  - Tampilkan semua field dari tabel Customer yang field CustNo-nya adalah 1221
- **select \* from parts where Description like 'Regulator%'**
  - Tampilkan semua field dari tabel parts yang dalam field Description diawali dengan kata Regulator
- **select \* from parts where Description like '%Regulator'**
  - Tampilkan semua field dari tabel parts yang dalam field Description diakhiri dengan kata Regulator



## Perintah Select 1 tabel

**select upper(FirstName), lower(LastName), SubString(FirstName from 1 for 3)as Tiga\_Huruf\_Pertama from Employee**

Tampilkan FirstName dalam bentuk kapital, Lastname dalam bentuk huruf kecil dan 3 Huruf pertama dari nama depan.

**select \* from vendors where Fax is null**

Tampilkan semua field dari tabel vendors yang field Fax belum diisi (yang kosong/is null).

**select \* from vendors where Fax is not null**

Tampilkan semua field dari tabel vendors yang field Fax telah diisi (is not null)

**select \* from parts where VendorNo in (3511,2014,7382)**

Tampilkan semua field dari tabel parts yang vendorNo ada dalam himpunan (3511,2014,7382)

**select \* from parts where VendorNo not in (3511,2014,7382)**

Tampilkan semua field dari tabel parts yang vendorNo tidak ada dalam himpunan (3511,2014,7382)

**select \* from employee where salary=(select max(salary)from employee)**

Tampilkan semua field dari tabel employee yang salarynya sama dengan salary paling besar [max(salary)] dari tabel employee



## Perintah Select 1 tabel

- **select OrderNo from Orders where Extract(Year from Saledate)=1988 and Extract(MONTH from Saledate)=5**
  - Tampilkan hanya field OrderNo dari tabel Orders yang tahun dari field SaleDate (Year from Saledate) adalah 1988 dan bulan dari field Saledate (month from Saledate) adalah 5 (mei).
- **select OrderNo,saledate from Orders where Extract(Year from Saledate)=1988 and (Extract(month from saledate)=4 or extract(month from saledate)=8)**
  - Tampilkan field OrderNo dan Saledate dari tabel Orders yang tahun dari field SaleDate (Year from Saledate) adalah 1988 dan bulan dari field Saledate (month from Saledate) adalah 4 (april) atau 8 (agustus).
- **select \* from employee where salary between 10000 and 40000**
  - Tampilkan semua field dari tabel employee yang salarynya antara 10000 - 40000
- **select \* from employee where salary not between 10000 and 40000**
  - Tampilkan semua field dari tabel employee yang salarynya diluar (not between) antara 10000 - 40000



## Perintah Select 2 tabel

- `select Orders.*, Customer.* from Orders, Customer where Orders.CustNo=Customer.CustNo`
- `select Orders.OrderNo, Orders.CustNo, Customer.Company from Orders, Customer where Orders.CustNo=Customer.CustNo`
- `select a.OrderNo, a.CustNo, b.Company from Orders a, Customer b where a.CustNo=b.CustNo`
- `select Orders.OrderNo, Orders.CustNo, Customer.Company from Orders join Customer on (Orders.CustNo=Customer.CustNo)`
- `select a.OrderNo, a.CustNo, b.Company from Orders a join Customer b on (a.CustNo=b.CustNo)`





## Perintah Select ke banyak tabel

Untuk membuat select ke banyak tabel sebaiknya harus melihat dulu skema relasi.

- `select a.OrderNo, a.CustNo, b.Company, a.EmpNo, c.FirstName from Orders a, Customer b, Employee c where a.CustNo=b.CustNo and a.EmpNo=c.EmpNo`
- `select a.OrderNo, a.CustNo, b.Company, a.EmpNo, c.FirstName from Orders a join Customer b on a.CustNo=b.CustNo join Employee c on a.EmpNo=c.EmpNo`
  - Tampilkan OrderNo, CustNo, dan EmpNo dari tabel Orders (a), Company dari tabel Customer (b) dan FirstName dari tabel Employee (c) dimana ketiga tabel tersebut mempunyai relasi :
    - Tabel Order berelasi dengan tabel Employee berdasarkan field EmpNo
    - Tabel Order berelasi dengan tabel Customer berdasarkan field CustNo





## Perintah Insert (Menambah Data)

- INSERT digunakan untuk menambah data baru ke suatu tabel.
- Struktur dasar perintah INSERT adalah :

```
INSERT INTO nama_tabel [(daftar_field)]  
VALUES (daftar_isi_field)
```

**Catatan :** Perintah yang diapit [ ] boleh tidak dicantumkan (tidak wajib ada) Daftar field hanya wajib disertakan ketika ingin menyisipkan data tidak sesuai dengan urutan struktur field, atau hanya ingin mengisi beberapa field saja (tidak semuanya)

- **insert into Parts values(9999,3511,'Aqua Gelas',10,35,1235,2500)**
  - Sisipkan record baru ke tabel Parts dimana PartNo=9999, VendorNo=3511, Description='Aqua Gelas', OnHand=10, OnOrder=35, Cost=1235 dan ListPrice=2500. Perhatikan bahwa urutan pengisian nilai sesuai dengan urutan struktur field
- **insert into Parts(VendorNo, PartNo, Description) values(2674,9998,'Nokia 6120 Classic')**
  - Sisipkan record baru hanya pada 3 field saja yaitu VendorNo=2674, PartNo=9998, Description='Nokia 6120 Classic'. Urutan di bagian value harus sesuai dengan urutan daftar field.

## Perintah Update (Mengedit Data)

- UPDATE digunakan untuk mengupdate data/record yang ada di suatu tabel.

- Struktur dasar perintah UPDATE adalah :

UPDATE nama\_tabel SET nama\_field = nilai\_baru

[nama\_field = nilai\_baru...]

[WHERE kriteria\_record\_yang\_akan\_diupdate]

**Catatan :** Perintah yang diapit [ ] boleh tidak dicantumkan (tidak wajib ada).

Diperbolehkan untuk mengupdate data lebih dari 1 field dalam sebuah query.

- **update customer set state=Upper(state), Country=lower(country)**
  - Update tabel customer, isi field state dengan isi field state yang sudah dikapitalkan, isi field country dengan field country yang sudah dihurufkecilkan. Dikarenakan tidak ada perintah where maka perubahan dilakukan ke semua record.
- **update parts set Onhand=100, onorder=0, cost=250000, listprice=400000 where PartNo=9998**
  - Update tabel parts, update field Onhand, OnOrder, Cost, dan ListPrice pada record yang PartNo-nya adalah 9998
- **update parts set OnOrder=OnOrder+5 where VendorNo=3511**
  - Update tabel Parts isi tambah field OnOrder dengan 5 Part yang dibeli dari VendorNo 3511

## Perintah Delete (Menghapus Data)

- DELETE digunakan untuk menghapus data/record yang ada di suatu tabel.
- Struktur dasar perintah DELETE adalah :

**DELETE FROM nama table**

**[ WHERE kriteria\_record\_yang\_akan\_dihapus ]**

**Catatan :** Perintah yang diapit [ ] boleh tidak dicantumkan (tidak wajib ada). Perintah where hanya dilakukan jika anda ingin menghapus record tertentu saja. **Jika perintah where tidak dibuat, maka semua record akan dihapus.**

- **[TIDAK USAH DICoba] Delete From Parts**
  - Hapus semua record dari tabel Parts. Karena tanpa where, maka penghapusan akan berlaku untuk semua record
- **Delete From Parts where PartNo=9999**
  - Hapus record yang ada di tabel Parts yang mempunyai isi field PartNo=9999 saja.

# Bekerja dengan Query di Delphi

• Buat aplikasi baru

## Atur Form1

- Caption : Pengolahan Data dengan Query 1
- Name : FQuery1
- Save dengan nama UFQuery1.pas

## Save Project

- Save Project nama : ProjQuery1.dpr

The screenshot shows a Delphi application window titled "Pengolahan Data dengan Query 1". The window contains a table of employee data and a form for data entry.

Empl	LastName	FirstName	PhoneExt	HireDate
2	Nelson	Roberto	250	28/12/1988
4	Young	Bruce	233	28/12/1988
5	Lambert	Kim	22	06/02/1989
8	Johnson	Leslie	410	05/04/1989
9	Forest	Phil	229	17/04/1989
11	Weston	K. J.	34	17/01/1990
12	Lee	Terri	256	01/05/1990

Below the table is a button labeled "TAMBAH".

Below the button is a section titled "Pengisian Data" with the following fields:

- No Pegawai: ENoPeg
- Nama Depan: ENamaDpn
- Nama Belakang: ENamaBlk
- Ekstensi Telepon: EExtTelp
- Tanggal Mulai Kerja: 04/06/2010
- Gaji: EGaji

# Bekerja dengan Query di Delphi

- **Tambahkan sebuah TQuery di form, atur propertinya :**
  - Nama : QBrowse
  - SQL : select \* from employee
  - Active : True
- **Tambahkan sebuah Datasource di form, atur propertinya :**
  - Name : DsQBrowse
  - Dataset : QBrowse
- **Tempatkan sebuah DbGrid, atur propertiesnya :**
  - DataSource : DsQBrowse
- **Tempatkan sebuah TQuery di form, atur propertinya :**
  - Name : QExec
- **Tempatkan beberapa TEdit di form. Atur properti seperti pada form.**
- **Tempatkan sebuah DateTimePicker (ada di tab Win32), atur propertinya :**
  - Name : DTPTanggal
- **Tempatkan sebuah Tombol di form. Atur propertinya :**
  - Name : TTambah
  - Caption : Tambah



# Bekerja dengan Query di Delphi

```
procedure TFQuery1.TTambahClick(Sender: TObject);
begin
  try
    QExec.SQL.Clear; // Kosongkan SQL yang ada
    QExec.SQL.Add('Insert into Employee values ('); // Perintah Insert
    QExec.SQL.Add(''+ENoPeg.Text+'','); // Isi Field EmpNo
    QExec.SQL.Add(''+ENamaBlk.Text+'','); // Isi Field LastName
    QExec.SQL.Add(QuotedStr(ENamaDpn.Text)+' ,'); // Isi Field FirstName
    QExec.SQL.Add(QuotedStr(EExtTelp.Text)+' ,'); // Isi PhoneExt
    QExec.SQL.Add(QuotedStr(DateToStr(DTPTanggal.Date))+' ,'); // Isi Hiredate
    QExec.SQL.Add(EGaji.Text+' '); // Isi field Salary
    QExec.ExecSQL; // Eksekusi Query
    Showmessage('Data Berhasil disimpan');
    QBrowse.DisableControls; // Nonaktifkan perubahan di Control
    QBrowse.Close; // Tutup Browse (QBrowse.Active:=False)
    QBrowse.Open; // Buka Browse (QBrowse.Active:=True)
    QBrowse.Locate('EmpNo',ENoPeg.Text,[]); // Cari EmpNo sesuai NoPeg
    QBrowse.EnableControls; // Aktifkan perubahan di Control
  except
    On E: Exception do
      ShowMessage('Ada Error : '+#13+#13+E.Message); // Tampilkan Error
  end;
end;
```

## Master - Detail dengan SQL

- Buat Aplikasi Baru (File → New → Application)
- Simpan Form dan Projectnya (di folder yang ada databasenya)
- Tempatkan sebuah TQuery di form, atur properti :
  - Name : QOrder (untuk menampilkan data order)
  - SQL : Select \* from Orders
  - Active : True
- Tempatkan sebuah TDataSource di form, atur propertinya :
  - Name : DsOrder
  - Dataset : QOrder
- Tempatkan sebuah TDbGrid di form, atur properti Datasource dengan DsOrder

## Master - Detail dengan SQL

- Tempatkan sebuah TQuery di form, atur properti :
  - Name : QDetailOrder (untuk menampilkan data detail order)
  - SQL : `select a.ItemNo, a.PartNo, b.Description, a.Qty, b.ListPrice, (a.Qty*b.ListPrice) as SubTotal from items a join parts b on a.PartNo=b.PartNo where a.OrderNo=:OrderNo`  
[:**OrderNo** adalah field yang berelasi ke tabel master (table order)]
  - DataSource : DsOrder (datasource master)
  - Active : True
- Tempatkan sebuah TDataSource di form, atur propertinya :
  - Name : DsDetailOrder
  - Dataset : QDetailOrder
- Tempatkan sebuah TDbGrid di form, atur properti Datasource dengan DsDetailOrder

## Master - Detail dengan SQL

- Tempatkan sebuah TQuery di form, atur properti :
  - Name : QRekapDetail (untuk menampilkan perhitungan rekap detail)
  - SQL : `select sum(a.Qty) as Banyak_Item, sum(a.Qty*b.ListPrice) as Total from items a join parts b on a.PartNo=b.PartNo where a.OrderNo=:OrderNo`  
[:OrderNo adalah field yang berelasi ke tabel master (table order)]
  - DataSource : DsOrder (datasource master)
  - Active : True
- Tempatkan sebuah TDataSource di form, atur propertinya :
  - Name : DsRekapDetail
  - Dataset : QRekapDetail
- Tempatkan sebuah TDbEdit di form, atur properti
  - Datasource : DsRekapDetail
  - DataField : Total
- Tempatkan sebuah TDbEdit di form, atur properti
  - Datasource : DsRekapDetail
  - FieldName : Banyak\_Item

# Master - Detail dengan SQL

Form1

SQL

OrderNo	CustNo	SaleDate	ShipDate	EmpNo	ShipToContact
1003	1351	4/12/1988	5/3/1988	114	
1004	2156	4/17/1988	4/18/1988	145	Maria Eventosh
1005	1356	4/20/1988	1/21/1988	110	

SQL

ItemNo	PartNo	Description	Qty	ListPrice	SubTotal
1	1313	Regulator System	5	250	1250

SQL

Banyak\_Item: 5

Total: 1250