
SQL (Structured Query Language) Dengan Delphi

Oleh : Andri Heryandi, ST

What is 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.
 - Untuk SQL yang lebih lanjut, dianjurkan untuk mempelajari SQL milik SQL Server, MySQL, Oracle, Interbase, PostgreSQL dan server-server database lain.
-

Jenis-Jenis perintah SQL

- DDL (Data Definition Language)
Perintah DDL digunakan untuk membuat definisi data seperti membuat struktur tabel (create table), membuat index (create index), mengganti struktur tabel (alter table) atau menghapus index (drop table).
- DML (Data Manipulation Language)
DML digunakan untuk melakukan akses terhadap data seperti pengambilan data (select), penambahan (insert), penghapusan (delete), pembaruan (update).

[Referensi SQL yang didukung oleh BDE](#)

DDL

- Di perkuliahan Delphi, tidak akan diterangkan mengenai DDL. Diasumsikan bahwa struktur tabel yang akan diakses oleh SQL telah tersedia.
 - DDL akan anda dapatkan di perkuliahan Basis Data dan Praktikum Basis Data
-

DML

- Perintah Select
Digunakan untuk mengambil data dari database.
 - Perintah Insert
Digunakan untuk menyisipkan record baru ke suatu tabel yang ada dalam database
 - Perintah Update
Mengupdate/Edit data yang ada di suatu tabel
 - Perintah Delete
Menghapus data yang ada di suatu tabel
-

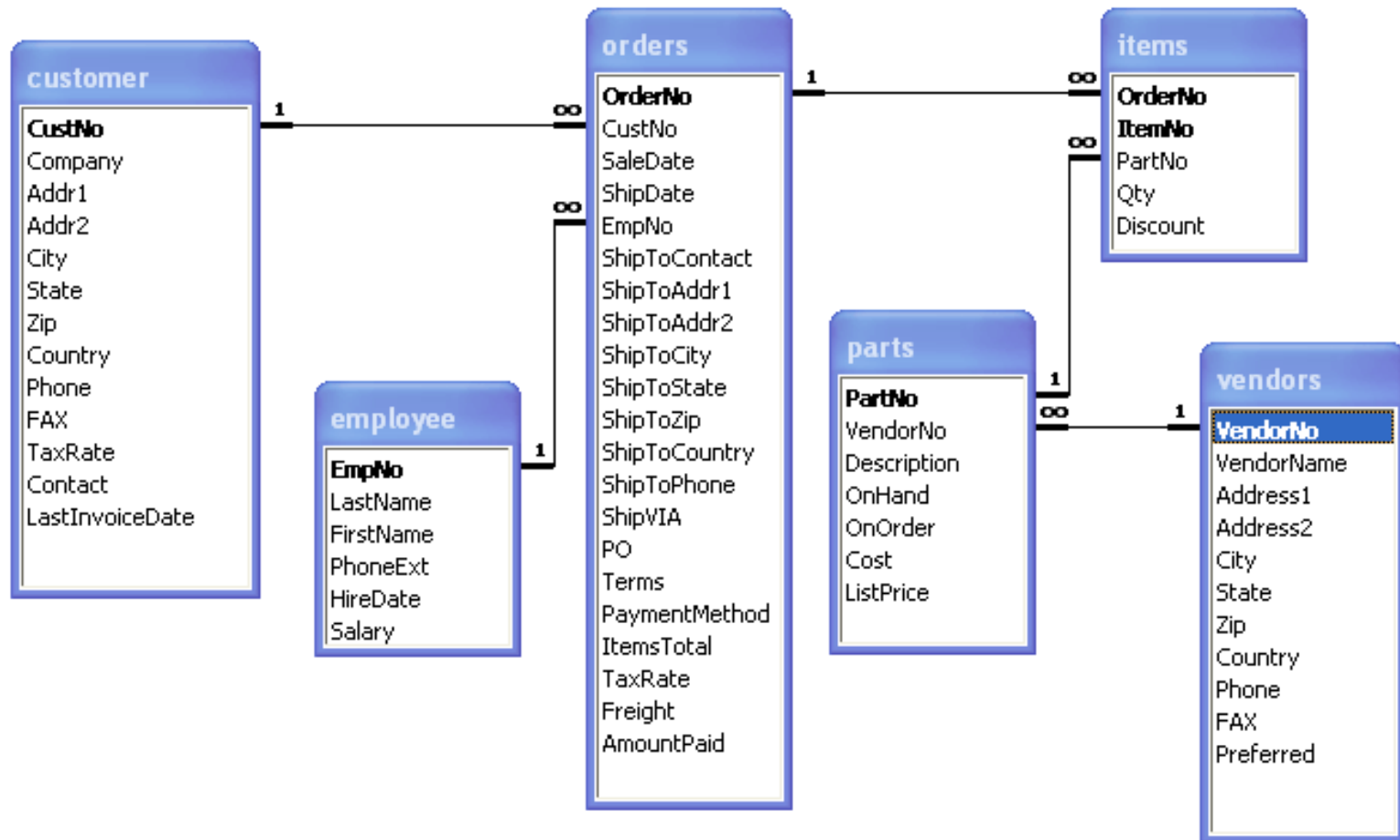
Contoh Database

- Gunakan tabel-tabel berikut untuk belajar memanipulasi data dengan menggunakan SQL.

File Contoh Database ada dalam file

[Contoh Database.zip](#)

Skema Relasi Pada Contoh Database



Membuat Program SQL Commander

- Buat aplikasi baru.
 - Ganti properti Form1
 - Name : FSQL
 - Caption: SQL Commander
 - Simpan form dengan nama file uFSQL.pas
 - Simpan project dengan nama SQLCommander.dpr
-

Membuat Program SQL Commander

- Tempatkan sebuah komponen **Query** yang ada di tab BDE ke Form
 - DatabaseName : [Kosongkan] atau isi alias
 - Name : Query1
 - Active : False
- Tempatkan sebuah komponen **DataSource** ke Form
 - 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

Layout SQL Commander



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;
```

Save, Compile and Run the SQLCommander

- 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 yang ada dalam daftar_field

Contoh 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.

Contoh Perintah Select 1 tabel

- `select min(salary) as Gaji_Terkecil, max(salary) as Gaji_Terb Besar, 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

Contoh 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)
-

Contoh 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

Contoh Perintah Select 1 tabel

- `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
-

Contoh Perintah Select 1 tabel dengan Fungsi Agregasi

- `Select VendorNo,Count(*) from Parts group by VendorNo`
 - Perintah di atas berguna untuk menampilkan banyak record barang (`Count(*)`) yang ada di tabel Parts dimana pengelompokan datanya (`group by`) didasarkan pada field `VendorNo`
- `select PaymentMethod, Sum(AmountPaid) as Total, Count(*) as Banyak_Record from Orders group by PaymentMethod order by Total`
 - Tampilkan jumlah field `AmountPaid` dan `Banyak_Recordnya` yang berasal dari tabel Order dimana data dikelompokkan/digroupkan berdasarkan field `PaymentMethod` dan diurutkan berdasarkan `Total (Sum(AmountPaid))`.
- `select State,City,count(*) as Banyak_Vendor from vendors group by State,City`
 - Tampilkan `State, City` dan `Banyak_Vendor` dari tabel Vendors dimana data dikelompokkan berdasarkan `State`, disubkelompokkan lagi berdasarkan `City`

Perintah Insert (Menambah Data)

Perintah Insert (Menambahkan 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)

Perintah Insert (Menambahkan Data)

- `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. Perintah where hanya dilakukan jika anda ingin mengupdate record tertentu saja. Jika perintah where tidak dibuat, maka semua record akan berubah sesuai dengan perubahannya

Perintah Update (Mengedit Data)

- `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.**

Perintah Delete (Menghapus Data)

- **[TIDAK USAH DICOBAB]** 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.
 - Delete From Orders where Extract(Year from SaleDate)=2007
 - Hapus record dari tabel Order yang Tahun dari Saledatonya adalah 2007.
-

Perintah Select ke 2 Tabel dengan menggunakan perintah WHERE

- `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`
-

Perintah Select ke 2 Tabel dengan menggunakan perintah JOIN

- `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 dengan menggunakan perintah WHERE

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`
-

Perintah Select ke Banyak Tabel dengan menggunakan perintah JOIN

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 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
-

Latihan SQL

- Carilah data Order yang customernya berada di United State (us). Tampilkan hanya OrderNo, CustNo dan Kota dari kustomer tersebut.

SQL

Select Banyak Tabel + Fungsi Agregasi

Carilah banyak data barang yang dijual kepada Customer. Tampilkan berdasarkan kota dan banyaknya barang

- `select customer.city, sum(items.Qty) from parts, items, orders, customer where parts.partno=items.partno and items.orderno=orders.orderno and orders.custno=customer.custno group by customer.city`
-

Latihan SQL

- Carilah banyak barang, dan total bayar order. Tampilkan hanya OrderNo, Banyak_barang dan Total_Bayarannya saja (gunakan fungsi sum)

SQL

Latihan SQL

- Carilah total penjualan yang dilakukan oleh pegawai (employee) yang bernama John Montgomery

SQL

BEKERJA DENGAN QUERY MENGUNAKAN DELPHI

Bekerja dengan Query di Delphi

CARA 1

MENYUSUN SQL DAN MENGEKSEKUSINYA

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 dengan nama : ProjQuery1.dpr
-

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 propertienya :
 - 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 di slide berikutnya
- 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

Pengolahan Data dengan Query 1

DATABASE DENGAN MENGGUNAKAN QUERY (1)

EmpNo	LastName	FirstName	PhoneExt	Hir
1	Herdis	Eris	123	6/2
2	Nelson	Roberto	250	12
3	Heryandi	Andri		6/1
4	Young	Bruce	233	12
5	Lambert	Kim	22	2/6
8	Johnson	Leslie	410	4/6
9	Forest	Phil	229	4/1
11	Weston	K. J.	34	1/1

Pengisian Data

No Pegawai: ENoPeg

Nama Depan: ENamaDpn

Nama Belakang: ENamaBlk

Ekstensi Telepon: EExtTelp

Tanggal Mulai Kerja: 6/ 1/2008

Gaji: EGaji

Tambah (1)

SQL

SQL

Bekerja dengan Query di Delphi

Isi Event Onclick dari TTambah dengan perintah di bawah ini :

```
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;
```

Bekerja dengan Query di Delphi

■ CARA 2

MENGGUNAKAN PARAMETER DATA

Bekerja dengan Query di Delphi

Untuk tiap proses sebaiknya menggunakan komponen TQuery masing-masing. Jadi untuk Tambah data memerlukan sebuah TQuery, untuk Edit data memerlukan sebuah query dan untuk Hapus data juga memerlukan query. Perintah SQL di setiap TQuery berbeda-beda, disesuaikan dengan kegunaannya.

-
- Tempatkan sebuah TUpdateSQL di form, atur propertinya :
 - Name : QTambah
 - SQL : insert into employee values(:EmpNo, :LastName, :FirstName, :PhoneExt, :HireDate, :Gaji)
 - Atur Parameternya.
 - Klik QTambah, Klik [...] di properti Params
 - Ganti DataType
 - Klik parameter EmpNo, properti DataType diisi ftInteger
 - Klik parameter LastName, properti DataType diisi ftString
 - Klik parameter FirstName, properti DataType diisi ftString
 - Klik parameter PhoneExt, properti DataType diisi ftString
 - Klik parameter HireDate, properti DataType diisi ftDate
 - Klik parameter Gaji, properti DataType diisi ftCurrency
 - Tempatkan sebuah Tombol di form, atur propertinya :
 - Name : TTambah2
-

Pengolahan Data dengan Query 1

DATABASE DENGAN MENGGUNAKAN QUERY (1)

EmpNo	LastName	FirstName	PhoneExt	Hir
1	Herdis	Eris	123	6/2
2	Nelson	Roberto	250	12
3	Heryandi	Andri		6/1
4	Young	Bruce	233	12
5	Lambert	Kim	22	2/6
8	Johnson	Leslie	410	4/5
9	Forest	Phil	229	4/1
11	Weston	K. J.	34	1/1

Pengisian Data

No Pegawai:

Nama Depan:

Nama Belakang:

Ekstensi Telepon:

Tanggal Mulai Kerja:

Gaji:

Tambah (1)

Tambah (2)

Isi event OnClick dari Tombol Tambah 2 dengan perintah di bawah ini

```
procedure TFQuery1.TTambah2Click(Sender: TObject);
begin
    try
        // Pengisian Parameter
        QTambah.ParamByName('EmpNo').AsInteger:=StrToInt(ENoPeg.Text);
        QTambah.ParamByName('FirstName').AsString:=ENamaDpn.Text;
        QTambah.ParamByName('LastName').AsString:=ENamaBlk.Text;
        QTambah.ParamByName('PhoneExt').AsString:=EExtTelp.Text;
        QTambah.ParamByName('HireDate').AsDate:=DTPTanggal.Date;
        QTambah.ParamByName('Gaji').AsCurrency:=StrToCurr(EGaji.Text);
        QTambah.ExecSQL; // Jalankan 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;
```

Untuk proses Edit, dan Hapus silahkan
berimprovisasi...

Bekerja dengan Query di Delphi

- **MASTER-DETAIL**
- **DENGAN SQL**



Aplikasi 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
-

Aplikasi 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

Aplikasi 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

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

Banyak_Item: 5

Total: 1250

Spread your wings

- Mulai gunakan database server yang untuk Client/Server (Oracle, Interbase, PostgreSQL, SQL Server, MySQL, FireBird)
 - Pelajari terus SQL, karena SQL adalah bahasa standar untuk berkomunikasi dengan database server
 - Be a MultiSkilled person (memahami banyak hal secara mendalam). Jangan hanya CrossSkilled (mengetahui banyak hal tapi hanya dasarnya saja)
-

Sehat Berapa

Kritik, Saran, Dukungan, Revisi dll
Kirimkan ke :

kuliahdelphi@yahoogroups.com

Atau

kuliahdelphi@yahoo.co.id

SEMOGA BERMANFAAT
