**Bab XI Pencarian Data**

Salah satu konsep pemograman berbasis data base proses pencarian data menjadi satu hal yang sangat penting didalam mendukung kesempurnaan hasil didalam sebuah aplikasi program. Fungsi secara umum adanya pencarian data adalah untuk mendapatkan secara cepat data yang diinginkan dari sebuah tabel didalm sebuah database.

# Konsep Pemograman Pencarian Data b Konsep Pencarian dengan Properti

Konsep ini merupakan metode pencarian data dimana memanfaatkan kemampuan property untuk mendapatkan data yang diinginkan

Ada 5 konsep pencarian data dengan property

# Locate

Locate adalah metode pencarian data untuk record yang sama dengan criteria yang sudah ada ataupun mendekati dengan kriteria yang ada. Pencarian locate dapat digunakan untuk tabel dengan index maupun yang tanpa index

Contoh

If not table1.locate (‘nama’, enama.text, []) then

Messagedlg(‘”’ + enama.text + ‘ “ Tdak ditemukan’, mterror, [mbok],0);

# Findkey

Metode finkey digunakan mencari record yang sama . Metode ini dapat digunakan untuk tabel dengan index.

# Contoh

Table1.indexname := ‘ ‘;

If not table1.findkey ([‘00099’]) then

Messagedlg(‘data Tdak ditemukan’, mterror, [mbok],0);

Proses pencarian data diatas digunakan untuk table dengan index sebagai primary key. Sedangkan untuk pencarian dengan indes secondary dapat terlihat seperti contoh berikut ini.

Table1.indexname := ‘nama’;

If not table1.findkey([enama.text]) then

Messagedlg(‘”’ + enama.text + ‘ “ Tdak ditemukan’, mterror, [mbok],0); Else

Tampil;

# Find Nearest

Metode ini digunakan untuk tabel dengan index selaian itu juga dapat digunakan untuk pencarian record yang paling mendekati.

Contoh

Table1.indexname ;= ‘nama’; Table1.findnearest([enama.text])

# Gotokey

Metode ini sama seperti dengan metode dengan findkey, tetapi dalam hal penulisan lebih rumit karena harus menjalankan terlebih dahulu event Setkey ataupun editkey. Konsep ini juga dapat digunakan untuk tabel dengan index maupun tidak.

# Contoh

Table1.setkey; Table1.fieldbyname(‘nama’).Asstring := enama.text; If not table1.gotokey then

Messagedlg(‘”’ + enama.text + ‘ “ Tdak ditemukan’, mterror, [mbok],0); Else

Tampil;

# Gotonearest

Metode ini hampis sama dengan metoded findnearest. Perbedaan utamanya hanya dalam hal penulisan.

Contoh

Table1.setkey

Table1.fieldbyname(‘nama’).Asstring := enama.text;

Table1.gotokey;

# b Konsep pencarian data dengan perintah SQL

Metode ini merupakan metode pencarian data yang banyak digunakan oleh para pengembang aplikasi karena konsep ini dapat digunakan untuk semua bahasa pemograman dengan berbagai macam database. Luwes merupakan salah satu keunggulan penggunaan konsep pencarian dengan perintah SQl. Selain itu konsep yang digunakan tidak perlu mengingat property yang digunakan.

# Contoh

Var strsql : string; Begin

Strsql := ‘Select \* from barang where nama like “’ + enama.text +’%” ‘; Query1.sql.clear;

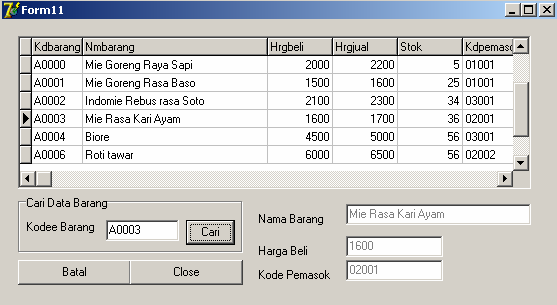
Query1.sql.add(strsql); Query.close; Query.open;

End;

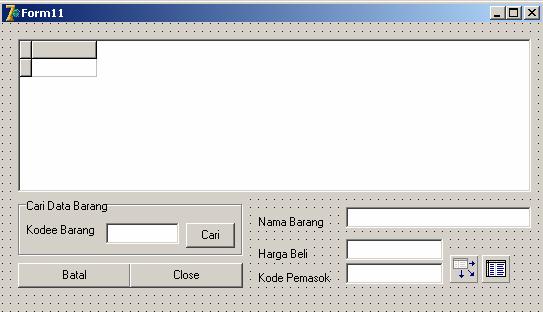
# 11.2. Aplikasi Program Pencarian Data

1. **Dengan Propeti**

# Hasil Form Setelah Dijalankan



* 1. **Desain Form**



# Desain Properties

Catatan = Koneksi table dan data base akan dilakukan secara programming, jadi untuk setting properties tidak perlu disetup.

|  |  |  |
| --- | --- | --- |
| **Object** | **Properties**  **Name Caption/text** | |
|  |  |
| Group Box1 | Group Box1 | Cari Data Barang |
| Edit1 | Ecari | - |
| Edit2 | Enmbarang | - |
| Edit3 | Ehrgbeli | - |
| Edit4 | Kdpemasok | - |
| Label1 | Label1 | Kode Barang |
| Label2 | Label2 | Nama Barang |
| Label3 | Label3 | Harga Beli |
| Label4 | Label4 | Kode Pemasok |
| Button1 | Ecari | Cari |
| Button2 | Ebatal | Batal |
| Button3 | Eclose | Close |
| Object | Data Base Name | Table Name |
| Table1 | - | - |
| Object | Data Set |  |
| Data Source1 | - |  |
| Object | Data Source |  |
| Dbgrid1 | - |  |
| Object | Active |  |
| Table1 | - |  |

# Listing Program

Ketika pada saat program dijalankan maka proses koneksi program akan diaktifkan selama form tersebut diaktifkan.

procedure TForm11.FormCreate(Sender: TObject); begin

table1.DatabaseName := 'penjualan' ; table1.TableName := 'barang.db'; table1.Active:= true; datasource1.DataSet:= table1; dbgrid1.DataSource := datasource1; end;

Program pencarian dimulai dengan input kode barang pada txtcari dan diakhiri dengan button cari. Jika data barang tidak ditemukan maka akan ditampilkan pesan, tetapi jika tditemukan maka data barang akan ditampilkan.

procedure TForm11.bcariClick(Sender: TObject); begin

table1.IndexName := '';

if not table1.FindKey([ecari.Text]) then begin

showmessage('Data Belum ada'); exit;

form11.ActiveControl := ecari; end

else

enmbarang.Text := table1['nmbarang']; ehrgbeli.Text := table1['hrgbeli']; ekdpemasok.Text := table1['kdpemasok']; end;

# Penjelasan

*Table1.indexname := ‘ ‘* menunjukan bahwa proses pencarian deengan memanfaatkan kunci utama (*primary key*), sehingga koneksi program dianjurkan dengan menggunakan metode *findkey*. *If Not table1.findkey([ecari.text])* menunjukan bahwa jika data tidak ada, maka akan ditampilkan pesan bahwa tidak ditemukan dan kursor dikembalikan ke txtcari, tetapi jika ada maka data akan ditampilkan.

Program untuk membatalkan proses pencarian dengan mengaktifkan txtcari kembali.

procedure TForm11.bbatalClick(Sender: TObject); begin

ecari.Text := ''; form11.ActiveControl := ecari; end;

Program untuk menutup form

procedure TForm11.bbatalClick(Sender: TObject); begin

if (application.MessageBox('Anda yakin form akan ditutup','Info',MB\_YESNO)=

IDYES) then close;

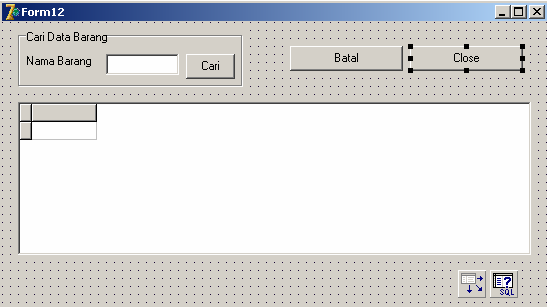
end;

# Dengan Metode SQl

* 1. **Hasil Setelah Form Dijalankan**



# Desain Form



* 1. **Desain Properties**

**Catatan =** untuk koneksi dengan perintah SQL maka icon table tidak dibutuhkan tetapi icon query yang dibutuhkan. Icon Query terdapat dalam tab BDE.

|  |  |  |
| --- | --- | --- |
| **Object** | **Properties**  **Name Caption/text** | |
|  |  |
| Group Box1 | Group Box1 | Cari Data Barang |
| Edit1 | Ecari | - |
| Label1 | Label1 | Nama Barang |
| Button1 | Ecari | Cari |
| Button2 | Ebatal | Batal |
| Button3 | Eclose | Close |
| Object | Data Base Name | Table Name |
| Query1 | - | - |
| Object | Data Set |  |
| Data Source1 | - |  |
| Object | Data Source |  |
| Dbgrid1 | - |  |
| Object | Active |  |
| Query | - |  |

# Listing Program

**Program koneksi dengan perintah SQl.**

procedure TForm12.FormCreate(Sender: TObject);

//var strsql : string; begin

query1.DatabaseName := 'penjualan'; query1.SQL.Add ('Select \* from barang'); query1.Active := true; datasource1.DataSet := query1; dbgrid1.DataSource := datasource1;

end;

# Penjelasan

Secara umum koneksi databse dan tabel hampir sama dengan koneksi dengan icon table, hanya terdapat perbedaan ketika harus mengkatifkan tabel. Dengan query harus dituliskan sintaks perintah SQL. Kondisi ini lebih luwes karena bisa menampilkan data lebih dari satu tabel.

# Program pencarian nama

procedure TForm12.bcariClick(Sender: TObject); var strsql : string;

begin

strsql := 'Select \* from barang where nmbarang like "' + ecari.Text + '%"';

query1.SQL.Clear; query1.SQL.Add(strsql); query1.Close; query1.Open;

end;

# Program Untuk mengembalikan data ke semua record

procedure TForm12.bbatalClick(Sender: TObject); var sql : string;

begin

sql := 'Select \* from barang' ; query1.SQL.Clear; query1.SQL.Add(sql); query1.Close;

query1.Open; end;

# Program menutup Form

procedure TForm12.bcloseClick(Sender: TObject); begin

close; end.