

Aplikasi Inventory Barang Dengan Visual Basic .Net

Khoirur Rozikin, M.Kom

Aplikasi Inventory Barang Dengan Visual Basic .Net

**DIGUNAKAN UNTUK MATAKULIAH
PEMROGRAMAN CLIENT SERVER**

Oleh: Khoirur Rozikin, M.Kom



YAYASAN PRIMA AGUS TEKNIK

Aplikasi Inventory Barang Dengan Visual Basic .Net

Penulis:

Khoirur Rozikin, M.Kom

ISBN : 978-623-6141-34-2 (PDF)

Editor:

Indra Ava Dianta, S.Kom., M.T

Penyunting :

Zaenal Mustofa, S.Kom.,M. Kom

Desain Sampul dan Tata Letak :

Khoirur Rozikin, M.Kom

Penerbit :

Yayasan Prima Agus Teknik
Redaksi: Jln Majapahit No 605 Semarang
Tlpn. (024) 6723456
Fax . 024-6710144
Email: penerbit_ypat@stekom.ac.id

Distributor Tunggal:

UNIVERSITAS STEKOM
Jln Majapahit No 605 Semarang
Tlpn. (024) 6723456
Fax . 024-6710144
Email: info@stekom.ac.id

Hak Cipta dilindungi Undang undang
Dilarang memperbanyak karya Tulis ini dalam bentuk dan dengan cara apapun
tanpa ijin tertulis dan penerbit.

KATA PENGANTAR

Visual Basic .NET merupakan bahasa pemrograman komputer berorientasi objek yang diimplementasikan pada .NET Framework. Visual Basic .Net merupakan pengembangan dari bahasa pemrograman Visual Basic 6. Bahasa pemrograman Visual Basic .Net banyak digunakan untuk pembuatan aplikasi baik aplikasi *single user* maupun *multi user*.

Buku ini di susun untuk dipersembahkan kepada mahasiswa Universitas STEKOM agar dapat membuat aplikasi dengan bahasa pemrograman Visual Basic .Net. Buku ini berisi contoh aplikasi Inventory Barang yang terdiri dari Form Barang untuk menyimpan data Barang, Form Suplier untuk menyimpan data Suplier, Form Pembelian untuk menyimpan transaksi Pembelian, Form Customer untuk menyimpan data Customer dan Form Pembelian untuk menyimpan data Pembelian. Pembuatan laporan tidak menggunakan program bantu tetapi menggunakan komponen *Print Document* yang sudah ada di Visual Basic .Net, laporan yang dibuat yaitu laporan data Barang, data Suplier, data Customer, data Pembelian PerFaktur, data Pembelian PerPeriode dan data Penjualan PerNota. Form Pemakai, form Login dan form Hak Akses digunakan untuk mengatur hak akses pemakai dalam mengakses form dan laporan yang ada di aplikasi Inventory. Untuk lebih memahami materi, pada buku ini terdapat *link* video pembelajaran untuk masing-masing materi.

Harapan penyusun, buku ini bisa bermanfaat dan bisa digunakan oleh mahasiswa dan pembaca pada umumnya untuk pembuatan aplikasi menggunakan bahasa pemrograman Visual Basic .Net.

Penulis

DAFTAR ISI

Cover	i
Halaman Judul	ii
Kata Pengantar	iv
BAB I APLIKASI INVENTORY BARANG	1
1.1 Form Barang	1
1.2 Form Suplier	1
1.3 Form Pembelian	2
1.4 Form Customer	2
1.5 Form Penjualan	3
1.6 Form Menu Utama	3
1.7 Cetak Data Barang	4
1.8 Cetak Data Suplier	4
1.9 Cetak Pembelian PerFaktur	4
1.10 Cetak Pembelian PerPeriode	5
1.11 Cetak Data Customer	5
1.12 Cetak Data Penjualan PerNota	5
1.13 Cetak data Penjualan PerPeriode	6
1.14 Form Pemakai	6
1.15 Form Login	6
1.16 Form Hak Akses	7
1.17 Tutorial Aplikasi Inventory Barang	7
BAB II FORM DATA BARANG	8
2.1 Capaian Pembelajaran	8
2.2 Form Data Barang	8
2.3 Tutorial Form Barang	13
2.4 Evaluasi	13
2.5 Referensi	13
BAB III FORM DATA SUPLIER	14
3.1 Capaian Pembelajaran	14
3.2 Form Data Suplier	14
3.3 Tutorial Form Suplier	19
3.4 Evaluasi	19
3.5 Referensi	19
BAB IV FORM TRANSAKSI PEMBELIAN	20
4.1 Capaian Pembelajaran	20
4.2 Form Transaksi Pembelian	20

4.3 Tutorial Form Pembelian	26
4.4 Evaluasi	27
4.5 Referensi	27
BAB V FORM DATA CUSTOMER	28
5.1 Capaian Pembelajaran	28
5.2 Form Data Customer	28
5.3 Tutorial Form Customer	33
5.4 Evaluasi	33
5.5 Referensi	33
BAB VI FORM TRANSAKSI PENJUALAN	34
6.1 Capaian Pembelajaran	34
6.2 Form Transaksi Penjualan	34
6.3 Tutorial Form Penjualan	41
6.4 Evaluasi	41
6.5 Referensi	41
BAB VII FORM MENU UTAMA	42
7.1 Capaian Pembelajaran	42
7.2 Form Menu Utama	42
7.3 Tutorial Form Menu Utama	44
7.4 Evaluasi	44
7.5 Referensi	44
BAB VIII MENCETAK DATA MASTER	45
8.1 Capaian Pembelajaran	45
8.2 Mencetak Data Barang	45
8.3 Mencetak Data Suplier	47
8.4 Mencetak Data Customer	49
8.3 Tutorial Mencetak data Master	50
8.4 Evaluasi	50
8.5 Referensi	50
BAB IX MENCETAK PEMBELIAN PERFAKTUR	51
9.1 Capaian Pembelajaran	51
9.2 Form Pembelian PerFaktur	51
9.3 Tutorial Pembelian PerFaktur	57
9.4 Evaluasi	57
9.5 Referensi	57
BAB X MENCETAK PEMBELIAN PERPERIODE	58
10.1 Capaian Pembelajaran	58

10.2 Form Pembelian PerPeriode	58
10.3 Tutorial Mencetak Pembelian PerPeriode	64
10.4 Evaluasi	64
10.5 Referensi	64
BAB XI MENCETAK PENJUALAN PERNOTA.....	65
11.1 Capaian Pembelajaran	65
11.2 Form Penjualan PerNota	65
11.3 Tutorial Mencetak Penjualan PerNota	65
11.4 Evaluasi	71
11.5 Referensi	71
BAB XII MENCETAK PENJUALAN PERPERIODE	72
12.1 Capaian Pembelajaran	72
12.2 Form Penjualan PerPeriode	72
12.3 Tutorial Mencetak Penjualan PerPeriode	78
12.4 Evaluasi	78
12.5 Referensi	78
BAB XIII FORM DATA PEMAKAI	79
13.1 Capaian Pembelajaran	79
13.2 Form Data Pemakai	79
13.3 Tutorial Form Pemakai	84
13.4 Evaluasi	84
13.5 Referensi	84
BAB XIV FORM LOGIN PEMAKAI	85
14.1 Capaian Pembelajaran	85
14.2 Form Login Pemakai	85
14.3 Tutorial Login Pemakai	89
14.4 Evaluasi	89
14.5 Referensi	89
BAB XV FORM HAK AKSES	90
15.1 Capaian Pembelajaran	90
15.2 Form Hak Akses.....	90
15.3 Tutorial Hak Akses	94
15.4 Evaluasi	94
15.5 Referensi	94

BAB I

APLIKASI INVENTORY BARANG

1.1 FORM BARANG

KODEBRG	NAMABRG	HRGBELI	HRGJUAL	JMLAWAL	JMLBELI	JMLJUAL	JMLSTOK
*							

Gambar 1.1 Form Data Barang

Form Barang digunakan untuk menyimpan data master Barang, data yang diinput yaitu Kode Barang, Nama Barang, Harga Beli, Harga Jual dan Jumlah Awal. Data Jumlah Beli akan terisi otomatis dari data Transaksi Pembelian, data Jumlah Jual akan terisi otomatis dari data Transaksi Penjualan dan data Jumlah Stok terisi dari Jumlah Awal ditambah Jumlah Beli dikurangi Jumlah Jual.

1.2 FORM SUPLIER

KODESP	NAMASP	ALAMATSP	TELPONSP
*			

Gambar 1.2 Form Data Suplier

Form Suplier digunakan untuk menyimpan data master Suplier yang terdiri dari Kode Suplier, Nama Suplier, Alamat Suplier dan Nomor Telpon Suplier.

1.3 FORM PEMBELIAN

Gambar 1.3 Form Transaksi Pembelian

Form Pembelian digunakan untuk menyimpan data transaksi Pembelian dengan menginput data Nomor Faktur, Tanggal Pembelian dan Kode Suplier. Satu Faktur Pembelian bisa terdiri dari beberapa item barang.

1.4 FORM CUSTOMER

Gambar 1.4 Form Data Customer

Form Customer digunakan untuk menyimpan data master Customer yang terdiri dari Kode Customer, Nama Customer, Alamat Customer dan Nomor Telpon Customer.

1.5 FORM PENJUALAN

KODEBRG	NAMABRG	HRGJUAL	JUMLAH	TOTAL
*				

NOTA	TANGGAL	KODECS	NAMACS	KODEBRG	NAMABRG	HRGJUAL	JUMLAH	TOTAL
*								

Gambar 1.5 Form Transaksi Penjualan

Form Penjualan digunakan untuk menyimpan data transaksi Penjualan dengan menginput data Nomor Nota, Tanggal Penjualan dan Kode Customer. Satu Nota Penjualan bisa terdiri dari beberapa item barang.

1.6 FORM MENU UTAMA

PROGRAM INVENTORY BARANG

Gambar 1.6 Form Menu Utama

Form Menu Utama untuk menjalankan Form Barang, Form Suplier, Form Pembelian, Form Customer dan Form Penjualan dari satu form.

1.7 CETAK DATA BARANG

LAPORAN DATA BARANG

NOMOR	KODEBRG	NAMA BARANG	HRG BELI	HRG JUAL	JML AWAL	JML BELI	JML JUAL	JML STOK
1	KYB001	KEYBOARD	65.000	70.000	25	35	18	42
2	MNT001	MONITOR 15"	175.000	200.000	10	5	12	3
3	MOU001	MOUSE LG	25.000	30.000	25	47	26	46

Gambar 1.7 Laporan Data Barang

Menampilkan data Barang di layar yang bisa pula dicetak ke printer.

1.8 CETAK DATA SUPLIER

LAPORAN DATA SUPLIER

NOMOR	KODE	NAMA SUPLIER	ALAMAT	NO. TELPON
1	SP001	CV. COMPUTAMA	JL. MAJAPAHIT 25	088802250099
2	SP002	PT. TEKNOCOMP	JL. GAJAHMADA	088802250066

Gambar 1.8 Laporan Data Suplier

Menampilkan data Suplier di layar yang bisa pula dicetak ke printer.

1.9 CETAK DATA PEMBELIAN PERFAKTUR

LAPORAN PEMBELIAN PERFAKTUR

NO. FAKTUR : FK002		KODE SUPLIER : SP002			
TANGGAL : 27/01/2021		NAMA SUPLIER : PT. TEKNOCOMP			
NOMOR	KODEBRG	NAMA BARANG	HRG BELI	JUMLAH	TOTAL
1	MNT001	MONITOR 15"	150.000	5	750.000
2	KYB001	KEYBOARD	65.000	15	975.000
3	MOU001	MOUSE LG	25.000	20	500.000
TOTAL :					2.225.000

Gambar 1.9 Laporan Pembelian PerFaktur

Menampilkan data Pembelian PerFaktur di layar yang bisa pula dicetak ke printer.

1.10 CETAK DATA PEMBELIAN PERPERIODE

LAPORAN PEMBELIAN PERPERIODE TANGGAL 27/01/2021 S/D 12/02/2021

NO.	FAKTUR	TANGGAL	NAMA SUPLIER	KODEBRG	NAMA BARANG	HRG BELI	JUMLAH	TOTAL
1	FK002	27/01/2021	PT. TEKNOCOMP	MNT001	MONITOR 15"	150.000	5	750.000
2	FK002	27/01/2021	PT. TEKNOCOMP	KYB001	KEYBOARD	65.000	15	975.000
3	FK002	27/01/2021	PT. TEKNOCOMP	MOU001	MOUSE LG	25.000	20	500.000
4	FK003	12/02/2021	CV. COMPUTAMA	KYB001	KEYBOARD	65.000	12	780.000
5	FK003	12/02/2021	CV. COMPUTAMA	MOU001	MOUSE LG	25.000	15	375.000
TOTAL :								3.380.000

Gambar 1.10 Laporan Pembelian PerPeriode

Menampilkan data Pembelian PerPeriode di layar yang bisa pula dicetak ke printer.

1.11 CETAK DATA CUSTOMER

LAPORAN DATA CUSTOMER

NOMOR	KODE	NAMA CUSTOMER	ALAMAT	NO. TELPON
1	CS001	STMIK NUSANTARA	JL. DIPONEGORO 25	024654321
2	CS002	SMK NUSA PUTRA	JL. PAHLWAWAN 125	024321456

Gambar 1.11 Laporan Data Customer

Menampilkan data Customer di layar yang bisa dicetak ke printer.

1.12 CETAK DATA PENJUALAN PERNOTA

LAPORAN PENJUALAN PERNOTA

NO. NOTA : NT001 KODE CUSTOMER : CS002
TANGGAL : 27/01/2020 NAMA CUSTOMER : SMK NUSA PUTRA

NOMOR	KODEBRG	NAMA BARANG	HRG JUAL	JUMLAH	TOTAL
1	KYB001	KEYBOARD	70.000	8	560.000
2	MOU001	MOUSE LG	30.000	6	180.000
TOTAL :					740.000

Gambar 1.12 Laporan Penjualan PerNota

Menampilkan data Penjualan PerNota di layar yang bisa pula dicetak ke printer.

1.13 CETAK DATA PENJUALAN PERPERIODE

LAPORAN PENJUALAN PERPERIODE
TANGGAL 30/01/2020 S/D 20/02/2020

NO.	NOTA	TANGGAL	NAMA CUSTOMER	KODEBRG	NAMA BARANG	HRG JUAL	JUMLAH	TOTAL
1	NT002	30/01/2020	STMIK NUSANTARA	KYB001	KEYBOARD	70.000	10	700.000
2	NT002	30/01/2020	STMIK NUSANTARA	MOU001	MOUSE LG	30.000	12	360.000
3	NT003	20/02/2020	SMK NUSA PUTRA	MNT001	MONITOR 15"	160.000	12	1.920.000
4	NT003	20/02/2020	SMK NUSA PUTRA	MOU001	MOUSE LG	30.000	8	240.000
TOTAL :								3.220.000

Gambar 1.13 Laporan Penjualan PerPeriode

Menampilkan data Penjualan PerPeriode di layar bisa dicetak ke printer.

1.14 FORM PEMAKAI

	KODEPK	NAMAPK	JABATAN	PASSWORD	FOTO
*					

Gambar 1.14 Form Pemakai

Form Pemakai menyimpan data pemakai program Inventory Barang.

1.15 FORM LOGIN

KODE PEMAKAI	NAMA PEMAKAI	JABATAN	PASSWORD

Gambar 1.15 Form Login

Form Login untuk mengatur hak akses pemakai sesuai jabatan.

1.16 FORM HAK AKSES

JABATAN	MASTER	TRANSAKSI	CETAK	HAKAKSES	SELESAI
MANAGER	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
KEUANGAN	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
ADMINISTRASI	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
»*	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Gambar 1.16 Form Hak Akses

Form Hak Akses untuk mengatur hak akses masing-masing pemakai sesuai dengan jabatannya tanpa merubah kode program.

1.17 TUTORIAL APLIKASI INVENTORY BARANG



https://youtu.be/AWjtwKle_Lc

BAB II

FORM DATA BARANG

2.1 CAPAIAN PEMBELAJARAN

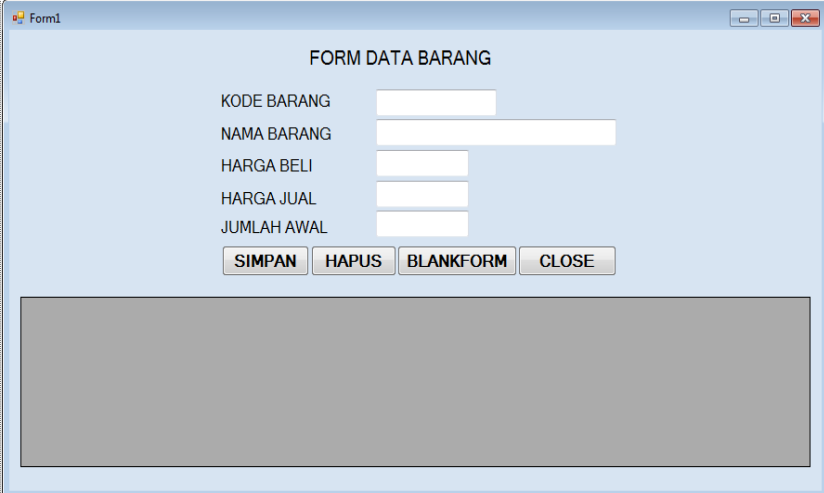
Capaian pembelajaran dari Form Data Barang yaitu mahasiswa memahami struktur tabel Barang, dapat mendesain Form Data Barang, memahami kode program yang digunakan dan cara mengoperasikan Form Data Barang yang meliputi menambahkan data baru, mengedit data dan menghapus data.

2.2 FORM DATA BARANG

A. MENDESAIN FORM DATA BARANG

Langkah-langkah mendesain Form Data Barang sebagai berikut :

1. Aktifkan Program Visual Basic .Net
2. Klik menu File ⇒ New Project
3. Ketik PRINV pada kotak Name, Klik tombol OK
4. Klik tempat kosong pada area Form
5. Klik propertie Name, Ketik FORMBARANG
6. Klik menu File ⇒ Save All, Klik tombol Browse
7. Tentukan folder lokasi penyimpanan file proyek
8. Klik tombol Select Folder, Klik tombol Save
9. Buat desain Form Data Barang di bawah ini




The image shows a screenshot of a Windows application window titled "Form1". The window contains a form titled "FORM DATA BARANG". The form has five text input fields arranged vertically, each with a label to its left: "KODE BARANG", "NAMA BARANG", "HARGA BELI", "HARGA JUAL", and "JUMLAH AWAL". Below these fields are four buttons: "SIMPAN", "HAPUS", "BLANKFORM", and "CLOSE". The form is set against a light blue background, and there is a large grey rectangular area at the bottom of the form, likely intended for a data grid or list.

Gambar 2.1 Desain Form Data Barang

Daftar Objek pada Form Data Barang

Komponen	Properties	Isi
Label	Text	FORM DATA BARANG
	Text	KODE BARANG
	Text	NAMA BARANG
	Text	HARGA BELI
	Text	HARGA JUAL
	Text	JUMLAH AWAL
TextBox	Name	TXTKODEBRG
	Name	TXTNAMABRG
	Name	TXTHRGBELI
	Name	TXTHRGJUAL
	Name	TXTJMLAWAL
Button	Name	BTSIMPAN
	Text	SIMPAN
	Name	BTHAPUS
	Text	HAPUS
	Name	BTBLANKFORM
	Text	BLANKFORM
	Name	BTCLOSE
	Text	CLOSE
DataGridView	Name	DGVBARANG

10. Buat database MS Access DBINV.MDB pada folder PRINV\PRINV\BIN\DEBUG
11. Buat struktur tabel berikut

	Field Name	Data Type	
	KODEBRG	Text	Field Size : 6
	NAMABRG	Text	Field Size : 25
	HRGBELI	Number	
	HRGJUAL	Number	
	JMLAWAL	Number	
	JMLBELI	Number	
	JMLJUAL	Number	
	JMLSTOK	Number	

12. Simpan struktur tabel dengan nama BARANG
13. Tutup program MS Access
14. Kembali ke Form Barang

B. KODE PROGRAM FORM DATA BARANG

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMBARANG
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DABRG As New OleDb.OleDbDataAdapter
    Public DTBRG As New DataTable
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DABRG untuk penghubung data tabel, variabel DTBRG menyimpan data hasil perintah query.

```
Private Sub FORMBARANG_Load(ByVal sender As Object, ByVal e As System.EventArgs) :
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE =DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()
End Sub
```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi.

```
Sub BLANKFORM()
    TXTKODEBRG.Text = ""
    TXTNAMABRG.Text = ""
    TXTHRGBELI.Text = 0
    TXTHRGJUAL.Text = 0
    TXTJMLAWAL.Text = 0
    TXTKODEBRG.Focus()
End Sub
Sub DGV()
    SQLSTR = "SELECT * FROM BARANG"
    DABRG = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTBRG.Clear()
    DABRG.Fill(DTBRG)
    DGVBRG.DataSource = DTBRG
End Sub
Private Sub FORMBARANG_Activated(ByVal sender As Object, I
    BLANKFORM()
    DGV()
End Sub
```

Sub BLANKFORM untuk mengosongkan Textbox input data Barang, sub DGV untuk menampilkan data Barang ke DataGridView DGVBRG. Saat form aktif akan menjalankan sub BLANKFORM dan sub DGV.

```

Private Sub TXTKODEBRG_Leave(ByVal sender As Object, ByVal e As System.EventArgs) Handles DTBERG.Leave
    SQLSTR = "SELECT * FROM BARANG WHERE KODEBRG='" & TXTKODEBRG.Text & "'"
    DABRG = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTBERG.Clear()
    DABRG.Fill(DTBERG)
    DGVBERG.DataSource = DTBERG
    If DTBERG.Rows.Count > 0 Then
        TXTNAMABRG.Text = DGVBERG.Rows.Item(DGVBERG.CurrentRow.Index).Cells(1).Value
        TXTHRGBELI.Text = DGVBERG.Rows.Item(DGVBERG.CurrentRow.Index).Cells(2).Value
        TXTHRGJUAL.Text = DGVBERG.Rows.Item(DGVBERG.CurrentRow.Index).Cells(3).Value
        TXTJMLAWAL.Text = DGVBERG.Rows.Item(DGVBERG.CurrentRow.Index).Cells(4).Value
    End If
    DGV()
End Sub

```

Saat kursor meninggalkan Textbox TXTKODEBRG akan dideteksi apakah Kode Barang yang telah diketik sudah ada di tabel Barang, jika Kode Barang sudah ada maka datanya akan ditampilkan ke masing-masing Textbox.

```

Private Sub BTSIMPAN_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BTSIMPAN.Click
    SQLSTR = "SELECT * FROM BARANG WHERE KODEBRG='" & TXTKODEBRG.Text & "'"
    DABRG = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTBERG.Clear()
    DABRG.Fill(DTBERG)
    DGVBERG.DataSource = DTBERG
    If DTBERG.Rows.Count > 0 Then
        SQLSTR = "UPDATE BARANG SET KODEBRG='" & TXTKODEBRG.Text & "', NAMAARG='" & _
            TXTNAMABRG.Text & "', HRGBELI=" & TXTHRGBELI.Text & ", HRGJUAL=" & TXTHRGJUAL.Text & _
            ", JMLAWAL=" & TXTJMLAWAL.Text & " WHERE KODEBRG='" & TXTKODEBRG.Text & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    Else
        SQLSTR = "INSERT INTO BARANG VALUES ('" & TXTKODEBRG.Text & "','" & TXTNAMABRG.Text & _
            "','" & TXTHRGBELI.Text & "','" & TXTHRGJUAL.Text & "','" & TXTJMLAWAL.Text & "','0,0,0)"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    BLANKFORM()
    DGV()
End Sub

```

Saat diklik tombol SIMPAN akan dideteksi apakah Kode Barang yang disimpan sudah ada di tabel Barang, jika Kode Barang sudah ada maka data Barang akan diupdate sedangkan jika belum ada maka data Barang yang diinput akan ditambahkan sebagai data baru.

```

Private Sub ETHAPUS_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ETHAPUS.Click
    Dim X As String
    Dim XKODEBRG As String
    XKODEBRG = TXTKODEBRG.Text
    X = MsgBox("DATA BARANG " & TXTNAMABRG.Text & " JADI DIHAPUS !", MsgBoxStyle.Yes)
    If X = vbYes Then
        SQLSTR = "DELETE FROM BARANG WHERE KODEBRG='" & XKODEBRG & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    BLANKFORM()
    DGV()
End Sub

```

Saat diklik tombol HAPUS akan ditampilkan konfirmasi apakah data Barang yang sedang tampil akan dihapus, jika diklik tombol Yes maka data Barang yang sedang tampil akan dihapus sedangkan jika diklik tombol No data Barang tidak akan dihapus.

```
Private Sub BTBLANKFORM_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BLANKFORM.Click
    BLANKFORM.Text = ""
End Sub
```

Saat diklik tombol BLANKFORM akan menjalankan sub BLANKFORM untuk mengosongkan Textbox.

```
Private Sub BTCLOSE_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles CLOSE.Click
    KONEKSI.Close()
    Me.Close()
End Sub
```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENGOPERASIKAN FORM DATA BARANG

1. Menambah data Barang

Cara menambahkan data Barang :

- a. Klik menu Debug => Start Debugging
 - b. Isikan data Kode Barang, Nama Barang, Harga Beli, Harga Jual dan Jumlah Awal
 - c. Klik tombol SIMPAN untuk menyimpan data Barang
- Tekan tombol Tab untuk memindahkan kursor ke text box berikutnya.

2. Mengedit data Barang

Cara mengedit data Barang :

- a. Klik menu Debug => Start Debugging
- b. Isikan Kode Barang yang datanya mau diedit
- c. Tekan tombol Tab (data barang yang kodenya sesuai akan tampil)
- d. Lakukan pengeditan data
- e. Klik tombol SIMPAN untuk pengeditan data Barang

3. Menghapus data Barang

Cara menghapus data Barang :

- a. Klik menu Debug => Start Debugging
- b. Isikan Kode Barang yang datanya mau dihapus
- c. Tekan tombol Tab (data barang yang kodenya sesuai akan tampil)
- d. Klik tombol HAPUS
- e. Tampil konfirmasi data jadi dihapus atau tidak
- f. Klik tombol Yes jika data jadi dihapus atau klik tombol No jika data tidak jadi dihapus

2.3 TUTORIAL FORM BARANG



<https://youtu.be/8wFP13Ium84>

2.4 EVALUASI

- a. Atur supaya data Kode Barang hanya bisa diisi maksimal 6 karakter
- b. Atur supaya data Nama Barang hanya bisa diisi maksimal 25 karakter

2.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB III FORM DATA SUPLIER

3.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari Form Data Suplier yaitu mahasiswa memahami struktur tabel Suplier, dapat mendesain Form Data Suplier, memahami kode program yang digunakan dan cara mengoperasikan Form Data Suplier yang meliputi menambahkan data baru, mengedit data dan menghapus data.

3.2 FORM DATA SUPLIER

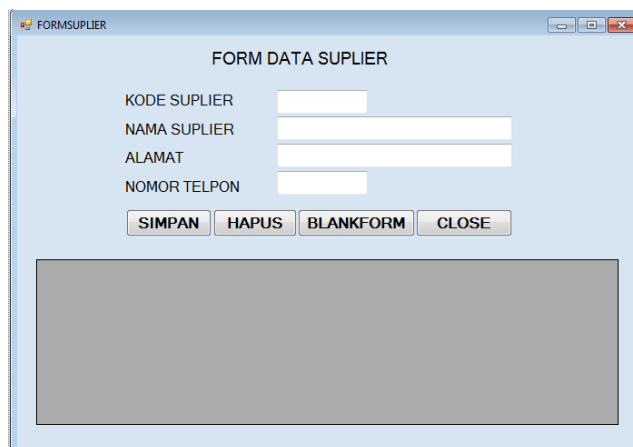
A. MENDESAIN FORM DATA SUPLIER

Langkah-langkah mendesain Form Data Suplier sebagai berikut :

1. Buka database DBINV.MDB pada folder PRINV\PRINV\BIN\DEBUG
2. Buat struktur tabel berikut :

Field Name	Data Type	
KODESP	Text	Field Size : 5
NAMASP	Text	Field Size : 25
ALAMATSP	Text	Field Size : 25
TELPONSP	Text	Field Size : 12

3. Simpan tabel dengan nama SUPLIER
4. Tutup program MS Access
5. Double klik file PRINV.SLN untuk membuka project Inventory
6. Klik menu Project ⇒ Add Windows Form
7. Ketik nama form FORMSUPLIER
8. Klik tombol Add, Buat desain Form Suplier berikut



Gambar 2.2 Desain Form Data Suplier

Daftar Objek pada Form Data Suplier

Komponen	Properties	Isi
Label	Text	FORM DATA SUPLIER
	Text	KODE SUPLIER
	Text	NAMA SUPLIER
	Text	ALAMAT
	Text	NOMORTELPON
TextBox	Name	TXTKODESP
	Name	TXTNAMASP
	Name	TXTALAMATSP
	Name	TXTTELPONSP
Button	Name	BTSIMPAN
	Text	SIMPAN
	Name	BTHAPUS
	Text	HAPUS
	Name	BTBLANKFORM
	Text	BLANKFORM
	Name	BTCLOSE
	Text	CLOSE
DataGridView	Name	DGVSP

B. KODE PROGRAM FORM DATA SUPLIER

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMSUPLIER
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DASP As New OleDb.OleDbDataAdapter
    Public DTSP As New DataTable
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DASP untuk penghubung data tabel, variabel DTSP menyimpan data hasil perintah query.

```

Private Sub FORMSUPLIER_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()
End Sub

```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi.

```

Sub BLANKFORM()
    TXTKODESP.Text = ""
    TXTNAMASP.Text = ""
    TXTALAMATSP.Text = ""
    TXTTELPONSP.Text = ""
    TXTKODESP.Focus()
End Sub
Sub DGV()
    SQLSTR = "SELECT * FROM SUPLIER"
    DASP = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTSP.Clear()
    DASP.Fill(DTSP)
    DGVSP.DataSource = DTSP
End Sub
Private Sub FORMSUPLIER_Activated(ByVal sender As Object, ByVal e As System.EventArgs)
    DGV()
    BLANKFORM()
End Sub

```

Sub BLANKFORM untuk mengosongkan Textbox input data Barang, sub DGV untuk menampilkan data Suplier ke DataGridView DGVSP. Saat form aktif akan menjalankan sub BLANKFORM dan sub DGV.

```

Private Sub TXTKODESP_Leave(ByVal sender As Object, ByVal e As System.EventArgs) Han
    On Error Resume Next
    SQLSTR = "SELECT * FROM SUPLIER WHERE KODESP='" & TXTKODESP.Text & "'"
    DASP = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTSP.Clear()
    DASP.Fill(DTSP)
    DGVSP.DataSource = DTSP
    If DTSP.Rows.Count > 0 Then
        TXTNAMASP.Text = DGVSP.Rows.Item(DGVSP.CurrentRow.Index).Cells(1).Value
        TXTALAMATSP.Text = DGVSP.Rows.Item(DGVSP.CurrentRow.Index).Cells(2).Value
        TXTTELPONSP.Text = DGVSP.Rows.Item(DGVSP.CurrentRow.Index).Cells(3).Value
    End If
    DGV()
End Sub

```

Saat kursor meninggalkan Textbox TXTKODESP akan dideteksi apakah Kode Suplier yang telah diketik sudah ada di tabel Suplier, jika Kode Suplier sudah ada maka datanya akan ditampilkan ke masing-masing Textbox.

```

Private Sub BTSIMPAN_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    SQLSTR = "SELECT * FROM SUPLIER WHERE KODESP='" & TXTKODESP.Text & "'"
    DASP = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTSP.Clear()
    DASP.Fill(DTSP)
    DGVSP.DataSource = DTSP
    If DTSP.Rows.Count > 0 Then
        SQLSTR = "UPDATE SUPLIER SET KODESP='" & TXTKODESP.Text & "',NAMASP='" & _
            TXTNAMASP.Text & "',ALAMATSP='" & TXTALAMATSP.Text & "',TELPONSP='" & _
            TXTTELPONSP.Text & "' WHERE KODESP='" & TXTKODESP.Text & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    Else
        SQLSTR = "INSERT INTO SUPLIER VALUES ('" & TXTKODESP.Text & "','" & _
            TXTNAMASP.Text & "','" & TXTALAMATSP.Text & "','" & TXTTELPONSP.Text & "')"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    BLANKFORM()
    DGV()
End Sub

```

Saat diklik tombol SIMPAN akan dideteksi apakah Kode Suplier yang akan disimpan sudah ada di tabel Suplier, jika Kode Suplier sudah ada maka data Suplier akan diupdate sedangkan jika belum ada maka data Suplier yang diinput akan ditambahkan sebagai data baru.

```

Private Sub BTHAPUS_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Ha
    Dim XKODESP As String
    Dim X As String
    XKODESP = TXTKODESP.Text
    X = MsgBox("DATA SUPLIER " & TXTNAMASP.Text & " JADI DIHAPUS !", MsgBoxStyle.YesNo)
    If X = vbYes Then
        SQLSTR = "DELETE FROM SUPLIER WHERE KODESP='" & XKODESP & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    BLANKFORM()
    DGV()
End Sub

```

Saat diklik tombol HAPUS akan ditampilkan konfirmasi apakah data Suplier yang sedang tampil akan dihapus, jika diklik tombol Yes maka data Suplier yang sedang tampil akan dihapus sedangkan jika diklik tombol No data Suplier tidak akan dihapus.

```

Private Sub BTBLANKFORM_Click(ByVal sender As System.Object, ByVal e As Sys
    BLANKFORM()
End Sub

```

Saat diklik tombol BLANKFORM akan menjalankan sub BLANKFORM untuk mengosongkan Textbox.


```

Private Sub BTCLOSE_Click(ByVal sender As System.Object, ByVal e As System
    KONEKSI.Close()
    Close()
End Sub

```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENGOPERASIKAN FORM DATA SUPLIER

1. Menambah data Suplier

Cara menambahkan data Suplier :

- a. Klik menu Debug => Start Debugging
 - b. Isikan data Kode Suplier, Nama Suplier, Alamat Suplier dan Nomor Telpon Suplier
 - c. Klik tombol SIMPAN untuk menyimpan data Suplier
- Tekan tombol Tab untuk memindahkan kursor ke text box berikutnya.

2. Mengedit data Suplier

Cara mengedit data Suplier :

- a. Klik menu Debug => Start Debugging
- b. Isikan Kode Suplier yang datanya mau diedit
- c. Tekan tombol Tab (data Suplier yang kodenya sesuai akan tampil)
- d. Lakukan pengeditan data
- e. Klik tombol SIMPAN untuk pengeditan data Suplier

3. Menghapus data Suplier

Cara menghapus data Suplier :

- a. Klik menu Debug => Start Debugging
- b. Isikan Kode Suplier yang datanya mau dihapus
- c. Tekan tombol Tab (data Suplier yang kodenya sesuai akan tampil)
- d. Klik tombol HAPUS
- e. Tampil konfirmasi data jadi dihapus atau tidak
- f. Klik tombol Yes jika data jadi dihapus atau klik tombol No jika data tidak jadi dihapus

3.3 TUTORIAL FORM SUPLIER



<https://youtu.be/HNZIMesVXFY>

3.4 EVALUASI

1. Atur lebar kolom di DataGridView, KODESP selebar 75, NAMASP selebar 150, ALAMATSP selebar 150 dan TELPONSP selebar 100
2. Tempatkan data Kode Suplier di tengah kolom

	KODESP	NAMASP	ALAMATSP	TELPONSP
	SP001	CV. COMPUTAMA	JL. MAJAPAHIT 25	088802250099
	SP002	PT. TEKNOCOMP	JL. GAJAHMADA	088802250066
▶*				

3.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB IV FORM TRANSAKSI PEMBELIAN

4.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari Form Transaksi Pembelian yaitu mahasiswa memahami struktur tabel transaksi Pembelian, dapat mendesain Form Transaksi Pembelian, memahami kode program yang digunakan dan cara mengoperasikan Form Transaksi Pembelian yang meliputi menambah item Barang pada Faktur Pembelian dan penyimpanan data transaksi Pembelian.

4.2 FORM TRANSAKSI PEMBELIAN

A. MENDESAIN FORM TRANSAKSI PEMBELIAN

Langkah-langkah mendesain Form Data Barang sebagai berikut :

1. Buka database DBINV.MDB pada folder PRINV\PRINV\BIN\DEBUG
2. Buat struktur tabel berikut :

Field Name	Data Type	
FAKTUR	Text	Field Size : 5
TANGGAL	Date/Time	
KODESP	Text	Field Size : 5

3. Simpan tabel dengan nama BELI
4. Buat struktur tabel berikut :

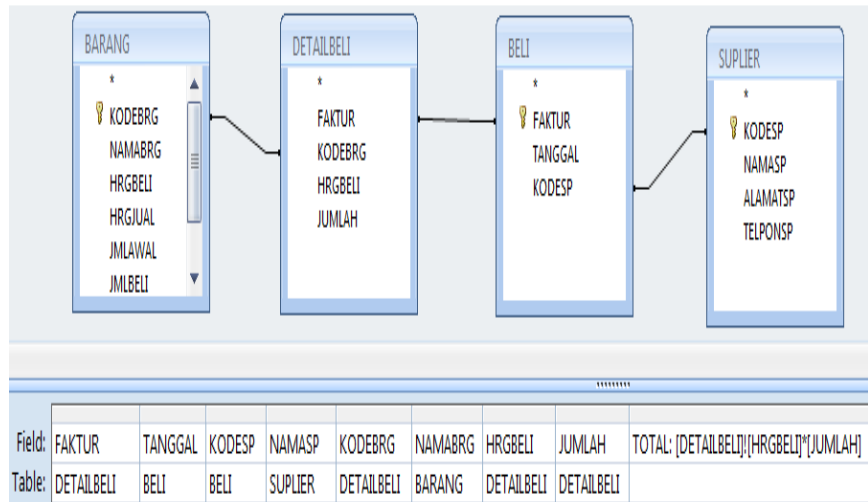
Field Name	Data Type	
FAKTUR	Text	Field Size : 5
KODEBRG	Text	Field Size : 6
HRGBELI	Number	
JUMLAH	Number	

5. Simpan tabel dengan nama DETAILBELI
6. Buat struktur tabel berikut :

Field Name	Data Type	
KODEBRG	Text	Field Size : 6
NAMABRG	Text	Field Size : 25
HRGBELI	Number	
JUMLAH	Number	
TOTAL	Number	

7. Simpan tabel dengan nama DETAILBELISEMU

8. Buat Desain Query berikut



9. Simpan Query dengan nama QUERYBELI

10. Tutup program MS Access

11. Double klik file PRINV.SLN untuk membuka project Inventory

12. Klik menu Project ⇒ Add Windows Form untuk menambahkan form baru

13. Ketik nama form FORMBELI

14. Klik tombol Add

15. Buat desain Form Transaksi Pembelian berikut

The screenshot shows a Microsoft Access form window titled "FORMBELI". The form has the following layout:

- Title: FORM TRANSAKSI PEMBELIAN
- Fields:
 - NO. FAKTUR: Text box
 - KODE SUPLIER: Dropdown menu
 - TANGGAL: Date picker (displaying 16/03/2021)
 - NAMA SUPLIER: Text box
- Buttons: TAMBAH, SIMPAN, BLANKFORM, CLOSE
- Below the buttons are several empty text boxes and a large empty rectangular area.

Gambar 4.1 Desain Form Transaksi Pembelian

Daftar Objek pada Form Transaksi Pembelian

Komponen	Properties	Isi
Label	Text	FORM TRANSAKSI PEMBELIAN
	Text	NOMOR FAKTUR
	Text	TANGGAL
	Text	KODE SUPLIER
	Text	NAMA SUPLIER
TextBox	Name	TXTFAKTUR
	Name	TXTNAMASP
DateTimePicker	Name	TXTTANGGAL
ComboBox	Name	TXTKODESP
Button	Name	BTTAMBAH
	Text	TAMBAH
	Name	BTSIMPAN
	Text	SIMPAN
	Name	BTBLANKFORM
	Text	BLANKFORM
	Name	BTCLOSE
	Text	CLOSE
ComboBox	Name	TXTKODEBRG
TextBox	Name	TXTNAMABRG
	Name	TXTHRGBELI
	Name	TXTJUMLAH
	Name	TXTTOTAL
DataGridView	Name	DGVDETAILBELI
	Name	DGVBELI

B. KODE PROGRAM FORM TRANSAKSI PEMBELIAN

Klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMBELI
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DADETAILBELI As New OleDb.OleDbDataAdapter
    Public DTDETAILBELI As New DataTable
    Public DAQUERYBELI As New OleDb.OleDbDataAdapter
    Public DTQUERYBELI As New DataTable
    Public DS As New DataSet
    Public DR As OleDbDataReader
    Public XFAKTUR As String
    Public XKODESP As String
    Public XKODEBRG As String
    Public XNAMABRG As String
    Public XHRGBELI As Integer
    Public XJUMLAH As Integer
    Public XTOTAL As Integer

```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DADETAILBELI dan DTDETAILBELI untuk menampilkan data detail Pembelian perfaktur, variabel DAQUERYBELI dan DTQUERYBELI untuk menampilkan seluruh data transaksi Pembelian. Variabel DR untuk membaca data tabel.

```
Private Sub FORMBELI_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()

    SQLSTR = "SELECT * FROM SUPLIER ORDER BY NAMASP"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    TXTKODESP.Items.Clear()
    While DR.Read
        TXTKODESP.Items.Add(DR("KODESP") & " " & DR("NAMASP"))
    End While
    DR.Close()

    SQLSTR = "SELECT * FROM BARANG ORDER BY NAMABRG"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    TXTKODEBRG.Items.Clear()
    While DR.Read
        TXTKODEBRG.Items.Add(DR("KODEBRG") & " " & DR("NAMABRG"))
    End While
    DR.Close()
End Sub
```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi. Combobox TXTKODESP diisi dengan data Kode Suplier dan Nama Suplier dari tabel tabel Suplier, combobox TXTKODEBRG diisi dengan data Kode Barang dan Nama Barang dari tabel Barang.

```
Sub KOSONG()
    TXTFAKTUR.Text = ""
    TXTKODESP.Text = ""
    TXTNAMASP.Text = ""
    TXTKODEBRG.Text = ""
    TXTNAMABRG.Text = ""
    TXTHRGBELI.Text = 0
    TXTJUMLAH.Text = 0
    TXTTOTAL.Text = 0
    SQLSTR = "DELETE FROM DETAILBELISEMU"
    DADETAILBELI = New OleDbDataAdapter(SQLSTR, KONEKSI)
    DTDETAILBELI.Clear()
    DADETAILBELI.Fill(DTDETAILBELI)
    DGVDetailedBELI.DataSource = DTDETAILBELI
    TXTFAKTUR.Focus()
End Sub
```

Sub KOSONG untuk mengosongkan Textbox input data Nomor Faktur, Kode Suplier, Nama Suplier, Kode Barang, Nama Barang, Harga Beli, Jumlah, Total dan mengosongkan data pada tabel DETAILBELISEMU yang digunakan untuk menyimpan sementara data Pembelian perFaktur.

```

Sub DGV()
SQLSTR = "SELECT * FROM DETAILBELISEMU"
DADETAILBELI = New OleDbDataAdapter(SQLSTR, KONEKSI)
DTDETAILBELI.Clear()
DADETAILBELI.Fill(DTDETAILBELI)
DGVDETAILBELI.DataSource = DTDETAILBELI
End Sub
Sub DGVBL()
SQLSTR = "SELECT * FROM QUERYBELI"
DAQQUERYBELI = New OleDbDataAdapter(SQLSTR, KONEKSI)
DTQUERYBELI.Clear()
DAQQUERYBELI.Fill(DTQUERYBELI)
DGVBELI.DataSource = DTQUERYBELI
End Sub
Private Sub FORMBELI_Activated(ByVal sender As Object, ByVal e
KOSONG()
DGV()
DGVBL()
End Sub

```

Sub DGV untuk menampilkan data pada tabel DetailBeliSemu ke Datagridview DgvDetailBeli, Sub DGVBL untuk menampilkan data pada query QueryBeli ke Datagridview DgvBeli. Saat form aktif akan menjalankan sub Kosong, Dgv dan DgvBeli.

```

Private Sub TXTKODESP_SelectedValueChanged(ByVal sender As System.Object, ByVal e
XKODESP = Microsoft.VisualBasic.Left(TXTKODESP.Text, 5)
SQLSTR = "SELECT KODESP,NAMASP FROM SUPPLIER WHERE KODESP='" & XKODESP & "'"
CMD = New OleDbCommand(SQLSTR, KONEKSI)
DR = CMD.ExecuteReader
While DR.Read
    TXTNAMASP.Text = DR("NAMASP")
End While
DR.Close()
TXTKODEBRG.Focus()
End Sub

```

Cari Kode Suplier yang diklik pada tabel Suplier dan tampilkan data Nama Suplier yang sesuai pada TextBox TXTNAMASP.

```

Private Sub TXTKODEBRG_SelectedValueChanged(ByVal sender As System.Object, ByVal e
XKODEBRG = Microsoft.VisualBasic.Left(TXTKODEBRG.Text, 6)
SQLSTR = "SELECT NAMABRG,HRGBELI FROM BARANG WHERE KODEBRG='" & XKODEBRG & "'"
CMD = New OleDbCommand(SQLSTR, KONEKSI)
DR = CMD.ExecuteReader
While DR.Read
    TXTNAMABRG.Text = DR("NAMABRG")
    TXTHRGBELI.Text = Format(DR("HRGBELI"), "#.##")
    XNAMABRG = DR("NAMABRG")
    XHRGBELI = DR("HRGBELI")
End While
DR.Close()
TXTJUMLAH.Focus()
End Sub

```

Cari Kode Barang yang diklik pada tabel Barang dan tampilkan data Nama Barang pada TextBox TXTNAMABRG dan Harga Beli pada TextBox TXTHRGBELI.

```

Private Sub TXTJUMLAH_TextChanged(ByVal sender As S
    If TXTJUMLAH.Text = "" Then TXTJUMLAH.Text = 0
    XTOTAL = XHRGBELI * TXTJUMLAH.Text
    TXTTOTAL.Text = Format(XTOTAL, "#, #")
End Sub

```

Saat diisikan data Jumlah Beli maka pada TextBox TXTTOTAL otomatis akan terisi data Harga Beli dikalikan Jumlah.

```

Private Sub BTTAMBAH_Click(ByVal sender As System.Object, ByVal e As
    XJUMLAH = TXTJUMLAH.Text
    SQLSTR = "INSERT INTO DETAILBELISEMU VALUES (' & XKODEBRG & _
        ',' & XNAMABRG & ',' & XHRGBELI & ',' & XJUMLAH & _
        ',' & XTOTAL & ')"
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()
    DGV()
    TXTKODEBRG.Text = ""
    TXTNAMABRG.Text = ""
    TXTHRGBELI.Text = 0
    TXTJUMLAH.Text = 0
    TXTTOTAL.Text = 0
    TXTKODEBRG.Focus()
End Sub

```

Setelah diklik tombol Tambah maka data Kode Barang, Nama Barang, Harga Beli, Jumlah dan Total akan ditambahkan ke tabel DetailBeliSemu dan TextBox input data akan dikosongkan untuk item barang berikutnya.

```

Private Sub BTSIMPAN_Click(ByVal sender As System.Object, ByVal e As System
    SQLSTR = "INSERT INTO BELI VALUES (' & TXTFAKTUR.Text & _
        ',' & Format(TXTTANGGAL.Value, "yyyy-MM-dd") & _
        ',' & XKODESP & ')"
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()

    SQLSTR = "INSERT INTO DETAILBELI ( KODEBRG, HRGBELI, JUMLAH )" & _
        " SELECT DETAILBELISEMU.KODEBRG, DETAILBELISEMU.HRGBELI, " & _
        " DETAILBELISEMU.JUMLAH FROM DETAILBELISEMU"
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()

    SQLSTR = "UPDATE DETAILBELI SET FAKTUR = ' & TXTFAKTUR.Text & _
        " WHERE DETAILBELI.FAKTUR Is Null"
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()

    KOSONG()
    DGVBL()
End Sub

```

Setelah diklik tombol Simpan, tambahkan data Nomor Faktur, Tanggal Pembelian dan Kode Suplier ke tabel Beli. Tambahkan data Kode Barang, Harga Beli dan Jumlah pada tabel DetailBeliSemu ke tabel DetailBeli. Update data Faktur yang masih kosong pada tabel DetailBeli dengan data Faktur pada TextBox TXTFAKTUR.


```
Private Sub BTCLOSE_Click(ByVal sender As System.Object,
    KONEKSI.Close()
    Close()
End Sub
```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENGOPERASIKAN FORM PEMBELIAN

Cara mengisi data transaksi Pembelian :

- a. Isikan data Nomor Faktur
- b. Isikan data Tanggal Pembelian
- c. Klik droplist pada Kode Suplier
- d. Klik salah satu Kode Suplier, Nama Suplier otomatis akan tampil
- e. Klik droplist pada Kode Barang
- f. Klik salah satu Kode Barang, Nama Barang dan Harga Barang otomatis akan tampil
- g. Isikan data Jumlah, data Total otomatis tampil
- h. Klik tombol Tambah untuk menambahkan item data Barang ke DetailBeliSemu
- i. Ulangi langkah e / h untuk menambah item Barang, satu Nomor Faktur bisa berisi beberapa item Barang
- j. Klik tombol Simpan untuk menyimpan Faktur Pembelian ke tabel Beli

4.3 TUTORIAL FORM PEMBELIAN



<https://youtu.be/9toA49qxp5Q>

4.4 EVALUASI

Tambahkan tombol HAPUS untuk menghapus item Barang yang telah ditambahkan di DataGridView DetailBeliSemu

FORM TRANSAKSI PEMBELIAN

NO. FAKTUR KODE SUPLIER

TANGGAL 29/03/2021 NAMA SUPLIER

TAMBAH HAPUS SIMPAN BLANKFORM CLOSE

4.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB V FORM DATA CUSTOMER

5.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari Form Data Customer yaitu mahasiswa memahai struktur tabel Customer, dapat mendesain Form Data Customer, memahami kode program yang digunakan dan cara mengoperasikan Form Data Customer yang meliputi manambah data baru, mengedit data dan menghapus data.

5.2 FORM DATA CUSTOMER

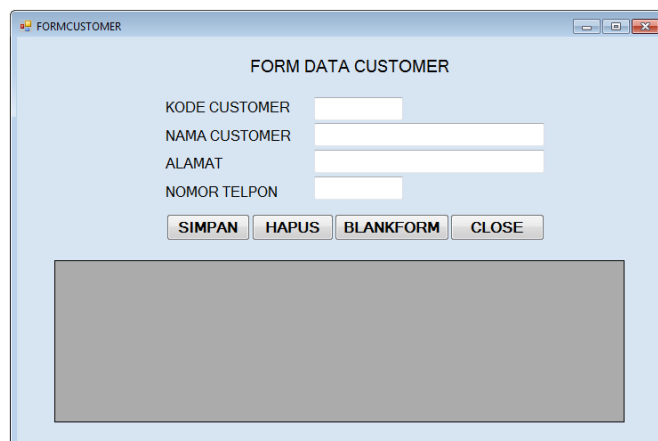
A. MENDESAIN FORM DATA CUSTOMER

Langkah-langkah mendesain Form Data Customer sebagai berikut :

1. Buka database DBINV.MDB pada folder PRINV\PRINV\BIN\DEBUG
2. Buat struktur tabel berikut :

Field Name	Data Type	
KODECS	Text	Field Size : 5
NAMACS	Text	Field Size : 25
ALAMATCS	Text	Field Size : 25
TELPONCS	Text	Field Size : 12

3. Simpan tabel dengan nama CUSTOMER
4. Tutup program MS Access
5. Double klik file PRINV.SLN untuk membuka project Inventory
6. Klik menu Project ⇒ Add Windows Form
7. Ketik nama form FORMCUSTOMER
8. Klik tombol Add, Buat desain Form Customer berikut



Gambar 5.1 Desain Form Data Customer

Daftar Objek pada Form Data Customer

Komponen	Properties	Isi
Label	Text	FORM DATA CUSTOMER
	Text	KODE CUSTOMER
	Text	NAMA CUSTOMER
	Text	ALAMAT
	Text	NOMORTELPON
TextBox	Name	TXTKODECS
	Name	TXTNAMACS
	Name	TXTALAMATCS
	Name	TXTTELPCONS
Button	Name	BTSIMPAN
	Text	SIMPAN
	Name	BTHAPUS
	Text	HAPUS
	Name	BTBLANKFORM
	Text	BLANKFORM
	Name	BTCLOSE
	Text	CLOSE
DataGridView	Name	DGVCS

B. KODE PROGRAM FORM DATA CUSTOMER

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMCUSTOMER
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DACS As New OleDb.OleDbDataAdapter
    Public DTCS As New DataTable
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DACS untuk penghubung data tabel, variabel DTCS menyimpan data hasil perintah query.

```

Private Sub FORMSUPLIER_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()
End Sub

```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi.

```

Sub KOSONG()
    TXTKODECS.Text = ""
    TXTNAMACS.Text = ""
    TXTALAMATCS.Text = ""
    TXTTELPONCS.Text = ""
    TXTKODECS.Focus()
End Sub
Sub DGV()
    SQLSTR = "SELECT * FROM CUSTOMER"
    DACS = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTCS.Clear()
    DACS.Fill(DTCS)
    DGVCS.DataSource = DTCS
End Sub
Private Sub FORMCUSTOMER_Activated(ByVal sender As Object, ByVal e As System.EventArgs)
    KOSONG()
    DGV()
End Sub

```

Sub KOSONG untuk mengosongkan Textbox input data Customer, sub DGV untuk menampilkan data Customer ke DataGridView DGVCS. Saat form aktif akan menjalankan sub KOSONG dan sub DGV.

```

Private Sub TXTKODECS_Leave(ByVal sender As Object, ByVal e As System.EventArgs)
    On Error Resume Next
    SQLSTR = "SELECT * FROM CUSTOMER WHERE KODECS='" & TXTKODECS.Text & "'"
    DACS = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTCS.Clear()
    DACS.Fill(DTCS)
    DGVCS.DataSource = DTCS
    If DTCS.Rows.Count > 0 Then
        TXTNAMACS.Text = DGVCS.Rows.Item(DGVCS.CurrentRow.Index).Cells(1).Value
        TXTALAMATCS.Text = DGVCS.Rows.Item(DGVCS.CurrentRow.Index).Cells(2).Value
        TXTTELPONCS.Text = DGVCS.Rows.Item(DGVCS.CurrentRow.Index).Cells(3).Value
    End If
    DGV()
End Sub

```

Saat kursor meninggalkan Textbox TXTKODECS akan dideteksi apakah Kode Customer yang telah diketik sudah ada di tabel Customer, jika Kode Customer sudah ada maka datanya akan ditampilkan ke masing-masing Textbox.

```

Private Sub BTSIMPAN_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    SQLSTR = "SELECT * FROM CUSTOMER WHERE KODECS='" & TXTKODECS.Text & "'"
    DACS = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DTCS.Clear()
    DACS.Fill(DTCS)
    DGWCS.DataSource = DTCS
    If DTCS.Rows.Count > 0 Then
        SQLSTR = "UPDATE CUSTOMER SET KODECS='" & TXTKODECS.Text & _
            "' ,NAMACS='" & TXTNAMACS.Text & "' ,ALAMATCS='" & _
            TXTALAMATCS.Text & "' ,TELPONCS='" & TXTTELPONCS.Text & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    Else
        SQLSTR = "INSERT INTO CUSTOMER VALUES ('" & TXTKODECS.Text & "','" & _
            TXTNAMACS.Text & "','" & TXTALAMATCS.Text & _
            "','" & TXTTELPONCS.Text & "')"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    KOSONG()
    DGV()
End Sub

```

Saat diklik tombol SIMPAN akan dideteksi apakah Kode Customer yang akan disimpan sudah ada di tabel Customer, jika Kode Customer sudah ada maka data Customer akan diupdate sedangkan jika belum ada maka data Customer yang diinput akan ditambahkan sebagai data baru.

```

Private Sub ETHAPUS_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Dim X As String
    X = MsgBox("DATA CUSTOMER JADI DIHAPUS !", MsgBoxStyle.YesNo)
    If X = vbYes Then
        SQLSTR = "DELETE FROM CUSTOMER WHERE KODECS='" & TXTKODECS.Text & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    KOSONG()
    DGV()
End Sub

```

Saat diklik tombol HAPUS akan ditampilkan konfirmasi apakah data Customer yang sedang tampil akan dihapus, jika diklik tombol Yes maka data Customer yang sedang tampil akan dihapus sedangkan jika diklik tombol No data Customer tidak akan dihapus.

```

Private Sub BTBLANKFORM_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    KOSONG()
End Sub

```

Saat diklik tombol BLANKFORM akan menjalankan sub KOSONG untuk mengosongkan Textbox.

```

Private Sub BTCLOSE_Click(ByVal sender As System.Object, ByVal e As System
    KONEKSI.Close()
    Close()
End Sub

```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENGOPERASIKAN FORM DATA CUSTOMER

1. Menambah data Customer

Cara menambahkan data Suplier :

- a. Klik menu Debug => Start Debugging
 - b. Isikan data Kode Customer, Nama Customer, Alamat Customer dan Nomor Telpon Customer
 - c. Klik tombol SIMPAN untuk menyimpan data Customer
- Tekan tombol Tab untuk memindahkan kursor ke text box berikutnya.

2. Mengedit data Customer

Cara mengedit data Customer :

- a. Klik menu Debug => Start Debugging
- b. Isikan Kode Customer yang datanya mau diedit
- c. Tekan tombol Tab (data Customer yang kodenya sesuai akan tampil)
- d. Lakukan pengeditan data
- e. Klik tombol SIMPAN untuk pengeditan data Customer

3. Menghapus data Customer

Cara menghapus data Customer :

- a. Klik menu Debug => Start Debugging
- b. Isikan Kode Customer yang datanya mau dihapus
- c. Tekan tombol Tab (data Suplier yang kodenya sesuai akan tampil)
- d. Klik tombol HAPUS
- e. Tampil konfirmasi data jadi dihapus atau tidak
- f. Klik tombol Yes jika data jadi dihapus atau klik tombol No jika data tidak jadi dihapus

5.3 TUTORIAL FORM CUSTOMER



<https://youtu.be/YXmm-MZC0u8>

5.4 EVALUASI

Buat kode program untuk mendeteksi saat input data Nomor Telpon jika diinput data selain angka tampilkan pesan “DATA NOMOR TELPON HANYA BISA DIISI ANGKA”

5.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*”, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB VI FORM TRANSAKSI PENJUALAN

6.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari Form Transaksi Penjualan yaitu mahasiswa memahami struktur tabel transaksi Penjualan, dapat mendesain Form Transaksi Penjualan, memahami kode program yang digunakan dan cara mengoperasikan Form Transaksi Penjualan yang meliputi menambahkan item Barang pada Nota Penjualan dan penyimpanan data Penjualan.

6.2 FORM TRANSAKSI PENJUALAN

A. MENDESAIN FORM TRANSAKSI PENJUALAN

Langkah-langkah mendesain Form Penjualan sebagai berikut :

1. Buka database DBINV.MDB pada folder PRINV\PRINV\BIN\DEBUG
2. Buat struktur tabel berikut :

Field Name	Data Type	
NOTA	Text	Field Size : 5
TANGGAL	Date/Time	
KODECS	Text	Field Size : 5

3. Simpan tabel dengan nama JUAL
4. Buat struktur tabel berikut :

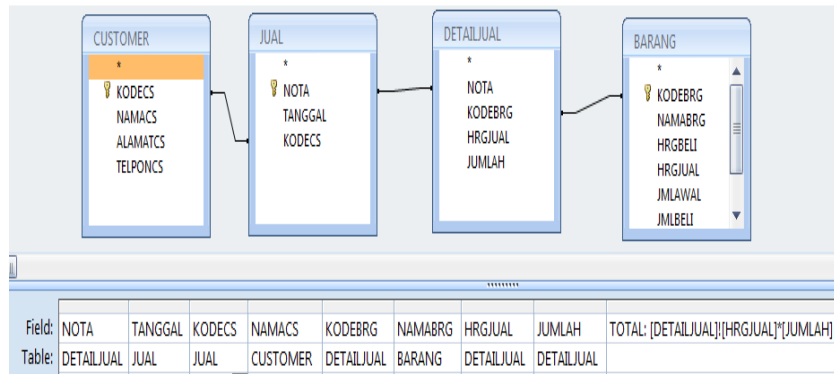
Field Name	Data Type	
NOTA	Text	Field Size : 5
KODEBRG	Text	Field Size : 6
HRGJUAL	Number	
JUMLAH	Number	

5. Simpan tabel dengan nama DETAILJUAL
6. Buat struktur tabel berikut :

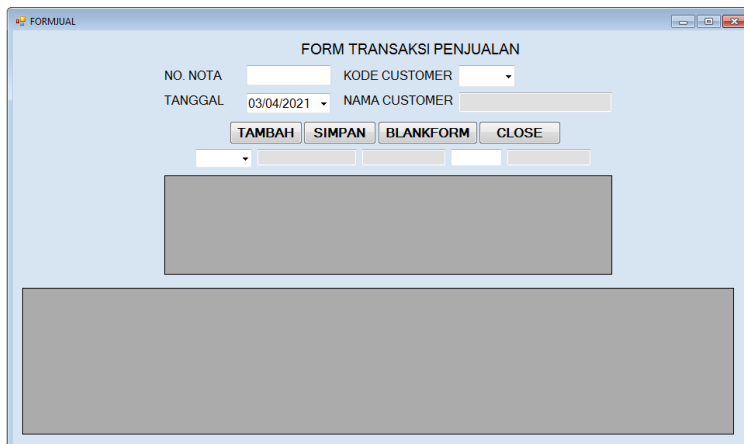
Field Name	Data Type	
KODEBRG	Text	Field Size : 6
NAMABRG	Text	Field Size : 25
HRGJUAL	Number	
JUMLAH	Number	
TOTAL	Number	

7. Simpan tabel dengan nama DETAILJUALSEMU

8. Buat Desain Query berikut



9. Simpan Query dengan nama QUERYJUAL
10. Tutup program MS Access
11. Double klik file PRINV.SLN untuk membuka project Inventory
12. Klik menu Project ⇒ Add Windows Form untuk menambahkan form baru
13. Ketik nama form FORMJUAL
14. Klik tombol Add
15. Buat desain Form Transaksi Penjualan berikut



Gambar 6.1 Desain Form Transaksi Penjualan

Daftar Objek pada Form Transaksi Penjualan

Komponen	Properties	Isi
Label	Text	FORM TRANSAKSI PENJUALAN
	Text	NOMOR NOTA

	Text	TANGGAL
	Text	KODE CUSTOMER
	Text	NAMA CUSTOMER
TextBox	Name	TXTNOTA
	Name	TXTNAMACS
DateTimePicker	Name	TXTTANGGAL
ComboBox	Name	TXTKODECS
Button	Name	BTTAMBAH
	Text	TAMBAH
	Name	BTSIMPAN
	Text	SIMPAN
	Name	BTBLANKFORM
	Text	BLANKFORM
	Name	BTCLOSE
	Text	CLOSE
ComboBox	Name	TXTKODEBRG
TextBox	Name	TXTNAMABRG
	Name	TXTHRGBELI
	Name	TXTJUMLAH
	Name	TXTTOTAL
DataGridView	Name	DGVDETAILJUAL
	Name	DGVJUAL

B. KODE PROGRAM FORM TRANSAKSI PENJUALAN

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMJUAL
    Public koneksi As String
    Public sqlstr As String
    Public conn As New OleDb.OleDbConnection
    Public CMD As New OleDb.OleDbCommand
    Public DADETAILJUAL As New OleDb.OleDbDataAdapter
    Public DTDETAILJUAL As New DataTable
    Public DAQUERYJUAL As New OleDb.OleDbDataAdapter
    Public DTQUERYJUAL As New DataTable
    Public DS As New DataSet
    Public DR As OleDbDataReader
    Public XNOTA As String
    Public XKODECS As String
    Public XKODEBRG As String
    Public XNAMABRG As String
    Public XHRGJUAL As Integer
    Public XJUMLAH As Integer
    Public XTOTAL As Integer

```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSI berisi string untuk koneksi ke database, variabel CONN untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DADETAILJUAL dan DTDETAILJUAL untuk menampilkan data detail Penjualan pernota, variabel DAQUERYJUAL dan DTQUERYJUAL untuk menampilkan seluruh data transaksi Penjualan. Variabel DR untuk membaca data dari suatu tabel.

```
Private Sub FORMJUAL_Load(ByVal sender As System.Object, ByVal e As System
koneksi = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
conn = New OleDb.OleDbConnection(koneksi)
conn.Open()

sqlstr = "SELECT * FROM CUSTOMER ORDER BY NAMACS"
CMD = New OleDbCommand(sqlstr, conn)
DR = CMD.ExecuteReader
TXTKODECS.Items.Clear()
While DR.Read
    TXTKODECS.Items.Add(DR("KODECS") & " " & DR("NAMACS"))
End While
DR.Close()

sqlstr = "SELECT * FROM BARANG ORDER BY NAMABRG"
CMD = New OleDbCommand(sqlstr, conn)
DR = CMD.ExecuteReader
TXTKODEBRG.Items.Clear()
While DR.Read
    TXTKODEBRG.Items.Add(DR("KODEBRG") & " " & DR("NAMABRG"))
End While
DR.Close()
End Sub
```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi. Combobox TXTKODECS diisi dengan data Kode Customer dan Nama Customer dari tabel Customer, combobox TXTKODEBRG diisi dengan data Kode Barang dan Nama Barang dari tabel Barang.

```
Sub KOSONG()
    TXTNOTA.Text = ""
    TXTKODECS.Text = ""
    TXTNAMACS.Text = ""
    TXTKODEBRG.Text = ""
    TXTNAMABRG.Text = ""
    TXTHRGJUAL.Text = 0
    TXTJUMLAH.Text = 0
    TXTTOTAL.Text = 0
    sqlstr = "DELETE FROM DETAILJUALSEMU"
    DADETAILJUAL = New OleDbDataAdapter(sqlstr, conn)
    DTDETAILJUAL.Clear()
    DADETAILJUAL.Fill(DTDETAILJUAL)
    DGVDetailedJUAL.DataSource = DTDETAILJUAL
    TXTNOTA.Focus()
End Sub
```

Sub KOSONG untuk mengosongkan Textbox input data Nomor Nota, Kode Customer, Nama Customer, Kode Barang, Nama Barang, Harga Jual, Jumlah, Total dan mengosongkan data pada tabel DETAILJUALSEMU yang digunakan untuk menyimpan sementara data Penjualan perNota.

```

Sub DGV()
    sqlstr = "SELECT * FROM DETAILJUALSEMU"
    DADETAILJUAL = New OleDbDataAdapter(sqlstr, conn)
    DTDETAILJUAL.Clear()
    DADETAILJUAL.Fill(DTDETAILJUAL)
    DGVDETAILJUAL.DataSource = DTDETAILJUAL
End Sub
Sub DGVJL()
    sqlstr = "SELECT * FROM QUERYJUAL"
    DAQUERYJUAL = New OleDbDataAdapter(sqlstr, conn)
    DTQUERYJUAL.Clear()
    DAQUERYJUAL.Fill(DTQUERYJUAL)
    DGVJUAL.DataSource = DTQUERYJUAL
End Sub
Private Sub FORMJUAL_Activated(ByVal sender As Object)
    KOSONG()
    DGV()
    DGVJL()
End Sub

```

Sub DGV untuk menampilkan data pada tabel DetailJualSemu ke Datagridview DgvDetailJual, Sub DGVJL untuk menampilkan data pada query QueryJual ke Datagridview DgvJual. Saat form aktif akan menjalankan sub Kosong, Dgv dan DgvBeli.

```

Private Sub TXTKODECS_SelectedValueChanged(ByVal sender As System.Object, ByVal e As EventArgs)
    XKODECS = Microsoft.VisualBasic.Left(TXTKODECS.Text, 5)
    sqlstr = "SELECT KODECS,NAMACS FROM CUSTOMER WHERE KODECS='" & XKODECS & "'"
    CMD = New OleDbCommand(sqlstr, conn)
    DR = CMD.ExecuteReader
    While DR.Read
        TXTNAMACS.Text = DR("NAMACS")
    End While
    DR.Close()
    TXTKODEBERG.Focus()
End Sub

```

Cari Kode Customer yang diklik pada tabel Customer dan tampilkan data Nama Customer yang sesuai pada TextBox TXTNAMACS.

```

Private Sub TXTKODEBRG_SelectedValueChanged(ByVal sender As System.Object, ByVal e
    XKODEBRG = Microsoft.VisualBasic.Left(TXTKODEBRG.Text, 6)
    sqlstr = "SELECT NAMABRG,HRGJUAL FROM BARANG WHERE KODEBRG='" & XKODEBRG & "'"
    CMD = New OleDbCommand(sqlstr, conn)
    DR = CMD.ExecuteReader
    While DR.Read
        TXTNAMABRG.Text = DR("NAMABRG")
        TXTHRGJUAL.Text = Format(DR("HRGJUAL"), "#, #")
        XNAMABRG = DR("NAMABRG")
        XHRGJUAL = DR("HRGJUAL")
    End While
    DR.Close()
    TXTJUMLAH.Focus()
End Sub

```

Cari Kode Barang yang diklik pada tabel Barang dan tampilkan data Nama Barang pada TextBox TXTNAMABRG dan Harga Jual pada TextBox TXTHRGJUAL.

```

Private Sub TXTJUMLAH_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)
    If TXTJUMLAH.Text = "" Then TXTJUMLAH.Text = 0
    XTOTAL = XHRGJUAL * TXTJUMLAH.Text
    TXTTOTAL.Text = Format(XTOTAL, "#, #")
End Sub

```

Saat diisikan data Jumlah Jual maka pada TextBox TXTTOTAL otomatis akan terisi data Harga Jual dikalikan Jumlah.

```

Private Sub BTTAMBAH_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    XJUMLAH = TXTJUMLAH.Text
    sqlstr = "INSERT INTO DETAILJUALSEMU VALUES ('" & XKODEBRG & "','" & XNAMABRG & "','" & XHRGJUAL & "','" & XJUMLAH & "','" & XTOTAL & "')"
    CMD = New OleDb.OleDbCommand(sqlstr, conn)
    CMD.ExecuteNonQuery()
    DGV.Refresh()
    TXTKODEBRG.Text = ""
    TXTNAMABRG.Text = ""
    TXTHRGJUAL.Text = 0
    TXTJUMLAH.Text = 0
    TXTTOTAL.Text = 0
    TXTKODEBRG.Focus()
End Sub

```

Setelah diklik tombol Tambah maka data Kode Barang, Nama Barang, Harga Jual, Jumlah dan Total akan ditambahkan ke tabel DetailJualSemu dan TextBox input data akan dikosongkan untuk item barang berikutnya.

```

Private Sub BTSIMPAN_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BTSIMPAN.
    sqlstr = "INSERT INTO JUAL VALUES ('" & TXTNOTA.Text & "','" & _
        Format(TXTTANGGAL.Value, "yyyy-MM-dd") & "','" & XKODECS & "'"")"
    CMD = New OleDb.OleDbCommand(sqlstr, conn)
    CMD.ExecuteNonQuery()

    sqlstr = "INSERT INTO DETAILJUAL ( KODEBRG, HRGJUAL, JUMLAH ) SELECT " & _
        "DETAILJUALSEM.U.KODEBRG, DETAILJUALSEM.U.HRGJUAL, DETAILJUALSEM.U.JUMLAH FROM DETAILJUALSEM.U"
    CMD = New OleDb.OleDbCommand(sqlstr, conn)
    CMD.ExecuteNonQuery()

    sqlstr = "UPDATE DETAILJUAL SET NOTA =" & TXTNOTA.Text & " WHERE DETAILJUAL.NOTA Is Null"
    CMD = New OleDb.OleDbCommand(sqlstr, conn)
    CMD.ExecuteNonQuery()

    KOSONG()
    DGWJL()
End Sub

```

Setelah diklik tombol Simpan, tambahkan data Nomor Nota, Tanggal Penjualan dan Kode Customer ke tabel Jual. Tambahkan data Kode Barang, Harga Jual dan Jumlah pada tabel DetailJualSemu ke tabel DetailJual. Update data Nota yang masih kosong pada tabel DetailJual dengan data Nota pada TextBox TXTNOTA.

```

Private Sub BTCLOSE_Click(ByVal sender As System.Object,
    KONEKSI.Close()
    Close()
End Sub

```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENGOPERASIKAN FORM PENJUALAN

Cara mengisi data transaksi Penjualan :

- a. Isikan data Nomor Nota
- b. Isikan data Tanggal Penjualan
- c. Klik droplist pada Kode Customer
- d. Klik salah satu Kode Customer, Nama Customer otomatis akan tampil
- e. Klik droplist pada Kode Barang
- f. Klik salah satu Kode Barang, Nama Barang dan Harga Barang otomatis akan tampil
- g. Isikan data Jumlah, data Total otomatis tampil
- h. Klik tombol Tambah untuk menambahkan item data Barang
- i. Ulangi langkah e s/d h untuk menambah item Barang
- j. Klik tombol Simpan untuk menyimpan data Nota Penjualan ke tabel Jual

6.5 TUTORIAL FORM PENJUALAN



<https://youtu.be/szwC0-Lsw6s>

6.6 EVALUASI

Buat kode program untuk menghitung Total Penjualan PerNota.

FORM TRANSAKSI PENJUALAN

NO. NOTA KODE CUSTOMER

TANGGAL NAMA CUSTOMER

	KODEBRG	NAMABRG	HRGJUAL	JUMLAH	TOTAL
	KYB001	KEYBOARD	100000	8	800000
	MOU001	MOUSE	55000	6	330000
▶▶					

TOTAL

6.7 REFERENSI

I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.

Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.

Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.

WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB VII

FORM MENU UTAMA

7.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari Form Menu Utama yaitu mahasiswa bisa membuat Form Menu untuk menjalankan Form Barang, Form Suplier, Form Pembelian, Form Customer dan Form Penjualan dari satu form yaitu Form Menu Utama.

7.2 FORM MENU UTAMA

A. MENDESAIN FORM MENU UTAMA

Langkah-langkah mendesain Form Penjualan sebagai berikut :

1. Double klik file PRINV.SLN untuk membuka project Inventory
2. Klik menu Project ⇒ Add Windows Form untuk menambahkan form baru
3. Ketik nama form MENUUTAMA
4. Klik tombol Add
5. Klik komponen MenuStrip pada ToolBox
6. Klik area kosong pada form
7. Buat desain Form Menu berikut



Gambar 7.1 Desain Form Menu Utama

B. KODE PROGRAM FORM MENU

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```

Private Sub DATABARANGToolStripMenuItem_Click(
    FORMBARANG.ShowDialog()
End Sub
Private Sub DATASUPLIERToolStripMenuItem_Click
    FORMSUPLIER.ShowDialog()
End Sub
Private Sub TRANSAKSIPEMBELIANToolStripMenuIte
    FORMBELI.ShowDialog()
End Sub
Private Sub DATACUSTOMERToolStripMenuItem_Clic
    FORMCUSTOMER.ShowDialog()
End Sub
Private Sub TRANSAKSIPENJUALANToolStripMenuIte
    FORMJUAL.ShowDialog()
End Sub
Private Sub KELUARDARIPROGRAMToolStripMenuItem
    Close()
End Sub

```

Perintah FORMBARANG.ShowDialog() untuk menampilkan Form Barang saat diklik menu Master => Data Barang, FORMSUPLIER.ShowDialog() untuk menampilkan Form Suplier saat diklik menu Master => Data Suplier, FORMBELI.ShowDialog() untuk menampilkan Form Beli saat diklik menu Transaksi => Transaksi Pembelian, FORMCUSTOMER.ShowDialog() untuk menampilkan Form Customer saat diklik menu Master => Data Customer, FORMJUAL.ShowDialog() untuk menampilkan Form Jual saat diklik menu Transaksi => Transaksi Penjualan, Close() untuk menutup Form saat diklik menu Selesai => Keluar Dari Program.

C. MENGOPERASIKAN FORM MENU

Cara mengoperasikan Form Menu :

- a. Klik menu Project => PRINV Properties
- b. Klik Droplist pada Startup Form
- c. Klik MENUUTAMA
- d. Klik menu Debug => Start Debugging
- e. Klik menu Master => Data Barang akan tampil Form Barang
- f. Pada Form Barang klik tombol Close untuk kembali ke Form Menu
- g. Klik menu Master => Data Suplier akan tampil Form Suplier
- h. Pada Form Suplier klik tombol Close untuk kembali ke Form Menu
- i. Klik menu Master => Data Customer akan tampil Form Customer
- j. Pada Form Customer klik tombol Close untuk ke Form Menu

- k. Klik menu Transaksi => Transaksi Pembelian untuk menampilkan Form Beli
- l. Pada Form Beli klik tombol Close untuk kembali ke Form Menu
- m. Klik menu Transaksi => Transaksi Penjualan untuk menampilkan Form Jual
- n. Pada Form Jual klik tombol Close untuk kembali ke Form Menu
- o. Klik menu Selesai => Keluar Dari Program untuk keluar

7.3 TUTORIAL FORM MENU UTAMA



<https://youtu.be/ZBfzEXIA-eo>

7.4 EVALUASI

Buat kode program pada Form Barang untuk menghitung Jumlah Beli sesuai dengan total jumlah masing-masing Barang dari transaksi Pembelian, Jumlah Jual sesuai dengan total jumlah masing-masing Barang dari transaksi Penjualan dan Jumlah Stok dengan Jumlah Awal ditambah Jumlah Beli dikurangi Jumlah Jual.

7.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB VIII

MENCETAK DATA MASTER

8.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari Mencetak Data Master yaitu mahasiswa bisa memahami kode program untuk mendesain dan mencetak data Barang, data Suplier dan data Customer menggunakan komponen PrintDocument.

8.2 MENCETAK DATA BARANG

Langkah-langkah mencetak data Barang sebagai berikut :

- a. Double klik file PRINV.SLN untuk membuka project Inventory
- b. Double klik form MENUUTAMA
- c. Klik komponen PrintDocument pada ToolBox
- d. Klik area kosong pada form
- e. Klik propertie Name
- f. Ketik PDBARANG
- g. Ulangi langkah c s/d f untuk menambahkan objek PrintDocument dengan nama PDSUPLIER dan PDCUSTOMER
- h. Klik menu View => Code
- i. Ketik kode program berikut

```
Public Class MENUUTAMA
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Dim DR As OleDbDataReader
    Dim BARIS As Integer
    Dim NOMOR As Integer
    Dim FONTJUDUL As Font = New Font("CALIBRI", 14, FontStyle.Bold, GraphicsUnit.Point)
    Dim FONTDATA As Font = New Font("CALIBRI", 12, FontStyle.Regular, GraphicsUnit.Point)
    Dim KIRI As StringFormat = New StringFormat
    Dim TENGAH As StringFormat = New StringFormat
    Dim KANAN As StringFormat = New StringFormat
```

Variabel BARIS untuk menentukan nilai baris data, variabel NOMOR untuk Nomor Urut data, variabel FONTJUDUL untuk menentukan ukuran font judul lapotan, variabel FONTDATA untuk menentukan ukuran font data, variabel KIRI untuk perataan kiri, variabel TENGAH untuk perataan tengah dan variabel KANAN untuk perataan kanan.

```

Private Sub MENUUTAMA_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()

```

```
End Sub
```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi

```

Private Sub PDBARANG_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.PrintPageEvent)
    KIRI.Alignment = StringAlignment.Near
    TENGAH.Alignment = StringAlignment.Center
    KANAN.Alignment = StringAlignment.Far

```

```

    e.Graphics.DrawString("LAPORAN DATA BARANG", FONTJUDUL, Brushes.Blue, 450, 25, TENGAH)
    e.Graphics.DrawLine(Pens.Black, 30, 70, 805, 70)
    e.Graphics.DrawString("NOMOR", FONTDATA, Brushes.Black, 100, 75, KANAN)
    e.Graphics.DrawString("KODEBERG", FONTDATA, Brushes.Black, 125, 75, KIRI)
    e.Graphics.DrawString("NAMA BARANG", FONTDATA, Brushes.Black, 200, 75, KIRI)
    e.Graphics.DrawString("HRG BELI", FONTDATA, Brushes.Black, 400, 75, KANAN)
    e.Graphics.DrawString("HRG JUAL", FONTDATA, Brushes.Black, 500, 75, KANAN)
    e.Graphics.DrawString("JML AWAL", FONTDATA, Brushes.Black, 575, 75, KANAN)
    e.Graphics.DrawString("JML BELI", FONTDATA, Brushes.Black, 650, 75, KANAN)
    e.Graphics.DrawString("JML JUAL", FONTDATA, Brushes.Black, 725, 75, KANAN)
    e.Graphics.DrawString("JML STOK", FONTDATA, Brushes.Black, 800, 75, KANAN)
    e.Graphics.DrawLine(Pens.Black, 30, 100, 805, 100)

```

```
BARIS = 105
```

```
NOMOR = 1
```

```
SQLSTR = "SELECT * FROM BARANG"
```

```
CMD = New OleDbCommand(SQLSTR, KONEKSI)
```

```
DR = CMD.ExecuteReader
```

```
While DR.Read
```

```

    e.Graphics.DrawString(NOMOR, FONTDATA, Brushes.Black, 100, BARIS, KANAN)
    e.Graphics.DrawString(DR("KODEBERG"), FONTDATA, Brushes.Black, 125, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMABRG"), FONTDATA, Brushes.Black, 200, BARIS, KIRI)
    e.Graphics.DrawString(Format(DR("HRGBELI"), "#,#"), FONTDATA, Brushes.Black, 400, BARIS, KANAN)
    e.Graphics.DrawString(Format(DR("HRGJUAL"), "#,#"), FONTDATA, Brushes.Black, 500, BARIS, KANAN)
    e.Graphics.DrawString(DR("JMLAWAL"), FONTDATA, Brushes.Black, 575, BARIS, KANAN)
    e.Graphics.DrawString(DR("JMLBELI"), FONTDATA, Brushes.Black, 650, BARIS, KANAN)
    e.Graphics.DrawString(DR("JMLJUAL"), FONTDATA, Brushes.Black, 725, BARIS, KANAN)
    e.Graphics.DrawString(DR("JMLSTOK"), FONTDATA, Brushes.Black, 800, BARIS, KANAN)
    NOMOR = NOMOR + 1
    BARIS = BARIS + 20

```

```
End While
```

```
e.Graphics.DrawLine(Pens.Black, 30, BARIS + 5, 805, BARIS + 5)
```

```
DR.Close()
```

```
End Sub
```

Perintah `e.Graphics.DrawString("LAPORAN DATA BARANG", FONTJUDUL, Brushes.Blue, 450, 25, TENGAH)` untuk mencetak string LAPORAN DATA BARANG menggunakan jenis FontJudul, warna Biru, kolom 450, baris 25, perataan Tengah. Perintah `e.Graphics.DrawLine(Pens.Black, 30, 70, 805, 70)` untuk mencetak garis mendatar warna Hitam, mulai Kolom 30 Baris 70 sampai Kolom 805 Baris 70. Variabel BARIS untuk menentukan awal Baris data pertama, variabel NOMOR untuk awal Nomor data pertama. Sebelum mencetak data berikutnya, nilai variabel BARIS akan bertambah 20 dan nilai variabel NOMOR bertambah 1.

```
Private Sub CETAKBARANGToolStripMenuItem_Click(By
    PDBARANG.Print()
End Sub
```

Perintah `PDBARANG.Print` untuk mencetak data yang telah didesain pada Private Sub `PDBARANG_PrintPage`.

- j. Klik menu Debug => Start Debugging
- k. Klik menu Cetak => Cetak Barang

LAPORAN DATA BARANG

NOMOR	KODEBRG	NAMA BARANG	HRG BELI	HRG JUAL	JML AWAL	JML BELI	JML JUAL	JML STOK
1	KYB001	KEYBOARD	90.000	100.000	50	35	18	67
2	MNT001	MONITOR	700.000	750.000	25	5	12	18
3	MOU001	MOUSE	50.000	55.000	50	47	26	71

8.3 MENCETAK DATA SUPLIER

Langkah-langkah mencetak data Suplier sebagai berikut :

- a. Double klik form MENUUTAMA
- b. Klik menu View => Code, ketik kode program berikut

```
Private Sub PDSUPLIER_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.I
    KIRI.Alignment = StringAlignment.Near
    TENGAH.Alignment = StringAlignment.Center
    KANAN.Alignment = StringAlignment.Far

    e.Graphics.DrawString("LAPORAN DATA SUPLIER", FONTJUDUL, Brushes.Blue, 400, 25, TENGAH)
    e.Graphics.DrawLine(Pens.Black, 30, 70, 805, 70)
    e.Graphics.DrawString("NOMOR", FONTDATA, Brushes.Black, 100, 75, KANAN)
    e.Graphics.DrawString("KODE", FONTDATA, Brushes.Black, 125, 75, KIRI)
    e.Graphics.DrawString("NAMA SUPLIER", FONTDATA, Brushes.Black, 200, 75, KIRI)
    e.Graphics.DrawString("ALAMAT", FONTDATA, Brushes.Black, 400, 75, KIRI)
    e.Graphics.DrawString("NO. TELPON", FONTDATA, Brushes.Black, 600, 75, KIRI)
    e.Graphics.DrawLine(Pens.Black, 30, 100, 805, 100)
```

```

BARIS = 105
NOMOR = 1
SQLSTR = "SELECT * FROM SUPLIER"
CMD = New OleDbCommand(SQLSTR, KONEKSI)
DR = CMD.ExecuteReader
While DR.Read
    e.Graphics.DrawString(NOMOR, FONTDATA, Brushes.Black, 100, BARIS, KANAN)
    e.Graphics.DrawString(DR("KODESP"), FONTDATA, Brushes.Black, 125, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMASP"), FONTDATA, Brushes.Black, 200, BARIS, KIRI)
    e.Graphics.DrawString(DR("ALAMATSP"), FONTDATA, Brushes.Black, 400, BARIS, KIRI)
    e.Graphics.DrawString(DR("TELPONSP"), FONTDATA, Brushes.Black, 600, BARIS, KIRI)
    NOMOR = NOMOR + 1
    BARIS = BARIS + 20
End While
e.Graphics.DrawLine(Pens.Black, 30, BARIS + 5, 805, BARIS + 5)
DR.Close()

```

End Sub

Perintah `e.Graphics.DrawString("LAPORAN DATA SUPLIER", FONTJUDUL, Brushes.Blue, 400, 25, TENGAH)` untuk mencetak string LAPORAN DATA SUPLIER menggunakan jenis FontJudul, warna Biru, kolom 400, baris 25, perataan Tengah. Perintah `e.Graphics.DrawLine(Pens.Black, 30, 70, 805, 70)` untuk mencetak garis mendatar warna Hitam, mulai Kolom 30 Baris 70 sampai Kolom 805 Baris 70. Variabel BARIS untuk menentukan awal Baris data pertama, variabel NOMOR untuk awal Nomor data pertama. Sebelum mencetak data berikutnya, nilai variabel BARIS akan bertambah 20 dan nilai variabel NOMOR bertambah 1.

```

Private Sub CETAKSUPLIERToolStripMenuItem_Click(ByV
    PDSUPLIER.Print()
End Sub

```

Perintah `PDSUPLIER.Print` untuk mencetak data yang telah didesain pada `Private Sub PDSUPLIER_PrintPage`.

- c. Klik menu Debug => Start Debugging
- d. Klik menu Cetak => Cetak Suplier

LAPORAN DATA SUPLIER

NOMOR	KODE	NAMA SUPLIER	ALAMAT	NO. TELPON
1	SP001	CV. COMPUTAMA	JL. MAJAPAHIT 25	088802250099
2	SP002	PT. TEKNOCOMP	JL. GAJAHMADA	088802250066

8.4 MENCETAK DATA CUSTOMER

Langkah-langkah mencetak data Suplier sebagai berikut :

- Double klik form MENUUTAMA
- Klik menu View => Code
- Ketik kode program berikut

```
Private Sub PDCUSTOMER_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.
    KIRI.Alignment = StringAlignment.Near
    TENGAH.Alignment = StringAlignment.Center
    KANAN.Alignment = StringAlignment.Far

    e.Graphics.DrawString("LAPORAN DATA CUSTOMER", FONTJUDUL, Brushes.Blue, 400, 25, TENGAH)
    e.Graphics.DrawLine(Pens.Black, 30, 70, 805, 70)
    e.Graphics.DrawString("NOMOR", FONTDATA, Brushes.Black, 100, 75, KANAN)
    e.Graphics.DrawString("KODE", FONTDATA, Brushes.Black, 125, 75, KIRI)
    e.Graphics.DrawString("NAMA CUSTOMER", FONTDATA, Brushes.Black, 200, 75, KIRI)
    e.Graphics.DrawString("ALAMAT", FONTDATA, Brushes.Black, 400, 75, KIRI)
    e.Graphics.DrawString("NO. TELPON", FONTDATA, Brushes.Black, 600, 75, KIRI)
    e.Graphics.DrawLine(Pens.Black, 30, 100, 805, 100)
    BARIS = 105
    NOMOR = 1
    SQLSTR = "SELECT * FROM CUSTOMER"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    While DR.Read
        e.Graphics.DrawString(NOMOR, FONTDATA, Brushes.Black, 100, BARIS, KANAN)
        e.Graphics.DrawString(DR("KODECS"), FONTDATA, Brushes.Black, 125, BARIS, KIRI)
        e.Graphics.DrawString(DR("NAMACS"), FONTDATA, Brushes.Black, 200, BARIS, KIRI)
        e.Graphics.DrawString(DR("ALAMATCS"), FONTDATA, Brushes.Black, 400, BARIS, KIRI)
        e.Graphics.DrawString(DR("TELPONCS"), FONTDATA, Brushes.Black, 600, BARIS, KIRI)
        NOMOR = NOMOR + 1
        BARIS = BARIS + 20
    End While
    e.Graphics.DrawLine(Pens.Black, 30, BARIS + 5, 805, BARIS + 5)
    DR.Close()
End Sub
```

Perintah `e.Graphics.DrawString("LAPORAN DATA CUSTOMER", FONTJUDUL, Brushes.Blue, 400, 25, TENGAH)` untuk mencetak string `LAPORAN DATA CUSTOMER` menggunakan jenis `FontJudul`, warna `Biru`, kolom `400`, baris `25`, perataan `Tengah`. Perintah `e.Graphics.DrawLine(Pens.Black, 30, 70, 805, 70)` untuk mencetak garis mendatar warna `Hitam`, mulai Kolom `30` Baris `70` sampai Kolom `805` Baris `70`. Variabel `BARIS` untuk menentukan awal Baris data pertama, variabel `NOMOR` untuk awal Nomor data pertama. Sebelum mencetak data berikutnya, nilai variabel `BARIS` akan bertambah `20` dan nilai variabel `NOMOR` bertambah `1`.

```
Private Sub CETAKCUSTOMERToolStripMenuItem_Click
    PDCUSTOMER.Print()
End Sub
```

Perintah `PDCUSTOMER.Print` untuk mencetak data yang telah didesain di `Private Sub PDCUSTOMER_PrintPage`.

- d. Klik menu Debug => Start Debugging
- e. Klik menu Cetak => Cetak Customer

LAPORAN DATA CUSTOMER

NOMOR	KODE	NAMA CUSTOMER	ALAMAT	NO. TELPON
1	CS001	STMIK NUSANTARA	JL. DIPONEGORO 25	024654321
2	CS002	SMK NUSA PUTRA	JL. PAHLWAWAN 125	024321456

8.5 TUTORIAL MENCETAK DATA MASTER



<https://youtu.be/sOObsQnq1lo>

8.6 EVALUASI

Buat Kode program untuk mencetak Total Jumlah Awal, Jumlah Beli, Jumlah Jual dan Jumlah Stok pada Laporan Data Barang.

8.7 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB IX

MENCETAK PEMBELIAN PERFAKTUR

9.1 CAPAIAN PEMBELAJARAN

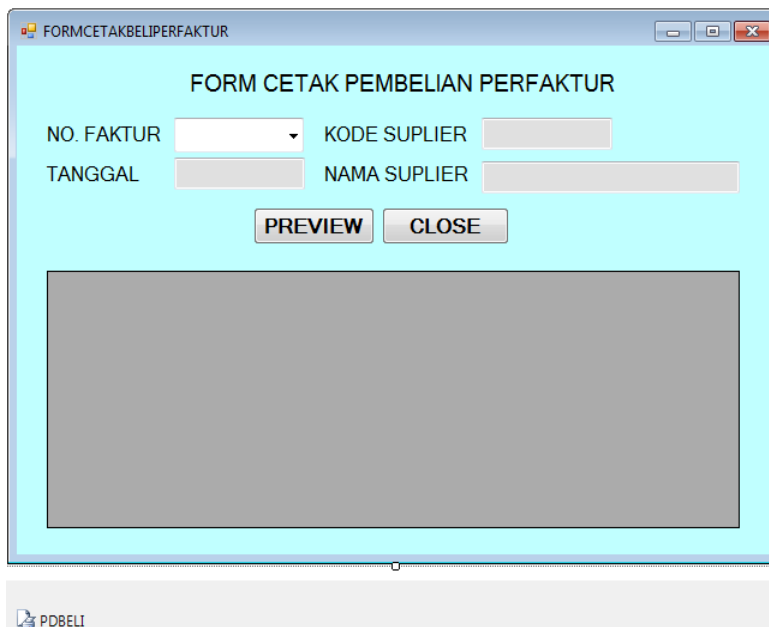
Capaian pembelajaran Mencetak Pembelian PerFaktur yaitu mahasiswa dapat mendesain form untuk memfilter data Pembelian PerFaktur dan memahami kode program untuk mencetak data Pembelian PerFaktur.

9.2 FORM PEMBELIAN PERFAKTUR

A. MENDESAIN FORM PEMBELIAN PERFAKTUR

Langkah-langkah mendesain Form Pembelian PerFaktur sebagai berikut :

1. Double klik file PRINV.SLN untuk membuka project Inventory
2. Klik menu Project ⇒ Add Windows Form
3. Ketik nama form FORMCETAKBELIPERFAKTUR
4. Klik tombol Add
5. Buat desain Form Pembelian PerFaktur berikut



The image shows a screenshot of a Windows application window titled "FORMCETAKBELIPERFAKTUR". The window has a light blue background and contains the following elements:

- Title Bar:** "FORMCETAKBELIPERFAKTUR" with standard minimize, maximize, and close buttons.
- Form Title:** "FORM CETAK PEMBELIAN PERFAKTUR" centered at the top.
- Fields:**
 - "NO. FAKTUR" with a dropdown menu.
 - "TANGGAL" with a date picker.
 - "KODE SUPLIER" with a text input field.
 - "NAMA SUPLIER" with a text input field.
- Buttons:** "PREVIEW" and "CLOSE" buttons.
- Content Area:** A large grey rectangular area below the buttons, likely intended for a preview of the invoice.

At the bottom of the screenshot, a taskbar icon for "PBELI" is visible.

Gambar 9.1 Desain Form Pembelian PerFaktur

Daftar Objek pada Form Pembelian PerFaktur

Komponen	Properties	Isi
Label	Text	FORM CETAK PEMBELIAN PERFAKTUR
	Text	NO. FAKTUR
	Text	TANGGAL
	Text	KODE SUPLIER
	Text	NAMA SUPLIER
ComboBox	Name	TXTFAKTUR
TextBox	Name	TXTTANGGAL
	Name	TXTKODESP
	Name	TXTNAMASP
Button	Name	BTPREVIEW
	Text	PREVIEW
	Name	BTCLOSE
	Text	CLOSE
DataGridView	Name	DGVBELI
PrintDocument	Name	PDBELI

B. KODE PROGRAM FORM PEMBELIAN PERFAKTUR

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Imports System.Drawing.Printing
Public Class FORMCETAKBELIPERFAKTUR
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DA As New OleDb.OleDbDataAdapter
    Public DT As New DataTable
    Dim DR As OleDbDataReader
    Dim BARIS As Integer
    Dim FONTJUDUL As Font = New Font("CALIBRI", 14, FontStyle.Bold, GraphicsUnit.Point)
    Dim FONTDATA As Font = New Font("CALIBRI", 12, FontStyle.Regular, GraphicsUnit.Point)
    Dim KIRI As StringFormat = New StringFormat
    Dim TENGAH As StringFormat = New StringFormat
    Dim KANAN As StringFormat = New StringFormat
    Dim NOMOR As Integer
    Dim TOTAL As Double
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR

untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DA untuk menampilkan data Pembelian PerFaktur, Variabel DR untuk membaca data dari suatu tabel. Variabel BARIS untuk menentukan nilai baris data, variabel NOMOR untuk Nomor Urut data, variabel FONTJUDUL untuk menentukan ukuran font judul lapotan, variabel FONTDATA untuk menentukan ukuran font data, variabel KIRI untuk perataan kiri, variabel TENGAH untuk perataan tengah dan variabel KANAN untuk perataan kanan.

```

Private Sub FORMCETAKBELIPERFAKTUR_Load(ByVal sender As Object, ByVal e As System
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()

    SQLSTR = "SELECT * FROM BELI ORDER BY FAKTUR"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    TXTFAKTUR.Items.Clear()
    While DR.Read
        TXTFAKTUR.Items.Add(DR("FAKTUR"))
    End While
    DR.Close()
End Sub

```

Membuat koneksi ke database DBINV,MDB dan membuka koneksi database. Mengisi data pada ComboBox TXTFAKTUR dengan data Nomor Faktur dari tabel BELI

```

Sub KOSONG()
    TXTFAKTUR.Text = ""
    TXTKODESP.Text = ""
    TXTNAMASP.Text = ""
    TXTTANGGAL.Text = ""
    SQLSTR = "DELETE FROM DETAILBELISEMU"
    DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVBELI.DataSource = DT
    TXTFAKTUR.Focus()
End Sub
Sub DGV()
    SQLSTR = "SELECT * FROM DETAILBELISEMU"
    DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVBELI.DataSource = DT
End Sub
Private Sub FORMCETAKBELIPERFAKTUR_Activated(By
    KOSONG()
    DGV()
End Sub

```

Sub KOSONG untuk mengosongkan ComboBox, TextBox dan menghapus semua data pada tabel DETAILBELISEMU. Sub DGV untuk menampilkan data pada tabel DETAILBELISEMU ke DataGridView DGVBELI. Saat form aktif akan menjalankan Sub KOSONG dan Sub DGV.

```
Private Sub TXTFAKTUR_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles
    SQLSTR = "SELECT * FROM QUERYBELI WHERE FAKTUR=" & TXTFAKTUR.Text & ""
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    While DR.Read
        TXTTANGGAL.Text = DR("TANGGAL")
        TXTKODESP.Text = DR("KODESP")
        TXTNAMASP.Text = DR("NAMASP")
    End While
    DR.Close()

    SQLSTR = "DELETE * FROM DETAILBELISEMU"
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()

    SQLSTR = "INSERT INTO DETAILBELISEMU ( KODEBERG, NAMABERG, HRGBELI, JUMLAH, TOTAL ) SELECT " & _
    "QUERYBELI.KODEBERG, QUERYBELI.NAMABERG, QUERYBELI.HRGBELI, QUERYBELI.JUMLAH, QUERYBELI.TOTAL " & _
    " FROM QUERYBELI WHERE FAKTUR=" & TXTFAKTUR.Text & ""
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()

    DGV()
    BTPREVIEW.Focus()
End Sub
```

Saat diklik Nomor Faktur akan dicari Nomor Faktur yang diklik pada query QUERYBELI lalu ditampilkan data Tanggal Pembelian, Kode Suplier dan Nama Suplier. Berikutnya akan dihapus semua data pada tabel DETAILBELISEMU dan data pada QUERYBELI yang Nomor Fakturnya sesuai akan ditambahkan ke tabel DETAILBELISEMU.

```
Private Sub PDBELI_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.PrintPageEventArgs)
    KIRI.Alignment = StringAlignment.Near
    TENGAH.Alignment = StringAlignment.Center
    KANAN.Alignment = StringAlignment.Far

    e.Graphics.DrawString("LAPORAN PEMBELIAN PERFAKTUR", FONTJUDUL, Brushes.Blue, 300, 25, TENGAH)
    e.Graphics.DrawString("NO. FAKTUR : " & TXTFAKTUR.Text, FONTDATA, Brushes.Black, 30, 70, KIRI)
    e.Graphics.DrawString("TANGGAL : " & TXTTANGGAL.Text, FONTDATA, Brushes.Black, 30, 90, KIRI)
    e.Graphics.DrawString("KODE SUPLIER : " & TXTKODESP.Text, FONTDATA, Brushes.Black, 240, 70, KIRI)
    e.Graphics.DrawString("NAMA SUPLIER : " & TXTNAMASP.Text, FONTDATA, Brushes.Black, 240, 90, KIRI)
    e.Graphics.DrawLine(Pens.Black, 30, 110, 600, 110)
    e.Graphics.DrawString("NOMOR", FONTDATA, Brushes.Black, 100, 115, KANAN)
    e.Graphics.DrawString("KODEBERG", FONTDATA, Brushes.Black, 125, 115, KIRI)
    e.Graphics.DrawString("NAMA BARANG", FONTDATA, Brushes.Black, 200, 115, KIRI)
    e.Graphics.DrawString("HRG BELI", FONTDATA, Brushes.Black, 400, 115, KANAN)
    e.Graphics.DrawString("JUMLAH", FONTDATA, Brushes.Black, 500, 115, KANAN)
    e.Graphics.DrawString("TOTAL", FONTDATA, Brushes.Black, 575, 115, KANAN)
    e.Graphics.DrawLine(Pens.Black, 30, 140, 600, 140)
```

Mendesain judul laporan yang terdiri dari Nomor Faktur, Tanggal, Kode Suplier dan Nama Suplier. Perintah e.Graphics.DrawLine(Pens.Black, 30, 110, 600, 110) untuk mencetak garis mendatar, berikutnya mencetak judul kolom yang terdiri dari Nomor, Kodebrg, Nama Barang, Harga Beli, Jumlah dan Total.

```

BARIS = 145
NOMOR = 1
TOTAL = 0
SQLSTR = "SELECT * FROM DETAILBELISEMU"
CMD = New OleDbCommand(SQLSTR, KONEKSI)
DR = CMD.ExecuteReader
While DR.Read
    e.Graphics.DrawString(NOMOR, FONTDATA, Brushes.Black, 100, BARIS, KANAN)
    e.Graphics.DrawString(DR("KODEBRG"), FONTDATA, Brushes.Black, 125, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMA BRG"), FONTDATA, Brushes.Black, 200, BARIS, KIRI)
    e.Graphics.DrawString(Format(DR("HRGBELI"), "#,#"), FONTDATA, Brushes.Black, 400, BARIS, KANAN)
    e.Graphics.DrawString(DR("JUMLAH"), FONTDATA, Brushes.Black, 475, BARIS, KANAN)
    e.Graphics.DrawString(Format(DR("TOTAL"), "#,#"), FONTDATA, Brushes.Black, 575, BARIS, KANAN)
    NOMOR = NOMOR + 1
    BARIS = BARIS + 20
    TOTAL = TOTAL + DR("TOTAL")
End While
e.Graphics.DrawLine(Pens.Black, 30, BARIS + 5, 600, BARIS + 5)
e.Graphics.DrawString("TOTAL :", FONTDATA, Brushes.Black, 500, BARIS + 10, KANAN)
e.Graphics.DrawString(Format(TOTAL, "#,#"), FONTDATA, Brushes.Black, 575, BARIS + 10, KANAN)
e.Graphics.DrawLine(Pens.Black, 30, BARIS + 30, 600, BARIS + 30)
DR.Close()
End Sub

```

Mencetak data pada tabel DETAILBELISEMU yang terdiri dari Nomor, Kode Barang, Nama Barang, Harga Beli, Jumlah dan Total. Variabel TOTAL untuk menghitung Total Pembelian PerFaktur.

```

Private Sub BTPREVIEW_Click(ByVal sender
    PDBELI.Print()
End Sub

```

Saat diklik tombol PREVIEW akan mencetak data Pembelian PerFaktur yang sudah didesain.

```

Private Sub BTCLOSE_Click(ByVal sender As System.Object,
    KONEKSI.Close()
    Close()
End Sub

```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENCETAK PEMBELIAN PERFAKTUR

Cara mencetak data Pembelian PerFaktur :

- Klik droplist pada Nomor Faktur
- Klik Nomor Faktur yang datanya mau dicetak, data item barang akan tampil di DataGridView
- Klik tombol PREVIEW

KODEBRG	NAMABRG	HRGBELI	JUMLAH	TOTAL
MNT001	MONITOR	150000	5	750000
KYB001	KEYBOARD	65000	15	975000
MOU001	MOUSE	25000	20	500000

LAPORAN PEMBELIAN PERFAKTUR

NO. FAKTUR : FK002 KODE SUPLIER : SP002
TANGGAL : 27/01/2021 NAMA SUPLIER : PT. TEKNOCOMP

NOMOR	KODEBRG	NAMA BARANG	HRG BELI	JUMLAH	TOTAL
1	MNT001	MONITOR	150.000	5	750.000
2	KYB001	KEYBOARD	65.000	15	975.000
3	MOU001	MOUSE	25.000	20	500.000
					TOTAL : 2.225.000

9.3 TUTORIAL PEMBELIAN PERFAKTUR



<https://youtu.be/6-cYdgi8l30>

9.4 EVALUASI

Tambahkan garis pembatas data pada laporan Pembelian PerFaktur.

LAPORAN PEMBELIAN PERFAKTUR

NO. FAKTUR : FK002 KODE SUPLIER : SP002
TANGGAL : 27/01/2021 NAMA SUPLIER : PT. TEKNOCOMP

NOMOR	KODEBRG	NAMA BARANG	HRG BELI	JUMLAH	TOTAL
1	MNT001	MONITOR	150.000	5	750.000
2	KYB001	KEYBOARD	65.000	15	975.000
3	MOU001	MOUSE	25.000	20	500.000
TOTAL :					2.225.000

9.5 REFERENSI

I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*”, Jakarta : Elex Media Komputindo.

Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.

Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.

WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB X MENCETAK PEMBELIAN PERPERIODE

10.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran Mencetak Pembelian PerPeriode yaitu mahasiswa dapat mendesain form untuk memfilter data Pembelian PerPeriode dan memahami kode program untuk mencetak data Pembelian PePeriode.

10.2 FORM PEMBELIAN PERPERIODE

A. DESAIN FORM PEMBELIAN PERPERIODE

Langkah-langkah mendesain Form Pembelian PerPeriode sebagai berikut :

1. Double klik file PRINV.SLN untuk membuka project Inventory
2. Klik menu Project ⇒ Add Windows Form
3. Ketik nama form FORMCETAKBELIPERPERIODE
4. Klik tombol Add
5. Buat desain Form Pembelian PerPeriode berikut

Gambar 10.1 Desain Form Pembelian PerPeriode

Daftar Objek pada Form Pembelian PerPeriode

Komponen	Properties	Isi
Label	Text	FORM Cetak Pembelian PERPERIODE
	Text	MULAI TANGGAL
	Text	SAMPAI TANGGAL

ComboBox	Name	TXTMULAI
	Name	TXTSAMPAI
Button	Name	BTPREVIEW
	Text	PREVIEW
	Name	BTCLOSE
	Text	CLOSE
DataGridView	Name	DGVBELI
PrintDocument	Name	PDBELI

B. KODE PROGRAM FORM PEMBELIAN PERPERIODE

Klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Imports System.Drawing.Printing
Public Class FORMCETAKBELIPERPERIODE
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DA As New OleDb.OleDbDataAdapter
    Public DT As New DataTable
    Dim DR As OleDbDataReader
    Dim BARIS As Integer
    Dim FONTJUDUL As Font = New Font("CALIBRI", 14, FontStyle.Bold, GraphicsUnit.Point)
    Dim FONTDATA As Font = New Font("CALIBRI", 12, FontStyle.Regular, GraphicsUnit.Point)
    Dim KIRI As StringFormat = New StringFormat
    Dim TENGAH As StringFormat = New StringFormat
    Dim KANAN As StringFormat = New StringFormat
    Dim NOMOR As Integer
    Dim TOTAL As Double
    Dim XMULAI As String
    Dim XSAMPAI As String
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DA untuk menampilkan data Pembelian PerFaktur, Variabel DR untuk membaca data dari suatu tabel. Variabel BARIS untuk menentukan nilai baris data, variabel NOMOR untuk Nomor Urut data, variabel FONTJUDUL untuk menentukan ukuran font judul lapotan, variabel FONTDATA untuk menentukan ukuran font data, variabel KIRI untuk perataan kiri, variabel TENGAH untuk perataan tengah dan variabel KANAN untuk perataan kanan, variabel XMULAI untuk tanggal awal periode, variabel XSAMPAI untuk tanggal akhir periode.

```

Private Sub FORMCETAKBELIPERPERIODE_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Load
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()

    SQLSTR = "SELECT * FROM BELI ORDER BY TANGGAL"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    TXTMULAI.Items.Clear()
    TXTSAMPALAI.Items.Clear()
    While DR.Read
        TXTMULAI.Items.Add(DR("TANGGAL"))
        TXTSAMPALAI.Items.Add(DR("TANGGAL"))
    End While
    DR.Close()
End Sub

```

Membuat koneksi ke database DBINV,MDB dan membuka koneksi database. Mengisi data pada ComboBox TXTMULAI dan TXTSAMPALAI dengan data Tanggal Pembelian dari tabel BELI

```

Sub KOSONG()
    TXTMULAI.Text = ""
    TXTSAMPALAI.Text = ""
    TXTMULAI.Focus()
End Sub

```

```

Sub DGV()
    SQLSTR = "SELECT * FROM QUERYBELI WHERE FAKTUR IS NULL"
    DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVBELI.DataSource = DT
End Sub

```

```

Private Sub FORMCETAKBELIPERPERIODE_Activated(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Activated
    KOSONG()
    DGV()
End Sub

```

Sub KOSONG untuk mengosongkan ComboBox TXTMULAI dan TXTSAMPALAI. Sub DGV untuk menampilkan data pada query QUERYBELI ke DataGridView DGVBELI. Saat form aktif akan menjalankan Sub KOSONG dan Sub DGV.

```

Private Sub TXTMULAI_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles TXTMULAI_SelectedValueChanged
    If TXTSAMPAL.Text <> "" Then
        XMULAI = Strings.Mid(TXTMULAI.Text, 4, 2) & "/" & Strings.Left(TXTMULAI.Text, 2) & "/" & Strings.Right(TXTMULAI.Text, 4)
        XSAMPAL = Strings.Mid(TXTSAMPAL.Text, 4, 2) & "/" & Strings.Left(TXTSAMPAL.Text, 2) & "/" & Strings.Right(TXTSAMPAL.Text, 4)
        SQLSTR = "SELECT * FROM QUERYBELI WHERE TANGGAL BETWEEN #" & XMULAI & "# AND #" & XSAMPAL & "#"
        DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
        DT.Clear()
        DA.Fill(DT)
        DGVBELI.DataSource = DT
    End If
End Sub

```

Saat diklik Tanggal Mulai, jika data pada ComboBox TXTSAMPAL tidak kosong maka data Pembelian sesuai dengan periode yang telah ditentukan akan tampil di DataGridView DGVBELI.

```

Private Sub TXTSAMPAL_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles TXTSAMPAL_SelectedValueChanged
    If TXTMULAI.Text <> "" Then
        XMULAI = Strings.Mid(TXTMULAI.Text, 4, 2) & "/" & Strings.Left(TXTMULAI.Text, 2) & "/" & Strings.Right(TXTMULAI.Text, 4)
        XSAMPAL = Strings.Mid(TXTSAMPAL.Text, 4, 2) & "/" & Strings.Left(TXTSAMPAL.Text, 2) & "/" & Strings.Right(TXTSAMPAL.Text, 4)
        SQLSTR = "SELECT * FROM QUERYBELI WHERE TANGGAL BETWEEN #" & XMULAI & "# AND #" & XSAMPAL & "#"
        DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
        DT.Clear()
        DA.Fill(DT)
        DGVBELI.DataSource = DT
    End If
End Sub

```

Saat diklik Tanggal Sampai, jika data pada ComboBox TXTMULAI tidak kosong maka data Pembelian sesuai dengan periode yang telah ditentukan akan tampil di DataGridView DGVBELI.

```

Private Sub PDBELI_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.PrintPageEventArgs) Handles PDBELI
    KIRI.Alignment = StringAlignment.Near
    TENGAH.Alignment = StringAlignment.Center
    KANAN.Alignment = StringAlignment.Far

    e.Graphics.DrawString("LAPORAN PEMBELIAN PERPERIODE", FONTJUDUL, Brushes.Blue, 400, 25, TENGAH)
    e.Graphics.DrawString("TANGGAL " & TXTMULAI.Text & " S/D " & TXTSAMPAI.Text, FONTJUDUL, Brushes.Blue, 400, 50, TENGAH)
    e.Graphics.DrawLine(Pens.Black, 10, 95, 800, 95)
    e.Graphics.DrawString("NO.", FONTDATA, Brushes.Black, 50, 100, KANAN)
    e.Graphics.DrawString("FAKTUR", FONTDATA, Brushes.Black, 60, 100, KIRI)
    e.Graphics.DrawString("TANGGAL", FONTDATA, Brushes.Black, 120, 100, KIRI)
    e.Graphics.DrawString("NAMA SUPPLIER", FONTDATA, Brushes.Black, 210, 100, KIRI)
    e.Graphics.DrawString("KODEBERG", FONTDATA, Brushes.Black, 350, 100, KIRI)
    e.Graphics.DrawString("NAMA BARANG", FONTDATA, Brushes.Black, 435, 100, KIRI)
    e.Graphics.DrawString("HRG BELI", FONTDATA, Brushes.Black, 620, 100, KANAN)
    e.Graphics.DrawString("JUMLAH", FONTDATA, Brushes.Black, 700, 100, KANAN)
    e.Graphics.DrawString("TOTAL", FONTDATA, Brushes.Black, 775, 100, KANAN)
    e.Graphics.DrawLine(Pens.Black, 10, 125, 800, 125)

```

Mendesain judul laporan Tanggal Periode data Pembelian dan judul kolom yang terdiri dari Nomor, Nomor Faktur, Tanggal, Nama Suplier, Kode Barang, Nama Barang, Harga Beli, Jumlah dan Total.

```

BARIS = 130
NOMOR = 1
TOTAL = 0
XMULAI = Strings.Mid(TXTMULAI.Text, 4, 2) & "/" & Strings.Left(TXTMULAI.Text, 2) & "/" & Strings.Right(TXTMULAI.Text, 4)
XSAMPAI = Strings.Mid(TXTSAMPAI.Text, 4, 2) & "/" & Strings.Left(TXTSAMPAI.Text, 2) & "/" & Strings.Right(TXTSAMPAI.Text, 4)
SQLSTR = "SELECT * FROM QUERYBELI WHERE TANGGAL BETWEEN #" & XMULAI & "# AND #" & XSAMPAI & "#"
CMD = New OleDbCommand(SQLSTR, KONEKSI)
DR = CMD.ExecuteReader
While DR.Read
    e.Graphics.DrawString(NOMOR, FONTDATA, Brushes.Black, 50, BARIS, KANAN)
    e.Graphics.DrawString(DR("FAKTUR"), FONTDATA, Brushes.Black, 60, BARIS, KIRI)
    e.Graphics.DrawString(DR("TANGGAL"), FONTDATA, Brushes.Black, 120, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMASUP"), FONTDATA, Brushes.Black, 210, BARIS, KIRI)
    e.Graphics.DrawString(DR("KODEBERG"), FONTDATA, Brushes.Black, 350, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMABERG"), FONTDATA, Brushes.Black, 435, BARIS, KIRI)
    e.Graphics.DrawString(Format(DR("HRGBELI"), "#.#"), FONTDATA, Brushes.Black, 620, BARIS, KANAN)
    e.Graphics.DrawString(DR("JUMLAH"), FONTDATA, Brushes.Black, 675, BARIS, KANAN)
    e.Graphics.DrawString(Format(DR("TOTAL"), "#.#"), FONTDATA, Brushes.Black, 775, BARIS, KANAN)
    NOMOR = NOMOR + 1
    BARIS = BARIS + 20
    TOTAL = TOTAL + DR("TOTAL")
End While
e.Graphics.DrawLine(Pens.Black, 10, BARIS + 5, 800, BARIS + 5)
e.Graphics.DrawString("TOTAL :", FONTDATA, Brushes.Black, 675, BARIS + 10, KANAN)
e.Graphics.DrawString(Format(TOTAL, "#.#"), FONTDATA, Brushes.Black, 775, BARIS + 10, KANAN)
e.Graphics.DrawLine(Pens.Black, 10, BARIS + 30, 800, BARIS + 30)
DR.Close()
End Sub

```

Mencetak data Pembelian sesuai dengan Periode yang ditentukan terdiri dari Nomor, Nomor Faktur, Tanggal, Nama Suplier, Kode Barang, Nama Barang, Harga Beli, Jumlah dan Total. Variabel TOTAL untuk menghitung Total Pembelian PerPeriode.

```
Private Sub BTPREVIEW_Click(ByVal sender
    PDBELI.Print()
End Sub
```

Saat diklik tombol PREVIEW akan mencetak data Pembelian PerPeriode yang sudah didesain.

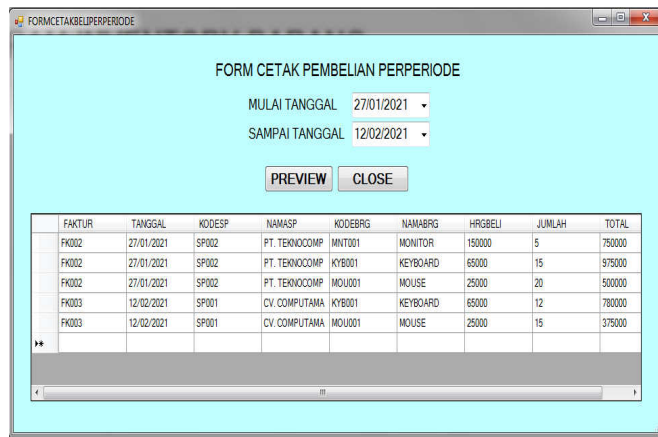
```
Private Sub BTCLOSE_Click(ByVal sender As System.Object,
    KONEKSI.Close()
    Close()
End Sub
```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENCETAK PEMBELIAN PERPERIODE

Cara mencetak data Pembelian PerPeriode :

- Klik droplist pada Mulai Tanggal, klik Tanggal Mulai Periode
- Klik droplist pada Sampai Tanggal, klik Tanggal Akhir Periode, data item barang akan tampil di DataGridView
- Klik tombol PREVIEW



LAPORAN PEMBELIAN PERPERIODE TANGGAL 27/01/2021 S/D 12/02/2021

NO.	FAKTUR	TANGGAL	NAMA SUPLIER	KODEBRG	NAMA BARANG	HRG BELI	JUMLAH	TOTAL
1	FK002	27/01/2021	PT. TEKNOCOMP	MNT001	MONITOR	150.000	5	750.000
2	FK002	27/01/2021	PT. TEKNOCOMP	KYB001	KEYBOARD	65.000	15	975.000
3	FK002	27/01/2021	PT. TEKNOCOMP	MOU001	MOUSE	25.000	20	500.000
4	FK003	12/02/2021	CV. COMPUTAMA	KYB001	KEYBOARD	65.000	12	780.000
5	FK003	12/02/2021	CV. COMPUTAMA	MOU001	MOUSE	25.000	15	375.000
							TOTAL :	3.380.000

10.3 TUTORIAL PEMBELIAN PERPERIODE



<https://youtu.be/iMW0AhYLVpg>

10.4 EVALUASI

Tambahkan Logo (gambar bebas) pada laporan Pembelian PerPeriode.

LAPORAN PEMBELIAN PERPERIODE
TANGGAL 27/01/2021 S/D 12/02/2021



NO.	FAKTUR	TANGGAL	NAMA SUPLIER	KODEBRG	NAMA BARANG	HRG BELI	JUMLAH	TOTAL
1	FK002	27/01/2021	PT. TEKNOCOMP	MNT001	MONITOR	150.000	5	750.000
2	FK002	27/01/2021	PT. TEKNOCOMP	KYB001	KEYBOARD	65.000	15	975.000
3	FK002	27/01/2021	PT. TEKNOCOMP	MOU001	MOUSE	25.000	20	500.000
4	FK003	12/02/2021	CV. COMPUTAMA	KYB001	KEYBOARD	65.000	12	780.000
5	FK003	12/02/2021	CV. COMPUTAMA	MOU001	MOUSE	25.000	15	375.000
TOTAL :								3.380.000

10.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*”, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB XI

MENCETAK PENJUALAN PERNOTA

11.1 CAPAIAN PEMBELAJARAN

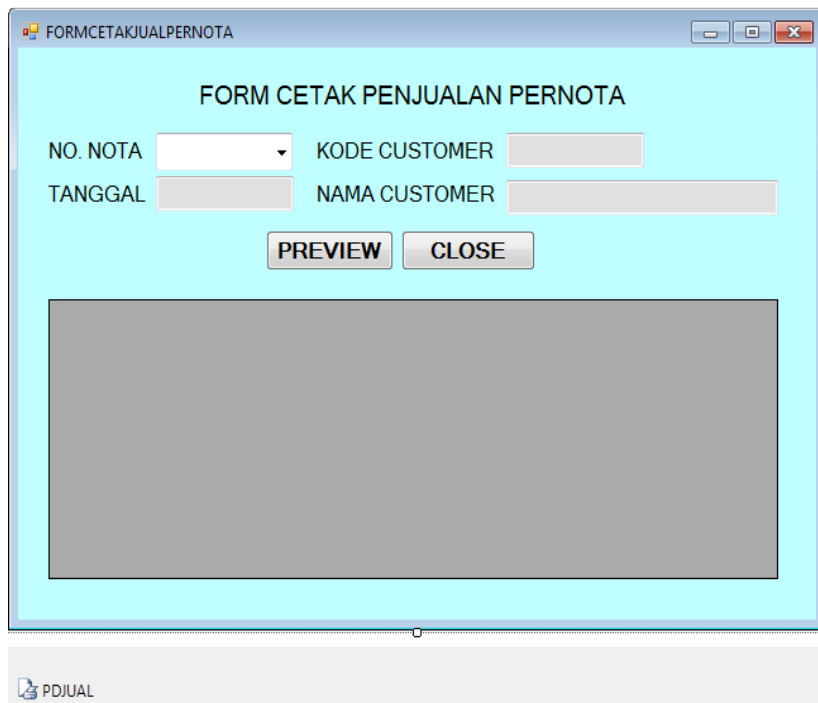
Capaian pembelajaran Mencetak Penjualan PerNota yaitu mahasiswa dapat mendesain form untuk memfilter data Penjualan PerNota dan memahami kode program untuk mencetak data Penjualan PerNota.

11.2 FORM PENJUALAN PERNOTA

A. MENDESAIN FORM PENJUALAN PERNOTA

Langkah-langkah mendesain Form Penjualan PerNota sebagai berikut :

1. Double klik file PRINV.SLN untuk membuka project Inventory
2. Klik menu Project ⇒ Add Windows Form
3. Ketik nama form FORMCETAKJUALPERNOTA
4. Klik tombol Add
5. Buat desain Form Penjualan PerNota berikut



The image shows a screenshot of a Windows application window titled "FORMCETAKJUALPERNOTA". The window has a light blue background and a title bar with standard Windows window controls (minimize, maximize, close). The main content area is titled "FORM CETAK PENJUALAN PERNOTA". Below the title, there are four input fields: "NO. NOTA" (a dropdown menu), "TANGGAL" (a text box), "KODE CUSTOMER" (a text box), and "NAMA CUSTOMER" (a text box). Below these fields are two buttons: "PREVIEW" and "CLOSE". At the bottom of the window, there is a large gray rectangular area, likely a placeholder for a preview or data output. The window is shown on a desktop with a taskbar at the bottom containing a file named "PDJUAL".

Gambar 11.1 Desain Form Penjualan PerNota

Daftar Objek pada Form Penjualan PerNota

Komponen	Properties	Isi
Label	Text	FORM CETAK PEMNJUALAN PERNOTA
	Text	NO. NOTA
	Text	TANGGAL
	Text	KODE CUSTOMER
	Text	NAMA CUSTOMER
ComboBox	Name	TXTNOTA
TextBox	Name	TXTTANGGAL
	Name	TXTKODECS
	Name	TXTNAMACS
Button	Name	BTPREVIEW
	Text	PREVIEW
	Name	BTCLOSE
	Text	CLOSE
DataGridView	Name	DGVJUAL
PrintDocument	Name	PDJUAL

B. KODE PROGRAM FORM PENJUALAN PERNOTA

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Imports System.Drawing.Printing
Public Class FORMCETAKBELIPERFAKTUR
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DA As New OleDb.OleDbDataAdapter
    Public DT As New DataTable
    Dim DR As OleDbDataReader
    Dim BARIS As Integer
    Dim FONTJUDUL As Font = New Font("CALIBRI", 14, FontStyle.Bold, GraphicsUnit.Point)
    Dim FONTDATA As Font = New Font("CALIBRI", 12, FontStyle.Regular, GraphicsUnit.Point)
    Dim KIRI As StringFormat = New StringFormat
    Dim TENGAH As StringFormat = New StringFormat
    Dim KANAN As StringFormat = New StringFormat
    Dim NOMOR As Integer
    Dim TOTAL As Double
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR

untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DA untuk menampilkan data Pembelian PerFaktur, Variabel DR untuk membaca data dari suatu tabel. Variabel BARIS untuk menentukan nilai baris data, variabel NOMOR untuk Nomor Urut data, variabel FONTJUDUL untuk menentukan ukuran font judul lapotan, variabel FONTDATA untuk menentukan ukuran font data, variabel KIRI untuk perataan kiri, variabel TENGAH untuk perataan tengah dan variabel KANAN untuk perataan kanan.

```

Private Sub FORMCETAKJUALPERNOTA_Load(ByVal sender As Object, ByVal e As System.
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()

    SQLSTR = "SELECT * FROM JUAL ORDER BY NOTA"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    TXTNOTA.Items.Clear()
    While DR.Read
        TXTNOTA.Items.Add(DR("NOTA"))
    End While
    DR.Close()
End Sub

```

Membuat koneksi ke database DBINV,MDB dan membuka koneksi database. Mengisi data pada ComboBox TXTNOTA dengan data Nomor Nota dari tabel JUAL

```

Sub KOSONG()
    TXTNOTA.Text = ""
    TXTKODECS.Text = ""
    TXTNAMACS.Text = ""
    TXTTANGGAL.Text = ""
    SQLSTR = "DELETE FROM DETAILJUALSEMU"
    DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVJUAL.DataSource = DT
    TXTNOTA.Focus()
End Sub
Sub DGV()
    SQLSTR = "SELECT * FROM DETAILJUALSEMU"
    DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVJUAL.DataSource = DT
End Sub
Private Sub FORMCETAKJUALPERNOTA_Activated(ByVa.
    KOSONG()
    DGV()
End Sub

```

Sub KOSONG untuk mengosongkan ComboBox, TextBox dan menghapus semua data pada tabel DETAILJUALSEMU. Sub DGV untuk menampilkan data pada tabel DETAILJUALSEMU ke DataGridView DGVJUAL. Saat form aktif akan menjalankan Sub KOSONG dan Sub DGV.

```
Private Sub TXTNOTA_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Hanc
    SQLSTR = "SELECT * FROM QUERYJUAL WHERE NOTA='" & TXTNOTA.Text & ""
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    While DR.Read
        TXTTANGGAL.Text = DR("TANGGAL")
        TXTKODECS.Text = DR("KODECS")
        TXTNAMACS.Text = DR("NAMACS")
    End While
    DR.Close()

    SQLSTR = "DELETE * FROM DETAILJUALSEMU"
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()

    SQLSTR = "INSERT INTO DETAILJUALSEMU ( KODEBRG, NAMABRG,HRGJUAL, JUMLAH,TOTAL ) SELECT " & _
    "QUERYJUAL.KODEBRG,QUERYJUAL.NAMABRG,QUERYJUAL.HRGJUAL,QUERYJUAL.JUMLAH,QUERYJUAL.TOTAL " & _
    " FROM QUERYJUAL WHERE NOTA='" & TXTNOTA.Text & ""
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    CMD.ExecuteNonQuery()

    DGV()
    BTPREVIEW.Focus()
End Sub
```

Saat diklik Nomor Nota akan dicari Nomor Nota yang diklik pada query QUERYJUAL lalu ditampilkan data Tanggal Penjualan, Kode Customer dan Nama Customer. Berikutnya akan dihapus semua data pada tabel DETAILJUALSEMU dan data pada QUERYJUAL yang Nomor Notanya sesuai akan ditambahkan ke tabel DETAILJUALSEMU.

```
Private Sub PDJUAL_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.PrintPageEventArgs)
    KIRI.Alignment = StringAlignment.Near
    TENGAH.Alignment = StringAlignment.Center
    KANAN.Alignment = StringAlignment.Far

    e.Graphics.DrawString("LAPORAN PENJUALAN PERNOTA", FONTJUDUL, Brushes.Blue, 300, 25, TENGAH)
    e.Graphics.DrawString("NO. NOTA : " & TXTNOTA.Text, FONTDATA, Brushes.Black, 30, 70, KIRI)
    e.Graphics.DrawString("TANGGAL : " & TXTTANGGAL.Text, FONTDATA, Brushes.Black, 30, 90, KIRI)
    e.Graphics.DrawString("KODE CUSTOMER : " & TXTKODECS.Text, FONTDATA, Brushes.Black, 240, 70, KIRI)
    e.Graphics.DrawString("NAMA CUSTOMER : " & TXTNAMACS.Text, FONTDATA, Brushes.Black, 240, 90, KIRI)
    e.Graphics.DrawLine(Pens.Black, 30, 110, 600, 110)
    e.Graphics.DrawString("NOMOR", FONTDATA, Brushes.Black, 100, 115, KANAN)
    e.Graphics.DrawString("KODEBRG", FONTDATA, Brushes.Black, 125, 115, KIRI)
    e.Graphics.DrawString("NAMA BARANG", FONTDATA, Brushes.Black, 200, 115, KIRI)
    e.Graphics.DrawString("HRG JUAL", FONTDATA, Brushes.Black, 400, 115, KANAN)
    e.Graphics.DrawString("JUMLAH", FONTDATA, Brushes.Black, 500, 115, KANAN)
    e.Graphics.DrawString("TOTAL", FONTDATA, Brushes.Black, 575, 115, KANAN)
    e.Graphics.DrawLine(Pens.Black, 30, 140, 600, 140)
```

Mendesain judul laporan yang terdiri dari Nomor Nota, Tanggal, Kode Customer dan Nama Customer. Perintah `e.Graphics.DrawLine(Pens.Black, 30, 110, 600, 110)` untuk mencetak garis mendatar, berikutnya mencetak judul kolom yang terdiri dari Nomor, Kode Barang, Nama Barang, Harga Jual, Jumlah dan Total.

```

BARIS = 145
NOMOR = 1
TOTAL = 0
SQLSTR = "SELECT * FROM DETAILJUALSEMU"
CMD = New OleDbCommand(SQLSTR, KONEKSI)
DR = CMD.ExecuteReader
While DR.Read
    e.Graphics.DrawString(NOMOR, FONTDATA, Brushes.Black, 100, BARIS, KANAN)
    e.Graphics.DrawString(DR("KODEBRG"), FONTDATA, Brushes.Black, 125, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMABRG"), FONTDATA, Brushes.Black, 200, BARIS, KIRI)
    e.Graphics.DrawString(Format(DR("HRGJUAL"), "#,#"), FONTDATA, Brushes.Black, 400, BARIS, KANAN)
    e.Graphics.DrawString(DR("JUMLAH"), FONTDATA, Brushes.Black, 475, BARIS, KANAN)
    e.Graphics.DrawString(Format(DR("TOTAL"), "#,#"), FONTDATA, Brushes.Black, 575, BARIS, KANAN)
    NOMOR = NOMOR + 1
    BARIS = BARIS + 20
    TOTAL = TOTAL + DR("TOTAL")
End While
e.Graphics.DrawLine(Pens.Black, 30, BARIS + 5, 600, BARIS + 5)
e.Graphics.DrawString("TOTAL :", FONTDATA, Brushes.Black, 500, BARIS + 10, KANAN)
e.Graphics.DrawString(Format(TOTAL, "#,#"), FONTDATA, Brushes.Black, 575, BARIS + 10, KANAN)
e.Graphics.DrawLine(Pens.Black, 30, BARIS + 30, 600, BARIS + 30)
DR.Close()
End Sub

```

Mencetak data pada tabel DETAILJUALSEMU yang terdiri dari Nomor, Kode Barang, Nama Barang, Harga Jual, Jumlah dan Total. Variabel TOTAL untuk menghitung Total Penjualan PerFaktur.

```

Private Sub BTPREVIEW_Click(ByVal sender As
    PDJUAL.Print()
End Sub

```

Saat diklik tombol PREVIEW akan mencetak data Penjualan PerNota yang sudah didesain.

```

Private Sub BTCLOSE_Click(ByVal sender As System.Object,
    KONEKSI.Close()
    Close()
End Sub

```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENCETAK PENJUALAN PERNOTA

Cara mencetak data Penjualan PerNota :

- Klik droplist pada Nomor Nota
- Klik Nomor Nota yang datanya mau dicetak, data item barang akan tampil di DataGridView
- Klik tombol PREVIEW

KODEBRG	NAMABRG	HRGJUAL	JUMLAH	TOTAL
MNT001	MONITOR	160000	12	1920000
MOU001	MOUSE	30000	8	240000

LAPORAN PENJUALAN PERNOTA

NO. NOTA : NT003

KODE CUSTOMER : CS002

TANGGAL : 20/02/2020

NAMA CUSTOMER : SMK NUSA PUTRA

NOMOR	KODEBRG	NAMA BARANG	HRG JUAL	JUMLAH	TOTAL
1	MNT001	MONITOR	160.000	12	1.920.000
2	MOU001	MOUSE	30.000	8	240.000
					TOTAL : 2.160.000

11.3 TUTORIAL MENCETAK PENJUALAN PERNOTA



https://youtu.be/C_-CIDEyBx4

11.4 EVALUASI

Cetak Tebal data Total Penjualan pada laporan Penjualan PerNota.

LAPORAN PENJUALAN PERNOTA

NO. NOTA : NT003	KODE CUSTOMER : CS002				
TANGGAL : 20/02/2020	NAMA CUSTOMER : SMK NUSA PUTRA				
NOMOR	KODEBRG	NAMA BARANG	HRG JUAL	JUMLAH	TOTAL
1	MNT001	MONITOR	160.000	12	1.920.000
2	MOU001	MOUSE	30.000	8	240.000
					TOTAL : 2.160.000

11.5 REFERENSI

I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*”, Jakarta : Elex Media Komputindo.

Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.

Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.

WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB XII MENCETAK PENJUALAN PERPERIODE

12.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran Mencetak Penjualan PerPeriode yaitu mahasiswa dapat mendesain form untuk memfilter data Penjualan PerPeriode dan memahami kode program untuk mencetak data Penjualan PePeriode.

12.2 FORM PENJUAIAN PERPERIODE

A. DESAIN FORM PENJUALAN PERPERIODE

Langkah-langkah mendesain Form Penjualan PerPeriode sebagai berikut :

1. Double klik file PRINV.SLN untuk membuka project Inventory
2. Klik menu Project ⇒ Add Windows Form
3. Ketik nama form FORMCETAKJUALPERPERIODE
4. Klik tombol Add, Buat desain Form Penjualan PerPeriode

Gambar 12.1 Desain Form Penjualan PerPeriode

Daftar Objek pada Form Penjualan PerPeriode

Komponen	Properties	Isi
Label	Text	FORM Cetak Penjualan PerPeriode
	Text	MULAI TANGGAL
	Text	SAMPAI TANGGAL
ComboBox	Name	TXTMULAI
	Name	TXTSAMPAI

Button	Name	BTPREVIEW
	Text	PREVIEW
	Name	BTCLOSE
	Text	CLOSE
DataGridView	Name	DGVBELI
PrintDocument	Name	PDBELI

B. KODE PROGRAM FORM PENJUALAN PERPERIODE

Klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Imports System.Drawing.Printing
Public Class FORMCETAKJUALPERPERIODE
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DA As New OleDb.OleDbDataAdapter
    Public DT As New DataTable
    Dim DR As OleDbDataReader
    Dim BARIS As Integer
    Dim FONTJUDUL As Font = New Font("CALIBRI", 14, FontStyle.Bold, GraphicsUnit.Point)
    Dim FONTDATA As Font = New Font("CALIBRI", 12, FontStyle.Regular, GraphicsUnit.Point)
    Dim KIRI As StringFormat = New StringFormat
    Dim TENGAH As StringFormat = New StringFormat
    Dim KANAN As StringFormat = New StringFormat
    Dim NOMOR As Integer
    Dim TOTAL As Double
    Dim XMULAI As String
    Dim XSAMPAI As String
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DA untuk menampilkan data Pembelian PerFaktur, Variabel DR untuk membaca data dari suatu tabel. Variabel BARIS untuk menentukan nilai baris data, variabel NOMOR untuk Nomor Urut data, variabel FONTJUDUL untuk menentukan ukuran font judul lapotan, variabel FONTDATA untuk menentukan ukuran font data, variabel KIRI untuk perataan kiri, variabel TENGAH untuk perataan tengah dan variabel KANAN untuk perataan kanan, variabel XMULAI untuk tanggal awal periode, variabel XSAMPAI untuk tanggal akhir periode.


```

Private Sub FORMCETAKJUALPERPERIODE_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Load
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()
    |
    SQLSTR = "SELECT * FROM JUAL ORDER BY TANGGAL"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    TXTMULAI.Items.Clear()
    TXTSAMPALAI.Items.Clear()
    While DR.Read
        TXTMULAI.Items.Add(DR("TANGGAL"))
        TXTSAMPALAI.Items.Add(DR("TANGGAL"))
    End While
    DR.Close()
End Sub

```

Membuat koneksi ke database DBINV,MDB dan membuka koneksi database. Mengisi data pada ComboBox TXTMULAI dan TXTSAMPALAI dengan data Tanggal Pembelian dari tabel JUAL

```

Sub KOSONG()
    TXTMULAI.Text = ""
    TXTSAMPALAI.Text = ""
    TXTMULAI.Focus()
End Sub

```

```

Sub DGV()
    SQLSTR = "SELECT * FROM QUERYJUAL WHERE NOTA IS NULL"
    DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVJUAL.DataSource = DT
End Sub

```

```

Private Sub FORMCETAKBELIPERPERIODE_Activated(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Activated
    KOSONG()
    DGV()
End Sub

```

Sub KOSONG untuk mengosongkan ComboBox TXTMULAI dan TXTSAMPALAI. Sub DGV untuk menampilkan data pada query QUERYJUAL ke DataGridView DGVJUAL. Saat form aktif akan menjalankan Sub KOSONG dan Sub DGV.

```

Private Sub TXTMULAI_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles '
    If TXTSAMPAL.Text <> "" Then
        XMULAI = Strings.Mid(TXTMULAI.Text, 4, 2) & "/" & Strings.Left(TXTMULAI.Text, 2) & _
            "/" & Strings.Right(TXTMULAI.Text, 4)
        XSAMPAL = Strings.Mid(TXTSAMPAL.Text, 4, 2) & "/" & Strings.Left(TXTSAMPAL.Text, 2) & _
            "/" & Strings.Right(TXTSAMPAL.Text, 4)
        SQLSTR = "SELECT * FROM QUERYPENJUAL WHERE TANGGAL BETWEEN #" & XMULAI & "# AND #" & XSAMPAL & "#"
        DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
        DT.Clear()
        DA.Fill(DT)
        DGVJUAL.DataSource = DT
    End If
End Sub

```

Saat diklik Tanggal Mulai, jika data pada ComboBox TXTSAMPAL tidak kosong maka data Penjualan sesuai dengan periode yang telah ditentukan akan tampil di DataGridView DGVJUAL.

```

Private Sub TXTSAMPAL_SelectedValueChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles '
    If TXTMULAI.Text <> "" Then
        XMULAI = Strings.Mid(TXTMULAI.Text, 4, 2) & "/" & Strings.Left(TXTMULAI.Text, 2) & _
            "/" & Strings.Right(TXTMULAI.Text, 4)
        XSAMPAL = Strings.Mid(TXTSAMPAL.Text, 4, 2) & "/" & Strings.Left(TXTSAMPAL.Text, 2) & _
            "/" & Strings.Right(TXTSAMPAL.Text, 4)
        SQLSTR = "SELECT * FROM QUERYPENJUAL WHERE TANGGAL BETWEEN #" & XMULAI & "# AND #" & XSAMPAL & "#"
        DA = New OleDbDataAdapter(SQLSTR, KONEKSI)
        DT.Clear()
        DA.Fill(DT)
        DGVJUAL.DataSource = DT
    End If
End Sub

```

Saat diklik Tanggal Sampai, jika data pada ComboBox TXTMULAI tidak kosong maka data Penjualan sesuai dengan periode yang telah ditentukan akan tampil di DataGridView DGVJUAL.

```
Private Sub PDJUAL_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.PrintPageEventArgs) Handles PDJUAL
    KIRI.Alignment = StringAlignment.Near
    TENGAH.Alignment = StringAlignment.Center
    KANAN.Alignment = StringAlignment.Far
```

```
e.Graphics.DrawString("LAPORAN PENJUALAN PERPERIODE", FONTJUDUL, Brushes.Blue, 400, 25, TENGAH)
e.Graphics.DrawString("TANGGAL " & TXTMULAI.Text & " S/D " & TXTSAMPAI.Text, FONTJUDUL, Brushes.Blue, 400, 50, TENGAH)
e.Graphics.DrawLine(Pens.Black, 10, 95, 800, 95)
e.Graphics.DrawString("NO.", FONTDATA, Brushes.Black, 50, 100, KANAN)
e.Graphics.DrawString("NOTA", FONTDATA, Brushes.Black, 60, 100, KIRI)
e.Graphics.DrawString("TANGGAL", FONTDATA, Brushes.Black, 120, 100, KIRI)
e.Graphics.DrawString("NAMA CUSTOMER", FONTDATA, Brushes.Black, 210, 100, KIRI)
e.Graphics.DrawString("KODEBRG", FONTDATA, Brushes.Black, 350, 100, KIRI)
e.Graphics.DrawString("NAMA BARANG", FONTDATA, Brushes.Black, 435, 100, KIRI)
e.Graphics.DrawString("HRG JUAL", FONTDATA, Brushes.Black, 620, 100, KANAN)
e.Graphics.DrawString("JUMLAH", FONTDATA, Brushes.Black, 700, 100, KANAN)
e.Graphics.DrawString("TOTAL", FONTDATA, Brushes.Black, 775, 100, KANAN)
e.Graphics.DrawLine(Pens.Black, 10, 125, 800, 125)
```

Mendesain judul laporan Tanggal Periode data Penjualan dan judul kolom yang terdiri dari Nomor, Nomor Nota, Tanggal, Nama Customer, Kode Barang, Nama Barang, Harga Jual, Jumlah dan Total.

```
BARIS = 130
NOMOR = 1
TOTAL = 0
XMULAI = Strings.Mid(TXTMULAI.Text, 4, 2) & "/" & Strings.Left(TXTMULAI.Text, 2) & "/" & Strings.Right(TXTMULAI.Text, 4)
XSAMPAI = Strings.Mid(TXTSAMPAI.Text, 4, 2) & "/" & Strings.Left(TXTSAMPAI.Text, 2) & "/" & Strings.Right(TXTSAMPAI.Text, 4)
SQLSTR = "SELECT * FROM QUERYJUAL WHERE TANGGAL BETWEEN #" & XMULAI & "# AND #" & XSAMPAI & "#"
CMD = New OleDbCommand(SQLSTR, KONEKSI)
DR = CMD.ExecuteReader
While DR.Read
    e.Graphics.DrawString(NOMOR, FONTDATA, Brushes.Black, 50, BARIS, KANAN)
    e.Graphics.DrawString(DR("NOTA"), FONTDATA, Brushes.Black, 60, BARIS, KIRI)
    e.Graphics.DrawString(DR("TANGGAL"), FONTDATA, Brushes.Black, 120, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMACS"), FONTDATA, Brushes.Black, 210, BARIS, KIRI)
    e.Graphics.DrawString(DR("KODEBRG"), FONTDATA, Brushes.Black, 350, BARIS, KIRI)
    e.Graphics.DrawString(DR("NAMAHRG"), FONTDATA, Brushes.Black, 435, BARIS, KIRI)
    e.Graphics.DrawString(Format(DR("HRGJUAL"), "#.#"), FONTDATA, Brushes.Black, 620, BARIS, KANAN)
    e.Graphics.DrawString(DR("JUMLAH"), FONTDATA, Brushes.Black, 675, BARIS, KANAN)
    e.Graphics.DrawString(Format(DR("TOTAL"), "#.#"), FONTDATA, Brushes.Black, 775, BARIS, KANAN)
    NOMOR = NOMOR + 1
    BARIS = BARIS + 20
    TOTAL = TOTAL + DR("TOTAL")
End While
e.Graphics.DrawLine(Pens.Black, 10, BARIS + 5, 800, BARIS + 5)
e.Graphics.DrawString("TOTAL .:", FONTDATA, Brushes.Black, 675, BARIS + 10, KANAN)
e.Graphics.DrawString(Format(TOTAL, "#.#"), FONTDATA, Brushes.Black, 775, BARIS + 10, KANAN)
e.Graphics.DrawLine(Pens.Black, 10, BARIS + 30, 800, BARIS + 30)
DR.Close()
End Sub
```

Mencetak data Penjualan sesuai dengan Periode yang ditentukan terdiri dari Nomor, Nomor Nota, Tanggal, Nama Customer, Kode Barang, Nama Barang, Harga Jual, Jumlah dan Total. Variabel TOTAL untuk menghitung Total Penjualan PerPeriode.

```
Private Sub BTPREVIEW_Click(ByVal sender As Object
    PDJUAL.Print()
End Sub
```

Saat diklik tombol PREVIEW akan mencetak data Pembelian PerPeriode yang sudah didesain.

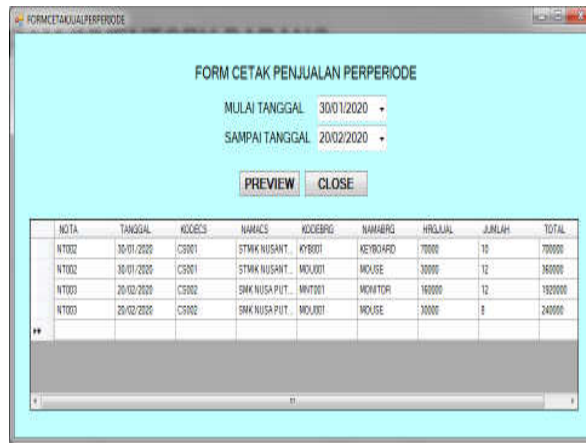
```
Private Sub BTCLOSE_Click(ByVal sender As System.Object,
    KONEKSI.Close()
    Close()
End Sub
```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENCETAK PENJUALAN PERPERIODE

Cara mencetak data Penjualan PerPeriode :

- Klik droplist pada Mulai Tanggal, klik Tanggal Mulai Periode
- Klik droplist pada Sampai Tanggal, klik Tanggal Akhir Periode, data item barang akan tampil di DataGridView
- Klik tombol PREVIEW



**LAPORAN PENJUALAN PERPERIODE
TANGGAL 30/01/2020 S/D 20/02/2020**

NO.	NOTA	TANGGAL	NAMA CUSTOMER	KODEBRG	NAMA BARANG	HRG JUAL	JUMLAH	TOTAL
1	NT002	30/01/2020	STMIK NUSANTARA	KYB001	KEYBOARD	70.000	10	700.000
2	NT002	30/01/2020	STMIK NUSANTARA	MOU001	MOUSE	30.000	12	360.000
3	NT003	20/02/2020	SMK NUSA PUTRA	MNT001	MONITOR	160.000	12	1.920.000
4	NT003	20/02/2020	SMK NUSA PUTRA	MOU001	MOUSE	30.000	8	240.000
TOTAL :								3.220.000

12.3 TUTORIAL MENCETAK PENJUALAN PERPERIODE



<https://youtu.be/iMW0AhYlWpg>

12.4 EVALUASI

Tampilkan data Tanggal pada judul laporan seperti berikut

LAPORAN PENJUALAN PERPERIODE TANGGAL 30 Januari 2020 S/D 20 Februari 2020

NO.	NOTA	TANGGAL	NAMA CUSTOMER	KODEBRG	NAMA BARANG	HRG JUAL	JUMLAH	TOTAL
1	NT002	30/01/2020	STMIK NUSANTARA	KYB001	KEYBOARD	70.000	10	700.000
2	NT002	30/01/2020	STMIK NUSANTARA	MOU001	MOUSE	30.000	12	360.000
3	NT003	20/02/2020	SMK NUSA PUTRA	MNT001	MONITOR	160.000	12	1.920.000
4	NT003	20/02/2020	SMK NUSA PUTRA	MOU001	MOUSE	30.000	8	240.000
TOTAL :								3.220.000

12.5 REFERENSI

I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.

Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.

Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.

WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB XIII FORM DATA PEMAKAI

13.1 CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari Form Data Pemakai yaitu mahasiswa memahami struktur tabel Pemakai, dapat mendesain Form Data Pemakai, memahami kode program yang digunakan dan cara mengoperasikan Form Data Pemakai yang meliputi menambah data baru, mengedit data dan menghapus data.

13.2 FORM DATA PEMAKAI

A. MENDESAIN FORM DATA PEMAKAI

Langkah-langkah mendesain Form Data Pemakai sebagai berikut :

1. Buka file database DBINV.MDB
2. Buat struktur tabel berikut

	Field Name	Data Type	
?	KODEPK	Text	Field Size : 5
	NAMAPK	Text	Field Size : 20
	JABATAN	Text	Field Size : 12
	PASSWORD	Text	Field Size : 10
	FOTO	Text	Field Size : 100

3. Simpan struktur tabel dengan nama PEMAKAI
4. Tutup program MS Access
5. Double klik file PRINV.SLN untuk membuka project Inventory
6. Klik menu Project ⇒ Add Windows Form
7. Ketik nama form FORMPEMAKAI
8. Klik tombol Add, buat Desain Form Pemakai berikut

The image shows a screenshot of a Windows application window titled "FORMPEMAKAI". The window contains a form titled "FORM DATA PEMAKAI". The form has four input fields: "KODE PEMAKAI", "NAMA PEMAKAI", "JABATAN" (a dropdown menu), and "PASSWORD". To the right of these fields is a large empty rectangular box. Below the input fields are four buttons: "SIMPAN", "HAPUS", "BLANKFORM", and "CLOSE". At the bottom of the form is a large grey rectangular area. The window title bar shows "FORMPEMAKAI" and standard Windows window controls. The taskbar at the bottom shows "OFD".

Gambar 13.1 Desain Form Data Pemakai

Daftar Objek pada Form Data Pemakai

Komponen	Properties	Isi
Label	Text	FORM DATA PEMAKAI
	Text	KODE PEMAKAI
	Text	NAMA PEMAKAI
	Text	JABATAN
	Text	PASSWORD
TextBox	Name	TXTKODEPK
	Name	TXTNAMAPK
	Name	TXTPASSWORD
	PasswordChar	*
ComboBox	Name	CBJABATAN
	Items	MANAGER ADMINISTRASI KEUANGAN
PictureBox	Name	PBFOTO
	SizeMode	StretchImage
Button	Name	BTSIMPAN
	Text	SIMPAN
	Name	BTHAPUS
	Text	HAPUS
	Name	BTBLANKFORM
	Text	BLANKFORM
	Name	BTCLOSE
Text	CLOSE	
DataGridView	Name	DGVPEMAKAI
OpenFileDialog	Name	OFD

B. KODE PROGRAM FORM DATA PEMAKAI

Klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMPEMAKAI
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DA As New OleDb.OleDbDataAdapter
    Public DT As New DataTable
    Public XFOTO As String
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DA untuk penghubung data tabel, variabel DT menyimpan data hasil perintah query, variabel XFOTO untuk menyimpan nama file foto.

```
Private Sub FORMPEMAKAI_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()
End Sub
```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi.

```
Sub KOSONG()
    TXTKODEPK.Text = ""
    TXTNAMAPK.Text = ""
    CBJABATAN.Text = ""
    TXTPASSWORD.Text = ""
    PBFOTO.Image = Nothing
    TXTKODEPK.Focus()
End Sub
Sub DGV()
    SQLSTR = "SELECT * FROM PEMAKAI"
    DA = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVPEMAKAI.DataSource = DT
End Sub
Private Sub FORMPEMAKAI_Activated(ByVal sender As Ob:
    KOSONG()
    DGV()
End Sub
```

Sub KOSONG untuk mengosongkan Textbox input data dan foto Pemakai, sub DGV untuk menampilkan data Pemakai ke DataGridView DGVPEMAKAI. Saat form aktif akan menjalankan sub KOSONG dan sub DGV.

```
Private Sub PBFOTO_Click(ByVal sender As System.
    OFD.ShowDialog()
    PBFOTO.Image = Image.FromFile(OFD.FileName)
    XFOTO = OFD.FileName
End Sub
```

Saat PictureBox PBFOTO diklik akan menampilkan window Open untuk memilih file gambar yang akan ditampilkan di PictureBox, nama file foto disimpan ke variabel XFOTO.


```

Private Sub TXTKODEPK_Leave(ByVal sender As Object, ByVal e As System.EventArgs) Handles TXTKODEPK.Leave
    SQLSTR = "SELECT * FROM PEMAKAI WHERE KODEPK='" & TXTKODEPK.Text & "'"
    DA = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVPEMAKAI.DataSource = DT
    If DT.Rows.Count > 0 Then
        TXTNAMAPK.Text = DGVPEMAKAI.Rows.Item(DGVPEMAKAI.CurrentRow.Index).Cells(1).Value
        CBJABATAN.Text = DGVPEMAKAI.Rows.Item(DGVPEMAKAI.CurrentRow.Index).Cells(2).Value
        TXTPASSWORD.Text = DGVPEMAKAI.Rows.Item(DGVPEMAKAI.CurrentRow.Index).Cells(3).Value
        PBFOTO.Image = Image.FromFile(DGVPEMAKAI.Rows.Item(DGVPEMAKAI.CurrentRow.Index).Cells(4).Value)
        XFOTO = DGVPEMAKAI.Rows.Item(DGVPEMAKAI.CurrentRow.Index).Cells(4).Value
    End If
    DGV()
End Sub

```

Saat kursor meninggalkan Textbox TXTKODEPK akan dideteksi apakah Kode Pemakai yang telah diketik sudah ada di tabel Pemakai, jika Kode Pemakai sudah ada maka datanya akan ditampilkan ke masing-masing Textbox.

```

Private Sub BHSIMPAN_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BHSIMPAN.Click
    SQLSTR = "SELECT * FROM PEMAKAI WHERE KODEPK='" & TXTKODEPK.Text & "'"
    DA = New OleDb.OleDbDataAdapter(SQLSTR, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVPEMAKAI.DataSource = DT
    If DT.Rows.Count > 0 Then
        SQLSTR = "UPDATE PEMAKAI SET KODEPK='" & TXTKODEPK.Text & "',NAMAPK='" & _
            TXTNAMAPK.Text & "',JABATAN='" & CBJABATAN.Text & "',PASSWORD='" & TXTPASSWORD.Text & _
            "',FOTO='" & XFOTO & "' WHERE KODEPK='" & TXTKODEPK.Text & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    Else
        SQLSTR = "INSERT INTO PEMAKAI VALUES ('" & TXTKODEPK.Text & "','" & TXTNAMAPK.Text & _
            "','" & CBJABATAN.Text & "','" & TXTPASSWORD.Text & "','" & XFOTO & "')"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    KOSONG()
    DGV()
End Sub

```

Saat diklik tombol SIMPAN akan dideteksi apakah Kode Pemakai yang akan disimpan sudah ada di tabel Pemakai, jika Kode Pemakai sudah ada maka data Pemakai akan diupdate sedangkan jika belum ada maka data Pemakai yang diinput akan ditambahkan sebagai data baru.

```

Private Sub BTHAPUS_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles BTHAPUS.Click
    Dim X As String
    X = MsgBox("DATA PEMAKAI " & TXTNAMAPK.Text & " JADI DIHAPUS!", MsgBoxStyle.YesNo)
    If X = vbYes Then
        SQLSTR = "DELETE FROM PEMAKAI WHERE KODEPK='" & TXTKODEPK.Text & "'"
        CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
        CMD.ExecuteNonQuery()
    End If
    KOSONG()
    DGV()
End Sub

```

Saat diklik tombol HAPUS akan ditampilkan konfirmasi apakah data Pemakai yang sedang tampil akan dihapus, jika diklik tombol Yes maka data Pemakai yang sedang tampil akan dihapus sedangkan jika diklik tombol No data Pemakai tidak akan dihapus.

```
Private Sub BTBLANKFORM_Click(ByVal sender  
    KOSONG()  
End Sub
```

Saat diklik tombol BLANKFORM maka akan menjalankan sub BLANKFORM untuk mengosongkan Textbox.

```
Private Sub BTCLOSE_Click(ByVal sender As Obje  
    KONEKSI.Close()  
    Close()  
End Sub
```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENGOPERASIKAN FORM DATA PEMAKAI

1. Menambah data Pemakai

Cara menambahkan data Pemakai :

- Klik menu Debug => Start Debugging
- Isikan data Kode Pemakai, Nama Pemakai, Jabatan, Foto dan Password Pemakai
- Klik tombol SIMPAN untuk menyimpan data Pemakai

2. Mengedit data Pemakai

Cara mengedit data Pemakai :

- Klik menu Debug => Start Debugging
- Isikan Kode Pemakai yang datanya mau diedit
- Tekan tombol Tab (data Pemakai yang kodenya sesuai akan tampil)
- Lakukan pengeditan data
- Klik tombol SIMPAN untuk pengeditan data Pemakai

3. Menghapus data Pemakai

Cara menghapus data Pemakai :

- Klik menu Debug => Start Debugging
- Isikan Kode Pemakai yang datanya mau dihapus
- Tekan tombol Tab (data Pemakai yang sesuai akan tampil)
- Klik tombol HAPUS
- Tampil konfirmasi data jadi dihapus atau tidak
- Klik tombol Yes jika data jadi dihapus atau klik tombol No jika data tidak jadi dihapus

13.3 TOTURIAL FORM PEMAKAI



<https://youtu.be/Zf5VeJZxcio>

13.4 EVALUASI

Tampilkan Data Password pada DataGridView dengan karakter * supaya tidak diketahui Passwordnya.

The screenshot shows a Windows application window titled 'FORMPEMAKAI'. The main form is titled 'FORM DATA PEMAKAI' and contains the following fields and controls:

- KODE PEMAKAI:
- NAMA PEMAKAI:
- JABATAN:
- PASSWORD:
- A photo placeholder box on the right.
- Buttons: SIMPAN, HAPUS, BLANKFORM, CLOSE.

Below the form is a DataGridView table with the following data:

	KODEPK	NAMAPK	JABATAN	PASWORD	FOTO
	PK001	RIZKI ATMAJA	MANAGER	****	G:\KULIAH ONLI...
	PK002	LINA	KEUANGAN	****	G:\KULIAH ONLI...
	PK003	ADI	ADMINISTRASI	***	G:\KULIAH ONLI...
▶▶					

13.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB XIV FORM LOGIN PEMAKAI

14.1 CAPAIAN PEMBELAJARAN

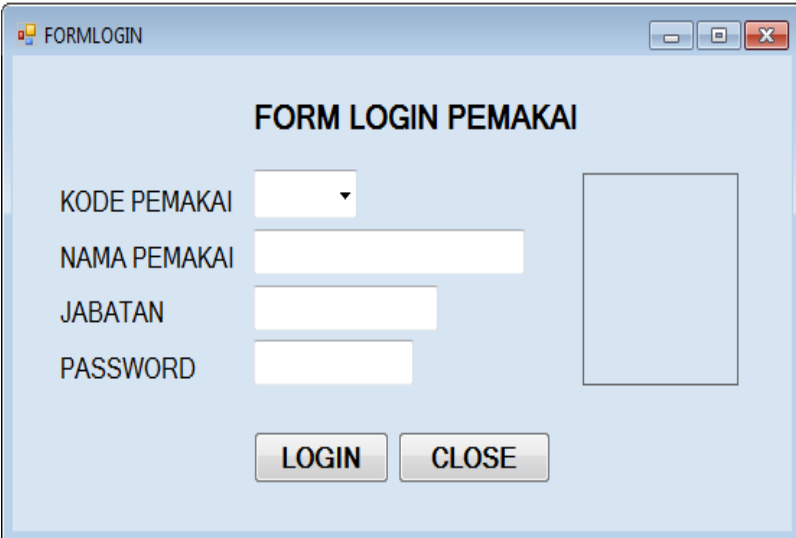
Capaian pembelajaran dari Form Login Pemakai yaitu mahasiswa dapat mendesain Form Login Pemakai untuk mengatur hak akses Pemakai sesuai jabatannya, memahami kode program yang digunakan dan cara mengoperasikan Form Login Pemakai.

14.2 FORM LOGIN PEMAKAI

A. MENDESAIN FORM LOGIN PEMAKAI

Langkah-langkah mendesain Form Login Pemakai sebagai berikut :

1. Double klik file PRINV.SLN untuk membuka project Inventory
2. Klik menu Project ⇒ Add Windows Form untuk menambahkan form baru
3. Ketik nama form FORMLOGIN
4. Klik tombol Add
5. Buat Desain Form Login Pemakai berikut



The image shows a screenshot of a Windows application window titled "FORMLOGIN". The window contains a form titled "FORM LOGIN PEMAKAI". The form has four input fields: "KODE PEMAKAI" (a dropdown menu), "NAMA PEMAKAI", "JABATAN", and "PASSWORD". To the right of these fields is a large empty rectangular box. At the bottom of the form are two buttons labeled "LOGIN" and "CLOSE".

Gambar 14.1 Desain Form Login Pemakai

Daftar Objek pada Form Login Pemakai

Komponen	Properties	Isi
Label	Text	FORM LOGIN PEMAKAI
	Text	KODE PEMAKAI
	Text	NAMA PEMAKAI
	Text	JABATAN
	Text	PASSWORD
ComboBox	Name	CBKODEPK
TextBox	Name	TXTNAMAPK
	Name	TXTJABATAN
	Name	TXTPASSWORD
	PasswordChar	*
PictureBox	Name	PBFOTO
	SizeMode	StretchImage
Button	Name	BTLOGIN
	Text	LOGIN
	Name	BTCLOSE
	Text	CLOSE

B. KODE PROGRAM FORM LOGIN PEMAKAI

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMLOGIN
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DA As New OleDb.OleDbDataAdapter
    Public DT As New DataTable
    Public DR As OleDbDataReader
    Public XKODEPK As String
    Public XPASSWORD As String
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DA untuk penghubung data tabel, variabel DT menyimpan data hasil perintah query, variabel XKODEPK untuk menyimpan Kode Pemakai yang diklik, variabel XPASSWORD untuk menyimpan teks Password yang diinput.

```

Private Sub FORMLOGIN_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()

    SQLSTR = "SELECT * FROM PEMAKAI"
    CMD = New OleDb.OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    CBKODEPK.Items.Clear()
    While DR.Read
        CBKODEPK.Items.Add(DR("KODEPK") & " " & DR("NAMAPK"))
    End While
    DR.Close()
End Sub

```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi. ComboBox CBKODEPK akan diisi dengan data Kode Pemakai dan Nama Pemakai dari tabel Pemakai.

```

Sub KOSONG()
    CBKODEPK.Text = ""
    TXTNAMAPK.Text = ""
    TXTJABATAN.Text = ""
    TXTPASSWORD.Text = ""
    PBFOTO.Image = Nothing
    CBKODEPK.Focus()
End Sub
Private Sub FORMLOGIN_Activated
    KOSONG()
End Sub

```

Sub KOSONG untuk mengosongkan ComboBox, Textbox dan foto Pemakai. Saat form aktif akan menjalankan sub KOSONG.

```

Private Sub CBKODEPK_SelectedValueChanged(ByVal sender As Object, By
    XKODEPK = Microsoft.VisualBasic.Left(CBKODEPK.Text, 5)
    SQLSTR = "SELECT * FROM PEMAKAI WHERE KODEPK='" & XKODEPK & "'"
    CMD = New OleDbCommand(SQLSTR, KONEKSI)
    DR = CMD.ExecuteReader
    While DR.Read
        TXTNAMAPK.Text = DR("NAMAPK")
        TXTJABATAN.Text = DR("JABATAN")
        XPASSWORD = DR("PASSWORD")
        PBFOTO.Image = Image.FromFile(DR("FOTO"))
    End While
    DR.Close()
    TXTPASSWORD.Focus()
End Sub

```

Saat diklik salah satu Kode Pemakai maka akan dicari Kode Pemakai yang diklik pada tabel Pemakai lalu ditampilkan data Nama Pemakai, Jabatan dan Foto Pemakai. Password Pemakai disimpan ke variabel XPASSWORD dan kursor ditempatkan ke TextBox TXTPASSWORD untuk mengisi Password Pemakai.

```

Private Sub BTLOGIN_Click(ByVal sender As System.Object, ByVal e As EventArgs)
    If TXTPASSWORD.Text = XPASSWORD Then
        Select Case TXTJABATAN.Text
            Case "MANAGER"
                MENUUTAMA.MASTERToolStripMenuItem.Enabled = False
                MENUUTAMA.TRANSAKSIToolStripMenuItem.Enabled = False
                MENUUTAMA.CETAKToolStripMenuItem.Enabled = True
                MENUUTAMA.SELESAIToolStripMenuItem.Enabled = True
            Case "KEUANGAN"
                MENUUTAMA.MASTERToolStripMenuItem.Enabled = False
                MENUUTAMA.TRANSAKSIToolStripMenuItem.Enabled = True
                MENUUTAMA.CETAKToolStripMenuItem.Enabled = True
                MENUUTAMA.SELESAIToolStripMenuItem.Enabled = True
            Case "ADMINISTRASI"
                MENUUTAMA.MASTERToolStripMenuItem.Enabled = True
                MENUUTAMA.TRANSAKSIToolStripMenuItem.Enabled = False
                MENUUTAMA.CETAKToolStripMenuItem.Enabled = True
                MENUUTAMA.SELESAIToolStripMenuItem.Enabled = True
        End Select
        MENUUTAMA.ShowDialog()
    Else
        Dim X As String
        X = MsgBox("PASSWORD SALAH !", vbOKOnly)
        TXTPASSWORD.Text = ""
        TXTPASSWORD.Focus()
    End If
End Sub

```

Saat diklik tombol LOGIN akan dideteksi apakah Password yang diinput sama dengan Password Pemakai yang disimpan di variabel XPASSWORD, jika Password sama maka akan diatur Menu yang diaktifkan sesuai dengan Jabatan Pemakai lalu ditampilkan Form Menu Utama, jika Password salah tampilkan pesan Password Salah dan diberikan kesempatan untuk input Password kembali.

```

Private Sub BTCLOSE_Click(ByVal sender As Object, ByVal e As EventArgs)
    KONEKSI.Close()
    Close()
End Sub

```

Saat diklik tombol CLOSE akan menutup koneksi database dan menutup form.

C. MENGOPERASIKAN FORM LOGIN PEMAKAI

Cara menambahkan data Pemakai :

- a. Klik droplist pada ComboBox Kode Pemakai
- b. Klik salah satu Kode Pemakai, data Nama Pemakai, Jabatan dan Foto Pemakai otomatis tampil

- c. Isikan Password Pemakai
- d. Klik tombol LOGIN, jika Password yang diisikan benar akan tampil form Menu Utama sedangkan jika Password salah akan tampil pesan Password Salah dan diberi kesempatan mengisi Password kembali.

14.3 TUTORIAL LOGIN PEMAKAI



<https://youtu.be/E6kkgZ1QxLo>

14.4 EVALUASI

Tambahkan kode program untuk mendeteksi jika input Password salah 3 kali tampilkan pesan Password sudah 3 Kali Salah dan form Login Pemakai ditutup.

14.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

BAB XV FORM HAK AKSES

15.1 CAPAIAN PEMBELAJARAN

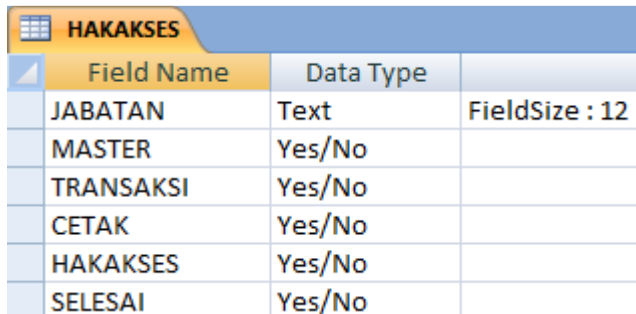
Capaian pembelajaran dari Form Hak Akses yaitu mahasiswa memahami struktur tabel Hak Akses, dapat mendesain Form Hak Akses, memahami kode program yang digunakan dan cara mengatur Hak Akses Pemakai menggunakan Form Hak Akses.

15.2 FORM HAK AKSES

A. MENDESAIN FORM HAK AKSES

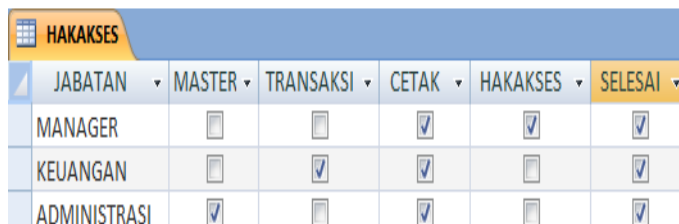
Langkah-langkah mendesain Form Hak Akses sebagai berikut :

1. Buka file database DBINV.MDB
2. Buat struktur tabel berikut



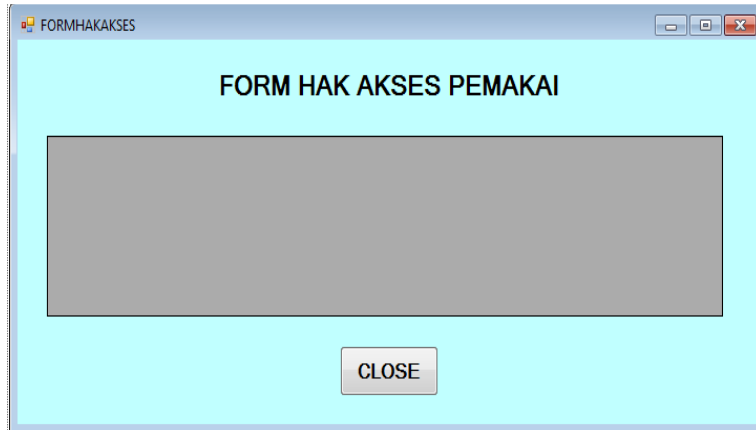
Field Name	Data Type	FieldSize
JABATAN	Text	FieldSize : 12
MASTER	Yes/No	
TRANSAKSI	Yes/No	
CETAK	Yes/No	
HAKAKSES	Yes/No	
SELESAI	Yes/No	

3. Simpan struktur tabel dengan nama HAKAKSES
4. Klik ikon View, isikan data Hak Akses berikut



JABATAN	MASTER	TRANSAKSI	CETAK	HAKAKSES	SELESAI
MANAGER	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
KEUANGAN	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
ADMINISTRASI	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

5. Tutup program MS Access
6. Double klik file PRINV.SLN untuk membuka project Inventory
7. Klik menu Project ⇒ Add Windows Form untuk menambahkan form baru
8. Ketik nama form FORMHAKAKSES
9. Klik tombol Add
10. Buat Desain Form Hak Akses berikut



Gambar 15.1 Desain Form Hak Akses

Daftar Objek pada Form Hak Akses

Komponen	Properties	Isi
Label	Text	FORM HAK AKSES PEMAKAI
DataGridView	Name	DGVAKSES
Button	Name	BTCLOSE
	Text	CLOSE

B. KODE PROGRAM FORM HAK AKSES

Langkah berikutnya klik menu View => Code lalu ketik kode program berikut :

```
Imports System.Data
Imports System.Data.OleDb
Public Class FORMHAKAKSES
    Public KONEKSISTRING As String
    Public KONEKSI As New OleDb.OleDbConnection
    Public SQLSTR As String
    Public CMD As New OleDb.OleDbCommand
    Public DA As New OleDb.OleDbDataAdapter
    Public DT As New DataTable
```

Perintah Public untuk membuat variabel Publik supaya variabel tersebut dikenal oleh semua sub program. Variabel KONEKSISTRING berisi string untuk koneksi ke database, variabel KONEKSI untuk membuat koneksi ke database, variabel SQLSTR untuk perintah Query, variabel CMD untuk menjalankan perintah Query, variabel DA untuk penghubung data tabel, variabel DT menyimpan data hasil perintah query..

```

Private Sub FORMHAKAKSES_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    KONEKSISTRING = "PROVIDER = MICROSOFT.JET.OLEDB.4.0;DATA SOURCE = DBINV.MDB"
    KONEKSI = New OleDb.OleDbConnection(KONEKSISTRING)
    KONEKSI.Open()
End Sub

```

Saat form dijalankan akan dilakukan koneksi ke database DBINV.MDB dan membuka koneksi.

```

Private Sub FORMHAKAKSES_Activated(ByVal sender As Object, ByVal e As System.EventArgs)
    sqlstr = "SELECT * FROM HAKAKSES"
    DA = New OleDbDataAdapter(sqlstr, KONEKSI)
    DT.Clear()
    DA.Fill(DT)
    DGVAKSES.DataSource = DT
End Sub

```

Saat form aktif, tampilkan data Hak Akses ke DataGridView DGVAKSES.

```

Private Sub BTCLOSE_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles BTCLOSE.Click
    Dim A As Integer
    For A = 1 To DGVAKSES.Rows.Count - 1
        SQLSTR = "UPDATE HAKAKSES SET MASTER=" & _
            DGVAKSES.Rows.Item(DGVAKSES.CurrentRow.Index).Cells(1).Value & _
            ",TRANSAKSI=" & DGVAKSES.Rows.Item(DGVAKSES.CurrentRow.Index).Cells(2).Value & _
            ",CETAK=" & DGVAKSES.Rows.Item(DGVAKSES.CurrentRow.Index).Cells(3).Value & _
            ",HAKAKSES=" & DGVAKSES.Rows.Item(DGVAKSES.CurrentRow.Index).Cells(4).Value & _
            ",SELESAI=" & DGVAKSES.Rows.Item(DGVAKSES.CurrentRow.Index).Cells(5).Value & _
            " WHERE JABATAN=" & DGVAKSES.Rows.Item(DGVAKSES.CurrentRow.Index).Cells(0).Value & ""
        CMD = New OleDb.OleDbCommand(sqlstr, KONEKSI)
        CMD.ExecuteNonQuery()
    Next
    KONEKSI.Close()
    Close()
End Sub

```

Saat diklik tombol CLOSE akan mengupdate data pada tabel HakAkses lalu menutup koneksi database dan menutup form.

Double klik FORMLOGIN, tambahkan kode program berikut :

```

Private Sub BTLOGIN_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    If TXTPASSWORD.Text = XPASSWORD Then
        SQLSTR = "SELECT * FROM HAKAKSES WHERE JABATAN='" & TXTJABATAN.Text & "'"
        CMD = New OleDbCommand(SQLSTR, KONEKSI)
        DR = CMD.ExecuteReader
        While DR.Read
            If DR("MASTER") = True Then
                MENUUTAMA.MASTERToolStripMenuItem.Enabled = True
            Else
                MENUUTAMA.MASTERToolStripMenuItem.Enabled = False
            End If
            If DR("TRANSAKSI") = True Then
                MENUUTAMA.TRANSAKSIToolStripMenuItem.Enabled = True
            Else
                MENUUTAMA.TRANSAKSIToolStripMenuItem.Enabled = False
            End If
            If DR("CETAK") = True Then
                MENUUTAMA.CETAKToolStripMenuItem.Enabled = True
            Else
                MENUUTAMA.CETAKToolStripMenuItem.Enabled = False
            End If
            If DR("HAKAKSES") = True Then
                MENUUTAMA.HAKAKSESToolStripMenuItem.Enabled = True
            Else
                MENUUTAMA.HAKAKSESToolStripMenuItem.Enabled = False
            End If
            If DR("SELESAI") = True Then
                MENUUTAMA.SELESAIToolStripMenuItem.Enabled = True
            Else
                MENUUTAMA.SELESAIToolStripMenuItem.Enabled = False
            End If
        End While
        DR.Close()
        MENUUTAMA.ShowDialog()
    Else
        Dim X As String
        X = MsgBox("PASSWORD SALAH !", vbOKOnly)
        TXTPASSWORD.Text = ""
        TXTPASSWORD.Focus()
    End If
End Sub

```

C. MENGOPERASIKAN FORM HAK AKSES

Cara menambahkan data Pemakai :

- a. Klik menu Debug => Start Debugging
- b. Login dengan jabatan MANAGER
- c. Klik menu Hak Akses => Hak Akses
- d. Lakukan perubahan Hak Akses

- e. Klik tombol CLOSE
- f. Klik menu Selesai => Keluar Dari Program
- g. Login dengan jabatan selain MANAGER, perhatikan perubahan Hak Akses

15.3 TUTORIAL HAK AKSES



<https://youtu.be/Ug173J1cpj4>

15.4 EVALUASI

Buat kode program pada Form Pemakai, jika ada Jabatan baru maka akan otomatis menambah data Jabatan pada tabel Hak Akses

15.5 REFERENSI

- I Gusti Ngurah Suryantara, (2014), *Merancang Aplikasi dengan VB .Net 2013*, Jakarta : Elex Media Komputindo.
- Uus Rusmawan, (2013), *Cara Cepat Belajar VB .Net*, Jakarta : Elex Media Komputindo.
- Jubilee Enterprise, (2014), *Visual Basic 2013 untuk Pemula*, Jakarta : Elex Media Komputindo.
- WahanaKomputer, ST, (2013), *Shortcourse Series : Visual Basic 2012*, Yogyakarta : Andi Publisher.

Visual Basic .NET merupakan bahasa pemrograman komputer berorientasi objek yang diimplementasikan pada .NET Framework. Visual Basic .Net merupakan pengembangan dari bahasa pemrograman Visual Basic 6.

Buku ini berisi contoh aplikasi Inventory Barang yang terdiri dari Form Barang untuk menyimpan data Barang, Form Suplier untuk menyimpan data Suplier, Form Pembelian untuk menyimpan transaksi Pembelian, Form Customer untuk menyimpan data Customer dan Form Pembelian untuk menyimpan data Pembelian. Pembuatan laporan tidak menggunakan program bantu tetapi menggunakan komponen *Print Document* yang sudah ada di Visual Basic .Net, laporan yang dibuat yaitu laporan data Barang, data Suplier, data Customer, data Pembelian PerFaktur, data Pembelian PerPeriode dan data Penjualan PerNota. Form Pemakai, form Login dan form Hak Akses digunakan untuk mengatur hak akses pemakai dalam mengakses form dan laporan yang ada di aplikasi Inventory. Untuk lebih memahami materi, pada buku ini terdapat *link* video pembelajaran untuk masing-masing materi.

Aplikasi Inventory Barang Dengan Visual Basic .Net



YAYASAN PRIMA AGUS TEKNIK

ISBN 978-623-6141-34-2 (PDF)

