

MODUL PRAKTIKUM PEMROGRAMAN BERBASIS DESKTOP



STMIK YADIKA BANGIL

2016

MODUL 1

PENGENALAN VISUAL BASIC

1. TUJUAN

- Memahami platform Microsoft .NET;
- Mengenal Integrated Development Environment (IDE) Visual Basic .NET;
- Memahami struktur project Visual Basic .NET;
- Memahami jenis-jenis aplikasi .NET;
- Mampu membangun Graphical User Interface (GUI) menggunakan Visual Basic .NET;

2. TEORI DASAR

Platform .NET Framework

Microsoft .NET yang awalnya disebut Next Generation Windows Services (NGWS) adalah suatu platform untuk membangun dan menjalankan generasi penerus aplikasi-aplikasi terdistribusi. Microsoft.NET merupakan framework (kerangka) pengembangan yang menyediakan antarmuka pemrograman baru untuk layanan Windows dan API (Application Programming Interface). Microsoft .NET merupakan strategi Microsoft untuk menghubungkan sistem, informasi, dan alat (device), sehingga orang dapat berkomunikasi serta berkolaborasi dengan lebih efektif. Teknologi .NET terintegrasi penuh melalui produk-produk Microsoft, dan menyediakan kemampuan untuk mengembangkan solusi dengan menggunakan Web service. Platform Microsoft .NET terdiri dari lima komponen utama yang tersusun dalam tiga lapisan (layer). Lapisan paling bawah adalah sistem operasi; lapisan kedua terdiri dari tiga komponen; lapisan teratas adalah Visual Studio .NET.

Visual Studio .NET

Microsoft Visual Studio .NET merupakan kumpulan lengkap tools pengembangan untuk membangun aplikasi Web ASP.NET, XML Web Services, aplikasi dekstop, dan aplikasi mobile. Di dalam Visual Studio inilah bahasa-bahasa pemrograman .NET seperti Visual Basic, Visual C++, Visual C# (CSharp), dan Visual J# (JSharp). Semuanya menggunakan lingkungan pengembangan

terintegrasi atau IDE yang sama sehingga memungkinkan untuk saling berbagi tools dan fasilitas.

Visual Basic .NET

Visual Basic .NET (atau VB.NET) merupakan salah satu bahasa pemrograman yang bisa digunakan untuk membangun aplikasi-aplikasi.NET di platform Microsoft .NET. Tidak seperti generasi sebelumnya Visual Basic versi 6.0 ke bawah yang lebih difokuskan untuk pengembangan aplikasi desktop, Visual Basic .NET memungkinkan para pengembang membangun bermacam aplikasi, baik desktop maupun aplikasi web. Seiring dengan perkembangan aplikasi perangkat lunak yang semakin kompleks, saat ini Visual Basic .NET terbaru (Visual Basic 2008).

IDE Visual Basic

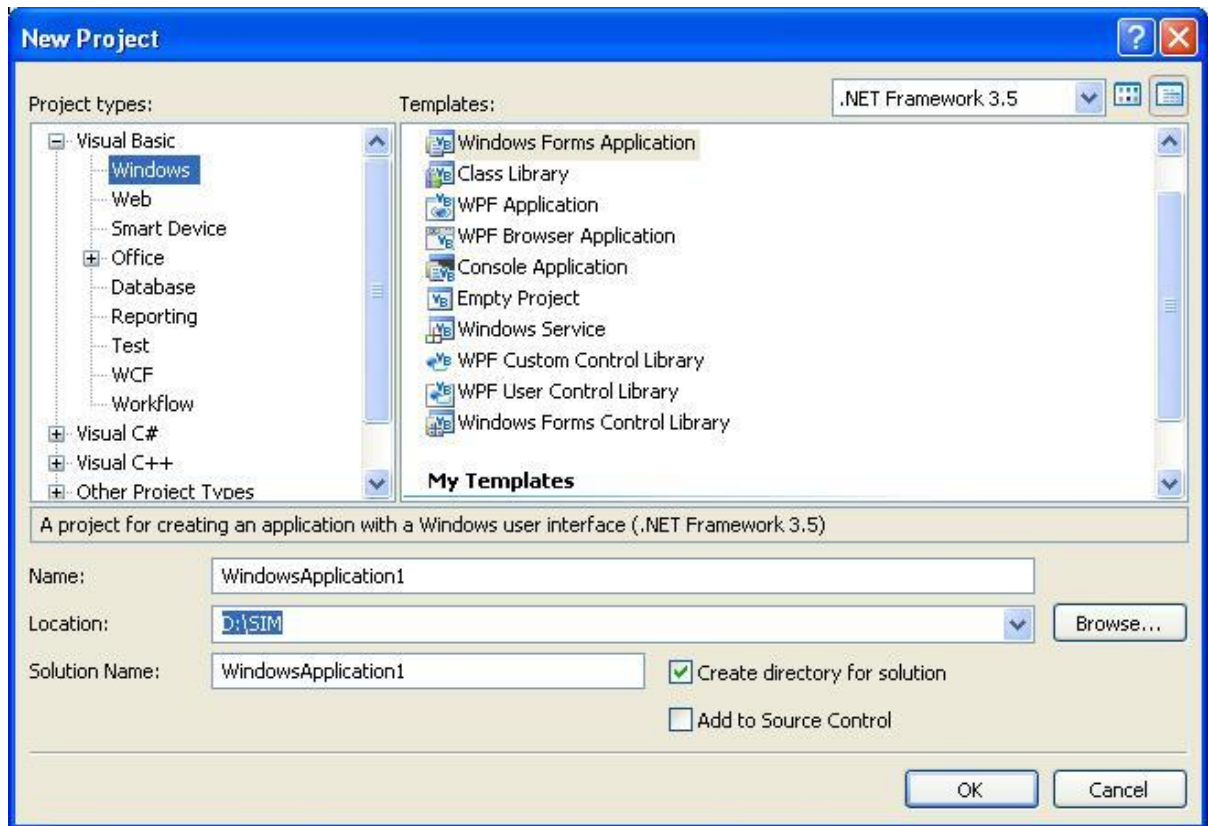
IDE (Integrated Development Environment), atau juga disebut sebagai Integrated Design/Debugging Environment, adalah perangkat lunak komputer yang berfungsi untuk membantu pemrogram dalam mengembangkan perangkat lunak. Singkatnya, IDE merupakan suatu lingkungan pengembangan aplikasi yang terintegrasi; lengkap dengan beragam tools atau utilitas pendukung.

3. PRAKTEK

Membuat Project Baru

Project merupakan kerangka dasar aplikasi yang sangat menentukan jenis aplikasi yang akan dibuat. Untuk memulai pembuatan project, ikuti langkah-langkah berikut:

1. Jalankan **Visual Basic 2008 Express Edition**.
2. Klik menu **File > New Project**, atau melalui ikon **New Project**, atau melalui kombinasi tombol **Ctrl+N**.
3. Pada kotak dialog **New Project**, pilih template **Windows Application**, kemudian klik **OK**.



Windows (Windows Form, Concole, Class Library, WPF dll)

Web (ASP Web Application, ASP.Net Web Servis dll)

Smart Device (Smart Device Project)

Database (Access dan SQL Server)

Report (Report Application dan Crystal Report)

WCF (Windows Communication Foundation) dll.

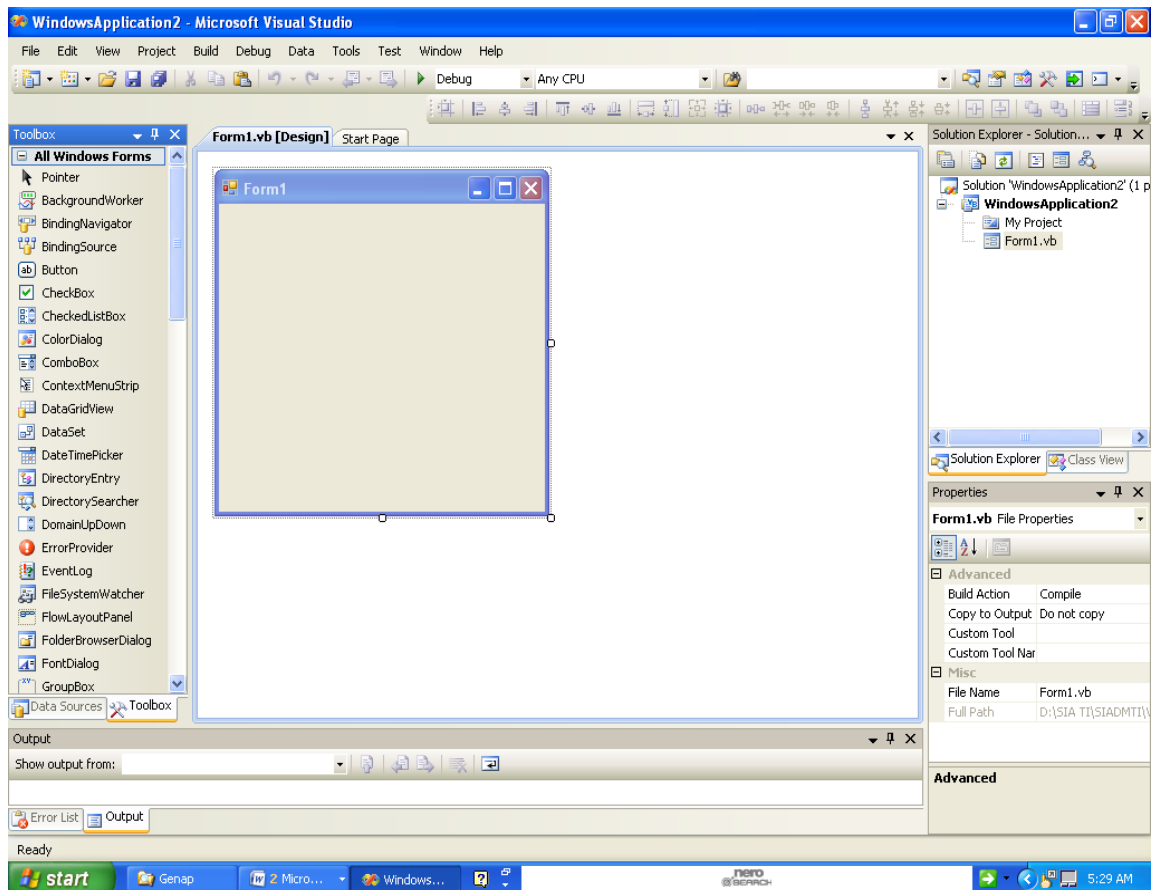
Windows Application. Untuk menciptakan aplikasi-aplikasi Windows yang berjalan secara lokal di komputer user.

Class Library. Untuk menciptakan kelas-kelas atau komponenkomponen reusable yang dapat di-share dengan project lainnya.

Console Application. Untuk membuat aplikasi-aplikasi command-line.

Mengenal IDE Visual Basic 2008

Dalam upaya memudahkan penggunaan IDE, maka diperlukan pemahaman yang baik mengenai IDE Visual Basic 2008.



Interface	Keterangan
Menu Bar	Menu standar pada visual Basic
Toolbar	Daftar tool yang digunakan untuk menjalankan perintah yang sering digunakan
Toolbox	Daftar control yang tambahkan ke dalam program sebagai interface
Form Design	Digunakan untuk mengedit tampilan form serta mengatur posisi control pada form
Solution Explorer	Digunakan untuk mengolah file dan project
Properties	Digunakan untuk mengedit property dari form dan control yang sedang diedit
Error List	Digunakan untuk menampilkan pesan error jika terjadi kesalahan

Membuat Aplikasi Windows

1. Buat **project baru**, atau lanjutkan project pertama Anda sebelumnya.

2. Tambahkan sebuah kontrol Label ke form, dengan cara mengklik ganda Label di Toolbox, atau men-drag Label ke form. Atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Frmlatihan
	Text	Latihan Visual Basic
Label	Name	lblText

3. Tambahkan **Event Load** pada form dengan salah satu cara berikut:
 - a. Klik ganda area form kosong.
 - b. Klik area form kosong, kemudian tekan **F7**.
4. Lengkapi kode event handler-nya seperti berikut:

```
Public Class Form1
Private Sub Form1_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
' Men-set teks kontrol label
Me.lblText.Text = "Belajar Pemrograman Visual Basic"
End Sub
End Class
```

5. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
6. Simpan aplikasi **Anda**.

Membuat Aplikasi Console

1. Buat **project baru**, pilih template **Console Application**, kemudian klik **OK**.
2. Lengkapi kode programnya seperti berikut:

```
Module Module1
Sub Main()
' Mencetak teks ke console
Console.WriteLine("Apa Kabar Teman-Teman ")
Console.WriteLine("Selamat Belajar Pemrograman Visual")
End Sub
End Module
```

```
Console.Read()  
End Sub  
End Module
```

3. Tekan **F5** untuk menjalankan aplikasi.
4. Simpan aplikasi Anda.

MODUL 2 FORM DAN OBJEK KONTROL (1)

1. TUJUAN

- Memahami dan mengenal Form dan Objek Kontrol;
- Dapat membuat aplikasi sederhana dengan menggunakan form dan objek control.

2. DASAR TEORI

Form merupakan media interaksi antara pengguna dengan aplikasi yang dibuat. Form dapat dikatakan sebagai wadah atau penampung objek control yang akan digunakan. Form juga dapat dikatakan objek karena dapat memberikan reaksi saat menemui suatu kejadian. Form dapat dikategorikan menjadi 2 yaitu :

1. Form Dinamis

Yaitu form yang dapat dimanipulasi atau diubah bentuk serta disisipi objek control yang berisi perintah-perintah yang diperlukan oleh aplikasi yang akan dibuat, contoh :

- Window (Windows Form, Console, Class Library, WPF dsb.)
- Web (ASP.Net Web Application, ASP.NET Web Servis dsb)
- Smart Device (Smart Device Project)
- Database(Access dan SQL Server)
- Report (Report Aplication dan Crystal Report)
- WCF (Windows Communication Foundation)
- dll

2. Form Statis

Yaitu form yang tidak dapat dimanipulasi atau diubah bentuk serta disisipi objek control. Form ini hanya dapat dipanggil melalui kode perintah, Contoh :

a. Form Pesan (MessageBox)

Form yang bertugas untuk menampilkan pesan keterangan terhadap suatu kejadian yang diterima oleh aplikasi.

b. InputBox (Kotak Input Pesan)

Form ini digunakan untuk interaksi antara pengguna dengan aplikasi yang dibuat, dimana pengguna tersebut memasukkan suatu nilai lalu mengklik suatu tombol dan menunggu efek yang ditimbulkan oleh aplikasi yang dibuat.

OBJEK KONTROL

Label, TextBox dan Button

Label merupakan objek control yang dapat menampilkan output tetapi tidak dapat memberikan input pada saat dijalankan.

TextBox Merupakan objek control yang dapat diberikan input pada saat program dijalankan.

Button merupakan objek control yang dapat mengeksekusi perintah-perintah yang telah diberikan.

CheckBox dan RadioButton

CheckBox merupakan objek control yang berfungsi untuk memilih beberapa item data. Dengan objek ini anda dapat lebih dari satu pilihan dan bahkan memilih semua pilihan tersedia.

RadioButton fungsi hampir sama dengan checkbox, tetapi hanya dapat memilih satu pilihan yang tersedia.

ComboBox dan ListBox

ComboBox merupakan objek control yang dapat digunakan untuk menampilkan daftar item dengan pilihan dropdown.

ListBox merupakan objek control yang jauh berbeda dengan ComboBox hanya saja pada listbox menu daftar pilihan ditampilkan secara keseluruhan.

Timer dan DateTimePicker

Timer merupakan objek control yang berfungsi untuk mengatur dan menampilkan waktu yang diperlukan.

DateTimerPicker objek control yang berfungsi untuk menampilkan tanggal dengan menu DropDown.

3. PRAKTEK

CheckBox dan RadioButton

1. Buat **project baru**, dengan nama **Prak2**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	CheckBox dan RadioButton
Label1	Name	Lbl_Font
	Autosize	False
	BorderStyle	Fixed3D

	Font	Microsoft Sans Serif, 16
	TextAlign	Center
	Text	CheckBox dan RadioButton
GroupBox1	Text	Efek
GroupBox2	Text	Font Style
CheckBox1	Text	StrikeOut
CheckBox2	Text	Underline
RadioButton1	Text	Regular
RadioButton2	Text	Italic
RadioButton3	Text	Bold
RadioButton4	Text	BoldItalic

4. Klik ganda pada CheckBox1, kemudian ketikkan kode program berikut ini :

```
Private Sub CheckBox1_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckBox1.CheckedChanged

    Lbl_Font.Font = New Font("Microsoft sans Serif", Lbl_Font.Font.Size, _
        Lbl_Font.Font.Style Xor FontStyle.Strikeout)

End Sub
```

5. Klik ganda pada CheckBox2, kemudian ketikkan kode program berikut ini :

```
Private Sub CheckBox2_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles CheckBox2.CheckedChanged

    Lbl_Font.Font = New Font("Microsoft sans Serif", Lbl_Font.Font.Size, _
        Lbl_Font.Font.Style Xor FontStyle.Underline)

End Sub
```

6. Klik ganda pada RadioButton1, kemudian ketikkan kode program berikut ini :

```
Private Sub RadioButton1_CheckedChanged(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles RadioButton1.CheckedChanged  
  
    Lbl_Font.Font = New Font("Microsoft sans Serif", Lbl_Font.Font.Size, _  
        Lbl_Font.Font.Style Xor FontStyle.Regular)  
  
End Sub
```

7. Klik ganda pada RadioButton2, kemudian ketikkan kode program berikut ini :

```
Private Sub RadioButton2_CheckedChanged(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles RadioButton2.CheckedChanged  
  
    Lbl_Font.Font = New Font("Microsoft sans Serif", Lbl_Font.Font.Size, _  
        Lbl_Font.Font.Style Xor FontStyle.Italic)  
  
End Sub
```

8. Klik ganda pada RadioButton3, kemudian ketikkan kode program berikut ini :

```
Private Sub RadioButton3_CheckedChanged(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles RadioButton3.CheckedChanged  
  
    Lbl_Font.Font = New Font("Microsoft sans Serif", _  
        Lbl_Font.Font.Size, Lbl_Font.Font.Style Xor FontStyle.Bold)  
  
End Sub
```

9. Klik ganda pada RadioButton4, kemudian ketikkan kode program berikut ini :

```
Private Sub RadioButton4_CheckedChanged(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles RadioButton4.CheckedChanged
```

```
Lbl_Font.Font = New Font("Microsoft sans Serif", _  
    Lbl_Font.Font.Size, Lbl_Font.Font.Style Xor FontStyle.Bold Xor  
    FontStyle.Italic)  
  
End Sub
```

10. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
11. Simpan aplikasi **Anda**.

MODUL 3

FORM DAN OBJEK KONTROL (2)

1. TUJUAN

- Memahami dan mengenal Form dan Objek Kontrol;
- Dapat membuat aplikasi sederhana dengan menggunakan form dan objek control.

2. TEORI DASAR

OBJEK KONTROL

MenuStrip, ContextMenuStrip dan RichTextBox

MenuStrip adalah merupakan objek control yang berfungsi untuk membuat menu pilihan dropdown.

ContextMenuStrip adalah merupakan objek control yang berfungsi untuk membuat menu pilihan yang akan tampil pada saat tombol kanan mouse di klik.

RichTextBox adalah merupakan objek control yang fungsinya hampir sama seperti textbox hanya saja di objek control ini lebih banyak menampung karakter dan bisa manipulasi karakter seperti menggaris bawah sbd.

OpenFileDialog dan SaveFileDialog

OpenFileDialog adalah merupakan objek control yang berfungsi untuk membuka kotak dialog Open.

SaveFileDialog berfungsi untuk membuka kotak dialog save yang dimiliki windows.

ToolStrip dan StatusStrip

ToolStrip adalah merupakan objek control yang berfungsi hampir sama dengan tombol pada umumnya hanya saja pada objek control ini setiap tombol memiliki gambar dan digunakan untuk mengakses gambar dengan cepat.

StatusStrip digunakan untuk menampilkan informasi seperti nama aplikasi, jam, tanggal dll. Pada umumnya objek control ini terletak di bagian bawah aplikasi.

3. PRAKTEK

ComboBox dan ListBox

1. Tambahkan Form baru, dengan dengan cara klik kanan pada **Prak2**. Klik Add, pilih Windows Form Beri nama **Form2**
2. Buatlah desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form2	Name Text Start Position	Form2 ComboBox dan ListBox CenterScreen
Label1	Text	Drive
Label2	Text	Keterangan
ComboBox1	Name	Cmb_Drive
ListBox1	Name	LstInfo

4. Klik icon view code pada solution Explorer, di atas Public Class Form1, ketikkan kode program berikut ini :

```
Imports System.IO
Public Class Form2
```

5. Klik ganda pada form, kemudian ketikkan kode program berikut ini :

```
Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    Cmb_Drive.Items.AddRange(IO.DriveInfo.GetDrives)
End Sub
```

6. Klik ganda pada ComboBox, kemudian ketik kode program berikut ini :

```
Private Sub Cmb_Drive_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
Cmb_Drive.SelectedIndexChanged
    LstInfo.Items.Clear()
    LstInfo.Items.Add("Nama Drive :" & Dir(" " & _
Cmb_Drive.Text & " ", FileAttribute.Volume) & " ")
    LstInfo.Items.Add("System :" & Dir(" " & _
Cmb_Drive.Text & " ", FileAttribute.System) & " ")
    Dim dvr As New DriveInfo(Cmb_Drive.Text)
    Dim kapasitas = CStr(dvr.TotalSize / 1000000) & " MB"
    Dim sisa = CStr(dvr.TotalFreeSpace / 1000000) & " MB"
    Dim Terpakai = CStr(Val(kapasitas) - Val(sisa)) & " MB"
    LstInfo.Items.Add("Kapasitas : " & kapasitas)
    LstInfo.Items.Add("Terpakai : " & Terpakai)
    LstInfo.Items.Add("Sisa : " & sisa)
End Sub
```

7. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
8. Simpan aplikasi **Anda**

Timer dan DateTimePicker

1. Tambahkan Form baru, dengan dengan cara klik kanan pada **Prak2**. Klik Add, pilih Windows Form Beri nama **Form3**
2. Buatlah desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertynya seperti tabel berikut:

Object	Properties	Nilai
Form3	Name	Form3
	Text	Timer dan DateTimePicker
	Start Position	CenterScreen
Label1	name	Lbl12jam
Label2	Name	Lbl24jam
DateTimePicker1	Name	Dtanggal
Timer1	Enable	True
	Interval	1000

4. Klik ganda pada form, ketikkan kode program berikut ini :

```

Private Sub Form3_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    DTtanggal.Format = DateTimePickerFormat.Custom
    DTtanggal.CustomFormat = "dddd,dd/MM/yyyy"
    DTtanggal.Value = Format(Now)
End Sub

```

5. Klik ganda pada Timer1, ketikkan kode program berikut ini :

```

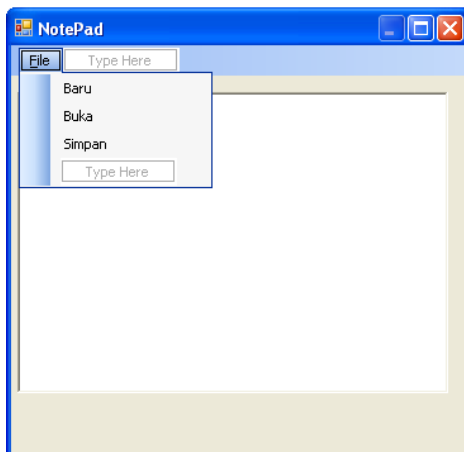
Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    Lbl12Jam.Text = Format(Now, "h:mm:ss tt")
    Lbl24Jam.Text = Format(Now, "hh:mm:ss")
End Sub

```

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
7. Simpan aplikasi **Anda**

MenuStrip, ContextMenuStrip dan RichTextBox

1. Tambahkan Form baru, dengan dengan cara klik kanan pada **Prak2**. Klik Add, pilih Windows Form Beri nama **Form4**
2. Buatlah desain Tampilan form seperti gambar berikut ini :



3. Tambahkan menuStrip seperti pada form di atas
4. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name Text Start Position	Form1 NotePad CenterScreen
MenuStrip1	Name	MenuStrip1
RichTextBox1	Name ContextMenuStrip	RchTeks ContextMenuStrip1
ContextMenuStrip1	Name	ContextMenuStrip1

5. Klik ganda pada menu buka, ketikkan kode program berikut ini :

```

Private Sub BukaToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles BukaToolStripMenuItem.Click
    Dim kotak_dialog As OpenFileDialog = New OpenFileDialog
    kotak_dialog.Filter = "File Text (*.txt) | *.txt"
    kotak_dialog.ShowDialog()
    Dim fileText As String
    =My.Computer.FileSystem.ReadAllText(kotak_dialog.FileName)
    RchTeks.Text = fileText
End Sub

```

6. Klik ganda pada menu simpan, ketikkan kode program berikut ini :

```

Private Sub SimpanToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
SimpanToolStripMenuItem.Click
    Dim kotak_dialog As SaveFileDialog = New SaveFileDialog
    kotak_dialog.Filter = "File Text (*.txt) | *.txt"
    kotak_dialog.ShowDialog()

```

```
My.Computer.FileSystem.WriteAllText("" & kotak_dialog.FileName &
"", RchTeks.Text, False)
End Sub
```

7. Klik ganda pada menu Baru, ketikkan kode program berikut ini :

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

8. Klik ganda pada menu ContextMenuStrip dengan menu regular, kemudian ketik kode program berikut ini :

```
Private Sub RegularToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
RegularToolStripMenuItem.Click
    RchTeks.SelectionFont = New Font(RchTeks.SelectionFont,
FontStyle.Regular)
End Sub
```

9. Klik ganda pada menu ContextMenuStrip dengan menu Bold, kemudian ketik kode program berikut ini :

```
Private Sub BoldToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles BoldToolStripMenuItem.Click
    RchTeks.SelectionFont = New Font(RchTeks.SelectionFont,
FontStyle.Bold)
End Sub
```

10. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.

11. Simpan aplikasi **Anda**

PRAKTIKUM 4

TIPE DATA, VARIABEL, KONSTANTA DAN OPERATOR

1. TUJUAN

- Memahami dan mengenal Tipe Data, Variabel, Konstanta dan Operator
- Dapat membuat aplikasi sederhana dengan menggunakan Tipe Data, Variabel, Konstanta dan Operator.

2. TEORI DASAR

Tipe Data

Tipe data merupakan suatu bentuk penggolongan jenis data berdasarkan kategori data, ukuran dan kegunaan data yang dapat ditampung oleh sebuah variabel. Tipe data pada visual basic 2008 dapat dilihat pada table berikut ini :

Tipe Data	.Net Tipe	Ukuran	Jangkauan
Decimal	System.Decimal	12 Bytes	0 – 28 bilangan decimal
Byte	System.Byte	1 Bytes	0 – 255
Double	System.Double	8 Bytes	Bilangan negative antara : -1.79769313486232e+308 sampai -4.94065645841247E-324 Bilangan positif antara : 4.94065645841247E-324 sampai 1.79769313486232e+308
Long	System.Int64	4 Bytes	-9.223.372.036.854.775.808 sampai 9.223.372.036.854.775.808
Single	System.Single	4 Bytes	Bilangan negative antara : -3.402823E sampai - 1.401298E Bilangan positif antara :

			1.401289E-45 sampai 3.402823e
Integer	System.Int32	4 Bytes	-2.147.483.648 sampai -2.147.483.647
Boolean	System.Boolean	2 Bytes	True or False
Date	System.DateTime	2 Bytes	1 Januari 0001 sampai 31 Desember 9999
Short	System.Int16	2 Bytes	-32.768 – 32.767
Char	System.Char	2 Bytes	0 to 65535
String	System.String	10 Bytes + (2 x jumlah karakter)	0 sampai kira-kira 2 milyar karakter Unicode
Object	System.Object	4 Bytes	Semua Tipe, jenis

Konversi Tipe Data

Konversi Tipe data adalah sebuah fungsi yang digunakan untuk mengkonversi (mengubah tipe dari satu tipe ke tipe data yang lain). Konversi tipe data dapat dilihat pada table berikut ini :

Konversi	Tipe Data	Keterangan
Cint	Integer	Integer
cBool	Boolean	Nilai 0 false, selain 0 true
cByte	Byte	Pembulatan nilai decimal
cLng	Long	Pembulatan nilai decimal
cCur	Currency	Money
cShort	Short	Pembulatan nilai decimal
cSng	Single	Pembulatan nilai decimal
cDate	Date	DateTime
CStr	String	Karakter Unicode
cDbl	Double	
cDec	Decimal	Bilangan decimal
cObj	Object	Object

Variabel

Variabel adalah merupakan tempat penyimpanan data dalam memori computer yang mengandung data atau nilai sementara dari sebuah proses pada suatu pemrograman.

Deklarasi Variabel

Deklarasi Variabel adalah merupakan fungsi untuk mengenal suatu variabel berikut dengan tipe data yang pakai pada variabel tersebut. Pendeklarasian variabel di bagi menjadi dua kategori antara lain :

1. Deklarasi Ekplisit

Suatu pendeklarasian umum yang sangat sering dilakukan dengan memberikan kata kunci as.

Bentuk umum :

{Private | Public | Dim | ...} Nama_Var [{As Tipe data}]

Contoh :

Dim Nim as Integer

Dim Nama as String

2. Deklarasi Emplisit

Deklarasi Emplisit adalah suatu pendeklarasian yang tidak memberikan kata kunci As pada semua variabel tersebut. Melainkan memberikan satu karakter atau symbol di akhir variabel. Tabel symbol deklarasi emplisit.

Karakter / Simbol	Tipe Data
%	Integer
\$	String
&	Long
@	Currency
!	Single
#	Double

Contoh :

Dim NIM% 'mewakili tipe data integer

Dim Nama\$ 'mewakili tipe data string

Ruang Lingkup Variabel

Ruang lingkup variabel sangat mempengaruhi sampai dimana nama variabel tersebut dapat dikenali oleh aplikasi. Ruang Lingkup Variabel dapat digolongkan menjadi :

1. Local Variabel

Local variabel merupakan suatu pendeklarasian variabel di dalam ruang lingkup objek saja. Tidak dapat digunakan di dalam objek lain.

2. Global Variabel

Global variabel adalah merupakan suatu pendeklarasian variabel tidak hanya di dalam ruang lingkup objek saja. Tetapi mencakup semua presedure dan objek.

Konstanta

Konstanta merupakan suatu variabel yang mempunyai nilai tetap dan tidak dapat diubah nilainya.

Deklarasi Konstanta

Deklarasi konstanta :

```
{Const} Nama_Var [{AS typedata}]=<nilai>
```

Contoh :

```
Const nama_bidang as string="persegi_panjang"
```

```
Public const Panjang As Integer=8
```

```
Private const Lebar As Integer=5
```

```
Private const luas As Integer=Panjang*Lebar
```

```
Private const Keliling As Integer=2*(Panjang+Lebar)
```


Deklarasi Konstanta Enumerasi

Deklarasi Konstanta Enumerasi merupakan pendeklarasian daftar konstanta yang bersipat global . Dengan deklarasi konstanta enumerasi dapat membuat daftar konstanta sendiri dapam bentuk seperti modul kelas dan dapat diakses oleh projek lain.

Format penulisannya :

```
{Private | Public | .... }[enum] Nama_Variabel_konstanta
```

Contoh :

```
Const nama_bidang as string="persegi_panjang"
```

```
Private const Panjang As Integer=8
```

```
Private const Lebar As Integer=5
```

```
Private const luas As Integer=Panjang*Lebar
```

```
Private const Keliling As Integer=2*(Panjang+Lebar)
```

```
Public enum Hitung
```

```
    Luas = Luas
```

```
    Keliling=Keliling
```

```
end Enum
```

```
A=Hitung.Luas 'menggunakan variabel enum luas
```

```
B=Hitung.Kelilin ' menggunakan variabel enum keliling
```

Operator

Operator Aritmatika

Operator Aritmatika adalah merupakan suatu operator perhitungan matematika yang digunakan untuk proses perhitungan matematika. Operator aritmatika dapat dilihat pada table berikut ini :

Operator	Fungsi
+	Penjumlahan

-	Pengurangan
*	Perkalian
/	Pembagian
Mod	Sisa hasil pembagian
^	Perpangkatan

Operator Perbandingan

Operator perbandingan adalah merupakan suatu operator yang menyatakan suatu kondisi itu benar atau salah. Operator perbandingan dapat dilihat pada table berikut ini :

Operator	Fungsi
<>	Tidak sama dengan
<	Lebih kecil
>	Lebih besar
>=	Lebih besar sama dengan
<=	Lebih kecil sama dengan
Like	Ciri yang sama
Is	keterangan

Operator Logika

Operator Logika adalah merupakan suatu operator yang dapat membandingkan dua ekspresi atau lebih data logika (Boolean) dan menghasilkan data logika (Boolean) baru. Operator logika dapat dilihat pada table berikut ini:

Operator	Fungsi
NOT	Tidak
AND	Dan
OR	Atau
XOR	Ekklusif atau

Operator Penugasan

Operator penugasan adalah merupakan operator yang bertugas untuk menentukan nilai yang dikeluarkan oleh suatu variabel terhadap kondisi tertentu.

Operator	Fungsi
=	Sama dengan

Operator Penggabungan (Concatenation)

Operator Penggabungan adalah merupakan suatu operator yang dapat digunakan untuk menggabungkan dua buah string atau lebih. Operator Penggabungan dapat dilihat pada table berikut ini :

Operator	Fungsi
+	Plus
&	Dan

3. PRAKTEK

Variabel Lokal

1. Buat **project baru**, dengan nama **Prak3**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Variabel Lokal
Button1	Name	BtnExit
	Text	E&xit

4. Klik ganda pada BtnExit1, kemudian ketikkan kode program berikut ini :

```

Dim tutup As String
tutup = MessageBox.Show("Anda yakin tutup form ini ?", "Lokal Variabel",
MessageBoxButtons.YesNo, MessageBoxIcon.Question)
    If tutup = MsgBoxResult.Yes Then
        End
    Else
    End If

```

5. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
6. Simpan aplikasi **Anda**.

Variabel Global

1. Tambahkan form baru , dengan nama **Form2**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form2	Name	Form2
	Text	Variabel Global
Button1	Name	BtnExit
	Text	E&xit

4. Klik ganda pada Form2, kemudian ketikkan kode program berikut ini :

```
Public Class Form2  
Dim tutup As String
```

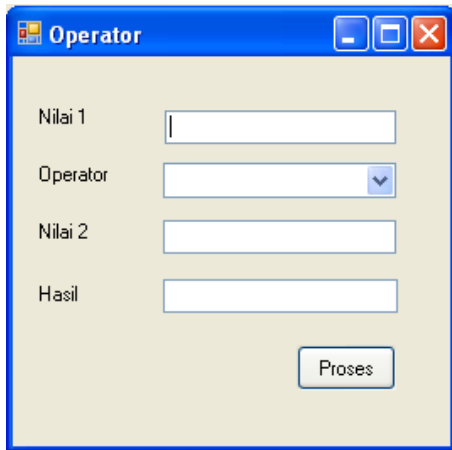
5. Klik ganda pada **BtnExit**, kemudian ketikkan kode program berikut ini :

```
tutup = MessageBox.Show("Anda yakin tutup form ini ?", "Global Variabel",  
MessageBoxButtons.YesNo, MessageBoxIcon.Question)  
If tutup = MsgBoxResult.Yes Then  
End  
Else  
End If
```

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
7. Simpan aplikasi **Anda**.

Operator

1. Tambahkan **Form baru**, dengan nama **Form3**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form3	Name	Form3
	Text	Operator
Label1	Text	Nilai1
Label2	Text	Operator
Label3	Text	Nilai2
Label4	Text	Hasil
TextBox1	Name	TxtNilai1
ComboBox1	Name	CmbOperator
TextBox2	Name	TxtNilai2
TextBox3	Name	TxtHasil
Button1	Name	BtnProses
	Text	Proses

4. Klik ganda pada Form3, kemudian ketikkan kode program berikut ini :

```
CmbOperator.Items.Add("+")
CmbOperator.Items.Add("-")
```

```
CmbOperator.Items.Add("x")
CmbOperator.Items.Add("/")
CmbOperator.Items.Add("Sisa")
CmbOperator.Items.Add("Pangkat")
CmbOperator.Items.Add("<>")
CmbOperator.Items.Add("<")
CmbOperator.Items.Add(">")
CmbOperator.Items.Add("<=")
CmbOperator.Items.Add(">=")
CmbOperator.Items.Add("Like")
CmbOperator.Items.Add("Not")
CmbOperator.Items.Add("And")
CmbOperator.Items.Add("Or")
CmbOperator.Items.Add("Xor")
CmbOperator.Items.Add("+p")
CmbOperator.Items.Add("&")
```

5. Klik ganda pada tombol Proses, lalu ketikkan kode program berikut ini :

```
Select Case CmbOperator.Text
Case "+"
    TxtHasil.Text = Val(TxtNilai1.Text) + Val(TxtNilai2.Text)
Case "-"
    TxtHasil.Text = Val(TxtNilai1.Text) - Val(TxtNilai2.Text)
Case "x"
    TxtHasil.Text = Val(TxtNilai1.Text) * Val(TxtNilai2.Text)
Case "/"
    TxtHasil.Text = Val(TxtNilai1.Text) / Val(TxtNilai2.Text)
Case "Sisa"
    TxtHasil.Text = Val(TxtNilai1.Text) mod Val(TxtNilai2.Text)
```

Case "Pangkat"

TxtHasil.Text = Val(TxtNilai1.Text) ^ Val(TxtNilai2.Text)

Case "<>"

TxtHasil.Text = TxtNilai1.Text <> TxtNilai2.Text

Case "<"

TxtHasil.Text = TxtNilai1.Text < TxtNilai2.Text

Case ">"

TxtHasil.Text = TxtNilai1.Text > TxtNilai2.Text

Case "<="

TxtHasil.Text = TxtNilai1.Text <= TxtNilai2.Text

Case ">="

TxtHasil.Text = TxtNilai1.Text >= TxtNilai2.Text

Case "Like"

TxtHasil.Text = TxtNilai1.Text Like TxtNilai2.Text

Case "Not"

TxtHasil.Text = Not TxtNilai1.Text = TxtNilai2.Text

Case "And"

TxtHasil.Text = TxtNilai1.Text = "STIKOM" And TxtNilai2.Text =
"STIKOM"

Case "Or"

TxtHasil.Text = TxtNilai1.Text = "STIKOM" Or TxtNilai2.Text =
"STIKOM"

Case "Xor"

TxtHasil.Text = TxtNilai1.Text = "STIKOM" Xor TxtNilai2.Text =
"STIKOM"

Case "+P"

TxtHasil.Text = TxtNilai1.Text + TxtNilai2.Text

Case "&"

TxtHasil.Text = TxtNilai1.Text & TxtNilai2.Text

End Select

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
7. Simpan aplikasi **Anda**.

PRAKTIKUM 5

PENCABANGAN

1. TUJUAN

- Memahami dan mengenal Pencabangan (if ... then dan Select Case)
- Dapat membuat aplikasi sederhana dengan menggunakan Pencabangan (if ... then dan Select Case ...)

2. TEORI DASAR

Pencabangan

Pencabangan adalah merupakan perintah yang dapat memberikan pilihan suatu kondisi, program akan menjalankan perintah apabila suatu kondisi memenuhi syarat tertentu. Pencabangan dapat dibedakan menjadi :

a. IF ... THEN

If ... then merupakan pencabangan yang mempunyai satu pencabangan atau satu blok perintah. Format penulisannya :

```
If kondisi then  
[perintah]  
End if
```

b. IF ... THEN ... ELSE

Suatu perintah pencabangan bersarang (Nested If) yang merupakan perkembangan dari perintah pencabangan IF.. THEN, yang dapat menjalankan satu blok perintah, yang memiliki dua nilai atau syarat bahkan lebih yang akan diuji untuk menjalankan suatu kondisi tertentu. Bila kondisi pertama benar maka jalankan perintah blok pertama, jika kondisi pertama salah maka jalankan perintah blok kedua dan selanjutnya.

Format Penulisannya :

```
IF (kondisi) then  
[perintah]  
ELSE  
[perintah]
```

```
End if
```

Dan

```
If (kondisi2) then  
[perintah]  
ELSEIF (kondisi2) Then  
[perintah]  
...  
ELSE  
[perintah]  
End If
```

c. **Select ... Case ...**

Select Case adalah control percabangan yang mempunyai fungsi hampir sama dengan percabangan if ... then... else. Select mempunyai penulisan dan pembacaan yang lebih mudah, efektif dan efisien. Namun mempunyai kelemahan yaitu tidak dapat menguji lebih dari satu ekspresi atau ungkapan. Format penulisannya adalah sebagai berikut :

Select Case kondisi

Case | Case is = ekspresi1

[perintah1]

Case | Case is = ekspresi2

[perintah2]

Case | Case is = ekspresi3

[perintah3]

...

Case Else

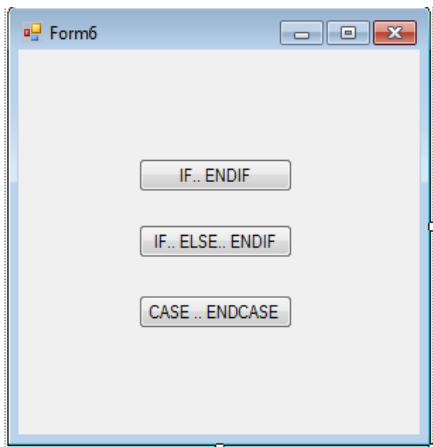
[perintah]

End Select

3. PRAKTEK

Pencabangan IF.. , IF.. ELSE, CASE..

1. Buat form baru.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Pencabangan IF
Button1	Name	Btn_if
	Text	IF.. ENDIF
Button2	Name	Btn_if_else
	Text	IF.. ELSE.. ENDIF
Button3	Name	Btn_case
	Text	CASE.. ENDCASE

4. Klik ganda pada Btn_if, kemudian ketikkan kode program berikut ini :

```
Private Sub Btn_if_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_if.Click
```

```

Dim usia As String, nilai As Single, thn As Integer
usia = InputBox("Usia Pegawai saat ini?", "Batasan usia pensiun", 0)
nilai = Val(usia)
If nilai >= 55 Then
    thn = nilai - 55
    MessageBox.Show("usia pegawai : " & usia & " Tahun", "Konfirmasi",
    MessageBoxButtons.OK, MessageBoxIcon.Information)
    MessageBox.Show("Keterangan : Pegawai harus pensiun " & thn & "
    tahun yang lalu", "Konfirmasi", MessageBoxButtons.OK,
    MessageBoxIcon.Information)
End If
End Sub

```

5. Klik ganda pada Btn_if_else, kemudian ketikkan kode program berikut ini :

```

Private Sub Btn_if_else_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btn_if_else.Click
    Dim usia As String, nilai As Single, thn As Integer
    usia = InputBox("Usia Pegawai saat ini?", "Batasan usia pensiun", 0)
    nilai = Val(usia)
    If nilai >= 55 Then
        thn = nilai - 55
        MessageBox.Show("usia pegawai : " & usia & " Tahun",
        "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
        MessageBox.Show("Keterangan : Pegawai harus pensiun " & thn &
        " tahun yang lalu", "Konfirmasi", MessageBoxButtons.OK,
        MessageBoxIcon.Information)
    Else
        thn = 55 - nilai
        MessageBox.Show("usia pegawai : " & usia & " Tahun",
        "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
        MessageBox.Show("Keterangan : Pegawai akan pensiun " & thn & "
        lagi", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
    End If
End Sub

```

```
End If
End Sub
```

6. Klik ganda pada Btn_case, kemudian ketikkan kode program berikut ini :

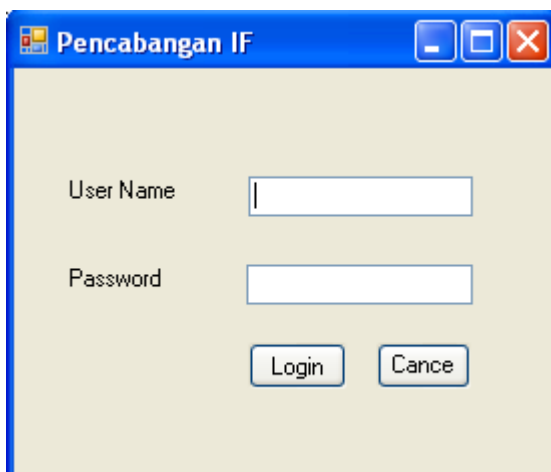
```
Private Sub Btn_case_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btn_case.Click
    Dim isian As String, usia As Integer
    isian = InputBox("Usia anda saat ini?", "Status Anda")
    usia = Val(isian)
    MessageBox.Show("Usia Anda saat ini : " & usia & " tahun",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
    Select Case usia
        Case 1 To 4
            MessageBox.Show("Anda sekarang tergolong balita",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
        Case 5 To 16
            MessageBox.Show("Anda sekarang tergolong anak-anak",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
        Case 17 To 24
            MessageBox.Show("Anda sekarang tergolong remaja",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
        Case 25 To 55
            MessageBox.Show("Anda sekarang tergolong dewasa",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
        Case Is >= 56
            MessageBox.Show("Anda sekarang tergolong manula",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
        Case Else
            MessageBox.Show("Anda salah memasukkan usia", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Information)
    End Select
```

End Sub

7. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
8. Simpan aplikasi **Anda**.

Pencabangan Bersarang (NESTED IF)

1. Buat **project baru**, dengan nama **Prak4**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Pencabangan IF
Label1	Text	User Name
Label2	Text	Password
TextBox1	Name	TxtUserName

TextBox2	Name PasswordChar	TxtPassword *
Button1	Name Text	BtnLogin Login
Button2	Name Text	BtnCancel Cancel

4. Klik ganda pada BtnLogin, kemudian ketikkan kode program berikut ini :

```

If TxtPassword.Text = "" And TxtUserName.Text = "" Then
    MessageBox.Show("User name dan Password harus diisi",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Question)
    TxtUserName.Focus()
Elseif TxtPassword.Text = "" Then
    MessageBox.Show("Password harus diisi", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Question)
    TxtPassword.Focus()
Elseif TxtUserName.Text = "" Then
    MessageBox.Show("User name harus diisi", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Question)
    TxtUserName.Focus()
Elseif TxtUserName.Text = "STIKOM" And TxtPassword.Text = "1234"
Then
    MessageBox.Show("Password dan User Name Benar", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Question)

Else
    MessageBox.Show("Password dan User Name Anda Salah, ulangi
lagi", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Question)
    TxtUserName.Text = ""
    TxtPassword.Text = ""
    TxtUserName.Focus()

```



```
End If  
End Sub
```

5. Klik ganda pada BtnCancel, kemudian ketikkan kode program berikut ini :

```
End
```

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start
7. Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
8. Simpan aplikasi **Anda**.

MODUL 6

Perulangan

1. TUJUAN

- Memahami dan mengenal Perulangan (For ... Next, While, Do While)
- Dapat membuat aplikasi sederhana dengan menggunakan Perulangan (For ... Next, While, Do While)

2. TEORI DASAR

PERULANGAN

Perintah perulangan adalah merupakan control perulangan yang berfungsi untuk melakukan perulangan (iterasi) dari suatu blok program secara berulang-ulang.

For ... Next

For ... Next merupakan suatu control perulangan yang mempunyai nilai tetap, jumlah perulangan telah diketahui sebelumnya. Format penulisannya adalah sebagai berikut :

```
For Counter=nilai_awal to nilai_akhir
```

```
[ Perintah ]
```

```
Next (Counter)
```

For Each ... In ... Next

Merupakan suatu control yang digunakan untuk perulangan yang jumlah pengulangannya sesuai dengan jumlah elemen dari suatu koleksi objek dan biasanya tidak diketahui dengan pasti berapa jumlahnya. Koleksi merupakan kumpulan data atau objek yang didefinisikan dengan kode program.

Format Penulisannya :

For Each elemen **In** Group

[Perintah]

[Perintah]

Next [elemen]

While ...

While merupakan suatu control perulangan yang berfungsi untuk melakukan perulangan yang memiliki satu syarat tertentu. Dan akan terus dijalankan selama syarat tersebut terpenuhi dan begitupun sebaliknya jika syarat tidak terpenuhi maka pernyataan tidak akan dijalankan.

Format Penulisannya :

While (kondisi)

[perintah1]

[perintah2]

...

End While

DO While ... LOOP

Merupakan suatu control perulangan yang melakukan perulangan terlebih dahulu dan bentuk pengujian dilakukan belakangan atau sebaliknya.

Format penulisannya :

DO While (kondisi)

[perintah1]

[Exit Do]

[perintah2]

LOOP

Bentuk 2

DO

[perintah1]

[Exit Do]

[perintah2]

LOOP While (kondisi)

DO UNTIL

Jika Do While melakukan perulangan selama kondisi memenuhi until akan melakukan perulangan hingga kondisi memenuhi. Selama kondisi tidak memenuhi, loop akan dijalankan terus.

Format penulisannya :

DO

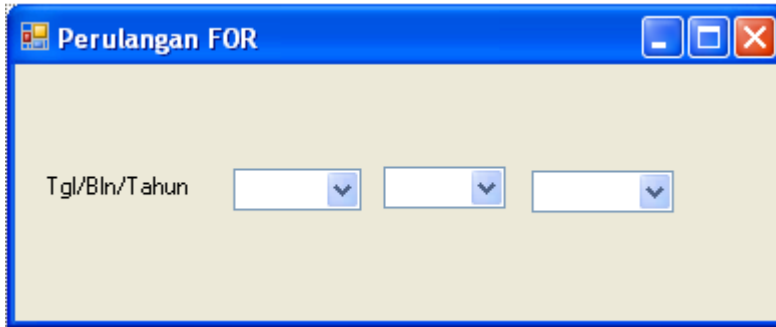
[perintah1]

LOOP UNTIL

3.LATIHAN

Perulangan FOR

1. Buat project baru dengan nama **Prak8**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Perulangan FOR
Label1	Text	Tgl/Bln/Thn
ComboBox1	Name	CmbTgl
ComboBox2	Name	CmbBln
ComboBox3	Name	CmbThn

4. Klik ganda pada Form1, kemudian ketikkan kode program berikut ini :

```
Dim tgl, bln, thn As Integer
```

```
For tgl = 1 To 31
```

```
    cmbTgl.Items.Add(tgl)
```

```
Next tgl
```

```
For bln = 1 To 12
```

```
    cmbBln.Items.Add(bln)
```

```
Next bln
```

```
For thn = 1900 To 9999
```

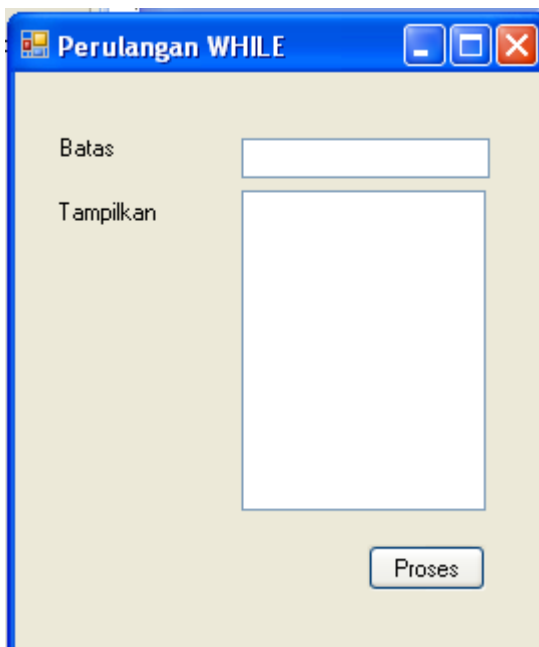
```
    CmbThn.Items.Add(thn)
```

Next thn

5. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
6. Simpan aplikasi **Anda**.

Perulangan WHILE

1. Tambahkan **Form baru**, dengan nama **Form3**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Perulangan WHILE
Label1	Text	Batas
Label2	Text	Tampilkan

TextBox1	Name	TxtBatas
ListBox1	Name	ListBox1
Button1	Name	Button1
	Text	Proses

4. Klik ganda pada Proses, kemudian ketikkan kode program berikut ini :

```

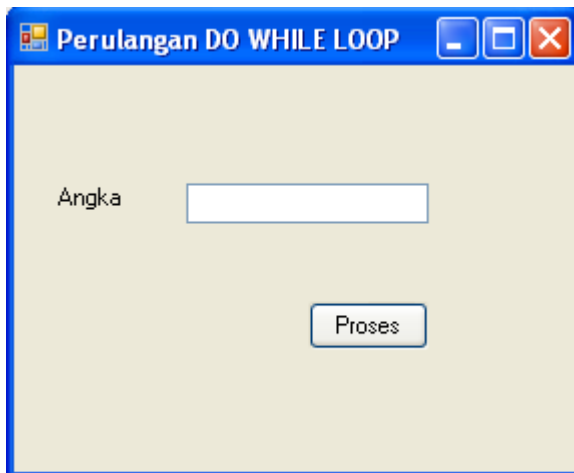
Dim angka As Integer
ListBox1.Items.Clear()
angka = 1
While angka <= Val(TxtBatas.Text)
    ListBox1.Items.Add(angka)
    angka = angka + 1
End While

```

5. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
6. Simpan aplikasi **Anda**.

Perulangan DO WHILE...LOOP

1. Tambahkan **Form baru**, dengan nama **Form4**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Perulangan DO WHILE ... LOOP
Label1	Text	Angka
TextBox1	Name	TxtAngka
Button1	Name	Button1
	Text	Proses

4. Klik ganda pada Proses, kemudian ketikkan kode program berikut ini :

```

Dim X%
X = TxtAngka.Text
Do
  X = X Mod 2
  If X = 0 Then
    MsgBox("Bilangan Genap")
  Exit Do
  ElseIf X = 1 Then

```

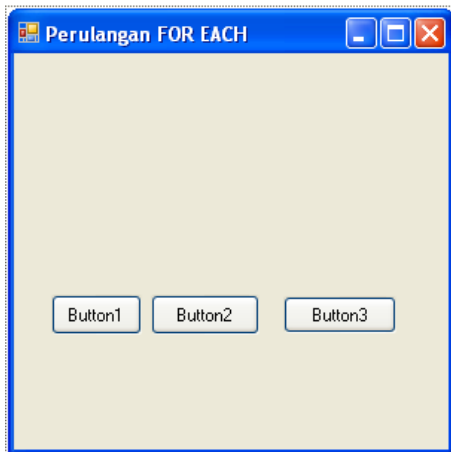


```
MsgBox("Bilangan Ganjil")
Exit Do
End If
Loop While Not X
```

5. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
6. Simpan aplikasi **Anda**.

Perulangan FOR EACH ... IN...NEXT

1. Tambahkan **Form baru**, dengan nama **Form5**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form5	Name	Form5
	Text	Perulangan FOR EACH ... IN...NEXT

Button1	Name Text	Button1 Button1
Button2	Name Text	Button2 Button2
Button3	Name Text	Proses Button3

4. Klik ganda pada Proses, kemudian ketikkan kode program berikut ini :

```
For Each ctrl In Controls  
    ctrl.Text = "Tombol"  
Next
```

5. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
6. Simpan aplikasi **Anda**.

VI. HASIL PRAKTIKUM

VII. ANALISA

VIII. KESIMPULAN

MODUL 7

ARRAY

1. TUJUAN

- Memahami dan mengenal Array (array satu dimensi dan array multi dimensi)
- Dapat membuat aplikasi sederhana dengan menggunakan array satu dimensi dan array multi dimensi

2. TEORI DASAR

Array adalah sekumpulan data yang memiliki tipe yang sama, sejumlah tetap, serta disusun secara terstruktur dan disimpan dalam satu variabel yang sama, dan diurutkan dengan index.

Array Satu Dimensi

Suatu array yang nilai dan ukurannya yang sudah ditentukan terlebih dahulu, dan memiliki satu dimensi.

Bentuk Umum :

Dim Array[indeks] As Tipe_Data

Contoh :

Dim Arr(2) As String

Arr(0) = TxtNim.Text

Arr(1) = TxtNama.Text

Arr(2) = TxtProdi.Text

Array Multi Dimensi

Suatu array yang fungsinya hampir sama dengan array satu dimensi hanya saja pada array multi dimensi ini mewakili nilai table yang terdiri dari informasi yang diatur dalam baris dan kolom. Untuk mendefenisikan elemen table tertentu, kita harus menentukan dua indeks, pertama mengidentifikasi elemen baris dan yang mengidentifikasi elemen kolom. Array multidimensi memiliki lebih dari dua dimensi.

Bentuk Umum :

Dim Array[indeks,Indeks] As Tipe_Data

Dim Array[indeks,indeks,indeks] As Tipe_Data

Contoh :

Dim Arr(3, 1) As String

Arr(0, 0) = "NIM"

Arr(0, 1) = "NAMA"

Arr(1, 0) = "JENIS KELAMIN"

Arr(1, 1) = "PRODI"

Arr(2, 0) = "Laki-laki"

Arr(2, 1) = "Perempuan"

Arr(3, 0) = "Manajemen Informatika"

Arr(3, 1) = "Teknik Informatika"

3. PRAKTEK

Array Satu Dimensi

1. Buat **project baru**, dengan nama **Prak9**.
2. Desain Tampilan form seperti gambar berikut ini :

NIM	NAMA	PRODI

3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Array Satu Dimensi
Label1	Text	NIM
Label2	Text	Nama
Label3	Text	Prodi
TextBox1	Name	TxtNIM
TextBox2	Name	TxtNama
TextBox3	Name	TxtProdi
ListView1	Name	ListView1
Button1	Name	BtnTambah
	Text	Tambah

4. Klik ganda pada Form Load, kemudian ketikkan kode program berikut ini :

```
ListView1.GridLines = True
ListView1.View = View.Details
ListView1.Columns.Add("NIM")
ListView1.Columns.Add("NAMA", 115)
ListView1.Columns.Add("PRODI", 110)
TxtNim.Text = 1
TxtNama.Focus()
```

5. Klik ganda pada BtnTambah, kemudian ketikkan kode program berikut ini :

```
Dim Arr(2) As String
Arr(0) = TxtNim.Text
Arr(1) = TxtNama.Text
```

```
Arr(2) = TxtProdi.Text
```

```
Dim listitem As ListViewItem
```

```
listitem = New ListViewItem
```

```
listitem = ListView1.Items.Add(Arr(0))
```

```
listitem.SubItems.Add(Arr(1))
```

```
listitem.SubItems.Add(Arr(2))
```

```
TxtNim.Text = TxtNim.Text + 1
```

```
TxtNama.Text = ""
```

```
TxtProdi.Text = ""
```

```
TxtNama.Focus()
```

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui icon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
7. Simpan aplikasi **Anda**.

Array Multi Dimensi

1. Tambahkan **Form baru**, dengan nama **Form2**.
2. Desain Tampilan form seperti gambar berikut ini :

NIM	NAMA	JENIS KELAMIN	PRODI

3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Array Multi Dimensi
Label1	Text	NIM
Label2	Text	Nama
Label3	Text	Kelamin
Label4	Text	Prodi
TextBox1	Name	TxtNIM
TextBox2	Name	TxtNama
Combo1	Name	CmbKelamin
Combo2	Name	CmbProdi
ListView1	Name	ListView1
Button1	Name	BtnProses
	Text	Proses

4. Klik ganda pada **Form Load**, kemudian ketikkan kode program berikut ini :

```
Dim Arr(3, 1) As String
Arr(0, 0) = "NIM"
Arr(0, 1) = "NAMA"
Arr(1, 0) = "JENIS KELAMIN"
Arr(1, 1) = "PRODI"
Arr(2, 0) = "Laki-laki"
Arr(2, 1) = "Perempuan"
Arr(3, 0) = "Manajemen Informatika"
Arr(3, 1) = "Teknik Informatika"

ListView1.GridLines = True
ListView1.View = View.Details
```

```

For Baris = 0 To 1
    For Kolom = 0 To 1
        ListView1.Columns.Add(Arr(Baris, Kolom), 100)
    Next Kolom
Next Baris

For Baris = 2 To 2
    For Kolom = 0 To 1
        CmbKelamin.Items.Add(Arr(Baris, Kolom))
    Next Kolom
Next Baris

For Baris = 3 To 3
    For Kolom = 0 To 1
        CmbProdi.Items.Add(Arr(Baris, Kolom))
    Next Kolom
Next Baris

```

5. Klik ganda pada **BtnProses**, kemudian ketikkan kode program berikut ini :

```

Dim Arr(3) As String
Arr(0) = TxtNIM.Text
Arr(1) = TxtNAMA.Text
Arr(2) = CmbKelamin.Text
Arr(3) = CmbProdi.Text

Dim listitem As ListViewItem
listitem = New ListViewItem
listitem = ListView1.Items.Add(Arr(0))
listitem.SubItems.Add(Arr(1))
listitem.SubItems.Add(Arr(2))
listitem.SubItems.Add(Arr(3))

```



```
TxtNIM.Text = TxtNIM.Text + 1  
TxtNAMA.Text = ""  
CmbKelamin.Text = ""  
CmbProdi.Text = ""  
TxtNAMA.Focus()
```

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui ikon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
7. Simpan aplikasi **Anda**.

MODUL 8

Procedure, Function dan Module

1. TUJUAN

- Memahami dan mengenal Procedure, function dan modul
- Dapat membuat aplikasi sederhana dengan menggunakan Procedure, function dan modul

2. TEORI DASAR

Sejauh ini, anda telah menuliskan sub procedure yang dibuat secara otomatis pada saat menggunakan event . Sub procedure merupakan blok kode yang mempunyai nama dan berisi perintah yang dapat dipanggil suatu waktu. Artinya, hanya sekali dibuat yang kemudian di panggil berkali-kali sesuai dengan kebutuhan.

Bentuk Umum :

```
[Private|Public] Sub subname[(argumentlist)]
```

```
    Statements
```

```
End sub
```

Sub Procedure dapat dipanggil dengan perintah :

```
[call] subname[(argumentlist)]
```

Keterangan :

Sub procedure (*Procedure*) sebaiknya menggunakan kata kerja (verb)

Pemanggilan procedure yang masih dalam satu kelas (class) atau modul (module) dapat dilakukan dengan kata kunci Me.

Procedure dapat menerima argument melalui nilai (by value) atau juga reference (by reference). Jika menggunakan by value, nilai pada argument tidak berubah. Sedangkan jika menggunakan by reference, nilai argument yang digunakan dapat berubah sesuai dengan perintah dalam procedure.

Secara default, argument dimasukkan melalui nilai (by value) dapat menggunakan kata kunci ByVal dan ByRef untuk By Reference.

Syntaks argument :

```
[ByVal|ByRef] variabelname as type
```

Contoh :

1. Procedure dengan argument di passing sebagai value

```
Sub HitungVal(ByVal as Decimal)
```

```
    A+=1
```

```
    textAkhir.text=a
```

```
end sub
```

2. Procedure dengan argument di passing sebagai value

```
Sub HitungRef(ByRef as Decimal)
```

```
    A+=1
```

```
    textAkhir.text=a
```

```
end sub
```

3. pemanggilan procedure

```
HitungVal(textAwal.Text)
```

Function (Fungsi)

Fungsi mempunyai banyak kemiripan dengan sub procedure. Bedanya, fungsi selalu mengembalikan nilai (return value)

Sintaks fungsi dituliskan sebagai berikut :

```
[Private|Public] function functionname[(argumentlist)] [As type]
```

```
    Statements{
```

```
        Functionname=expression | {return expression}
```

```
End function
```

Keterangan :

- Perbedaan antara fungsi dengan procedure hanyalah pada pengembalian nilai saja (return value). Sebagai contoh :
Menggunakan argument berupa nilai (by value) dan reference (by reference)
- Pengembalian nilai dapat dilakukan dengan menggunakan operator assignment “=” pada fungsi atau biasanya dengan perintah return.

Contoh :

Private Function cekData() as Boolean

```

if IsNumeric(TextAwal.text) then
    return true
else
    return false
endif

```

end function

Pemanggilan fungsi

```

if cekData()=true then
    HitungRef(textAwal.text)
else
    messageBox.Show("Data yang dimasukkan bukan angka", "Kesalahan data",
    MessageBoxButtons.OK,
    MessageBoxIcon.Error)
endif

```

Modul (Module)

Modul merupakan bagian yang sengaja dipisahkan untuk memudahkan pemrograman. Dalam module dapat dimasukkan procedure dan fungsi dan kemudian digunakan oleh beberapa form.

- Umumnya modul dideklarasikan secara public (public) agar dapat digunakan di dalam kode dimanapun
- Modul dapat diisi dengan main procedure yang akan dijalankan pertama kali saat program dimulai

- Modul merupakan salah satu argumentasi code reuse yang bertujuan agar program lebih efisien.

Sintaks pembuatan modul sebagai berikut :

```
Module moduleName
```

```
    Statement
```

```
End module
```

Contoh :

```
Module Module
```

```
    Public intResponse, RoleId, Msg As Integer
```

```
    Public Username, Password, Nama As String
```

```
End Module
```

3. PRAKTEK

Procedure dan Function

1. Buat **project baru**, dengan nama **Prak10**.
2. Desain Tampilan form seperti gambar berikut ini :

The image shows a screenshot of a Windows application window titled "BARANG". The window has a blue title bar with standard Windows window controls (minimize, maximize, close). The main content area is light beige and contains a form with the following elements:

- Label "Kode Barang" followed by a text input field.
- Label "Nama Barang" followed by a text input field.
- Label "Satuan" followed by a text input field.
- Label "Harga Satuan" followed by a text input field.
- Label "Jumlah" followed by a text input field.
- Label "Total Harga" followed by a text input field.
- At the bottom of the form, there are three buttons: "Hitung", "Clear", and "Exit". The "Hitung" button is highlighted with a dashed border, indicating it is the active or selected button.

3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Barang
Label1	Text	Kode Barang
Label2	Text	Nama Barang
Label3	Text	Satuan
Label4	Text	Harga
Label5	Text	Jumlah
Label6	Text	Total Harga
TextBox1	Name	TxtKodeBrg
TextBox2	Name	TxtNamaBrg
TextBox3	Name	TxtSatuan
TextBox4	Name	TxtHrgSatuan
TextBox5	Name	TxtHarga
TextBox6	Name	TxtTot
Button1	Name	BtnHitung
	Text	Hitung
Button2	Name	BtnClear
	Text	Clear
Button3	Name	BtnExit
	Text	Exit

4. Klik ganda pada Form, kemudian buatlah sub procedure **Bersih** berikut ini :

```
Sub Bersih()  
    TxtKodeBrg.Text = ""  
    TxtNamaBrg.Text = ""  
    TxtSatuan.Text = ""  
    TxtHarga.Text = ""  
    TxtHrgSatuan.Text = ""
```

```
TxtTot.Text = ""  
TxtKodeBrg.Focus()  
TxtTot.Enabled = False  
End Sub
```

5. Panggil procedure yang anda buat diatas dengan cara, klik ganda pada form load ketikkan kode program berikut ini :

```
Call Bersih()
```

6. Panggil procedure yang anda buat diatas pada BtnClear dengan cara, klik ganda pada BtnClear kemudian ketikkan kode program berikut ini :

```
Call Bersih()
```

7. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui icon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
8. Buatlah Function Hitung, berikut ini :

```
Private Function Hitung()  
    Dim Harga, HrgSatuan, Total As Integer  
    Harga = TxtHarga.Text  
    HrgSatuan = TxtHrgSatuan.Text  
    Total = Harga * HrgSatuan  
    TxtTot.Text = Total  
    Return Total  
End Function
```

9. Buatlah **Procedure CekDataKosong** dan sekaligus memanggil **function Hitung**, berikut ini :

```
Sub CekDataKosong()  
    If TxtKodeBrg.Text = "" Then
```

```

MessageBox.Show("Kode Barang harus diisi", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Warning)
TxtKodeBrg.Focus()
Elseif TxtNamaBrg.Text = "" Then
MessageBox.Show("Nama Barang harus diisi", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Warning)
TxtNamaBrg.Focus()
Elseif TxtSatuan.Text = "" Then
MessageBox.Show("Satua Barang harus diisi", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Warning)
TxtSatuan.Focus()
Elseif TxtHarga.Text = "" Then
MessageBox.Show("Harga Barang harus diisi", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Warning)
TxtHarga.Focus()
Elseif TxtHrgSatuan.Text = "" Then
MessageBox.Show("Harga Satuan Barang harus diisi", "Konfirmasi",
MessageBoxButtons.OK, MessageBoxIcon.Warning)
TxtHrgSatuan.Focus()
Else
    Call Hitung()
End If
End Sub

```

10. Memanggil procedure CekDataKosong dengan cara, klik ganda pada tombol BtnHitung:

```
Call CekDataKosong()
```

11. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui icon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.

12. Klik ganda pada tombol BtnExit, Kemudian ketikkan kode program berikut ini :

Dim Tutup As String

```
Tutup = MessageBox.Show("Yakin tutup form ini ?", "Konfirmasi",  
MessageBoxButtons.YesNo, MessageBoxIcon.Question)
```

```
If TUTUP = MsgBoxResult.Yes Then
```

```
    End
```

```
Else
```

```
    Exit Sub
```

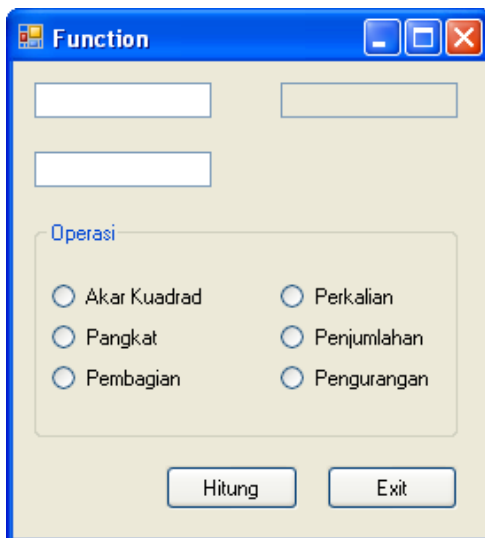
```
End If
```

13. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui icon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.

14. Simpan aplikasi **Anda**.

Function

1. Tambahkan **Form baru**, dengan nama **Form2**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name Text	Form1 Function
TextBox1	Name	TxtVar1
TextBox2	Name	TxtVar2
TextBox3	Name	TxtHasil
GroupBox1	Text	Operasi
RadioButton1	Name Text	RdbAkar Akar Kuadrat
RadioButton2	Name Text	RdbPangkat Pangkat
RadioButton3	Name Text	RdbPembagian Pembagian
RadioButton4	Name Text	RdbPerkalian Perkalian
RadioButton5	Name Text	RdbPenjumlahan Penjumlahan
RadioButton6	Name Text	RdbPengurangan Pengurangan
Button1	Name Text	BtnHitung Hitung
Button2	Name Text	BtnExit Exit

4. Klik kanan pada form, kemudian pilih menu view, kemudian ketik deklarasi masing-masing fungsi sebagai berikut ini :

```
Function Akar(ByVal a As Decimal) As String
    Return Math.Sqrt(a)
End Function
```

```
Function Pangkat(ByVal a As Decimal, ByVal b As Decimal) As String
```

```

    Return a ^ b
End Function
Function Pembagian(ByVal a As Decimal, ByVal b As Decimal) As String
    Return a / b
End Function
Function Perkalian(ByVal a As Decimal, ByVal b As Decimal) As String
    Return a * b
End Function
Function Penjumlahan(ByVal a As Decimal, ByVal b As Decimal) As
String
    Return a + b
End Function
Function Pengurangan(ByVal a As Decimal, ByVal b As Decimal) As
String
    Return a - b
End Function

```

5. Klik ganda pada Tombol Hitung, kemudian ketik kode program berikut ini :

```

If RdbAkar.Checked = True Then
    TxtHasil.Text = Akar(TxtVar1.Text)
Elseif RdbPangkat.Checked = True Then
    TxtHasil.Text = Pangkat(TxtVar1.Text, TxtVar2.Text)
Elseif RdbPembagian.Checked = True Then
    TxtHasil.Text = Pembagian(TxtVar1.Text, TxtVar2.Text)
Elseif RdbPerkalian.Checked = True Then
    TxtHasil.Text = Perkalian(TxtVar1.Text, TxtVar2.Text)
Elseif RdbPenjumlahan.Checked = True Then
    TxtHasil.Text = Penjumlahan(TxtVar1.Text, TxtVar2.Text)
Elseif RdbPengurangan.Checked = True Then
    TxtHasil.Text = Pengurangan(TxtVar1.Text, TxtVar2.Text)
End If

```

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui icon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
7. Simpan aplikasi **Anda**

PRAKTIKUM 9

Exception Handling

1. TUJUAN

- Memahami dan mengenal Exception Handling
- Dapat membuat aplikasi sederhana dengan menggunakan Exception Handling.

2. TEORI DASAR

EXCEPTION HANDLING

Perintah penanganan kesalahan atau yang lebih dikenal dengan sebutan Exception Handling Perintah ini digunakan untuk menangani kesalahan dalam menjalankan aplikasi, dengan adanya perintah ini setiap kesalahan akan diatasi secara otomatis sesuai dengan perintah penanganan kesalahan yang telah dibuat, sehingga tidak terjadi kemacetan aplikasi yang sedang berjalan.

Struktur Exception Handling

Struktur Exception Handling atau penanganan kesalahan pada visual basic 2008 antara lain :

Try ... Catch ... Finally ... End Try

Merupakan perintah penanganan kesalahan yang berfungsi untuk menangani kesalahan dalam menjalankan aplikasi. Bentuk penulisan :

Try

[Perintah]

Catch

[Tampilkan kesalahan]

Finally

[Kode program setelah perintah Try dan Catch]

End Try

Contoh :

```
Try
```

```
Dim X%
```

```
X = TxtAngka.Text
```

```
Catch ex As Exception
```

```
MsgBox(ex.ToString)
```

```
MsgBox("Input angka saja")
```

```
Finally
```

```
TxtAngka.Text=""
```

```
TxtAngka.Focus()
```

```
End Try
```

Throw

Throw merupakan perintah penanganan kesalahan yang berfungsi untuk menangani kesalahan apabila perintah try...catch ... Finally ... End. Try tidak dapat menangani suatu kesalahan dari jawaban aplikasi.

Contoh penulisan :

Try

```
[ Perintah ]
```

Catch ex As Exception

```
[Tampilkan kesalahan]
```

Throw ex

```
// mengembalikan ke kode pemanggil
```

Finally

[Kode program setelah perintah Try dan Catch]

Perintah ini bersifat optional bias dibuat bias tidak

End Try

Contoh :

```
Try
Dim X%
X = TxtAngka.Text
Catch ex As Exception
MsgBox(ex.ToString)
MsgBox("Input angka saja")
Throw ex
Finally
TxtAngka.Text=""
TxtAngka.Focus()
End Try
```

On Error Resume Next

Merupakan perintah penanganan kesalahan apabila terjadi kesalahan terhadap suatu baris tertentu maka baris tersebut akan di abaikan.

Contoh :

```
On Error Resume Next
Dim A As Integer
X = TextBox1.Text
TextBox1.Text="String"
MsgBox(ex.ToString)
```

MsgBox("Abaikan Kesalahan")

On Error Goto

Merupakan perintah penanganan kesalahan apabila terjadi kesalahan terhadap suatu baris tertentu maka perintah akan melompat ke baris yang dituju akan dijalankan.

Contoh :

On Error Goto Pesan

Dim A As Integer

X = TextBox1.Text

TextBox1.Text="String"

Exit Sub

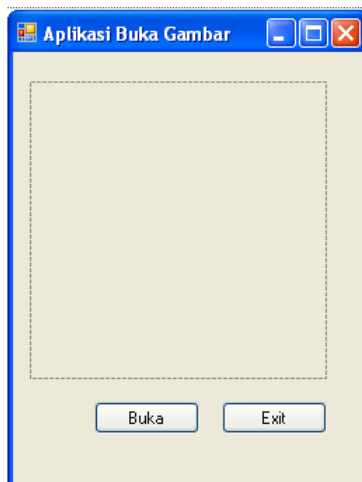
Pesan :

MsgBox("Abaikan Kesalahan")

3. PRAKTEK

Latihan 1

1. Buat **project baru**, dengan nama **Prak12**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Aplikasi Buka Gambar
OpenFileDialog1	Text	ofdBuka
PictureBox1	Name	PcbGambar
Button1	Name	BtnBuka
	Text	Buka
Button2	Name	BtnExit
	Text	Exit

4. Klik ganda pada Tombol Buka, kemudian ketik kode program berikut ini :

```
If OfdBuka.ShowDialog() = Windows.Forms.DialogResult.OK Then
    Try
        PcbGambar.Image = Image.FromFile(OfdBuka.FileName)
        Me.Text = "Membuka File " + OfdBuka.FileName
    Catch ex As Exception
        MessageBox.Show("File Gagal dibuka", "Error", MessageBoxButtons.OK,
            MessageBoxIcon.Error)
    End Try
End If
```

5. Klik ganda pada tombol BtnExit, Kemudian ketikkan kode program berikut ini :

```
Dim Tutup As String
Tutup = MessageBox.Show("Yakin tutup form ini ?", "Konfirmasi",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question)
```

```

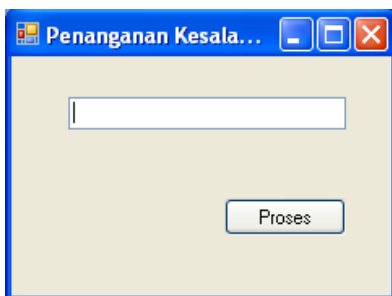
If Tutup = MsgBoxResult.Yes Then
    End
Else
    Exit Sub
End If

```

6. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui icon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
7. Simpan aplikasi **Anda**.

Latihan 2

1. Tambahkan **Form baru**, dengan nama **Form2**.
2. Desain Tampilan form seperti gambar berikut ini :



3. Tambahkan kontrol ke form seperti pada gambar diatas dan atur propertinya seperti tabel berikut:

Object	Properties	Nilai
Form1	Name	Form1
	Text	Penanganan Kesalahan
TextBox1	Name	TxtAngka
Button1	Name	BtnProses
	Text	Proses

4. Double klik pada tombol BtnProses, kemudian ketik kode Program berikut ini :

```
Try
  Dim X As Integer
  X = TxtAngka.Text
  Do
    X = X Mod 2
    If X = 0 Then
      MessageBox.Show("Bilangan Genap", "Informasi")
      Exit Do
    ElseIf X = 1 Then
      MessageBox.Show("Bilangan Ganjil", "Informasi")
      Exit Do
    End If
  Loop While Not X

  Catch ex As Exception
    MsgBox("Salah menginput data / input angka")
  Finally
    TxtAngka.Text=""
    TxtAngka.Focus()
  End Try
```

5. Jalankan aplikasi dengan menekan tombol **F5** (di keyboard), atau melalui icon Start Debugging di toolbar, atau melalui menu **Debug > Start Debugging**.
6. Simpan aplikasi **Anda**