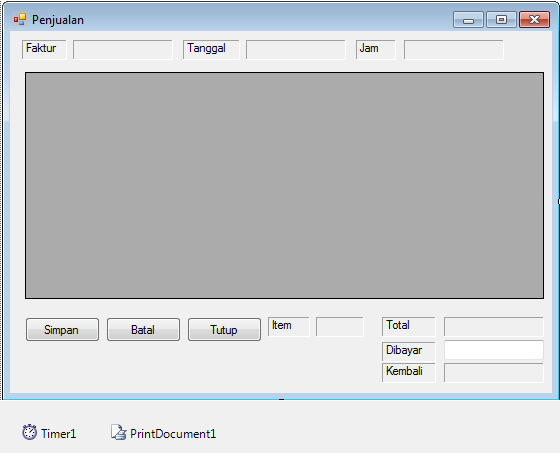
**LATIHAN GRIDVIEW-PRINTER**



|  |  |  |
| --- | --- | --- |
| **Object** | **Properties** | |
| **Name** | **Text** |
| Form | FormCetak.vb | Cetak |
| Label1 | Label1 | Faktur |
| Label2 | Faktur |  |
| Label3 | Label3 | Tanggal |
| Label4 | Tanggal |  |
| Label5 | Label5 | Jam |
| Label6 | Jam |  |
| Label7 | Label7 | Item |
| Label8 | Item |  |
| Label9 | Label9 | Total |
| Label10 | Total |  |
| Label11 | Label11 | Dibayar |
| TextBox1 | Dibayar |  |
| Label13 | Label13 | Kembali |
| Label14 | Kembali |  |
|  |  |  |
| Button1 | BTNSimpan | Simpan |
| Button2 | BTNBatal | Batal |
| Button3 | BTNTutup | Tutup |
|  |  |  |
| Timer | Timer1 |  |
| PrintDocument | PrintDocument1 |  |
|  | | |
| Class | RawPrinterHelper.vb |  |
| Modul | Sistem\_Module1.vb |  |
|  |  |  |
| DataGridView | DGV |  |
|  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Database** | **Nama Table** | **Field Name** | **Data Type** | **Field Size** |
| datatransaksi | Barang | Kode\_Barang | Text | 13 |
| Nama\_barang | Text | 30 |
| Harga\_beli | Number | Long Integer |
| Harga\_Jual | Number | Long Integer |
| Jumlah\_Barang | Number | Integer |
| Satuan | Text | 15 |
|  |  |  |  |
| DetailJual | Faktur | Text | 10 |
| Kode\_Barang | Text | 13 |
| Nama\_Barang | Text | 30 |
| Harga\_Jual | Number | Long Integer |
| Jumlah | Number | Integer |
| Subtotal | Number | Long Integer |
|  |  |  |  |
| Penjualan | Faktur | Text | 10 |
| Tanggal | Date/Time |  |
| Item | Number | Integer |
| Total | Number | Long Integer |
| Dibayar | Number | Long Integer |
| Kembali | Number | Long Integer |
| KodePtg | Text | 5 |

Form: Penjualan

Imports System.Data.OleDb

Imports System.Windows.Forms

Imports System.Drawing.Printing

Public Class Penjualan

Sub BersihkanText()

Total.Text = ""

Dibayar.Text = ""

Kembali.Text = ""

Item.Text = ""

End Sub

Sub BuatKolomBaru()

DGV.Columns.Add("Kode", "Kode")

DGV.Columns.Add("Nama", "Nama Barang")

DGV.Columns.Add("Harga", "Harga")

DGV.Columns.Add("Jumlah", "Jumlah")

DGV.Columns.Add("Total", "Total")

Call AturLebarKolom()

End Sub

Sub AturLebarKolom()

DGV.Columns(0).Width = 75

DGV.Columns(1).Width = 150

DGV.Columns(2).Width = 75

DGV.Columns(3).Width = 75

DGV.Columns(4).Width = 100

End Sub

Sub FakturOtomatis()

strSql = "Select \* from penjualan where faktur in (select max(faktur) from penjualan) order by faktur desc"

Dim urutan As String

Dim hitung As Long

If Not Valid\_table(strSql) Then

urutan = Format(Now, "yyMMdd") + "0001"

Else

hitung = Posisi\_Baris(strSql) + 1

urutan = Format(Now, "yyMMdd") + Microsoft.VisualBasic.Right("0000" & hitung, 4)

End If

Faktur.Text = urutan

End Sub

Private Sub Penjualan1\_Activated(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Activated

Call FakturOtomatis()

Tanggal.Text = Today

End Sub

Private Sub Penjualan1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

Call Koneksi()

Call BuatKolomBaru()

End Sub

Private Sub Timer1\_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick

Jam.Text = TimeOfDay

End Sub

Private Sub BTNBatal\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BTNBatal.Click

Call BersihkanText()

DGV.Columns.Clear()

Call BuatKolomBaru()

DGV.Focus()

End Sub

Sub kena(ByVal myGrid As DataGrid)

' Set the current cell to cell 1, row 1.

myGrid.CurrentCell = New DataGridCell(1, 1)

End Sub

Private Sub DGV\_CellEndEdit(ByVal sender As Object, ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles DGV.CellEndEdit

If e.ColumnIndex = 0 Then

strSql = "select \* from barang where kode\_barang='" & DGV.Rows(e.RowIndex).Cells(0).Value & "'"

If Valid\_table(strSql) Then

DGV.Rows(e.RowIndex).Cells(1).Value = Isi\_Kolom(strSql, 1)

DGV.Rows(e.RowIndex).Cells(2).Value = Isi\_Kolom(strSql, 3)

DGV.Rows(e.RowIndex).Cells(3).Value = 1

DGV.Rows(e.RowIndex).Cells(4).Value = DGV.Rows(e.RowIndex).Cells(2).Value \* DGV.Rows(e.RowIndex).Cells(3).Value

Call TotalItem()

Call TotalHarga()

'DGV.CurrentCell = DGV(3, 0)

'DGV.Rows(e.RowIndex).Cells(3).Value = Keys.Up()

Else

MsgBox("Kode barang tidak terdaftar")

End If

End If

If e.ColumnIndex = 3 Then

strSql = "select \* from barang where kode\_barang='" & DGV.Rows(e.RowIndex).Cells(0).Value & "'"

If Valid\_table(strSql) Then

If DGV.Rows(e.RowIndex).Cells(3).Value > Isi\_Kolom(strSql, 4) Then

MsgBox("Stok barang hanya ada " & Isi\_Kolom(strSql, 4) & "")

DGV.Rows(e.RowIndex).Cells(3).Value = 1

DGV.Rows(e.RowIndex).Cells(4).Value = DGV.Rows(e.RowIndex).Cells(2).Value \* DGV.Rows(e.RowIndex).Cells(3).Value

Call TotalItem()

Call TotalHarga()

Else

DGV.Rows(e.RowIndex).Cells(4).Value = DGV.Rows(e.RowIndex).Cells(2).Value \* DGV.Rows(e.RowIndex).Cells(3).Value

Call TotalItem()

Call TotalHarga()

End If

End If

DGV.CurrentCell = DGV.Rows(0).Cells(0)

End If

End Sub

Sub TotalItem()

Dim HitungItem As Integer = 0

For I As Integer = 0 To DGV.Rows.Count - 1

HitungItem = HitungItem + Val(DGV.Rows(I).Cells(3).Value)

Item.Text = HitungItem

Next

End Sub

Sub TotalHarga()

Dim HitungHarga As Integer = 0

For I As Integer = 0 To DGV.Rows.Count - 1

HitungHarga = HitungHarga + Val(DGV.Rows(I).Cells(4).Value)

Total.Text = HitungHarga

Next

End Sub

Private Sub BTNTutup\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BTNTutup.Click

Me.Close()

End Sub

Sub HapusBaris()

On Error Resume Next

'DGV.Rows.RemoveAt(DGV.CurrentCell.RowIndex)

Dim baris As Integer = DGV.CurrentCell.RowIndex

DGV.Rows(baris).Cells(0).Value = ""

Chr(30)

End Sub

Private Sub DGV\_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles DGV.KeyDown

If e.KeyCode = Keys.Up Then

DGV.CurrentCell = DGV.Rows(0).Cells(3)

End If

End Sub

Private Sub DGV\_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles DGV.KeyPress

On Error Resume Next

If e.KeyChar = Chr(27) Then

DGV.Rows.RemoveAt(DGV.CurrentCell.RowIndex)

Call TotalItem()

Call TotalHarga()

Dibayar.Clear()

Kembali.Text = ""

End If

End Sub

Private Sub Dibayar\_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles Dibayar.KeyPress

If e.KeyChar = Chr(13) Then

If Val(Dibayar.Text) < Val(Total.Text) Then

MsgBox("Pembayaran kurang")

Kembali.Text = ""

Dibayar.Focus()

Exit Sub

ElseIf Val(Dibayar.Text) = Val(Total.Text) Then

Kembali.Text = 0

BTNSimpan.Focus()

Else

Kembali.Text = Val(Dibayar.Text) - Val(Total.Text)

BTNSimpan.Focus()

End If

End If

If Not ((e.KeyChar >= "0" And e.KeyChar <= "9") Or e.KeyChar = vbBack) Then e.Handled() = True

End Sub

Private Sub BTNSimpan\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BTNSimpan.Click

If Total.Text = "" Or Dibayar.Text = "" Or Kembali.Text = "" Or Item.Text = "" Then

MsgBox("Data belum lengkap, tidak ada transaksi atau pembayaran masih kosong")

Exit Sub

End If

cetak\_struk()

'simpan ke tabel penjualan

strSql = "Insert into penjualan(faktur,tanggal,item,total,dibayar,kembali,KodePtg) values " & \_

"('" & Faktur.Text & "','" & Tanggal.Text & "','" & Item.Text & "','" & Total.Text & "','" & Dibayar.Text & "','" & Kembali.Text & "','" & "PTG01" & "')"

Exe\_table(strSql)

For baris As Integer = 0 To DGV.Rows.Count - 2

'simpan ke tabel detail

strSql = "Insert into detailjual (faktur,kode\_Barang,nama\_Barang,harga\_Jual,jumlah,subtotal) values " & \_

"('" & Faktur.Text & "','" & DGV.Rows(baris).Cells(0).Value & "','" & DGV.Rows(baris).Cells(1).Value & "','" & DGV.Rows(baris).Cells(2).Value & "','" & DGV.Rows(baris).Cells(3).Value & "','" & DGV.Rows(baris).Cells(4).Value & "')"

Exe\_table(strSql)

'kurangi stok barang

strSql = "select \* from barang where kode\_barang='" & DGV.Rows(baris).Cells(0).Value & "'"

If Valid\_table(strSql) Then

strSql = "update barang set jumlah\_barang= '" & Isi\_Kolom(strSql, 4) - DGV.Rows(baris).Cells(3).Value & "' where kode\_barang='" & DGV.Rows(baris).Cells(0).Value & "'"

Exe\_table(strSql)

End If

Next baris

DGV.Columns.Clear()

Call BuatKolomBaru()

Call FakturOtomatis()

Call BersihkanText()

End Sub

Sub cetak\_struk()

Dim s As String

Const sLN As String = ControlChars.CrLf

Const tabx As String = ControlChars.Tab

Dim pd As New PrintDialog

Dim oPS As New System.Drawing.Printing.PrinterSettings

Dim sps As String

s &= " Toko Brownis Fortuna" & sLN

s &= "Otlate : " & sLN

s &= "Kasir :" & sLN

s &= "No struk :" & Faktur.Text & sLN

s &= "tanggal : " & TimeOfDay & " " & sLN

s &= "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" & sLN

For baris As Integer = 0 To DGV.Rows.Count - 2

's &= "" & DGV.Rows(baris).Cells(0).Value + vbTab + DGV.Rows(baris).Cells(1).Value + vbTab + DGV.Rows(baris).Cells(2).Value + vbTab + DGV.Rows(baris).Cells(3).Value + vbTab + DGV.Rows(baris).Cells(4).Value & sLN

s &= "" & DGV.Rows(baris).Cells(0).Value + vbTab + DGV.Rows(baris).Cells(1).Value + vbTab + FormatD(DGV.Rows(baris).Cells(2).Value) & sLN

Next baris

s &= " " & sLN

s &= " " & sLN

s &= " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" & sLN

s &= " Subtotal : " & Total.Text & sLN

s &= " Cash : " & Dibayar.Text & sLN

s &= " Kembali : " & Kembali.Text & sLN

s &= " " & sLN

s &= " " & sLN

s &= " " & sLN

s &= "\*\*\*\*\*\*Terima Kasih atas kunjungan anda\*\*\*\*\*" & sLN

'MessageBox.Show(oPS.PrinterName)

MessageBox.Show(s)

pd.PrinterSettings = New PrinterSettings

Dim prname As String

RawPrinterHelper.SendStringToPrinter(oPS.PrinterName, s)

End Sub

End Class

CLASS: RawPrinterHelper

Imports System.IO

Imports System.Drawing.Printing

Imports System.Runtime.InteropServices

Public Class RawPrinterHelper

<StructLayout(LayoutKind.Sequential, CharSet:=CharSet.Unicode)> \_

Structure DOCINFOW

<MarshalAs(UnmanagedType.LPWStr)> Public pDocName As String

<MarshalAs(UnmanagedType.LPWStr)> Public pOutputFile As String

<MarshalAs(UnmanagedType.LPWStr)> Public pDataType As String

End Structure

<DllImport("winspool.Drv", EntryPoint:="OpenPrinterW", \_

SetLastError:=True, CharSet:=CharSet.Unicode, \_

ExactSpelling:=True, CallingConvention:=CallingConvention.StdCall)> \_

Public Shared Function OpenPrinter(ByVal src As String, ByRef hPrinter As IntPtr, ByVal pd As Long) As Boolean

End Function

<DllImport("winspool.Drv", EntryPoint:="ClosePrinter", \_

SetLastError:=True, CharSet:=CharSet.Unicode, \_

ExactSpelling:=True, CallingConvention:=CallingConvention.StdCall)> \_

Public Shared Function ClosePrinter(ByVal hPrinter As IntPtr) As Boolean

End Function

<DllImport("winspool.Drv", EntryPoint:="StartDocPrinterW", \_

SetLastError:=True, CharSet:=CharSet.Unicode, \_

ExactSpelling:=True, CallingConvention:=CallingConvention.StdCall)> \_

Public Shared Function StartDocPrinter(ByVal hPrinter As IntPtr, ByVal level As Int32, ByRef pDI As DOCINFOW) As Boolean

End Function

<DllImport("winspool.Drv", EntryPoint:="EndDocPrinter", \_

SetLastError:=True, CharSet:=CharSet.Unicode, \_

ExactSpelling:=True, CallingConvention:=CallingConvention.StdCall)> \_

Public Shared Function EndDocPrinter(ByVal hPrinter As IntPtr) As Boolean

End Function

<DllImport("winspool.Drv", EntryPoint:="StartPagePrinter", \_

SetLastError:=True, CharSet:=CharSet.Unicode, \_

ExactSpelling:=True, CallingConvention:=CallingConvention.StdCall)> \_

Public Shared Function StartPagePrinter(ByVal hPrinter As IntPtr) As Boolean

End Function

<DllImport("winspool.Drv", EntryPoint:="EndPagePrinter", \_

SetLastError:=True, CharSet:=CharSet.Unicode, \_

ExactSpelling:=True, CallingConvention:=CallingConvention.StdCall)> \_

Public Shared Function EndPagePrinter(ByVal hPrinter As IntPtr) As Boolean

End Function

<DllImport("winspool.Drv", EntryPoint:="WritePrinter", \_

SetLastError:=True, CharSet:=CharSet.Unicode, \_

ExactSpelling:=True, CallingConvention:=CallingConvention.StdCall)> \_

Public Shared Function WritePrinter(ByVal hPrinter As IntPtr, ByVal pBytes As IntPtr, ByVal dwCount As Int32, ByRef dwWritten As Int32) As Boolean

End Function

Public Shared Function SendBytesToPrinter(ByVal szPrinterName As String, ByVal pBytes As IntPtr, ByVal dwCount As Int32) As Boolean

Dim hPrinter As IntPtr

Dim dwError As Int32

Dim di As DOCINFOW

Dim dwWritten As Int32

Dim bSuccess As Boolean

With di

.pDocName = "Toko Komputer"

.pDataType = "RAW"

End With

bSuccess = False

If OpenPrinter(szPrinterName, hPrinter, 0) Then

If StartDocPrinter(hPrinter, 1, di) Then

If StartPagePrinter(hPrinter) Then

bSuccess = WritePrinter(hPrinter, pBytes, dwCount, dwWritten)

EndPagePrinter(hPrinter)

End If

EndDocPrinter(hPrinter)

End If

ClosePrinter(hPrinter)

End If

If bSuccess = False Then

dwError = Marshal.GetLastWin32Error()

End If

Return bSuccess

End Function

Public Shared Function SendFileToPrinter(ByVal szPrinterName As String, ByVal szFileName As String) As Boolean

Dim fs As New FileStream(szFileName, FileMode.Open)

Dim br As New BinaryReader(fs)

Dim bytes(fs.Length) As Byte

Dim bSuccess As Boolean

Dim pUnmanagedBytes As IntPtr

bytes = br.ReadBytes(fs.Length)

pUnmanagedBytes = Marshal.AllocCoTaskMem(fs.Length)

Marshal.Copy(bytes, 0, pUnmanagedBytes, fs.Length)

bSuccess = SendBytesToPrinter(szPrinterName, pUnmanagedBytes, fs.Length)

Marshal.FreeCoTaskMem(pUnmanagedBytes)

Return bSuccess

End Function

Public Shared Function SendStringToPrinter(ByVal szPrinterName As String, ByVal szString As String)

Dim pBytes As IntPtr

Dim dwCount As Int32

dwCount = szString.Length()

pBytes = Marshal.StringToCoTaskMemAnsi(szString)

SendBytesToPrinter(szPrinterName, pBytes, dwCount)

Marshal.FreeCoTaskMem(pBytes)

End Function

End Class

MODUL: Sistem\_Module1

Imports System.Data.OleDb

Imports CrystalDecisions.CrystalReports.Engine

Imports CrystalDecisions.Shared

Module Sistem\_Module1

'Memasan Variabel Public yang dibutuhkan

Public dConn As OleDbConnection

Public dAdapter As OleDbDataAdapter

Public dSet As DataSet

Public dcmd As OleDbCommand

Public dRead As OleDbDataReader

Public strSql As String

Public rsDataSource As System.Data.DataTable

'Public cryRpt As New ReportDocument

'Public crtableLogoninfos As New TableLogOnInfos

'Public crtableLogoninfo As New TableLogOnInfo

'Public crConnectionInfo As New ConnectionInfo

'Public CrTables As Tables

'Public CrTable As Table

'Procedure untuk meng-Koneksikan database

Public Sub Koneksi()

'Untuk Access 2003

'Conn = New OleDbConnection("provider=microsoft.jet.oledb.4.0;data source=E:\Percobaan\dbkoperasi.mdb")

'Conn.Open()

'Untuk Access 2007

dConn = New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;data source=D:\LatihanVB9\PrinterTransaksi\Printer02\Printer02\dataTransaksi.accdb")

dConn.Open()

End Sub

'Procedure untuk menampilkan isi Table

Public Sub Tampil\_Table(ByVal strsql As String, ByVal nmtable As String)

dAdapter = New OleDbDataAdapter(strsql, dConn)

dSet = New DataSet

dSet.Clear()

dAdapter.Fill(dSet, nmtable)

rsDataSource = (dSet.Tables(nmtable))

End Sub

'Procedure untuk meng-eksekusi Table Database

Public Sub Exe\_table(ByVal strsql As String)

dcmd = New OleDbCommand(strsql, dConn)

dcmd.ExecuteNonQuery()

End Sub

'Function untuk meng-Validasi data pada Table Database

Public Function Valid\_table(ByVal strsql As String) As Boolean

dcmd = New OleDbCommand(strsql, dConn)

dRead = dcmd.ExecuteReader

dRead.Read()

Return dRead.HasRows

End Function

Public Function Posisi\_Baris(ByVal strsql As String) As Long

dcmd = New OleDbCommand(strsql, dConn)

dRead = dcmd.ExecuteReader

dRead.Read()

'rd.GetString(0)

Return dRead.GetString(0)

End Function

Public Function Isi\_Kolom(ByVal strsql As String, ByVal kolom As Integer)

dcmd = New OleDbCommand(strsql, dConn)

dRead = dcmd.ExecuteReader

dRead.Read()

Return dRead.Item(kolom)

End Function

'Function untuk menentukan Path data

Public Function app\_path() As String

Return System.AppDomain.CurrentDomain.BaseDirectory()

End Function

'Function untuk menentukan terbilang dengan kata-kata

Public Function Terbilang(ByVal nilai As Long) As String

Dim bilangan As String() = {" ", "Satu", "Dua", "Tiga", "Empat", "Lima", \_

"Enam", "Tujuh", "Delapan", "Sembilan", "Sepuluh", "Sebelas"}

If nilai < 12 Then

Return " " & bilangan(nilai)

ElseIf nilai < 20 Then

Return Trim(Terbilang(nilai - 10)) & " Belas "

ElseIf nilai < 100 Then

Return (Trim(Terbilang(CInt((nilai \ 10)))) & " Puluh") + Terbilang(nilai Mod 10)

ElseIf nilai < 200 Then

Return "Seratus " & Trim(Terbilang(nilai - 100))

ElseIf nilai < 1000 Then

Return (Trim(Terbilang(CInt((nilai \ 100)))) & " Ratus ") + Terbilang(nilai Mod 100)

ElseIf nilai < 2000 Then

Return "Seribu " & Trim(Terbilang(nilai - 1000))

ElseIf nilai < 1000000 Then

Return (Trim(Terbilang(CInt((nilai \ 1000)))) & " Ribu ") + Terbilang(nilai Mod 1000)

ElseIf nilai < 1000000000 Then

Return (Trim(Terbilang(CInt((nilai \ 1000000)))) & " Juta ") + Terbilang(nilai Mod 1000000)

ElseIf nilai < 1000000000000 Then

Return (Trim(Terbilang(CInt((nilai \ 1000000000)))) & " Milyar ") + Terbilang(nilai Mod 1000000000)

ElseIf nilai < 1000000000000000 Then

Return (Trim(Terbilang(CInt((nilai \ 1000000000000)))) & " Trilyun ") + Terbilang(nilai Mod 1000000000000)

Else

Return ""

End If

End Function

'Function untuk memformat angka model-1

Public Function FormatD(ByVal var) As String

Return Format(FormatG(var), "###,###,##0.00")

End Function

'Function untuk memformat angka model-2

Public Function FormatD2(ByVal var) As String

Return Format(FormatG(var), "###,###,##0")

End Function

'Function untuk menghilangkan format angka

Public Function FormatG(ByVal var As String) As Double

Dim nVar As Double

nVar = Val(Microsoft.VisualBasic.Str(var))

Return nVar

End Function

End Module