

Form Transaksi Penjualan

Pembuatan form transaksi penjualan (menggunakan VFP 8):

Skenario :

Akan dibuat suatu form transaksi penjualan untuk melayani proses penjualan. Untuk menyederhanakan pada proses ini tidak diberikan data user/kasir maupun fasilitas login bagi user/kasir. Nomor nota mempunyai format "JL-99999" dengan 99999 adalah nomor urut misal JL-00001 dst.

Pengisian item pembelian mengacu pada kode barang, jika kode barang yang diisikan salah/tidak ditemukan di tabel barang maka akan memunculkan form untuk pencarian barang dengan fasilitas filter nama barang. User tidak diperbolehkan untuk mengubah harga jual.

Setiap penyimpanan transaksi penjualan akan disimpan di tabel penjualan (HJual dan DJual), mengupdate stok di tabel barang dan menambah nomor nota penjualan sebanyak 1.

Tabel yang dibutuhkan :

1. Tabel Untuk Menyimpan data Barang :
 - Barang.dbf
2. Tabel Untuk Penjualan :
 - Hjual.dbf → Header Jual
 - DJual.dbf → Detail Jual
3. Tabel Temporary Untuk pembuatan grid nota transaksi penjualan :
 - Tmpjual.dbf
4. Tabel Untuk Menyimpan Nomor Nota :
 - Nomor.dbf

Struktur Tabel :

Barang.dbf

Nama Field	Type	Lebar	Desimal	Index	Keterangan
Kodebrg	Character	8	-	Ya	Kode Barang
Namabrg	Character	35	-	Ya	Nama Barang
Satuan	Character	10	-		Satuan Barang
HrgBeli	Integer	4	0		Harga Beli
HrgJual	Integer	4	0		Harga Jual
Stok	Integer	4	0		Stok Barang

Hjual.dbf

Nama Field	Type	Lebar	Desimal	Index	Keterangan
Nonota	Character	10	-	Ya	Nomor Nota Transaksi
Tanggal	Date	8	-	Ya	Tanggal Transaksi
Total	Integer	4	0		Total Nilai Transaksi
Bayar	Integer	4	0		Besar Pembayaran

Djual.dbf

Nama Field	Type	Lebar	Desimal	Index	Keterangan
Nonota	Character	10	-	Ya	Nomor Transaksi
Kodebrg	Character	8	-		Kode Barang yang dijual
HrgJual	Integer	4	0		Harga Jual saat terjadi transaksi
Qty	Integer	4	0		Jumlah Barang dijual

Tmpjual.dbf

Nama Field	Type	Lebar	Desimal	Index	Keterangan
Kodebrg	Character	8	-	Ya	Kode Barang
Namabrg	Character	35	-	Ya	Nama Barang
Satuan	Character	10	-		Satuan Barang
HrgJual	Integer	4	0		Harga Jual
Qty	Integer	4	0		Jumlah Barang dijual

Nomor.dbf

Nama Field	Type	Lebar	Desimal	Index	Keterangan
Jual	Integer	4	0		Nomor Urut Transaksi Penjualan

I. Desain Form untuk pencarian barang yang dilengkapi dengan filter.

Simpan dengan nama : **FCARIBARANG**

Pengaturan Properties

Form	
AutoCenter	.T. - True
BorderStyle	2 - Fixed Dialog
Caption	Pencarian Barang
MaxButton	.F. - False
MinButton	.F. - False
WindowType	1 - Modal → ini penting !!
Data Environment	
Tabel yang diisikan	Tidak ada tabel yang diisikan / data Environment dikosongkan
1. Label	
Caption	Nama Barang
2. TextBox	
Name	txtCari
Format	!
TabIndex	1

3. Grid	
ColumnCount	4
Name	grdBarang → sebenarnya nama bebas, tapi konsisten
AllowCellSelection	.F. - False
ReadOnly	.T. - True
RecordSourceType	1 - Alias
RecordSource	Barang
ScrollBars	2 -Verticals
TabIndex	2
Berikut ini adalah control yang terletak di dalam grid (bila menggunakan builder akan lebih mudah)	
3.1. Column1	
ControlSource	Barang.kodebrg
3.2. Column2	
ControlSource	Barang.namabrg
3.3. Column3	
ControlSource	Barang.hrgjual
InputMask	999,999,999
Alignment	1 -Middle Right
3.4. Column4	
ControlSource	Barang.satuan

Pengaturan Method :

Agar setiap kali form dijalankan, focus cursor langsung terletak di txtCari, maka tadi pada properties txtCari untuk TabIndex diisi 1.

grdBarang.When

Untuk memudahkan user, maka user dapat menekan ENTER di data yang dipilih di grid. Untuk itu pada Method When diisi :

```

Object: [grdBarang] Procedure: [When]
*- diatur agar saat ditekan ENTER
*- form FCARIBARANG ditutup
ON KEY LABEL ENTER fCariBarang.release()

```

txtCari.GotFocus

Untuk Menghilangkan Efek penekanan ENTER jika focus kembali di txtCari (misal dengan melakukan klik di txtCari) dibuat method :

```

Object: [txtCari] Procedure: [GotFocus]
*- on key label ENTER tanpa tulisan
*- tambahan untuk mengembalikan
*- efek ENTER kembali ke default
ON KEY LABEL ENTER

```

txtCari.InteractiveChange

Setiap kali penulisan sesuatu di txtCari (saat isian txtcari berubah, **InteractiveChange**), maka akan dilakukan filter untuk nama barang yang diawali dengan kata seperti yang ditulis di txtCari, misal jika ditulis "T" maka otomatis daftar barang yang muncul di grid hanya yang diawali dengan huruf "T", kemudian kalau user menekan "A" sehingga di txtCari tertulis "TA" maka otomatis daftar barang yang muncul di grid hanya yang diawali dengan huruf "TA", demikian seterusnya, termasuk jika user menekan tombol BackSpace

untuk menghapus isian di txtCari. Perintah yang digunakan untuk melakukan filter adalah SET FILTER TO isi filternya, sedangkan untuk filternya digunakan fungsi / operator LIKE.

Contoh Penggunaan : SET FILTER TO LIKE("TA*",namabrg) → berarti dilakukan filter untuk namabrg yang 2 huruf depannya TA (tanda * berarti huruf apapun/bebas).

```

Object: [abl]txtCari Procedure: [InteractiveChange] View Parent Code
SELECT barang
*- lakukan filter nama barang yang berhuruf awal sesuai isi txtCari
SET FILTER TO LIKE(ALLTRIM(thisform.txtCari.Value)+"*",namabrg)
*- Refresh tampilan grid agar hasil filter langsung terlihat
thisform.grdBarang.Refresh()
    
```

Form1.Unload

Saat form ditutup (unload), kondisi filter dan efek tombol ENTER dinormalkan

```

Object: [Form1] Procedure: [Unload]
ON KEY LABEL ENTER
SET FILTER TO |
    
```

II. Desain Form Untuk Transaksi Penjualan

Simpan dengan nama : **FJUAL**

Pengaturan Properties

Form	
AutoCenter	.T. - True
BorderStyle	2 - Fixed Dialog
Caption	Form Penjualan
MaxButton	.F. - False
MinButton	.F. - False

DataEnvironment	
Diisi Tabel	tmpjual, barang, nomor
Tabel tmpjual	Exclusive diisi : .T. - True
1. Label	
Caption	Tanggal
2. TextBox	
Name	txtTanggal
Enabled	.F. - False
3. Label	
Caption	Nomor
4. TextBox	
Name	txtNomor
Enabled	.F. - False
5. Shape	
SpecialEffect	0 - 3D
6. Timer	
Name	Timer1
Interval	1000
7. Label	
Name	lblJam
Caption	=TIME()
8. TextBox	
Alignment	1 - Right
DisableBackColor	0,128,64 → Bisa dipilih sendiri
DisabledForeColor	255,255,0 → Bisa dipilih sendiri
Enabled	.F. - False
FontName	LCD → Bisa dipilih sendiri
FontSize	55
InputMask	999,999,999
Name	txtDisplay
9. Grid	
Name	Grid1
ColumnCount	6
RecordSourceType	1 - Alias
RocordSource	Tmpjual
Scrollbars	2 - Vertical
HeaderHeight	30 → bebas, hanya sekedar tampilan
HighLightStyle	1 - Current Row Highlighting Enable (hanya di VFP8 keatas)
Berikut ini adalah control yang terletak di dalam grid (bila tidak ditulis, gunakan defaultnya)	
9.1. Column1	
ControlSource	Tmpjual.kodebrg
9.2. Column2	
ControlSource	Tmpjual.namabrg
Enabled	.F. - False
9.3. Column3	
ControlSource	Tmpjual.satuan
Enabled	.F. - False
9.4. Column4	
ControlSource	Tmpjual.hrgjual
Enabled	.F. - False
Alignment	1 - MiddleRight
InputMask	999,999,999
9.5. Column5	
ControlSource	Tmpjual.qty
Alignment	1 - MiddleRight
InputMask	999
9.6. Column6	
ControlSource	Tmpjual.hrgjual * tmpjual.qty
Alignment	1 - MiddleRight
InputMask	999,999,999
Enabled	.F. - False
10. TextBox	
Name	txtBayar
Enabled	.F. - False
InputMask	999,999,999

Alignment	1 - Right
11. TextBox	
Name	txtBayar
InputMask	999,999,999
SelectOnEntry	.T. - True
Alignment	1 - Right
12. TextBox	
Name	txtKembali
Enabled	.F. - False
InputMask	999,999,999
Alignment	1 - Right
13. Label	
Caption	Total
14. Label	
Caption	Bayar
15. Label	
Caption	Kembali
16. Shape	
SpecialEffect	0 - 3D
17. CommandButton	
Name	cmdHapus
Picture	Delete.bmp → bebas, gunakan yang sesuai
Caption	\<Hapus Item
SpecialEffect	2 - Hot Tracking
MousePointer	15 - Hand → VFP ke atas
PicturePosition	7 - Above Caption, centered relative to caption → VFP ke atas
18. CommandButton	
Name	cmdSimpan
Picture	Save.bmp → bebas, gunakan yang sesuai
Caption	\<Simpan
SpecialEffect	2 - Hot Tracking
MousePointer	15 - Hand → VFP ke atas
PicturePosition	7 - Above Caption, centered relative to caption → VFP ke atas
19. CommandButton	
Name	cmdBatal
Picture	Cancel.bmp → bebas, gunakan yang sesuai
Caption	\<Batal
SpecialEffect	2 - Hot Tracking
MousePointer	15 - Hand → VFP ke atas
PicturePosition	7 - Above Caption, centered relative to caption → VFP ke atas
20. CommandButton	
Name	cmdKeluar
Picture	Close.bmp → bebas, gunakan yang sesuai
Caption	\<Keluar
SpecialEffect	2 - Hot Tracking
MousePointer	15 - Hand → VFP ke atas
PicturePosition	7 - Above Caption, centered relative to caption → VFP ke atas
21. Shape	
SpecialEffect	0 - 3D

Pengaturan Method :

Terdapat 2 kejadian yang dijalankan lebih dari 1 kali, yaitu :

- Mengosongkan tampilan, sekaligus mengisi txtTanggal dengan tanggal sekarang dan nomor nota dengan nomor transaksi terakhir yang tersimpan + 1
 - Terjadi saat : Awal **form** dibuka (**Init**), Pembatalan (**cmdBatal** di **klik**), selesai menyimpan (**cmdSimpan** di **klik**)
- Menghitung total nilai faktur, sekaligus menampilkan hasilnya di txtDisplay
 - Terjadi saat : Setiap kali jumlah barang selesai diisikan (Pengisian Item pada QTY), Saat dilakukan penghapusan (**cmdHapus** di **klik**)

Untuk kedua hal diatas akan dibuat method baru di dalam form bernama : kosongkan dan hitung_total, caranya Pilih menu Form | New Method, kemudian ketikkan kosongkan, klik Add, ketikkan hitung_total, klik Add, klik Close

Untuk mengisikan methodnya, pilih methodnya di form1.

Form1.kosongkan

```
Object: Form1 Procedure: kosongkan View Parent Cod

*-- Mengisi tanggal dengan tanggal sekarang
thisform.txtTanggal.Value = DATE()
*-- pilih tabel nomor, buat variabel nomorjual
*-- yang berisi nomor transaksi terakhir + 1
SELECT nomor
nomorjual = jual + 1
*-- Mengisi txtNomor dengan nomor nota
*-- berformat JL-99999
thisform.txtNomor.Value = "JL-"+PADL(nomorjual,5,'0')
*-- Kosongkan tampilan lain
thisform.txtDisplay.Value=0
thisform.txtTotal.Value=0
thisform.txtBayar.Value=0
thisform.txtKembali.Value=0
*-- Kosongkan tabel tmpjual, sehingga grid item kosong
SELECT tmpjual
DELETE ALL
*-- kemudian tambahkan 1 baris kosong
APPEND BLANK
*-- Fokuskan ke grid item
thisform.grid1.SetFocus()
```

Form1.hitung_total

```
Object: Form1 Procedure: hitung_total View Parent Code

SELECT tmpjual
*-- simpan posisi Record di tmpjual saat ini
nRecordSekarang = RECNO()
*-- buat variabel nTotal berisi nilai awal 0
nTotal = 0
*-- SCAN - ENDSCAN digunakan untuk menyisir tabel
*-- dari Record paling atas ke record paling bawah
SCAN
    *- akumulasikan qty * hrgjual ke variabel nTotal
    nTotal = nTotal + (qty * hrgjual)
ENDSCAN
*- selesai hitung total, kembalikan ke posisi Record semula
GO (nRecordSekarang)
*- tampilkan nilai nTotal di txtDisplay dan txtTotal
thisform.txtDisplay.Value=nTotal
thisform.txtTotal.Value = nTotal
```

Saat di awal Form dijalankan (**Init**), dilakukan proses pengosongan

Form1.Init

```
Object: Form1 Procedure: Init  
thisform.kosongkan()
```

Pengaturan method di grid item.

Grid1.Column1.Text1 (Pengisian Kode Barang)

Pada **Grid1.Column1.Text1.Lostfocus**, setiap kali user mengetikkan kode barang kemudian menekan ENTER atau TAB untuk berpindah kolom maka dilakukan proses pengecekan data kode barang

```
Object: Text1 Procedure: LostFocus  
*-- Jika User mengisikan Sesuatu/Kode barang  
IF !EMPTY(this.Value)  
    SELECT barang  
    *-- cari kodebrg yang sama dengan yang diketikkan user  
    LOCATE FOR kodebrg = this.Value  
    IF !FOUND()  
        *-- kalau tak ketemu buka form pencarian barang  
        DO FORM fCariBarang  
    ENDIF  
    *-- masukkan isi field barang sesuai hasil pencarian  
    *-- sehingga terbentuk m.kodebrg, m.namabrg, dst  
    SCATTER MEMVAR  
    *-- masukkan hasil scatter (m.kodebrg, dst) ke tmpjual  
    SELECT tmpjual  
    GATHER MEMVAR  
ENDIF
```

Pada **Grid1.Column1.Text1.Keypressed**, dilakukan pengecekan tombol apa yang ditekan user, kalau user menekan tombol panah ke bawah dan saat itu posisi kursor di baris paling bawah dari grid item maka focus.kursor akan dipindahkan ke posisi pembayaran (**txtBayar**)

```
Object: Text1 Procedure: KeyPress View Parent  
LPARAMETERS nKeyCode, nShiftAltCtrl  
  
*-- jika ditekan tombol panah Bawah (24)  
IF nKeyCode = 24  
    SKIP && maju 1 record  
    IF EOF() && jika sudah melebihi record terbawah  
        thisform.txtBayar.SetFocus() && cursor letakkan di txtbayar  
    ELSE  
        *-- kalau tak melebihi record terbawah  
        *-- kembalikan posisi record (mundur 1 record)  
        SKIP -1  
    ENDIF  
ENDIF
```

Grid1.Column5.Text1 (QTY) (Catatan : QTY = Quantity)

Pada **Grid1.Column5.Text1.Lostfocus**, setiap kali user selesai mengetikkan jumlah QTY penjualan, kemudian menekan ENTER maka akan dihitung nilai faktur yang terbaru, selain itu, kalau saat itu posisi cursor terletak di baris terbawah maka akan ditambahkan satu baris baru secara otomatis, dan user siap untuk mengisi kode barang berikutnya

```
Object: [abl]Text1 Procedure: ⚡ LostFocus

*-- jika setelah mengisi QTY menekan ENTER (13)
IF LASTKEY()=13
    SKIP    && masju 1 record
    IF EOF() && jika sudah melebihi record terbawah
        *-- letakkan fokus / cursor di kolom pertama grid
        thisform.grid1.column1.SetFocus()
        *-- tambah 1 record baru
        APPEND BLANK
    ENDIF
ENDIF
*-- hitung total nilai faktur
thisform.hitung_total()
*-- refresh tampilan form
thisform.Refresh()
```

Proses pembayaran

Proses pembayaran terjadi di **txtBayar**, setiap kali user mengetikkan angka pembayaran (berarti isi **txtBayar** berubah, **InteractiveChange**) maka isi yang diketikkan tersebut akan dimunculkan dengan tampilan besar di **txtDisplay**. Selain itu setelah user selesai mengetikkan jumlah pembayaran dan menekan ENTER atau TAB untuk berpindah ke control selanjutnya akan dilakukan pengecekan/ validasi, apakah jumlah pembayarannya sudah benar, kalau belum benar (dalam hal ini pembayaran terlalu kecil) maka akan muncul peringatan dan bahwa pembayaran kurang besar dan user harus mengisi lagi dengan jumlah pembayaran yang benar (tidak bisa berpindah dari **txtBayar** jika jumlah pembayaran belum benar)

txtBayar.InteractiveChange

```
Object: [abl]txtBayar Procedure: ⚡ InteractiveChange

*-- apapun yang ditulis di txtBayar ini
*-- akan dimunculkan di txtDisplay
thisform.txtDisplay.Value = this.Value
```

txtbayar.Valid

```
Object: [ab]txtBayar Procedure: ⚡ Valid View Parent Code
*-- jika isi pembayaran lebih kecil dari total nilai faktur
IF (this.Value < thisform.txtTotal.Value)
    MESSAGEBOX("Pembayaran Kurang Besar",0+16,"Peringatan")
    *-- return 0 pada valid menyebabkan user tidak bisa
    *-- keluar / berpindah dari txtBayar
    RETURN 0
ELSE
    *-- jika pembayaran benar, hitung kembalian
    thisform.txtKembali.Value = ;
        thisform.txtBayar.Value - thisform.txtTotal.Value
    *-- tampilkan hasil kembalian di txtDisplay
    thisform.txtDisplay.Value = ;
        thisform.txtKembali.Value
    *-- return .T. berarti cursor berpindah ke control lain
    RETURN .T.
ENDIF
```

Proses Penyimpanan

Pada proses penyimpanan (melakukan **klik** pada **cmdSimpan**), maka hal yang dilakukan adalah : menghapus data di Grid Item yang mempunyai jumlah jual (QTY) = 0, karena kalau tidak ada qty dianggap tidak ada yang dijual sehingga tak perlu dicatat. Setelah itu dilakukan penyimpanan data di **Hjual**. Kemudian dengan menggunakan SCAN-ENDSCAN, menyisir semua isi **tmpjual**, satu persatu data item yang terdapat di grid item disimpan ke dalam **Djual** dan mengubah stok di tabel **barang**, dengan mengurangi stok di tabel barang saat sebelum penjualan dikurangi jumlah penjualan (QTY). Setelah semua selesai dilakukan update tabel **nomor** untuk field **jual** (nomor transaksi jual), isinya ditambah 1.

Di Materi ini juga diberikan Teknik untuk pencetakan nota sederhana, dikarenakan tidak menggunakan printer maka pencetakan dilakukan dengan mengarahkan pencetakan ke file menggunakan **SET PRINTER TO**.

SET PRINTER TO nota.txt → mencetak ke file nota.txt

SET PRINTER TO *nama_printer* → mencetak ke Printer yang dituju

SET PRINTER ON → menyalakan fasilitas pencetakan

Syntax ? (tanda tanya) adalah untuk menulisi ke dalam hal yang dituju oleh SET PRINTER, tanda ?? → sama dengan ?, untuk menulisi tetapi tidak berganti baris lebih dahulu.

TRANSFORM → untuk merubah tampilan dengan format tertentu, misal TRANSFORM(12345,'999,999') akan menghasilkan 12,345 → terdapat pemisah ribuan

Setelah semua penyimpanan beres, maka tampilan kembali dikosongkan untuk transaksi berikutnya

cmdSimpan.Click

Object: cmdSimpan

Procedure: Click

View Parent C

```

*- Siapkan Variabel untuk penyimpanan
dTanggal = thisform.txtTanggal.Value
cNonota   = thisform.txtNomor.Value
nTotal    = thisform.txtTotal.Value
nBayar    = thisform.txtBayar.Value
*- Hapus semua data di tmpjual yang qtynya NOL
DELETE FROM tmpjual WHERE qty = 0
*- Simpan data di Hjual
INSERT INTO hjual (tanggal, nonota, total, bayar) ;
VALUES ;
(dTanggal, cNonota, nTotal, nBayar)

SELECT tmpjual
SCAN
    cKodeBrg = kodebrg
    nQty      = qty
    nHrgJual = hrgjual
    *-- simpan data di Djual
    INSERT INTO djual (nonota, kodebrg, qty, hrgjual) ;
VALUES ;
(cNonota, cKodebrg, nQty, nHrgjual)
    *-- Update stok barang, stok dikurangi jumlah jual
    UPDATE barang SET stok = stok - nQty ;
WHERE kodebrg = cKodebrg

    SELECT tmpjual
ENDSCAN
*-- nomor Penjualan ditambah 1
UPDATE nomor SET jual = jual + 1
*-- BUAT NOTA
SET CONSOLE OFF && agar tak muncul hasil ? di layar
SET PRINTER TO nota.txt && arahkan pencetakan ke file nota.txt
SET PRINTER ON          && pencetakan di hidupkan
*-- catatan : TRANSFORM untuk mencetak angka dengan format tertentu
?" -- NOTA PENJUALAN -- "
?"=====
?"Tgl:",dTanggal," NO:",cNonota
?"-----"

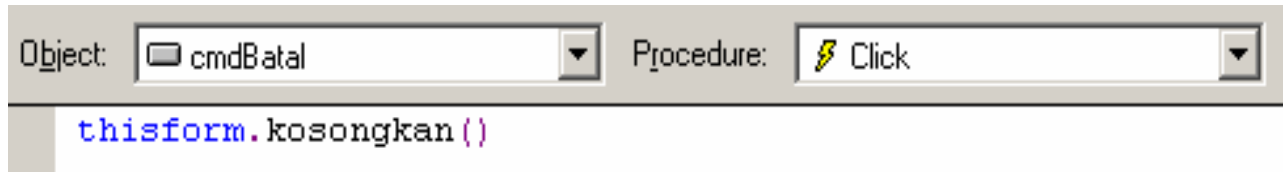
SELECT tmpjual
SCAN
    ?[""+kodebrg+""], " ",namabrg
    ?qty," x ", TRANSFORM(hrgjual,'9,999,999')," = "
    ??TRANSFORM(hrgjual * qty,'99,999,999')
ENDSCAN
?"-----"
?"TOTAL    :",TRANSFORM(nTotal,'99,999,999')
?"BAYAR    :",TRANSFORM(nBayar,'99,999,999')
?"KEMBALI :",TRANSFORM(nBayar - nTotal,'99,999,999')
?"      :: TERIMA KASIH :: "
SET PRINTER TO && kembalikan arah pencetakan ke default
SET PRINTER OFF && matikan pencetakan
*-- menampilkan isi file nota.txt (tulis jika dibutuhkan)
MODIFY FILE nota.txt NOEDIT
*--- kosongkan data
thisform.kosongkan()

```

Beberapa tombol lain yang dibutuhkan adalah :

cmdBatal.Click

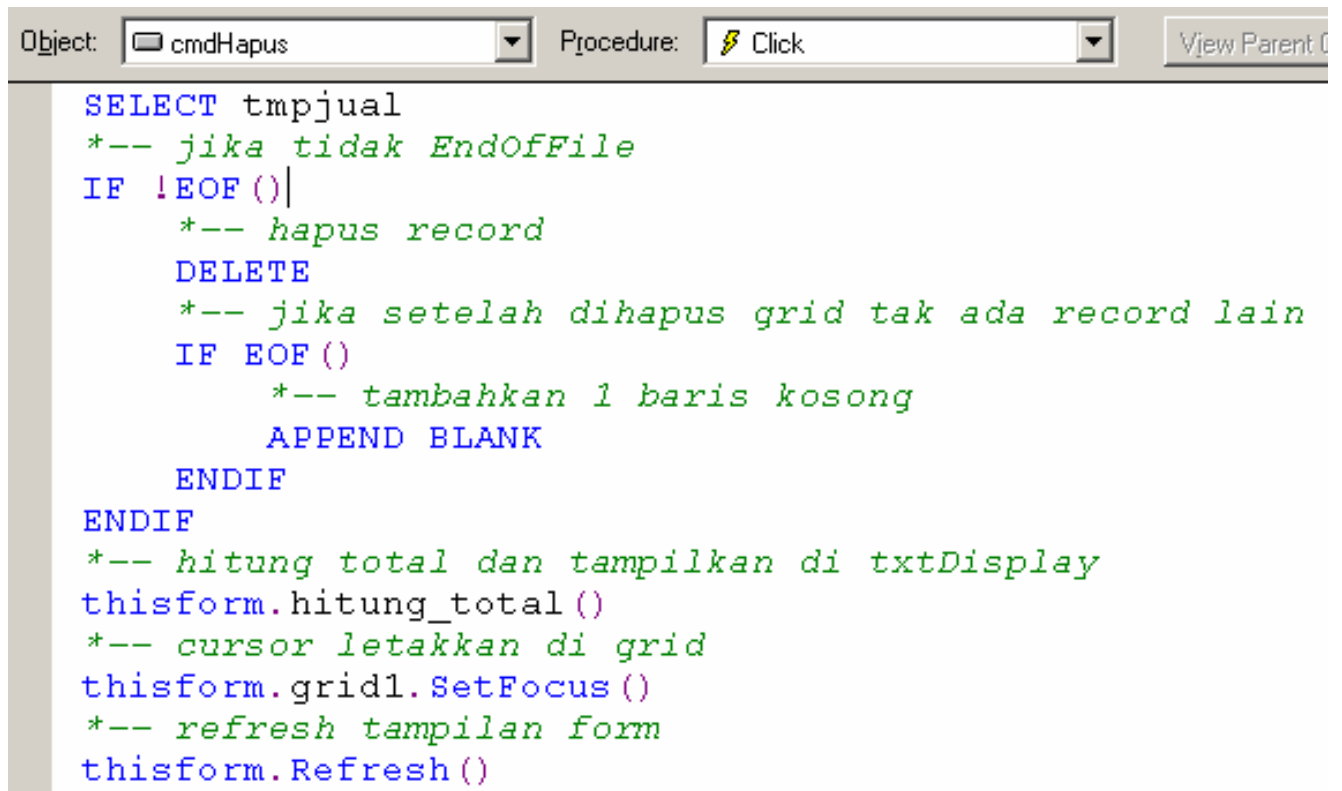
Untuk membatalkan proses penjualan yang sedang berlangsung, tinggal mengosongkan tampilan



```
Object: cmdBatal Procedure: Click  
thisform.kosongkan()
```

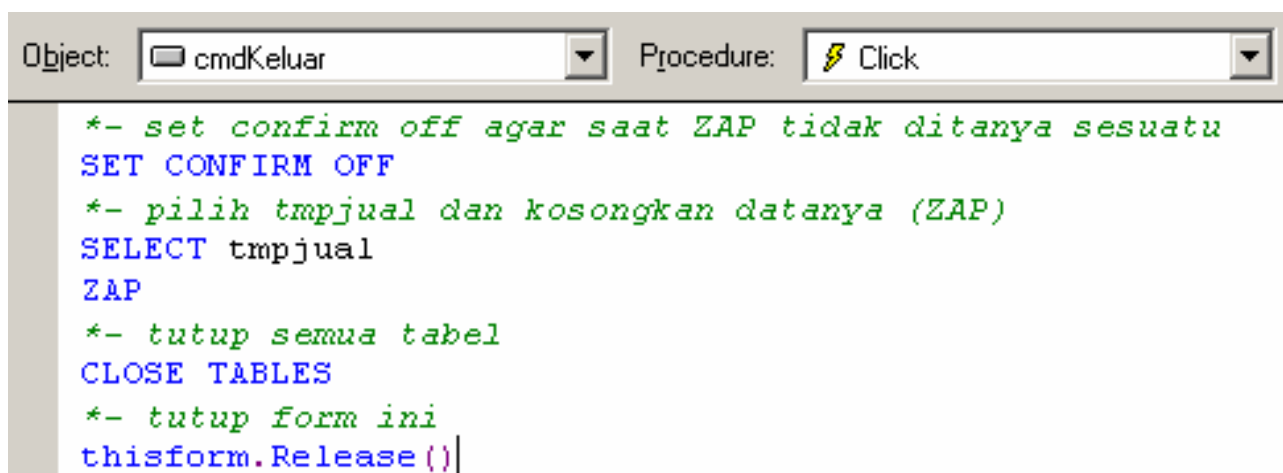
cmdHapus.Click

Untuk menghapus item penjualan yang sudah terlanjur diisikan di dalam grid item



```
Object: cmdHapus Procedure: Click View Parent  
SELECT tmpjual  
*-- jika tidak EndOfFile  
IF !EOF()|  
    *-- hapus record  
    DELETE  
    *-- jika setelah dihapus grid tak ada record lain  
    IF EOF()|  
        *-- tambahkan 1 baris kosong  
        APPEND BLANK  
    ENDIF  
ENDIF  
*-- hitung total dan tampilkan di txtDisplay  
thisform.hitung_total()|  
*-- cursor letakkan di grid  
thisform.grid1.SetFocus()|  
*-- refresh tampilan form  
thisform.Refresh()
```

cmdKeluar.Click



```
Object: cmdKeluar Procedure: Click  
*-- set confirm off agar saat ZAP tidak ditanya sesuatu  
SET CONFIRM OFF  
*-- pilih tmpjual dan kosongkan datanya (ZAP)  
SELECT tmpjual  
ZAP  
*-- tutup semua tabel  
CLOSE TABLES  
*-- tutup form ini  
thisform.Release()
```

Oke... saya rasa cukup sampai disini dahulu, memang proses diatas cukup sederhana akan tetapi sudah betul-betul bisa diterapkan untuk transaksi yang nyata. Anda bisa menambahkan sendiri hal-hal lain sesuai kebutuhan, yang penting terus berlatih, berlatih dan berlatih..... lebih baik mencobanya 1 jam di depan komputer dari pada 10 kali membacanya.... Good Luck and GBU.....