

IX. STRUCTURED QUERY LANGUAGE (SQL)

SQL adalah bahasa yang digunakan untuk membuat dan memanipulasi beberapa tabel data yang saling berhubungan. Keunggulan SQL adalah memungkinkan Anda untuk mencari dan mengurutkan data tabel yang tidak memiliki index. Perintah dalam SQL sangat banyak, dan pada modul ini hanya akan dibahas perintah yang sedang digunakan. Untuk menggunakan SQL dalam pemrograman Delphi, digunakan komponen QUERY yang ada dalam kategori dataaccess.

Perintah dalam SQL adalah:

1. SELECT

Perintah ini digunakan untuk menampilkan data atau record berdasarkan kriteria tertentu.

Format penulisan perintah Select adalah:

```
SELECT [DISTINCT] kolom FROM table
    [WHERE kondisi_pencarian]
    [ORDER BY Pengurutan]
    [GROUP BY Group_list]
    [HAVING Having_condition]
```

Contoh:

```
Select * from country
```

```
Select Name,Capital from country where population>8000000
```

A. WHERE

Klausula WHERE digunakan untuk memilih record yang akan ditampilkan

Contoh:

```
Select * From Country where name like 'A%' → untuk menampilkan
record yang namanya diawali dengan huruf 'A'
```

```
Select * From Country where name like '%a' → untuk menampilkan
record yang namanya diakhiri dengan huruf 'a'
```

B. ORDER BY

Klausula ORDER BY digunakan untuk mengurutkan table berdasarkan salah satu field, baik terurut menaik (*ascending*) maupun terurut menurun (*descending*).

Contoh:

```
Select * from country order by capital  
Select * from country order by capital Desc
```

C. GROUP BY dan HAVING

Klausula GROUP BY digunakan untuk mengelompokkan record menurut field tertentu.

Contoh:

```
Select continent from country group by continent.
```

Klausula HAVING mempunyai kegunaan mirip dengan WHERE tetapi klausula ini digunakan untuk memberi syarat pada klausula GROUP BY.

2. INSERT

Perintah ini digunakan untuk menambah data atau record pada table.

Syntax:

```
INSERT INTO Table ( Field-Field)  
Values(Data)
```

Contoh:

```
INSERT INTO COUNTRY (Name, Capital, Continent, Area, Population)  
Values ("Indonesia", "Jakarta", "Asia", "5200000,215000000")
```

Untuk mengeksekusi perintah sql dalam delphi digunakan perintah **ExecSQL**.

3. DELETE

Menghapus data dengan DELETE tidak berbeda jauh dengan perintah INSERT.

Syntax:

```
DELETE FROM Table WHERE Field = Data
```

Contoh:

```
DELETE FROM Country WHERE Name = 'INDONESIA'
```

4. UPDATE

Perintah UPDATE digunakan untuk mengganti nilai field dari sebuah record.

Syntax:

UPDATE Table SET Field = data WHERE Field = Data

Contoh:

```
UPDATE Country SET Population = 23000000 WHERE NAME =  
"INDONESIA"
```

Query Berparameter

Untuk meningkatkan fleksibilitas, Delphi mempunyai query yang berparameter.

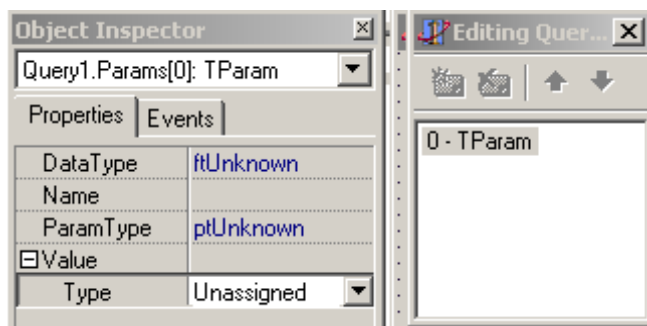
Contoh:

```
Select * From Country Where Name like :Nama
```

Parameter nama pada perintah ini dapat diganti dengan string lain. Ada dua cara untuk mengisi parameter Nama.

1. dengan menggunakan Object Inspector.

Pilih properties SQL pada komponen Query isi dengan perintah select seperti diatas. Kemudian isi properties params, isi properties NAME, VALUE, dan TYPE



2. dengan menggunakan method parambyname

contoh:

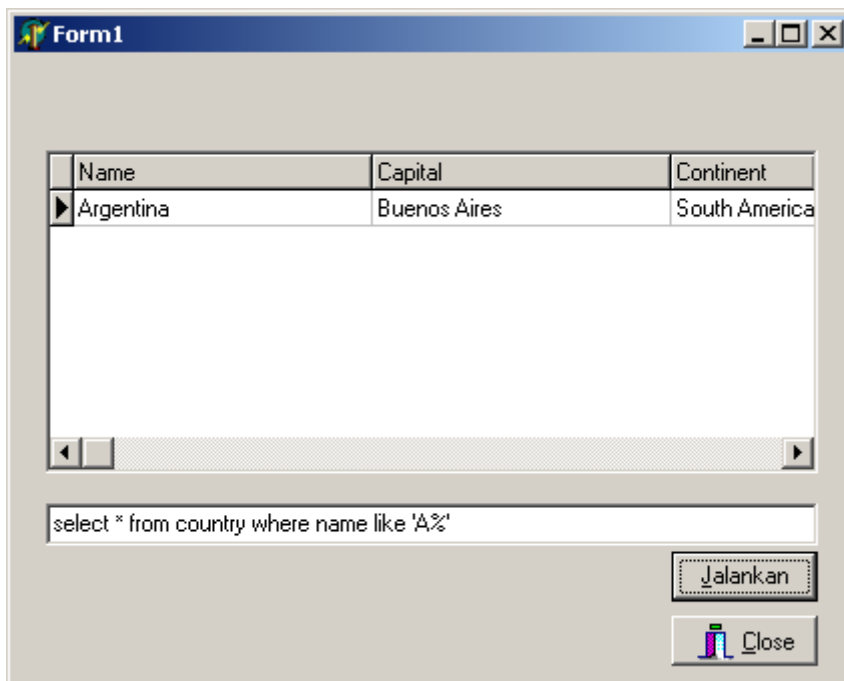
```
Query1.ParamByName ('NAME').AsString:='C%';
```

Method Komponen Query

1. Close : digunakan untuk menutup query
2. Open : digunakan untuk mengeksekusi query
3. Prepare : digunakan untuk meningkatkan performan Query
4. SQL
 - a. Add untuk menambah perintah SQL, seperti perintah select

- b. Clear untuk menghapus perintah yang telah ada

Contoh Program:



Kode program:

Kode untuk project

```
program Projectsql1;  
  
    uses  
        Forms,  
        Unitsql1 in 'Unitsql1.pas' {Form1};  
  
    {$R *.RES}  
  
    begin  
        Application.Initialize;  
        Application.CreateForm(TForm1, Form1);  
        Application.Run;  
    end.
```

Kode untuk Unit

```
unit Unitsql1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms,
  Dialogs,
  Buttons, StdCtrls, Grids, DBGrids, Db, DBTables;

type
  TForm1 = class(TForm)
    DataSource1: TDataSource;
    Query1: TQuery;
    DBGrid1: TDBGrid;
    Edit1: TEdit;
    Button1: TButton;
    BitBtn1: TBitBtn;
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.DFM}

procedure TForm1.Button1Click(Sender: TObject);
begin
```

```
query1.close;  
query1.SQL.Clear;  
query1.sql.Add(edit1.text);  
query1.open;
```

```
end;
```

```
end.
```