

Pengembangan Aplikasi Bisnis

Menjelajahi Visual C# Toolbox



MIKROSKIL
INFORMATION SYSTEMS PROGRAM

Overview

- Aplikasi windows dibangun dengan menggunakan fitur Visual C# yang memiliki Graphical User Interface (GUI). Pengguna dapat berinteraksi dengan seperangkat tools visual (seperti: button, textbox, toolbar, menu, dll) untuk membuat aplikasi yang dapat bekerja sesuai dengan persyaratan dan kebutuhan pengguna.
- Dalam sebuah Program biasanya user dapat bekerja dengan pilihan-pilihan. Pilihan tersebut dapat menggunakan kontrol RadioButton maupun CheckBox. Kedua kontrol tersebut disebut sebagai kontrol ‘point and click’ yang memudahkan user untuk membuat pilihan dengan meng-klik menggunakan mouse sehingga mengurangi tingkat error pada aplikasi. Pilihan tersebut dapat dikelompokkan berdasarkan kriteria tertentu pada kontrol GroupBox maupun Panel.

Overview (Cont.)

- Kontrol CheckBox memang berguna untuk memilih item dari suatu daftar. Namun, bagaimana kalau daftar tersebut terdiri dari 100 item? Apakah kita akan menggunakan 100 CheckBox? ListBox dan ComboBox merupakan kontrol yang dapat menyelesaikan permasalahan tersebut. ListBox dan ComboBox digunakan sebagai tempat untuk menampilkan daftar pilihan. Dengan ListBox dan ComboBox user dapat memilih pilihan yang ada pada ListBox maupun ComboBox tersebut.
- Dalam aplikasi berbasis windows form, terdapat banyak kontrol yang disediakan oleh Visual C#, seperti MessageBox dan DialogBox. MessageBox digunakan untuk menyajikan pesan kepada pengguna. DialogBox digunakan untuk berinteraksi dengan pengguna dan mengambil informasi.

Objectives

Setelah menyelesaikan Unit ini, mahasiswa diharapkan mampu:

- Membedakan Object, Property, Method, dan Event
- Mengetahui langkah-langkah pengembangan suatu aplikasi bisnis
- Memahami dasar-dasar pengoperasian Microsoft Visual Studio 2019
- Memahami jenis-jenis kontrol dan cara penamaan pada sebuah kontrol
- Memahami karakteristik dan penggunaan dari Form
- Memahami karakteristik dan penggunaan dari kontrol Button, Label dan TextBox
- Memahami karakteristik dan penggunaan dari kontrol GroupBox, Panel, CheckBox, dan RadioButton
- Memahami karakteristik dan penggunaan dari kontrol ListBox dan ComboBox
- Memahami karakteristik dan penggunaan dari kontrol MessageBox
- Memahami karakteristik dan penggunaan dari kontrol DialogBox

Contents

- Object, Property, Method & Event.
- Langkah-langkah pengembangan aplikasi bisnis.
- IDE Microsoft Visual Studio 2019
- Mengenal Kontrol
- Form
- Button, Label, dan TextBox
- GroupBox, Panel, CheckBox, dan RadioButton
- ListBox dan ComboBox
- MessageBox dan DialogBox

Object, Property, Method & Event

MIKROSKIL

Lesson Overview

- Aplikasi windows Visual C# bersifat event-driven, yang berarti aplikasi tidak akan melakukan apapun sampai aplikasi tersebut dipanggil untuk merespon suatu event (mis: menekan tombol, memilih menu, dst.).



Objectives

Setelah mempelajari lesson ini, mahasiswa diharapkan mampu:

- Memahami Istilah Object, Property, Method, dan Event.

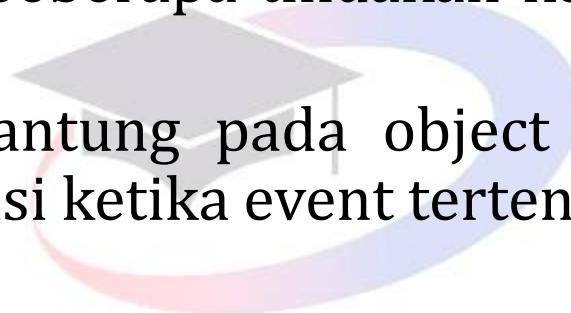


Project

- Aplikasi (project) terdiri dari:
 - Objects
 - Forms : windows yang dirancang sebagai antarmuka aplikasi
 - Controls : fitur grafis yang dimasukkan dalam form untuk memungkinkan interaksi pengguna (mis: textbox, label, button, dll.)
 - Properties : setiap karakteristik dari form atau control. Contoh: names, captions, size, color, position, dll. Dalam Visual C#, sudah terdapat nilai default pada property. Nilai pada property dapat diganti ketika mendesain aplikasi ataupun dengan koding.

Project (Cont.)

- Methods : metode built-in yang dapat dipanggil untuk memberikan beberapa tindakan ke kontrol atau objek tertentu
- Events : tergantung pada object atau control. Kode akan dieksekusi ketika event tertentu terjadi.



MIKROSKIL

Project (Cont.)

Project

Form 1

Control

Control

Control

{Code}

Form 2

Control

Control

Control

{Code}

Form 3

Control

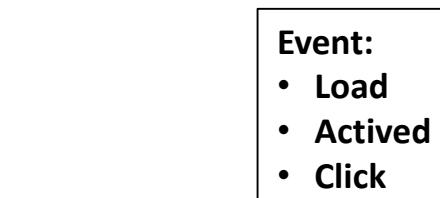
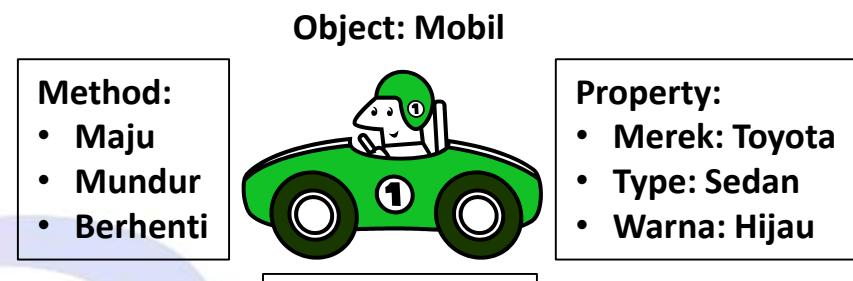
Control

Control

{Code}

Object, Property, Method & Event

- **Object** (Objek): Komponen di dalam sebuah program.
- **Property** (Properti): sifat atau karakteristik yang dimiliki objek.
- **Method** (Metoda): aksi, tindakan, atau perbuatan yang dapat dilakukan oleh objek.
- **Event** (Kejadian): Kejadian yang dapat diterima atau dialami oleh objek (biasanya dilakukan oleh user terhadap object).



Lesson Summary

Mahasiswa sekarang mampu:

- Memahami Istilah Object, Property, Method, dan Event.

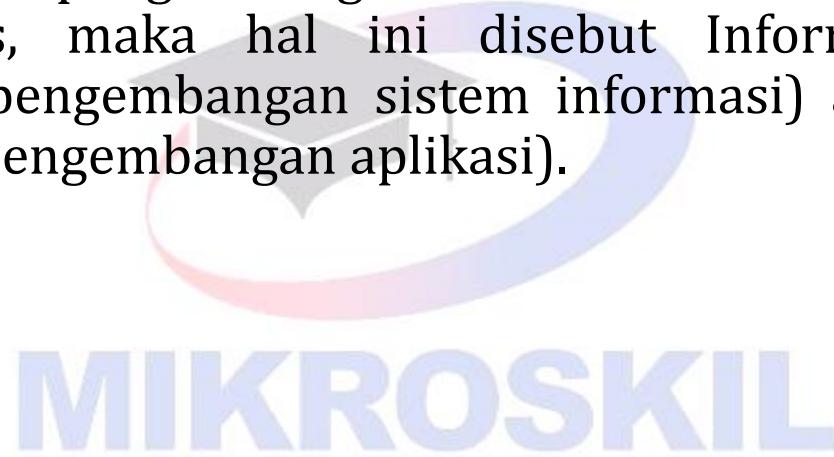


Langkah-langkah Pengembangan Aplikasi Bisnis

MIKROSKIL

Lesson Overview

Ketika pengembangan sistem untuk penyelesaian masalah diterapkan untuk pengembangan solusi sistem informasi terhadap masalah bisnis, maka hal ini disebut Information Systems Development (pengembangan sistem informasi) atau Application Development (pengembangan aplikasi).



Objectives

Setelah mempelajari Lesson ini, mahasiswa diharapkan mampu mengetahui langkah-langkah dalam mengembangkan suatu aplikasi bisnis.



Langkah-Langkah Pengembangan Aplikasi Bisnis

1. Merancang *user interface* dengan memasukkan control pada form.
2. Mengatur *property* yang dibutuhkan pada kontrol.
3. Menulis kode program berdasarkan *event* kontrol.



Lesson Summary

- Sekarang mahasiswa sudah mampu mengetahui langkah-langkah dalam mengembangkan suatu aplikasi bisnis.



IDE Microsoft Visual Studio 2019

MIKROSKIL

Lesson Overview

- Beberapa fitur dari Visual C#:
 - Integrated Development Environment (IDE) yang mudah digunakan
 - Kontrol yang lengkap
 - Merespon terhadap aksi mouse dan keyboard
 - Akses ke clipboard dan printer
 - Fungsi matematika, string, dan grafik yang lengkap
 - Dapat bekerja dengan sekumpulan array, variabel dan objek
 - Mendukung sequential file
 - Fasilitas debugger dan penanganan kesalahan
 - Tools grafik yang mudah digunakan
 - Tools akses database yang berguna
 - Kemampuan untuk mengembangkan aplikasi Windows dan internet dengan teknik yang sama
 - Module common language runtime baru membuat distribusi aplikasi menjadi hal yang mudah

Objectives

Setelah mempelajari Lesson ini, mahasiswa diharapkan mampu:

- Memahami dan mengerti cara penggunaan dan pengoperasian Microsoft Visual Studio 2019

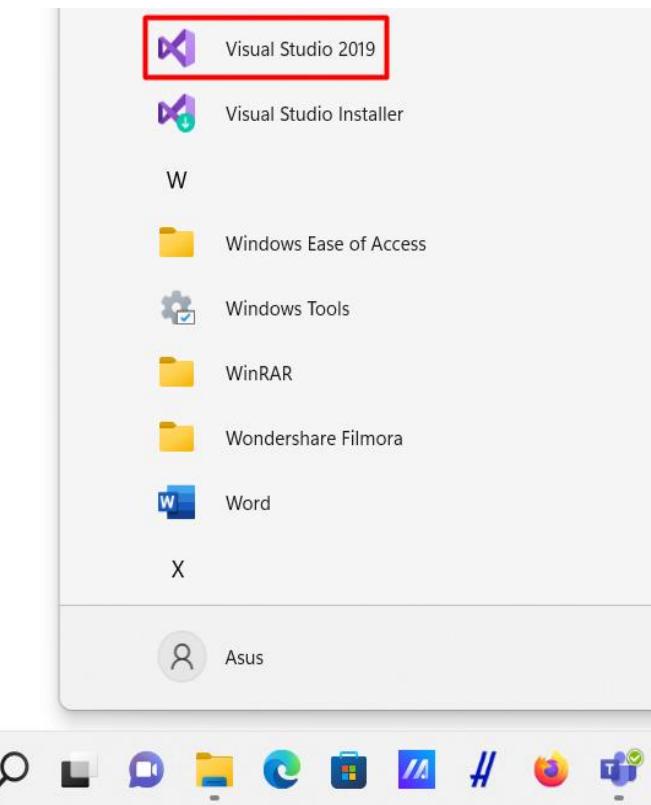


Antarmuka Program

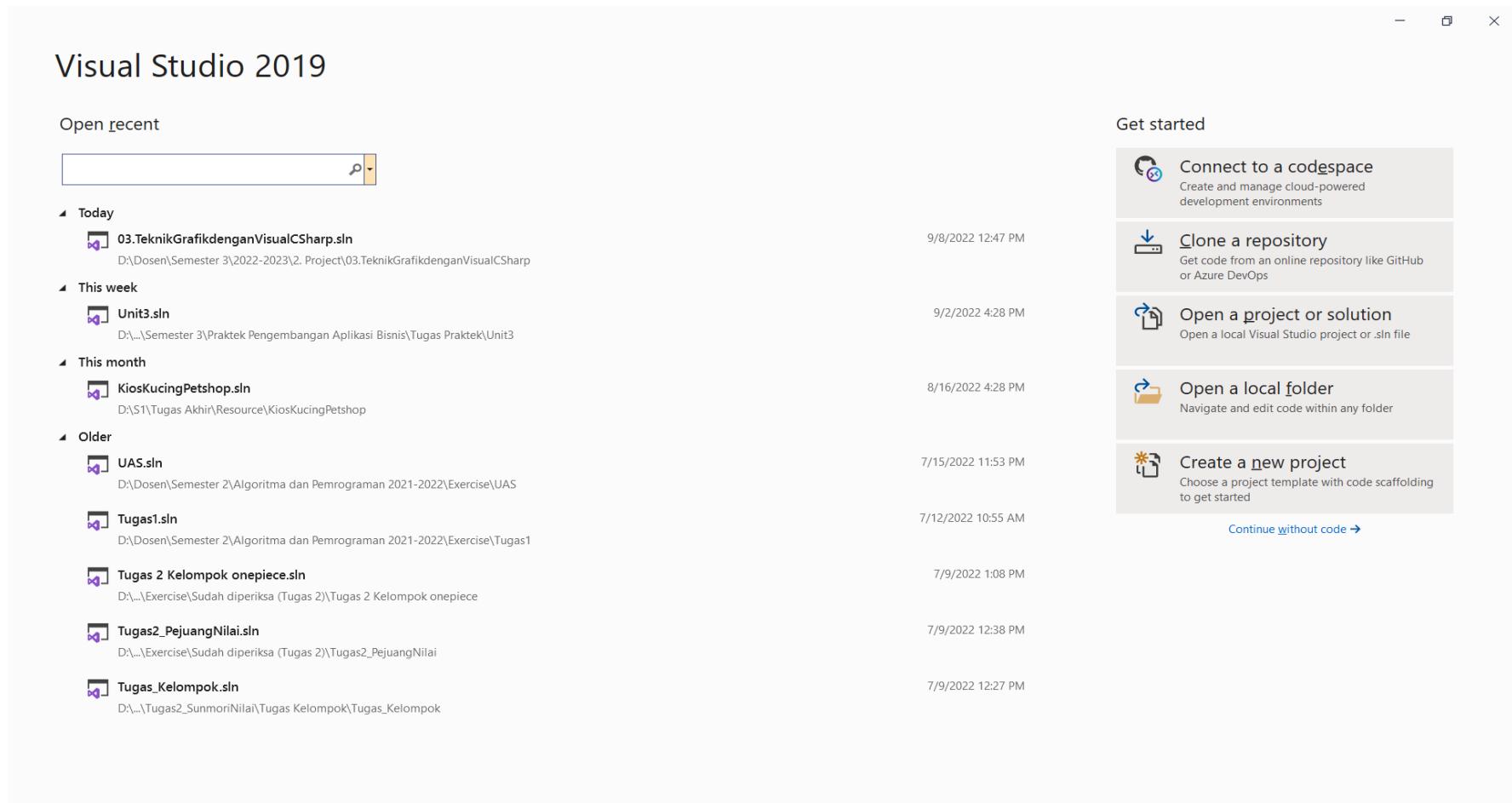
- Terdiri dari:
 - Windows Form.
 - Web Service dan Web Forms (ASP .NET).
- Bahasa pemrograman yang dapat digunakan untuk membangun aplikasi .NET: APL (A Programming Language), Visual C#, Cobol, Component Pascal, Curriculum, Eiffel, Forth, Fortran (Formula Translator), Haskell, Java, Jscript, Mercury, Mondrian, Oberon, Oz, Pascal, Perl, Python, RPG, Scheme, Smalltalk, Standar ML, Visual Basic, dan Visual C++.

IDE Microsoft Visual Studio 2019

- IDE = Integrated Development Environment
- Untuk mengaktifkan IDE dari Microsoft Visual Studio 2019 dapat dilakukan melalui menu start.
 - Pilih All Apps
 - Kemudian cari Visual Studio 2019
 - Klik pada Visual Studio 2019



Tampilan Halaman Awal



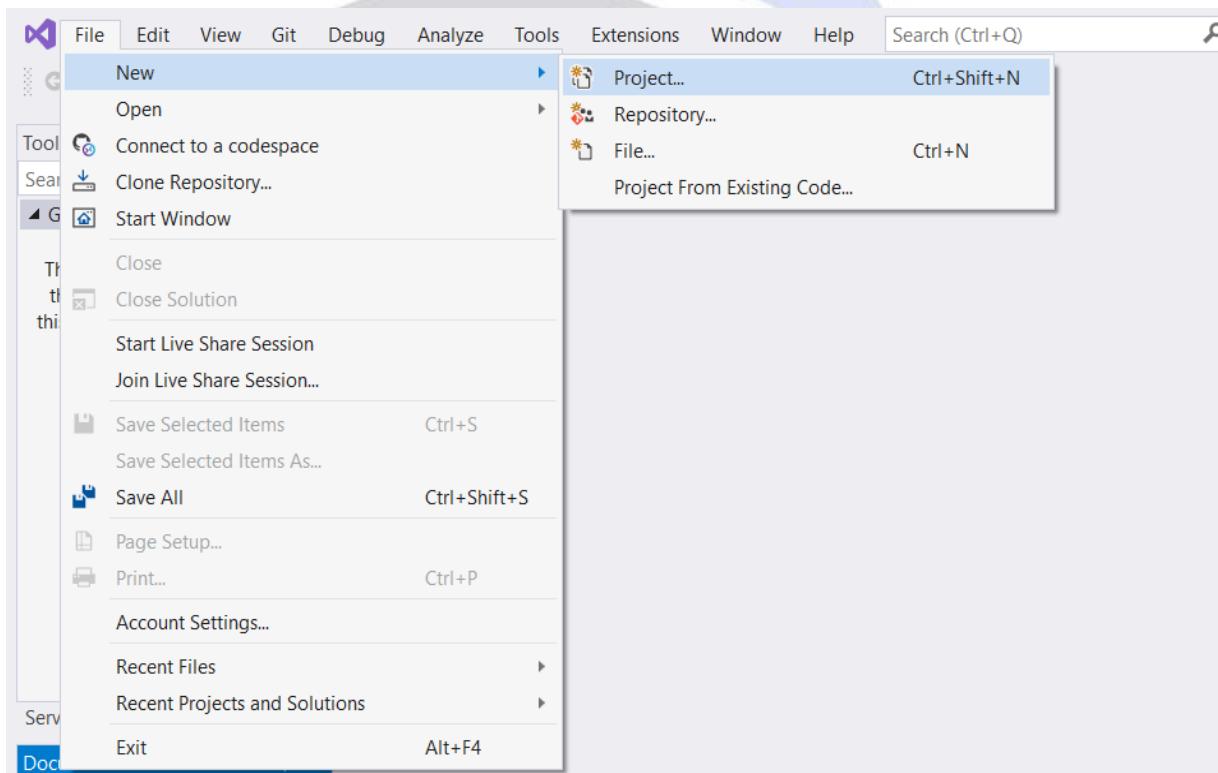
Tampilan Halaman Awal (Cont.)

Ada banyak informasi yang berguna di halaman awal (start page)

1. Pada bagian kiri, terdapat fitur untuk menampilkan project-project yang terakhir kali dibuka.
2. Pada bagian kanan terdapat fitur untuk:
 - Connect to a codespace: membuat dan mengelola lingkungan pengembangan yang didukung cloud
 - Clone a repository: mendapatkan kode dari repositori online seperti GitHub atau Azure DevOps
 - Open a project or solution: membuka project Visual Studio lokal atau file .sln
 - Open a local folder: membuka dan mengedit kode di dalam folder mana pun
 - Create a new project: membuat project baru dengan memilih project template

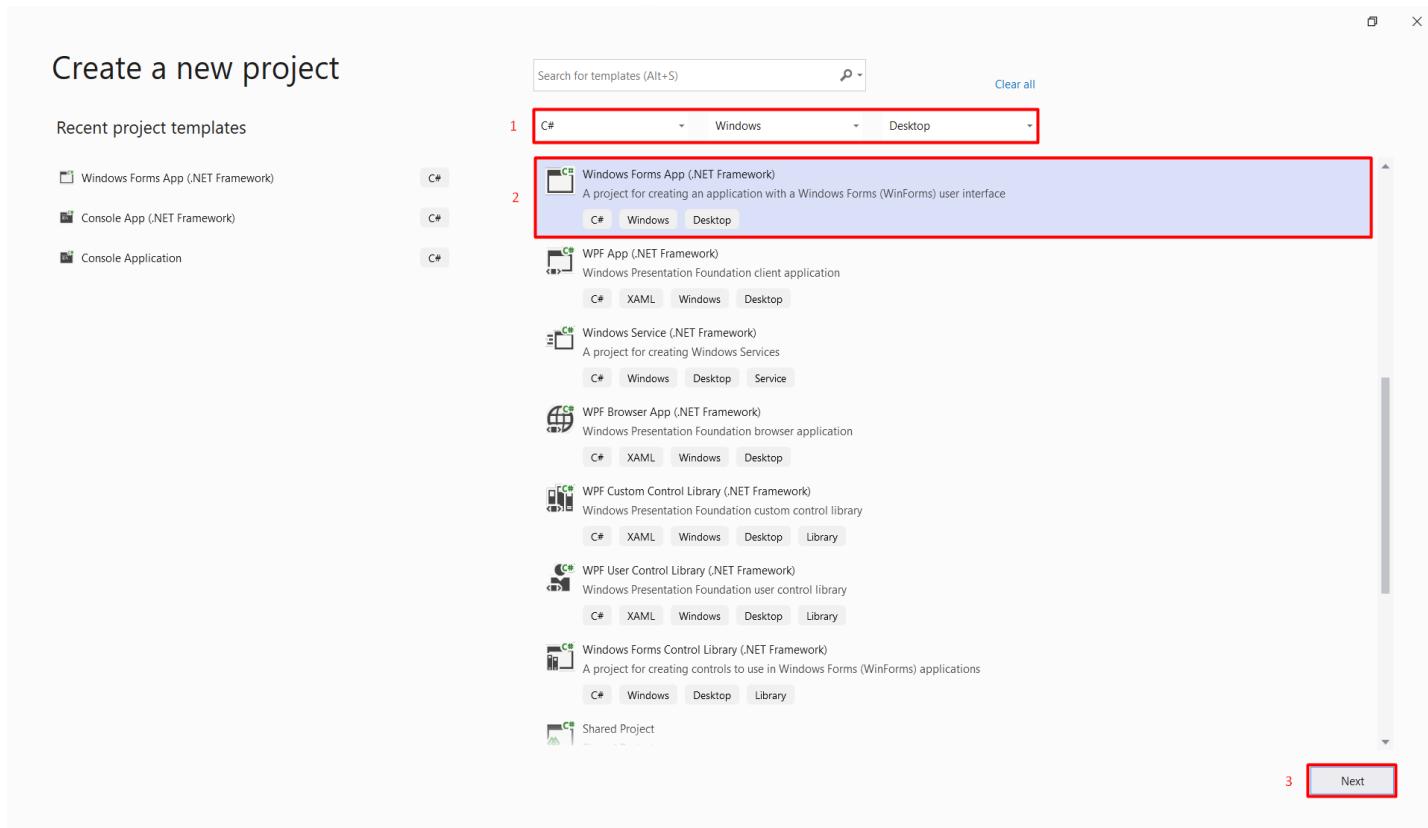
Membuat Project Baru

- Untuk membuat project baru, kita dapat menekan tombol **Create a new Project** yang berada di sebelah kanan Start Page atau dengan memilih menu **File → New → Project (Ctrl + Shift + N)** untuk menampilkan kotak dialog new project.



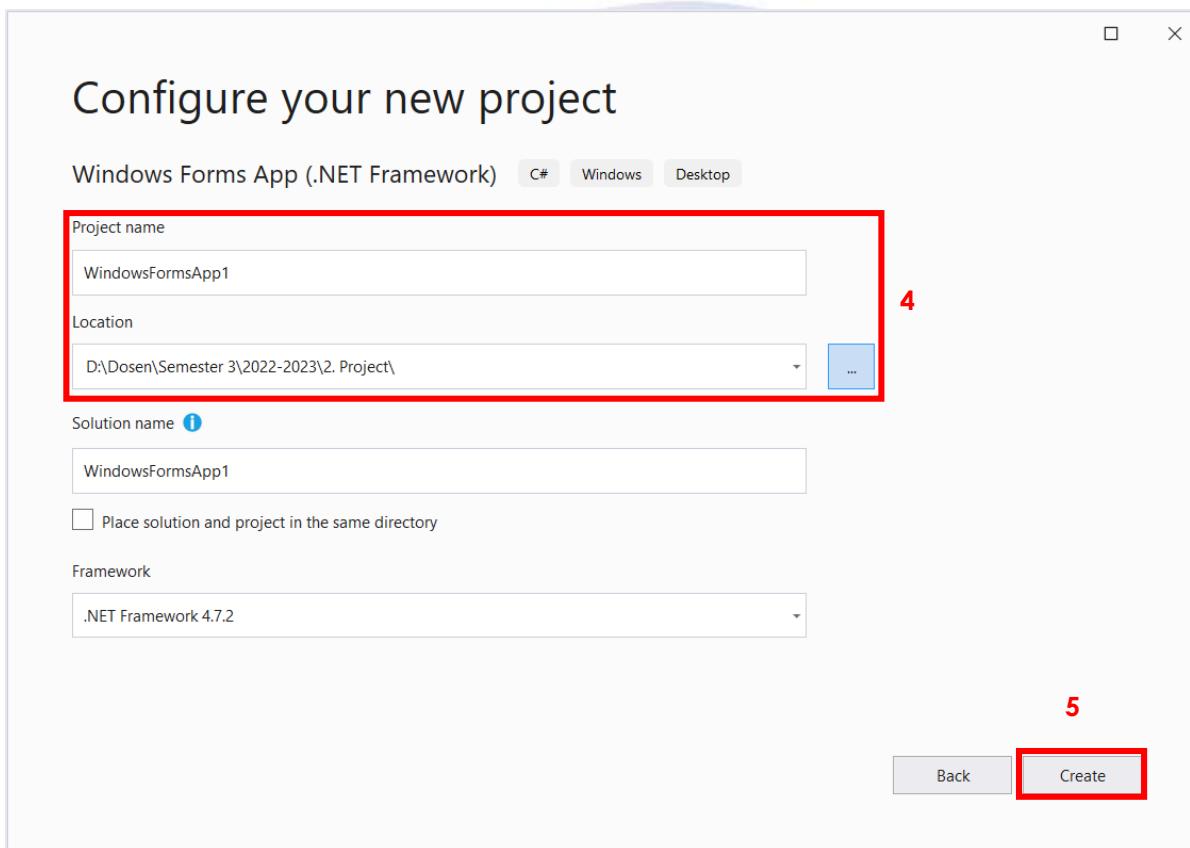
Membuat Project Baru (Cont.)

2. Tentukan bahasa pemrograman **C#**, platform **Windows**, dan jenis project **Desktop** → pilih template **Windows Forms App (.NET Framework)**. klik Next.



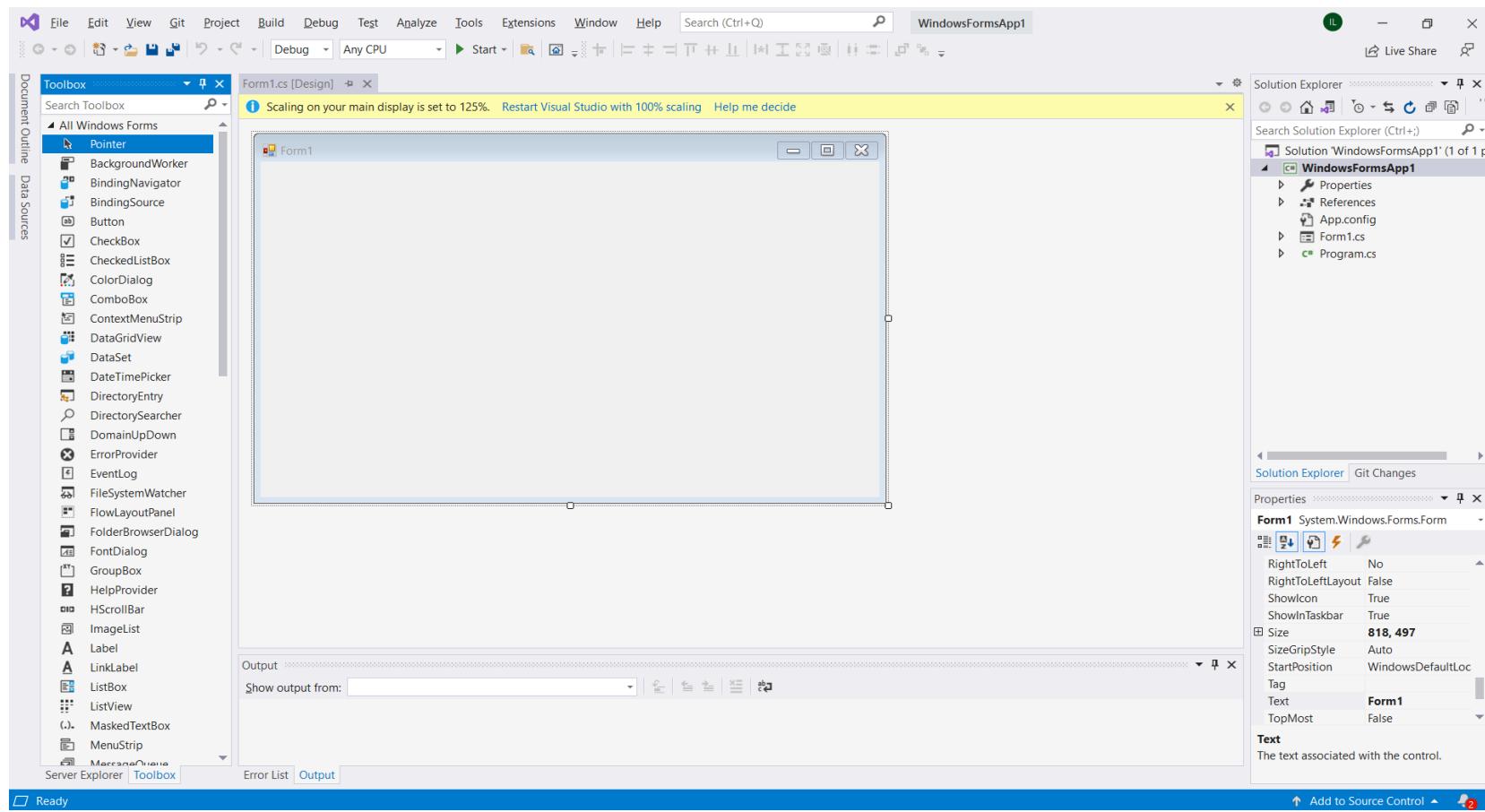
Membuat Project Baru (Cont.)

3. Pada kotak input **Project name**, ketikkan nama project. Masukkan tempat penyimpanan project pada **Location** dengan menekan tombol **Browse**. Klik Create.

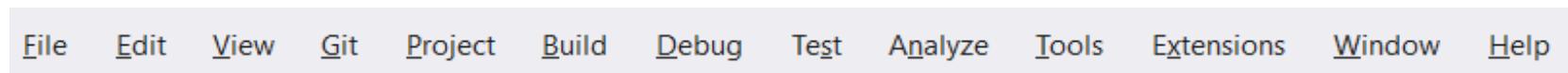


Membuat Project Baru (Cont.)

- IDE Microsoft Visual Studio akan menampilkan template aplikasi windows form.



Menu Bar



Menu	Fungsi
File	Digunakan untuk membuka / menutup project dan file, juga untuk keluar dari Visual Studio
Edit	Digunakan ketika menulis kode program untuk melakukan pengeditan seperti copy, cut, paste, dan menghapus teks
View	Menyediakan akses ke berbagai windows dalam IDE
Git	Digunakan untuk berinteraksi dengan folder dan file repositori
Project	Digunakan untuk menambah / menghapus komponen dari project
Build	Mengontrol proses kompilasi

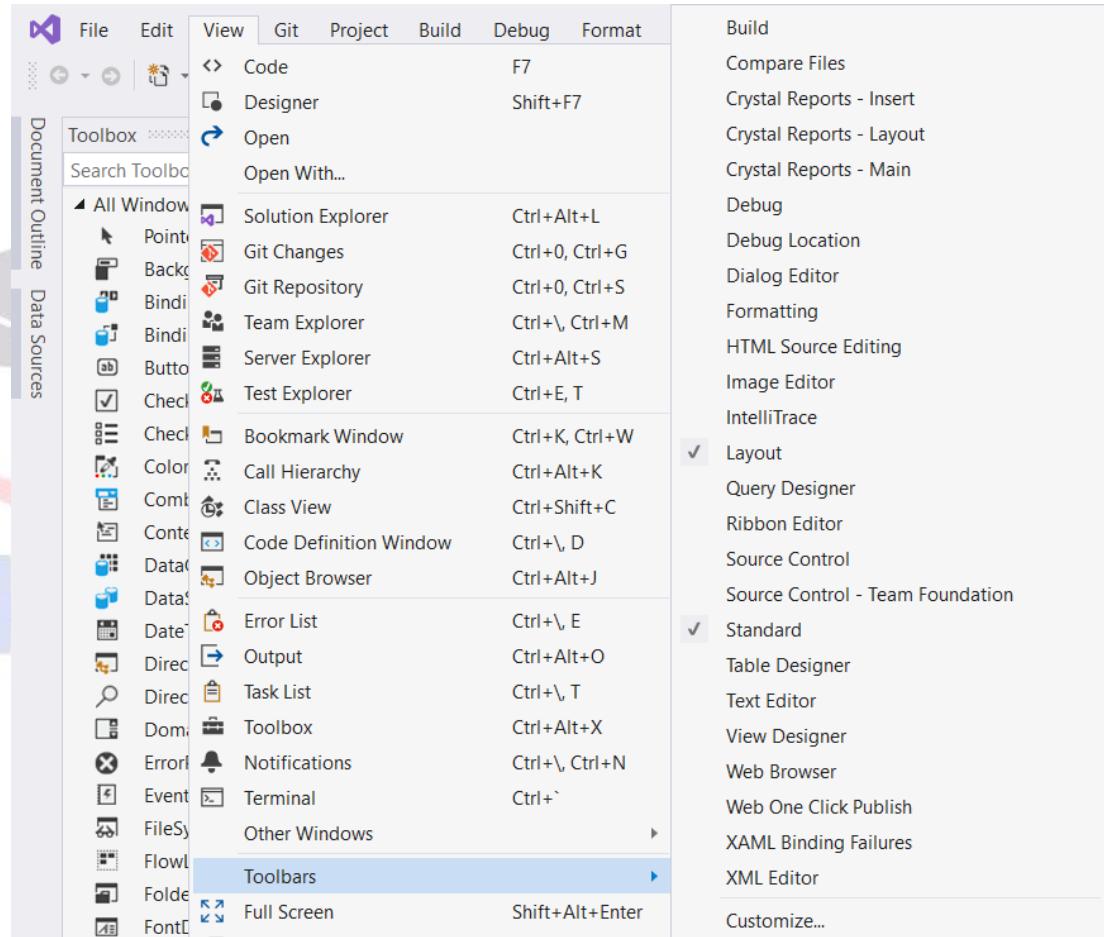
Menu Bar (Cont.)



Menu	Fungsi
Debug	Membantu melacak kesalahan / error pada kode program
Test	Digunakan untuk mengkompilasi dan menjalankan aplikasi yang lengkap
Analyze	Melakukan analisis terhadap kode program
Tools	Memungkinkan konfigurasi kustom terhadap IDE. Submenu Options paling banyak dipakai untuk memodifikasi IDE sesuai dengan kebutuhan personal masing-masing orang.
Window	Digunakan untuk mengubah layout windows dalam IDE
Extensions	Digunakan untuk menyesuaikan ekstensi
Help	Menyediakan akses ke dokumentasi online Visual C# melalui konten bantuan, indeksataupun pencarian.

Toolbar

- Menu View juga memungkinkan kita untuk memilih berbagai toolbar yang tersedia pada Visual C# IDE.



Toolbar (Cont.)

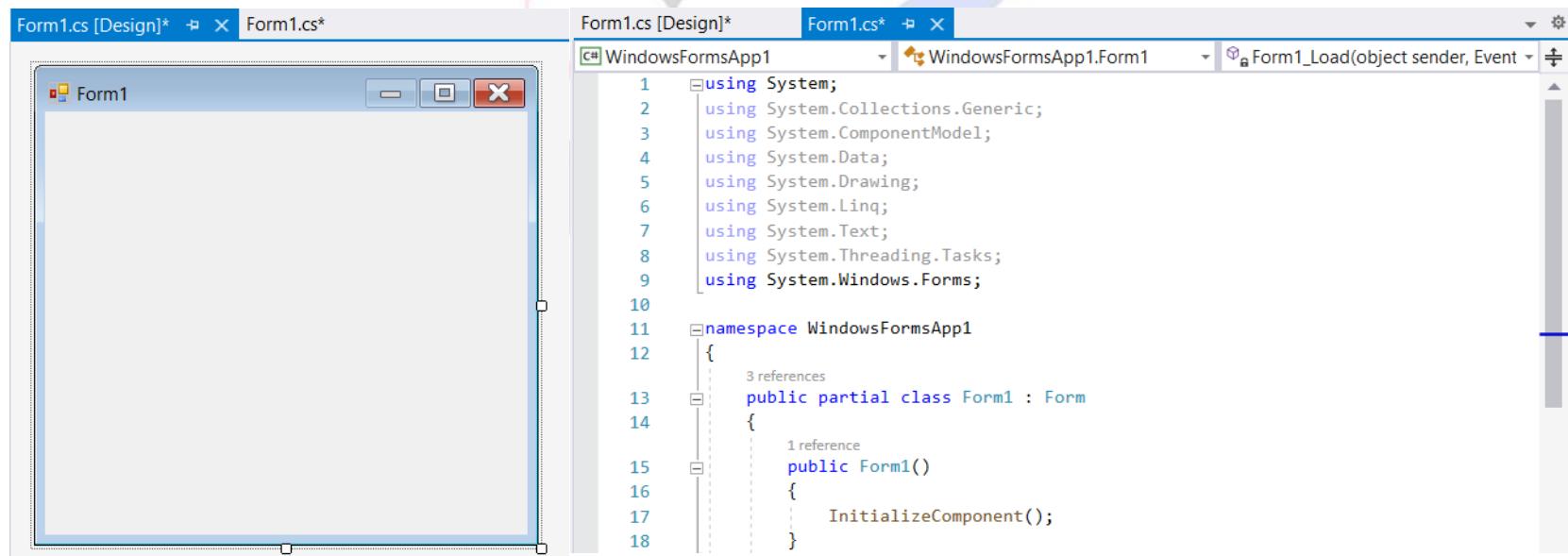
- Toolbar menyediakan akses yang cepat ke banyak fitur.
- Standard (Default) Toolbar dapat dilihat pada gambar dan letaknya berada di bawah menu bar.



- Apabila kita tidak mengetahui fungsi tombol toolbar, kita dapat mengarahkan mouse ke tombol tersebut dan tooltip yang berisi keterangan tombol akan muncul.

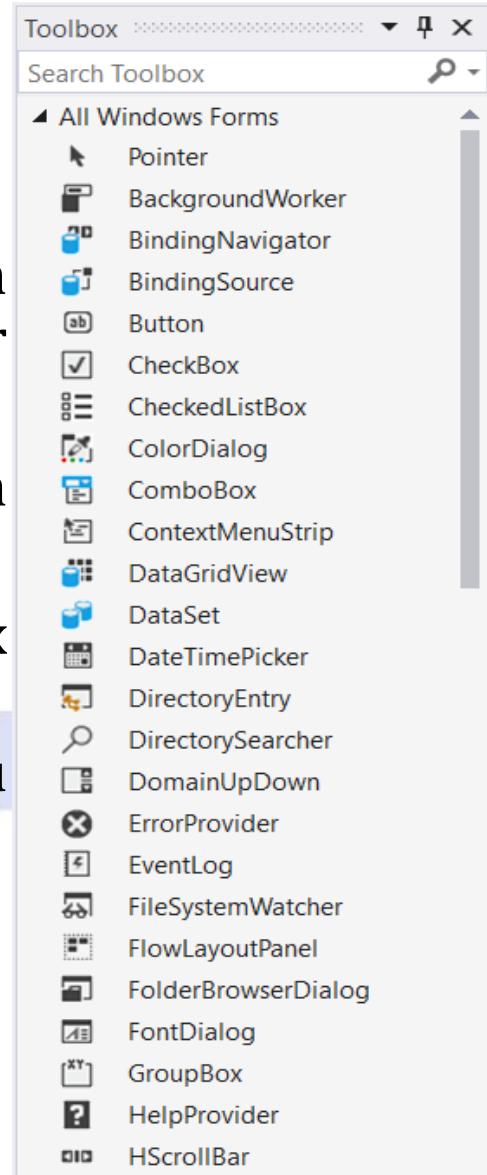
Form

- Merupakan jendela utama pengembangan Visual C# yang terdiri dari dua bagian yaitu:
 - Layar design (Form Designer)
 - Layar penulisan kode program (Code Editor)



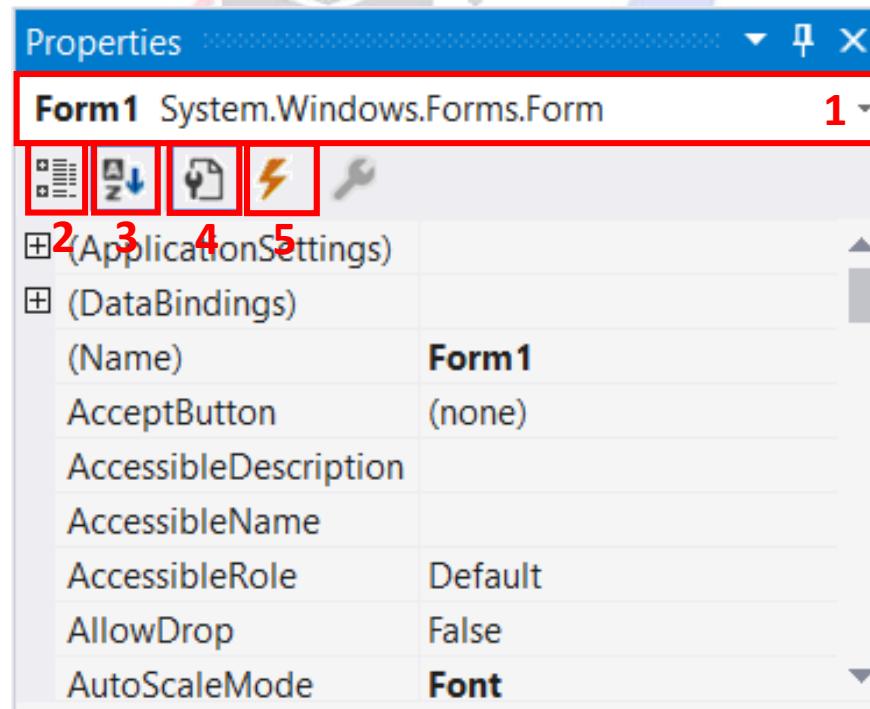
ToolBox

- Berisi kontrol-kontrol yang digunakan dalam aplikasi yang berbasis graphical user interface.
- Toolbox hanya aktif ketika form ditampilkan dalam layar design (Form Designer).
- Untuk menambahkan kontrol dari Toolbox ke form dapat digunakan dengan cara:
 - Klik pada kontrol yang diinginkan, lalu klik atau gambarkan pada form.
 - Drag kontrol dari toolbox.
 - Klik ganda pada kontrol yang diinginkan.



Properties Window

- Mengatur suatu nilai pada property objek dan event.
- Sangat membantu saat bekerja dengan Form Designer, karena kita dapat mengatur properti dengan mudah tanpa harus berurusan langsung dengan pemrograman.

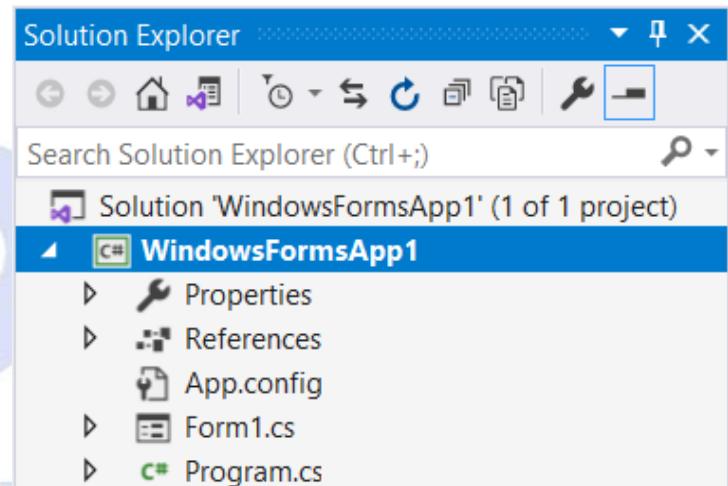


Properties Window (Cont.)

1. Drop-down pada bagian atas berisi daftar semua kontrol yang terdapat dalam form.
2. Property dapat disusun berdasarkan kategori (Categorized).
3. Property dapat disusun berdasarkan alphabet dari A sampai Z (Alphabetic).
4. Menampilkan semua karakteristik (properties) yang dimiliki oleh kontrol yang terpilih.
5. Menampilkan semua event yang dimiliki oleh kontrol yang terpilih.

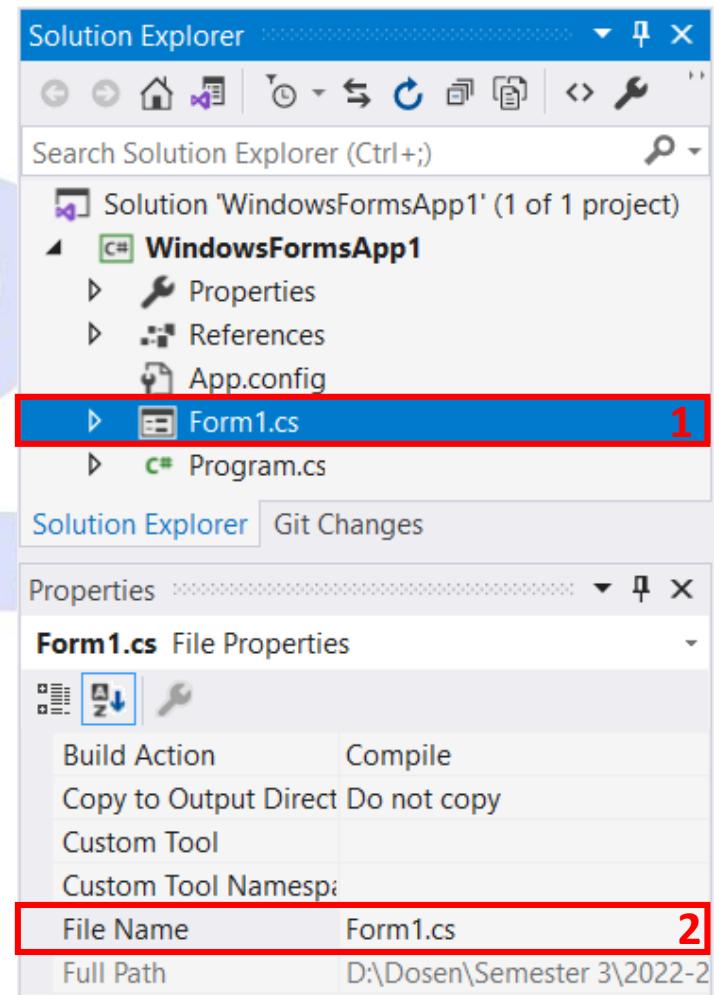
Solution Explorer

- Merupakan window yang developer dalam project maupun solution yang sedang aktif.
- Menampilkan semua form dan file yang terdapat dalam project.
- Untuk menampilkan form, double klik pada nama form di Solution Explorer.
- Kita dapat memilih tampilan design ataupun kode (View Code).



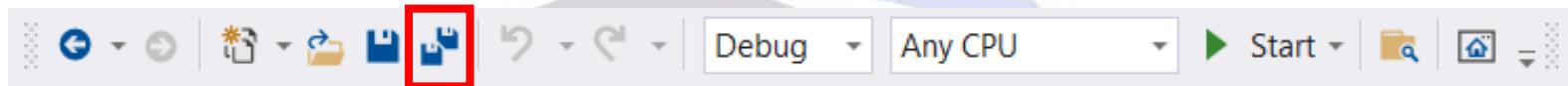
Mengganti Nama File

- Untuk mengganti nama file, klik Form1 di jendela **Solution Explorer**. Selanjutnya nama file akan ditampilkan di jendela **Properties**. Ganti properti **File Name** dengan nama file yang diinginkan.
- Alternatif: klik kanan pada Form1 di jendela **Solution Explorer**, pilih **Rename**, lalu ganti nama filenya.



Menyimpan Project

- Pilih menu **Save All** pada Toolbar atau dapat juga dengan memilih dari menu **File → Save All** .

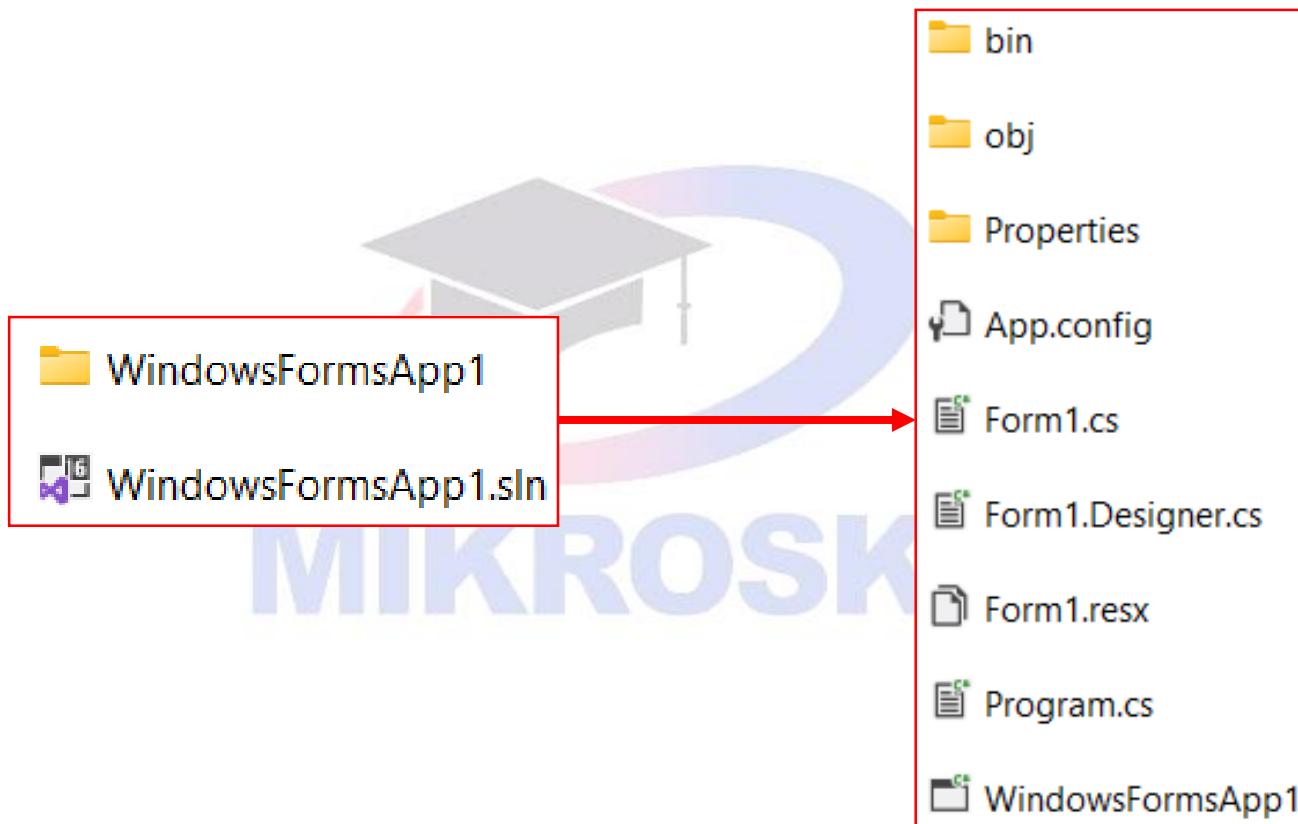


- Pastikan untuk menyimpan project sebelum menjalankan aplikasi atau keluar dari Visual Studio.

Struktur Project

- Visual C# menggunakan struktur folder yang sangat spesifik untuk menyimpan keseluruhan komponen pada suatu aplikasi.
- Ketika menyimpan suatu project baru (new project / solution), anda akan diminta untuk memasukkan **Name** dan **Location**. Folder dengan nama **Name** akan terbentuk pada lokasi **Location**. Folder tersebut akan berisi seluruh solution files, project files, dan file-file lain.
- Pada folder tersebut, terdapat 2 folder dengan nama: **Bin** dan **Obj**.

Struktur Project (Cont.)

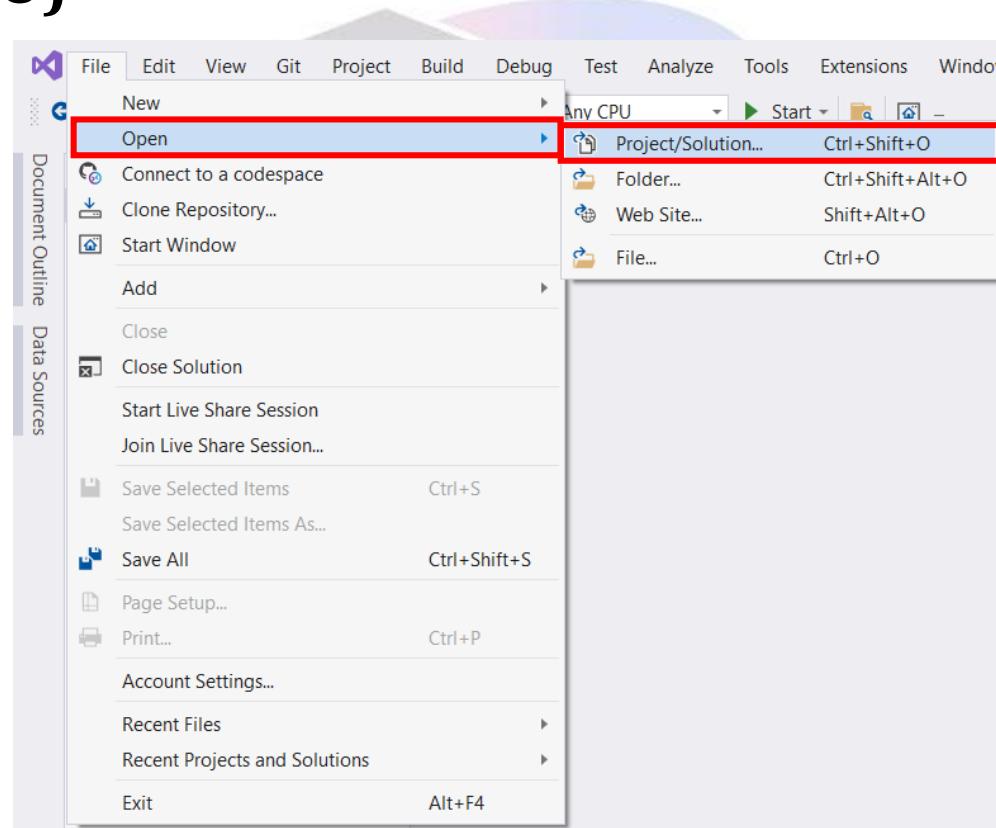


Struktur Project (Cont.)

File	Keterangan
Program.cs	Information on how things fit together
SolutionName.sln	Solution file for solution named SolutionName
SolutionName.suo	Solution options file
ProjectName.resx	Project file – one for each project in solution
FormName.resx	Form resources file – one for each form
FormName.cs	Form code file – one for each form
FormName.Designer.cs	File holding design information for form
App.ico	Icon used to represent the application

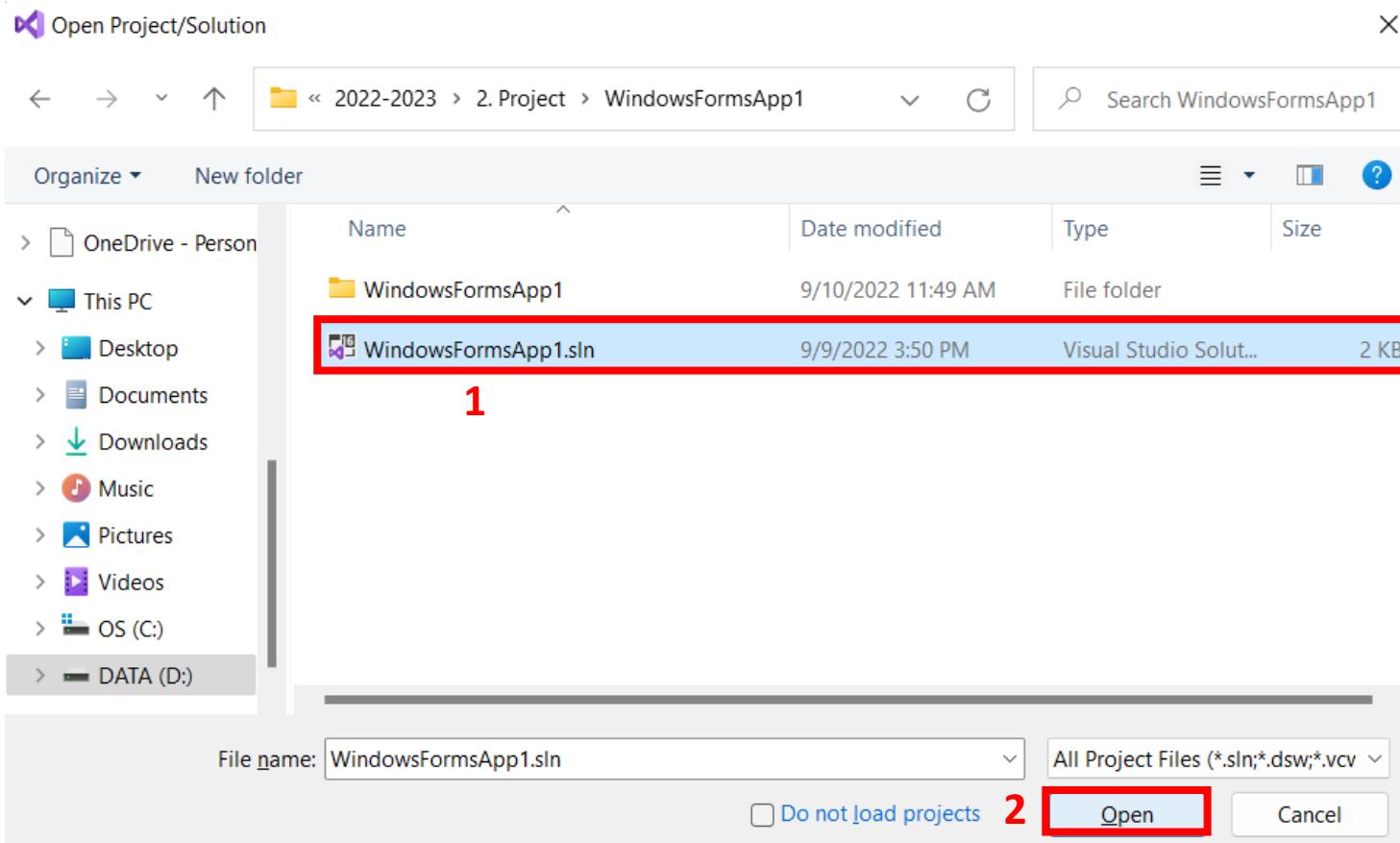
Membuka Saved Project

1. Pilih menu **File** → **Open** → **Project/Solution** (**Ctrl + Shift + O**)



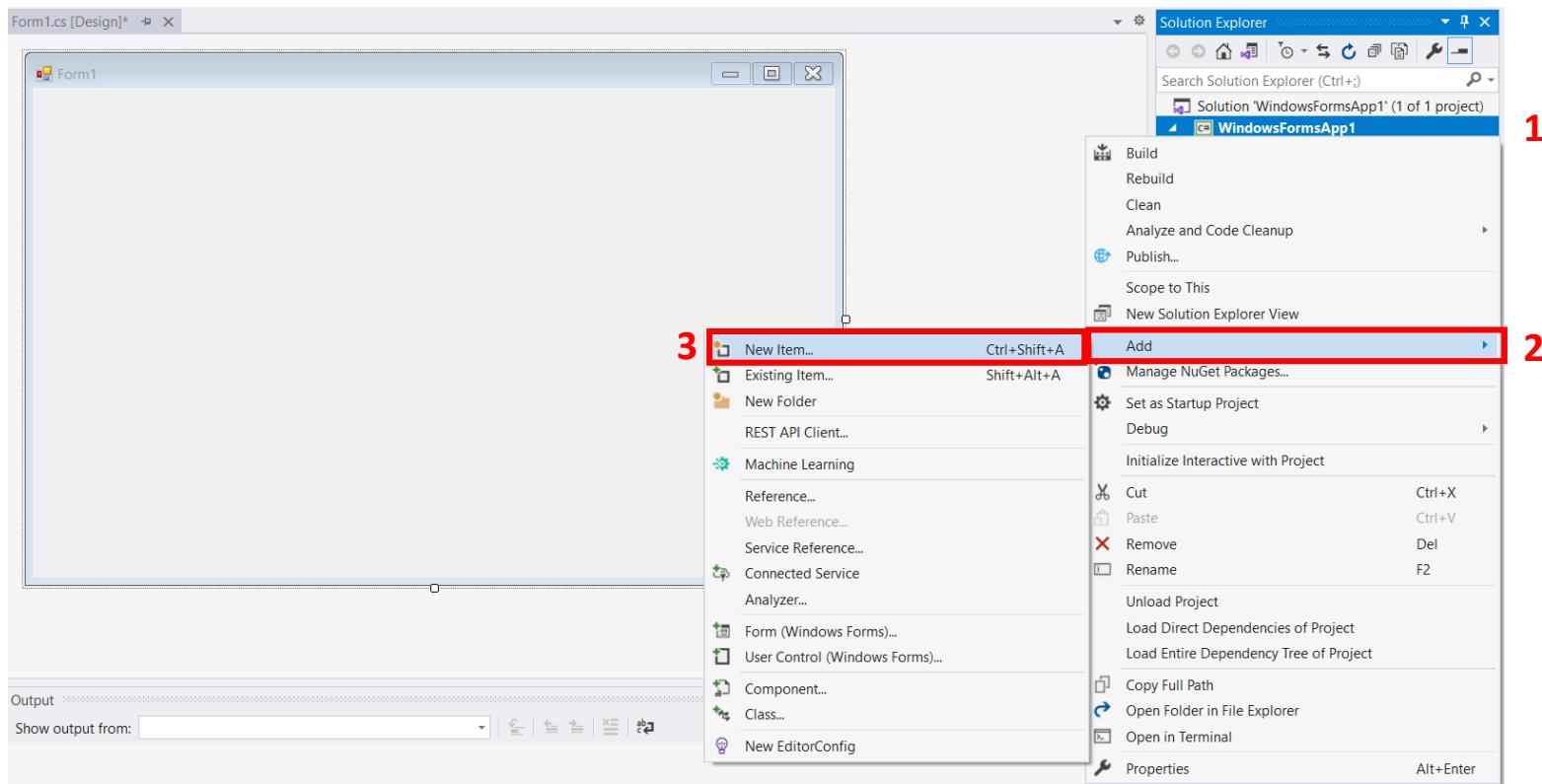
Membuka Saved Project (Cont.)

2. Kotak dialog Open Project akan terbuka. Cari lokasi penyimpanan project. Kemudian pilih file Solution (.sln), dan klik **Open**.



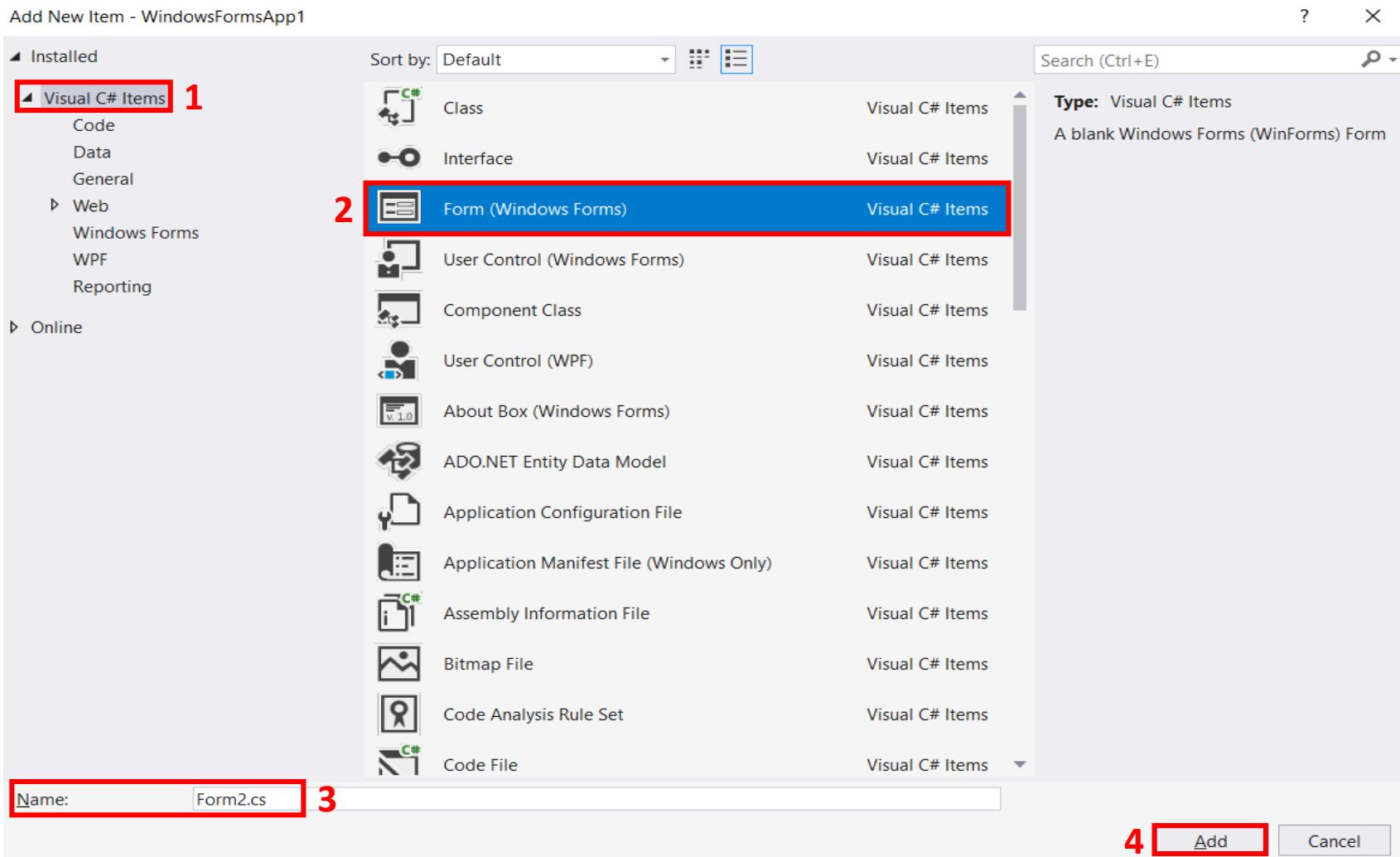
Menambahkan Item Baru

1. Klik kanan pada **nama project** pada **solution explorer** kemudian klik **add → new item** untuk menampilkan jendela **add new item**.



Menambahkan Item Baru (Cont.)

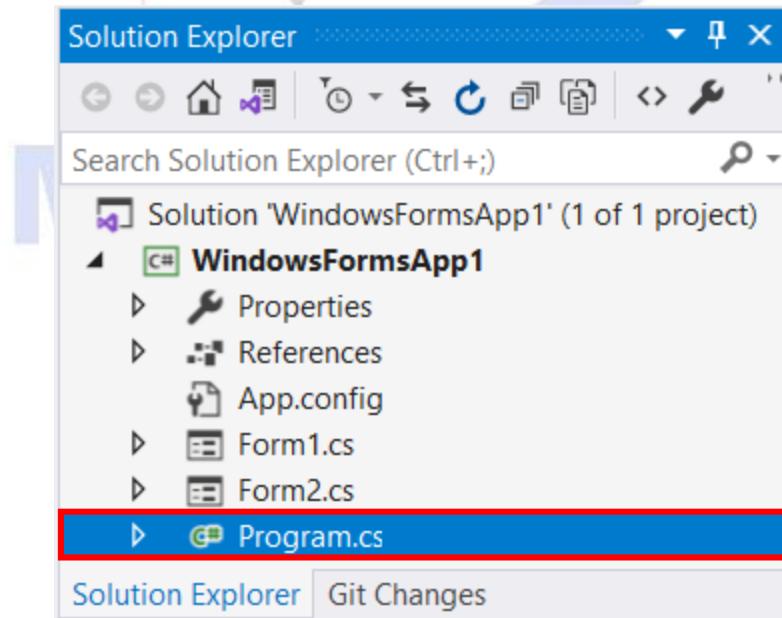
2. Pilih item yang akan ditambahkan, ubah nama item, kemudian klik button Add.



Memilih Form StartUp

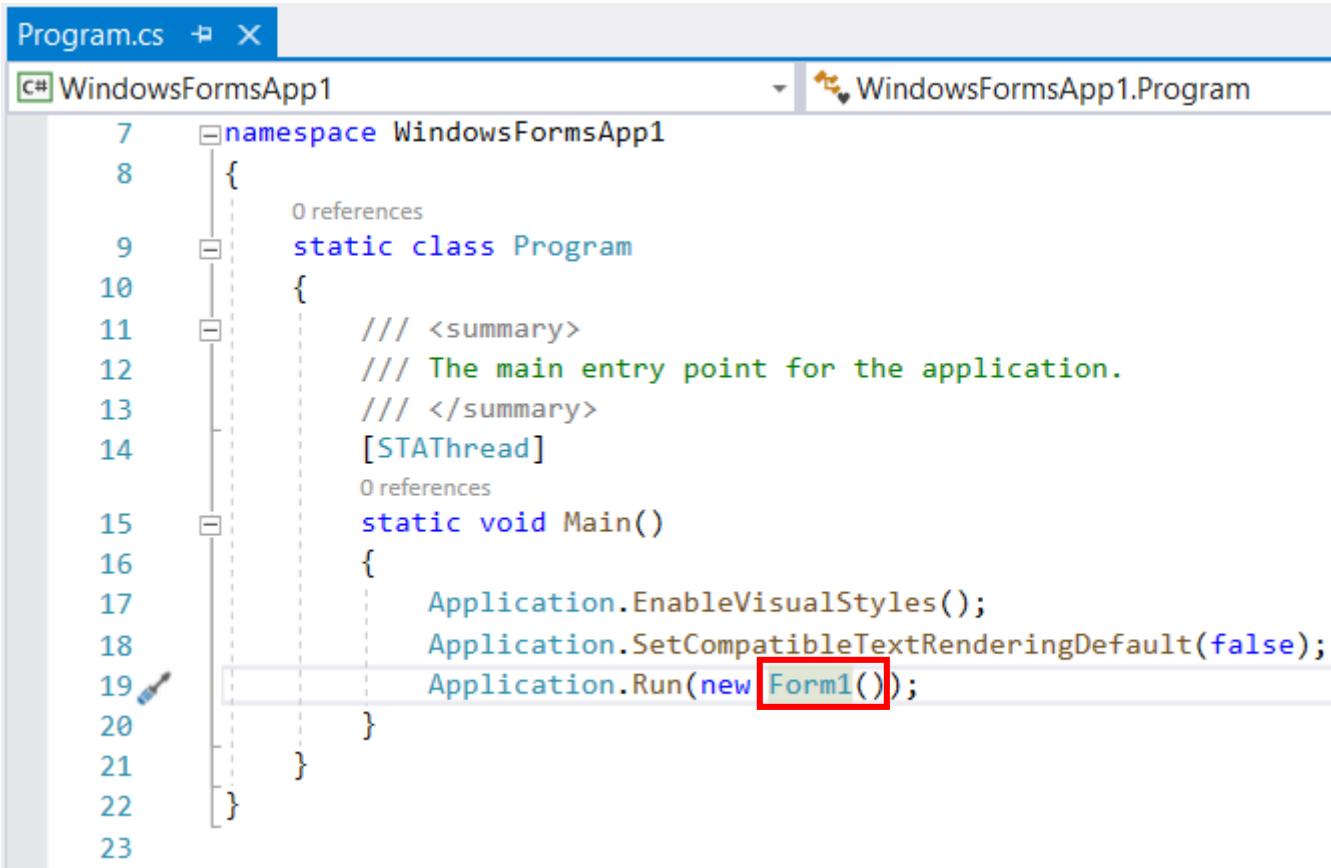
Apabila dalam suatu project terdapat beberapa form, kita dapat mengatur form mana yang muncul ketika aplikasi dijalankan.

1. Double klik **Program.cs** pada Solution Explorer



Memilih Form StartUp (Cont.)

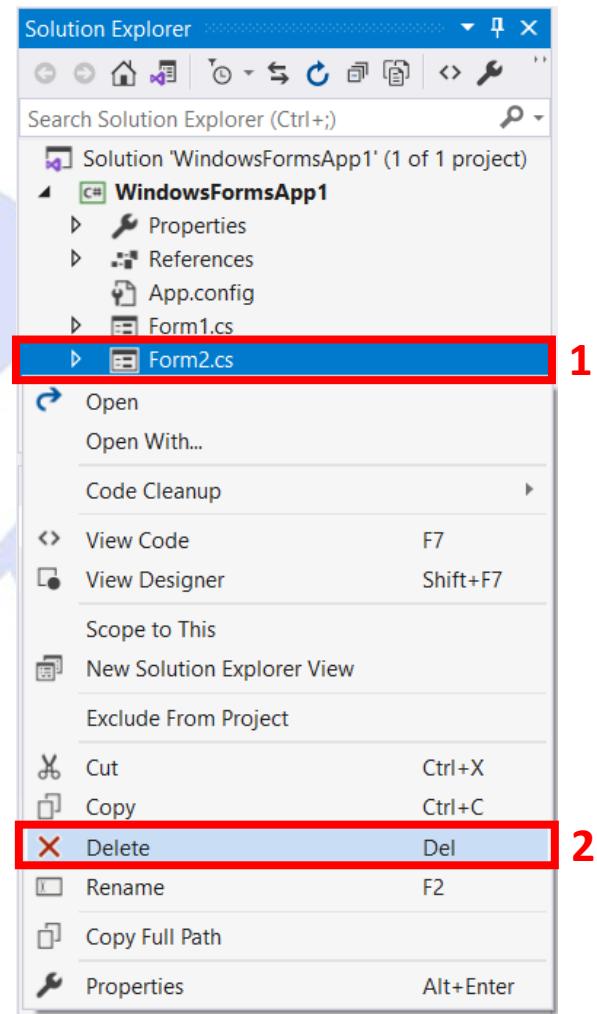
- Ubah nama form pada kotak berwarna merah menjadi nama form yang ingin dijalankan.



```
Program.cs  X
C# WindowsFormsApp1  WindowsFormsApp1.Program
7   namespace WindowsFormsApp1
8   {
9       static class Program
10      {
11          /// <summary>
12          /// The main entry point for the application.
13          /// </summary>
14          [STAThread]
15          static void Main()
16          {
17              Application.EnableVisualStyles();
18              Application.SetCompatibleTextRenderingDefault(false);
19              Application.Run(new Form1()); // Form1 is highlighted with a red box
20          }
21      }
22  }
23
```

Menghapus Item

- Klik kanan pada **nama item** pada solution explorer yang akan dihapus kemudian klik **delete**.



Lesson Summary

Sekarang mahasiswa mampu:

- Memahami dan mengerti cara penggunaan dan pengoperasian Microsoft Visual Studio 2019





Mengenal Kontrol

MIKROSKIL

Lesson Overview

- Kontrol adalah objek yang terkait dengan user interface yang dapat ditambahkan pada form, baik untuk menampilkan informasi, menerima input user, atau keduanya. Kontrol disediakan oleh .NET Framework Class Library. Visual C# dapat menggunakan class library yang disediakan oleh .NET Framework untuk Windows Forms maupun kontrol. Kontrol terkait erat dengan class System.Windows.Forms.Control, karena semua class untuk kontrol diturunkan dari class tersebut. Kontrol dapat ditambahkan dengan mudah pada Form apabila menggunakan Form Designer. Namun, adakalanya harus berurusan dengan kode pemrograman secara langsung dan tidak menggunakan Form Designer maupun Properties Window. Selain kontrol untuk user interface, terdapat juga kontrol nonuser interface dan kontrol ActiveX.

Objectives

Setelah menyelesaikan Lesson ini, mahasiswa diharapkan mampu:

- Memahami cara penamaan sebuah kontrol dan penggunaannya dalam baris kode



Penamaan Control

- Property yang paling penting dari setiap kontrol adalah property **Name**. Pada saat menulis kode program, property Name digunakan untuk merujuk pada kontrol yang diinginkan.
- Terdapat aturan dalam pemberian nama kontrol. Aturan ini diterapkan dengan menambah awalan sebanyak 3 huruf kecil (lowercase) yang menunjukkan jenis kontrol, kemudian diikuti dengan nama kontrol yang ingin diberikan.

Penamaan Control (Cont.)

Kontrol	Prefix	Contoh
Form	frm	frmMahasiswa
Button	btn	btnHitung
Label	lbl	lblNama
TextBox	txt	txtAlamat
Menu	mnu	mnuSimpan
CheckBox	chk	chkMerah

- Disarankan untuk menggabungkan huruf besar dan huruf kecil supaya lebih mudah dibaca, tetapi ingat bahwa C# bersifat case-sensitive, yang berarti lblNama berbeda dengan LBLNAMA

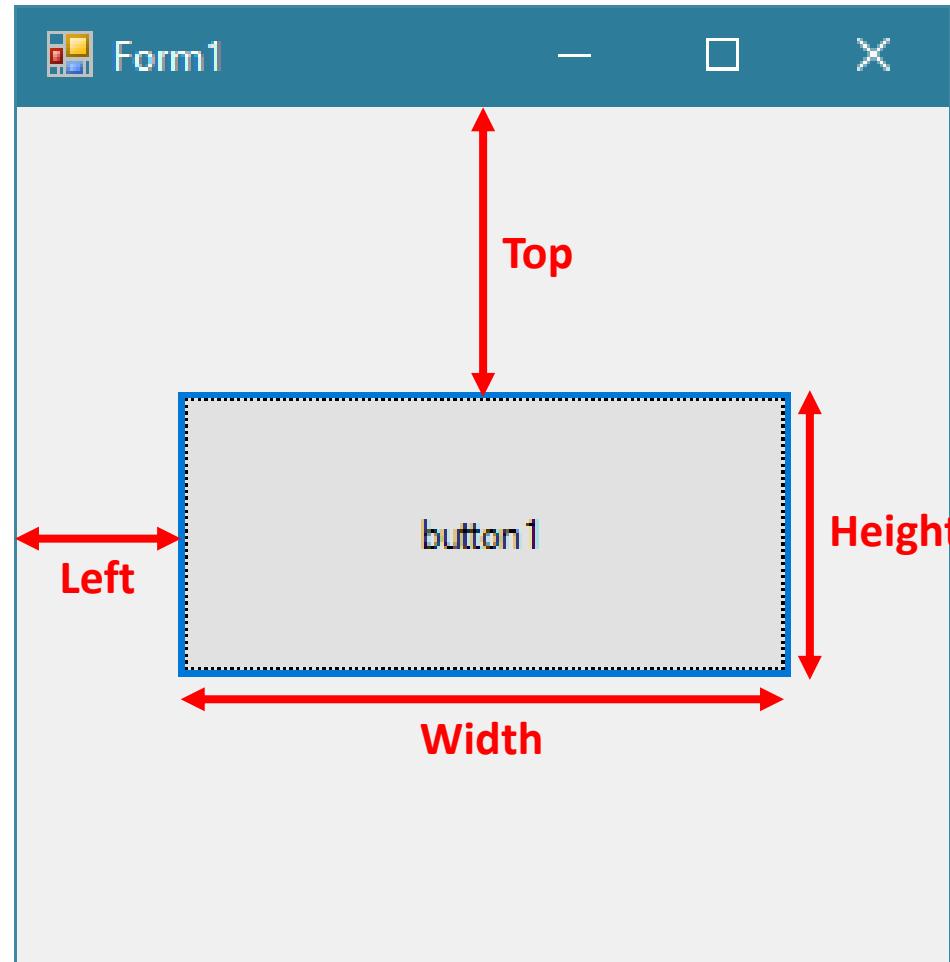
Penamaan Control (Cont.)

- Property Name pada kontrol dapat menampung 40 karakter dan harus dimulai dengan huruf dan hanya boleh terdiri dari huruf, angka, dan karakter underscore “_”
 -
- Berikan property name yang bermakna yang dapat membantu kita ataupun orang lain untuk memahami jenis kontrol dan fungsi dari kontrol bersangkutan.

Property Umum Class Control

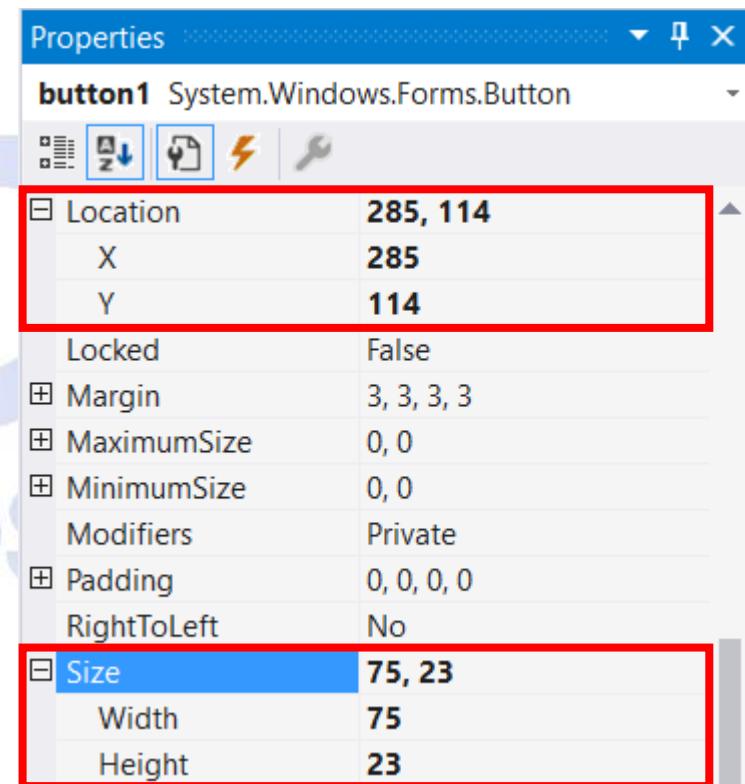
- Property yang umumnya terdapat pada kontrol adalah property yang berhubungan dengan posisi (**position**) dan ukuran (**size**).
- Terdapat 4 property (dalam satuan pixel), yaitu:
 - **Left** : jarak dari sisi kiri form ke sisi kiri kontrol
 - **Top** : jarak dari judul form ke sisi atas kontrol
 - **Width** : lebar kontrol
 - **Height** : tinggi kontrol

Property Umum Class Control (Cont.)



Property Umum Class Control (Cont.)

- Untuk melihat property Left dan Top pada Properties Window, tekan tanda + di samping property **Location**. Nilai X adalah property Left, sedangkan nilai Y adalah property Top.
- Untuk melihat property Width dan Height pada Properties Window, tekan tanda + di samping property **Size**. Nilai Width dan Height akan ditampilkan, dan nilai ini dapat diubah.



Mengatur Nilai Properties

- Mengatur nilai properties dapat dilakukan pada saat design melalui Properties Window ataupun ketika aplikasi dijalankan melalui kode program.
- Format penulisan kode program:
`objectName.PropertyName = NewValue;`
- Contoh:
`btnStart.BackColor = Color.Blue;`



Property Name Pada Event

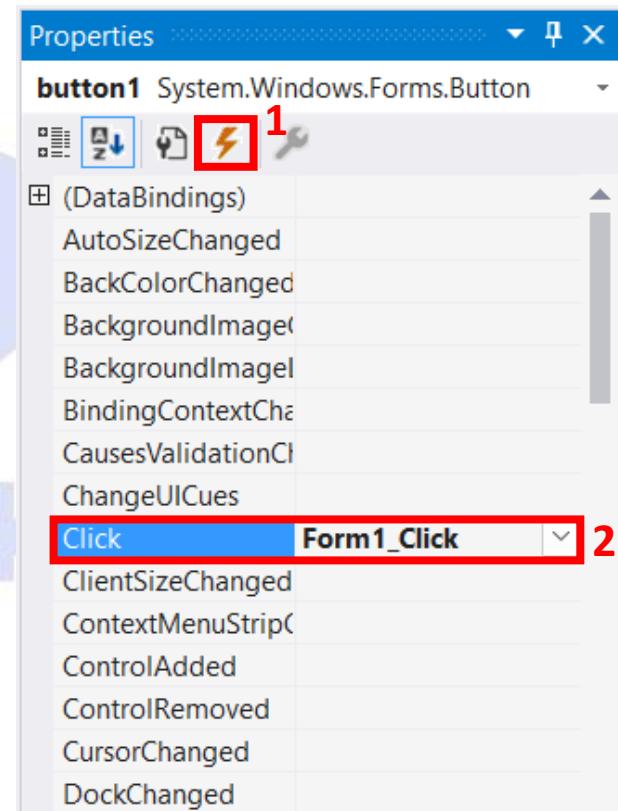
- Untuk menambahkan event baru pada kontrol, dapat melalui 2 cara.
- Cara pertama yaitu dengan double klik pada kontrol untuk memunculkan event default dari kontrol tersebut.
- Kode yang muncul ketika kita melakukan double klik pada kontrol akan mengikuti format sebagai berikut:

```
private void ControlName_Event(Arguments)  
{  
}  
}
```

- Keseluruhan baris koding yang dijalankan ketika event terjadi, diletakkan di antara kedua kurung kurawal.

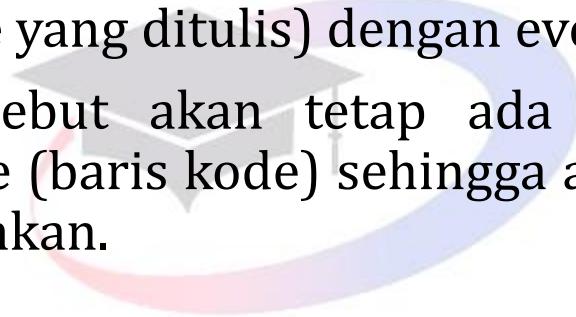
Property Name Pada Event (Cont.)

- Cara kedua adalah melalui **Properties Window**. Cara ini terutama digunakan apabila kita ingin menggunakan event yang lain dari kontrol (bukan event default).
- Pilih tombol **event** pada Properties Window dan double klik pada event yang diinginkan. Kode program akan langsung muncul pada Code Window.



Property Name Pada Event (Cont.)

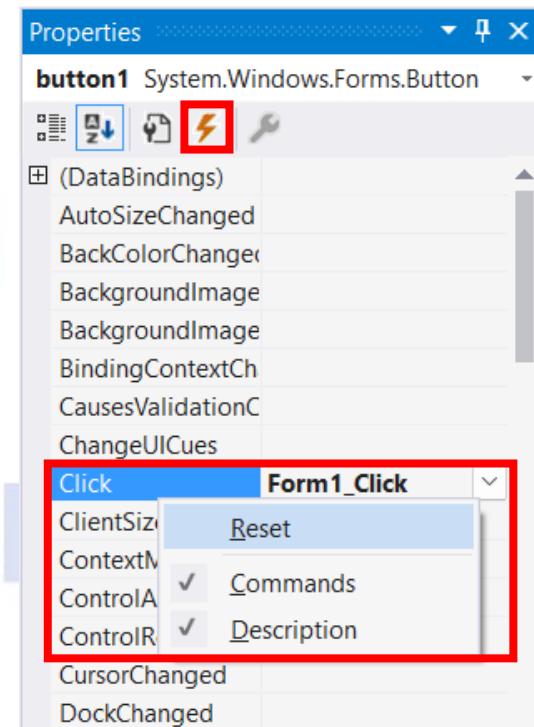
- Setelah kita menambahkan sebuah event pada aplikasi, Visual C# secara otomatis menuliskan baris koding untuk menghubungkan metode (baris kode yang ditulis) dengan event yang dipakai.
- Baris koding tersebut akan tetap ada walaupun kita telah menghapus metode (baris kode) sehingga akan terjadi error pada saat aplikasi dijalankan.



MIKROSKIL

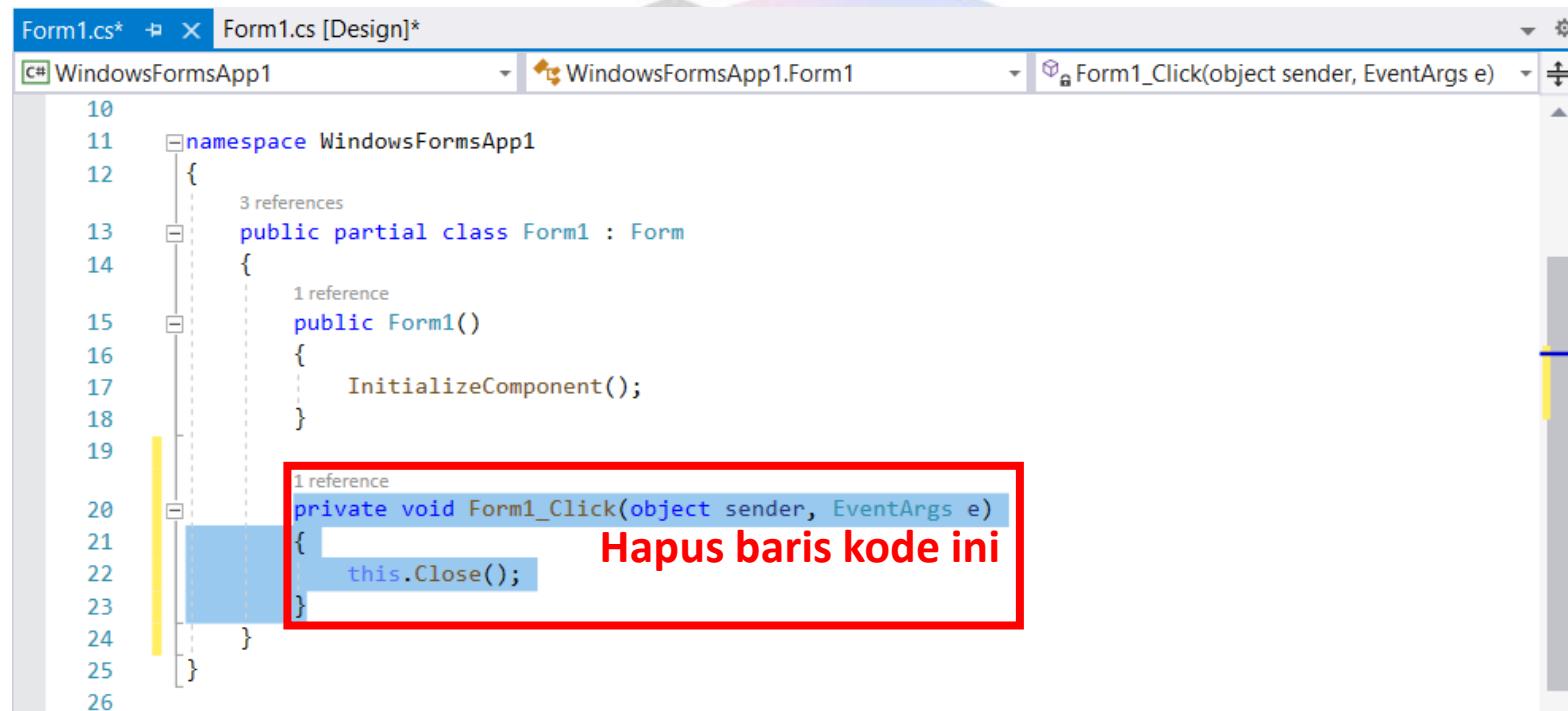
Property Name Pada Event (Cont.)

- Untuk mengatasinya, pertama-tama tekan tombol Event pada Properties Window, kemudian klik kanan pada event yang ingin dihapus, lalu pilih **Reset**.
- Proses ini akan menghapus baris koding yang menghubungkan metode dengan event.



Property Name Pada Event (Cont.)

- Setelah itu, baru hapus metode (baris kode) yang berhubungan dengan event tersebut pada Code Window.



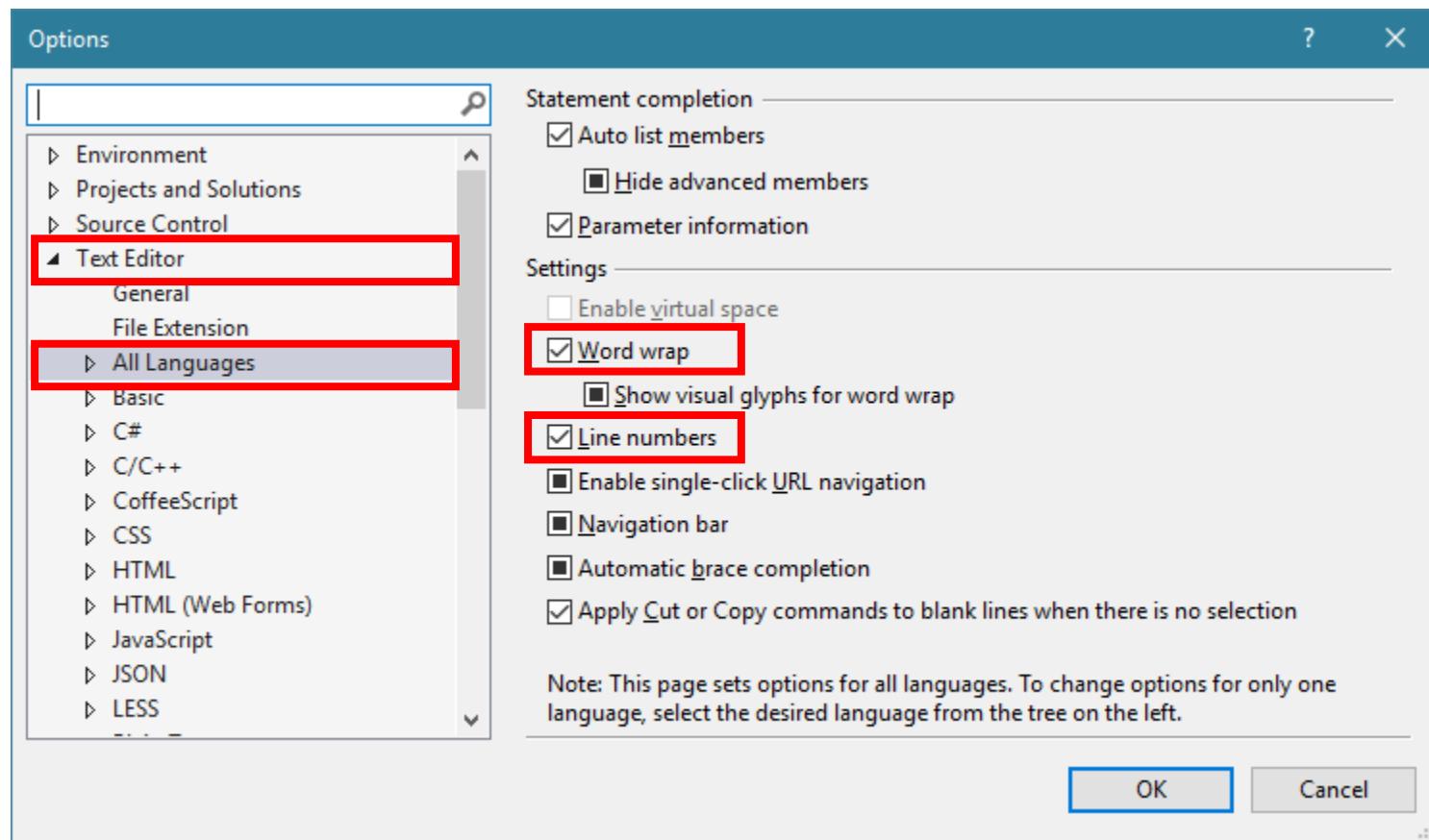
```
Form1.cs*  Form1.cs [Design]*  
C# WindowsFormsApp1  WindowsFormsApp1.Form1  Form1_Click(object sender, EventArgs e)  
10  
11  namespace WindowsFormsApp1  
12  {  
13      public partial class Form1 : Form  
14      {  
15          public Form1()  
16          {  
17              InitializeComponent();  
18          }  
19  
20          private void Form1_Click(object sender, EventArgs e)  
21          {  
22              this.Close();  
23          }  
24      }  
25  }  
26
```

Hapus baris kode ini

Penulisan Kode Program

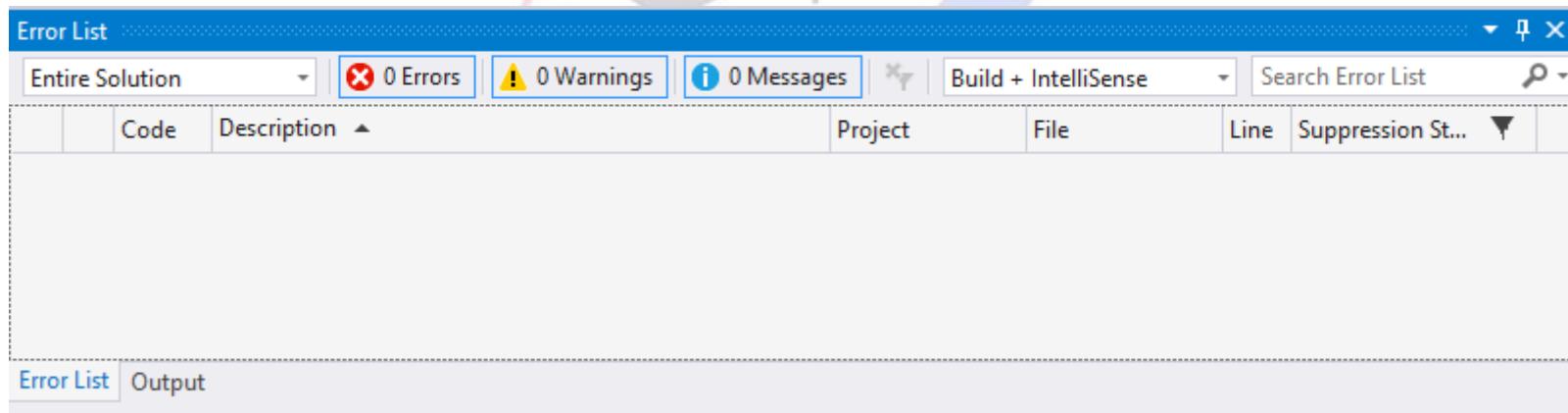
- Langkah terakhir dalam membangun aplikasi berbasis Visual C# adalah menuliskan kode program dengan menggunakan Bahasa C#.
- Fitur yang berguna dalam penulisan kode program adalah dengan menampilkan nomor baris (line number) di samping kode program yang ditulis.
- Cara menampilkannya adalah dengan memilih menu **Tools → Options**. Pada bagian kiri windows, pilih **Text Editor**, kemudian **All Languages**, lalu centang **Line numbers** pada sisi sebelah kanan.
- Kita juga dapat mencentang pilihan **Word wrap** agar penulisan kode program bisa secara otomatis enter ke bawah apabila telah mencapai ujung kanan layar.

Penulisan Kode Program (Cont.)



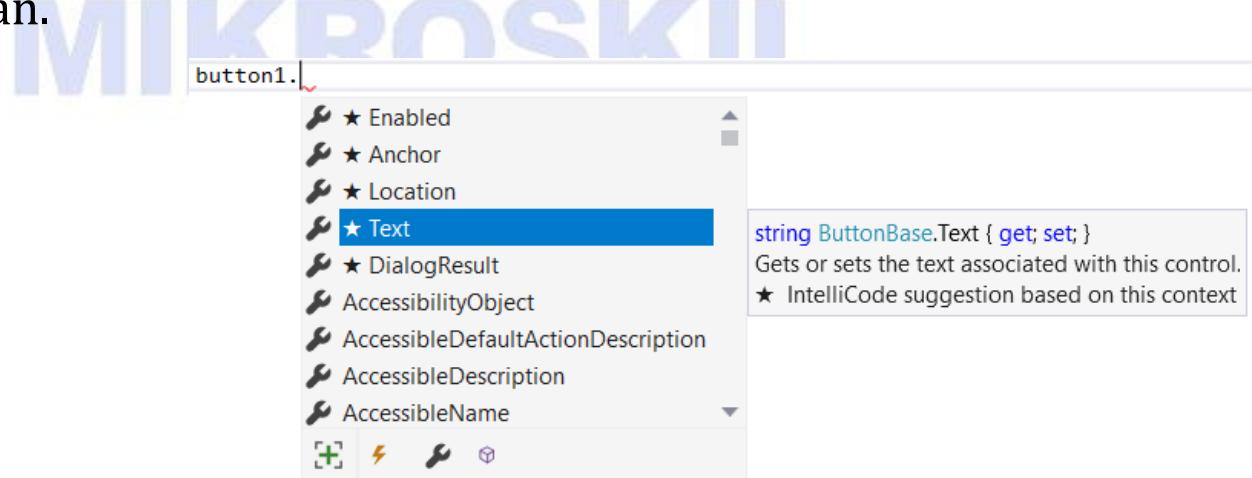
Penulisan Kode Program (Cont.)

- Line numbers tersebut akan berguna untuk mengidentifikasi di baris nomor berapa suatu error terjadi pada kode program.
- Informasi tersebut dapat dilihat pada **Error List Window**. Untuk mengaktifkannya, pilih menu **View → Error List**.



Fitur Intellisense

- Merupakan salah satu fasilitas yang terdapat pada layar kode.
- Berguna untuk menyelesaikan penulisan baris kode.
- Contoh, ketika kita mengetikkan nama kontrol diikuti dengan tanda titik “.”, maka Code Editor akan menampilkan informasi properties dan methods yang berhubungan dengan kontrol tersebut secara otomatis. Apabila kita menggunakan methods, informasi mengenai arguments (parameter) akan dimunculkan. Kesalahan penulisan syntax akan diidentifikasi dan error yang mungkin muncul akan diinformasikan.



Lesson Summary

Sekarang mahasiswa mampu:

- Memahami cara penamaan sebuah kontrol dan penggunaannya dalam baris kode



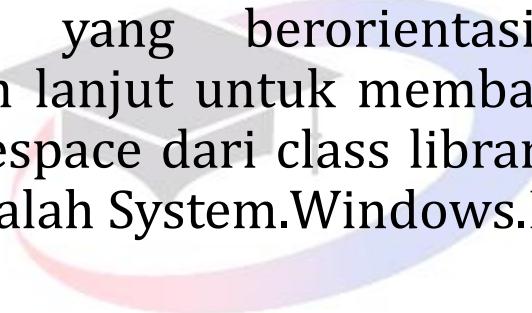


Mengenal Form

MIKROSKIL

Lesson Overview

- Windows Form adalah platform baru untuk pembuatan aplikasi Windows berbasis .NET Framework. Framework ini menyediakan sekelompok class yang berorientasi objek dan dapat dikembangkan lebih lanjut untuk membangun aplikasi windows yang lengkap. Namespace dari class library yang disediakan oleh .NET Framework adalah System.Windows.Forms.



MIKROSKIL

Objectives

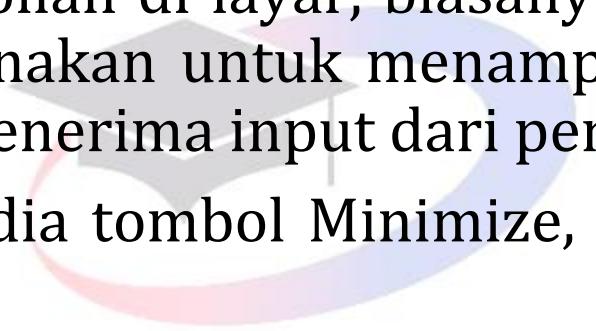
Setelah menyelesaikan Lesson ini, mahasiswa diharapkan mampu:

- Memahami karakteristik dan penggunaan dari form



Form

- Form merupakan tempat mendesain user interface. Bagian dari tampilan di layar, biasanya berbentuk kotak, yang dapat digunakan untuk menampilkan informasi ke pengguna dan menerima input dari pengguna.
- Pada form tersedia tombol Minimize, Maximize/Restore, dan Close.



MIKROSKIL

Form (Cont.)

- Property umum form:

Property	Keterangan
Name	Mengatur nama form (Form diawali dengan frm)
AcceptButton	Mengatur button mana dalam form yang ditekan apabila user menekan tombol Enter
BackColor	Mengatur warna latar form
CancelButton	Mengatur button mana dalam form yang ditekan apabila user menekan tombol Esc
ControlBox	Mengatur apakah control box (Minimize, Maximize/Restore, Close) ditampilkan pada form atau tidak
Enabled	Jika diatur menjadi False, maka semua kontrol dalam form menjadi tidak aktif
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan form

Form (Cont.)

Property	Keterangan
FormBorderStyle	Mengatur bingkai form apakah ukurannya bisa diubah atau tidak (fixed atau sizeable)
Height	Mengatur tinggi form dalam satuan pixel
Icon	Mengatur icon form
Left	Mengatur jarak dari sisi kiri layar ke sisi kiri form dalam satuan pixel
MaximizeBox	Mengatur apakah tombol Maximize/Restore ditampilkan pada form atau tidak
MinimizeBox	Mengatur apakah tombol Minimize ditampilkan pada form atau tidak
StartPosition	Mengatur posisi form pada saat aplikasi dijalankan
Text	Mengatur judul form
Top	Mengatur jarak dari sisi atas layar ke sisi atas form dalam satuan pixel
Width	Mengatur lebar form dalam satuan pixel

Form (Cont.)

- Metode umum form:

Method	Keterangan
Close	Menutup form
Focus	Mengatur fokus pada form
Hide	Menyembunyikan form
Show	Menampilkan form

- Sintaks penggunaan metode khusus form:

this.method();

Form (Cont.)

- Event umum form:

Event	Keterangan
Activated	Terjadi ketika form diaktifkan melalui koding atau oleh user
Click	Terjadi ketika form di-klik oleh user
FormClosing	Terjadi ketika form sedang ditutup
DoubleClick	Terjadi ketika form di-doubleklik oleh user
Load (default)	Terjadi ketika form dijalankan pertama kali

MIKROSKIL

Form (Cont.)

- Form dikenal sebagai objek **container**, karena berfungsi menampung kontrol-kontrol yang lain.
- Dampaknya adalah kontrol-kontrol yang diletakkan dalam form akan mewarisi property **BackColor**, **ForeColor**, dan **Font** dari form.
- Selain itu, apabila container diatur property **Enabled** menjadi False, maka semua kontrol yang ada di dalam container akan menjadi Disabled. Demikian pula dengan property **Visible**.

Lesson Summary

Sekarang mahasiswa mampu:

- Memahami karakteristik dan penggunaan dari form



Button, Label & Texthbox

MIKROSKIL

Lesson Overview

- Kontrol yang paling umum untuk digunakan dalam sebuah program adalah Button, Label dan Textbox



Objectives

Setelah menyelesaikan Lesson ini, mahasiswa diharapkan mampu:

- Memahami karakteristik dan penggunaan dari kontrol Button, Label dan TextBox



Button

- Biasanya digunakan untuk mengeksekusi suatu aksi dengan cara mengklik.
- Property umum pada button:

Property	Keterangan
Name	Mengatur nama button (Button diawali dengan btn)
BackColor	Mengatur warna latar button
Enabled	Jika diatur menjadi False, button kelihatan tetapi tidak dapat menerima klik
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan
Image	Mengatur gambar yang akan ditampilkan pada button
Text	Mengatur tulisan yang tampil pada button
TextAlign	Mengatur perataan teks pada button

Button (Cont.)

- Metode umum pada button:

Method	Keterangan
Focus	Mengatur fokus ke button
PerformClick	Menjalankan event click pada button

- Event umum pada button:

Event	Keterangan
Click (default)	Terjadi ketika button ditekan

Label

- Kontrol Label digunakan untuk menampilkan teks atau gambar yang tidak dapat diedit oleh user.
- Property umum pada label:

Property	Keterangan
Name	Mengatur nama label (Label diawali dengan lbl)
AutoSize	Mengatur apakah ukuran label berubah secara otomatis sesuai dengan isi label atau tidak. Secara default, kontrol label akan berubah secara otomatis mengikuti isi label sehingga untuk mengubah ukuran label, property AutoSize harus bernilai False
BackColor	Mengatur warna latar label
BorderStyle	Mengatur tipe bingkai pada label

Label (Cont.)

Property	Keterangan
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan
Text	Mengatur tulisan yang akan ditampilkan pada label
TextAlign	Mengatur perataan teks pada label
• Event umum pada label:	
Event	Keterangan
Click (default)	Terjadi ketika user menekan label
DoubleClick	Terjadi ketika user menekan label 2x secara cepat (double klik)

TextBox

- Digunakan untuk mendapatkan input dari user atau menampilkan teks.
- Teks di dalam TextBox dapat di-edit dan dapat juga dibuat menjadi read-only.
- TextBox dapat menampilkan satu baris atau banyak baris.
- Property umum pada textbox:

Property	Keterangan
Name	Mengatur nama TextBox (TextBox dimulai dengan txt)
AutoSize	Mengatur apakah ukuran tinggi dari textbox akan secara otomatis mengikuti tinggi tulisan
BackColor	Mengatur warna latar textbox
BorderStyle	Mengatur tipe bingkai textbox

TextBox (Cont.)

Property	Keterangan
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan
HideSelection	Mengatur apakah tulisan yang diblok pada textbox akan tetap dalam kondisi terblok setelah textbox lost focus
MaxLength	Mengatur jumlah karakter maksimum yang dapat ditampung textbox
MultiLine	Mengatur apakah textbox menampilkan 1 baris atau dapat lebih dari 1 baris
PasswordChar	Mengatur karakter yang akan dimunculkan dalam textbox
ReadOnly	Mengatur apakah tulisan dalam textbox dapat diedit atau tidak
ScrollBars	Mengatur apakah scrollbar akan muncul pada textbox multiline

TextBox (Cont.)

Property	Keterangan
Text	Mengatur tulisan yang tampil pada textbox
TextAlign	Mengatur perataan tulisan pada textbox
TextLength	Mengatur panjang tulisan pada textbox
• Metode umum pada textbox:	
Method	Keterangan
AppendText	Menambahkan teks baru pada teks yang sudah ada di textbox
Clear	Membersihkan textbox
Copy	Meng-copy teks yang terpilih ke clipboard
Cut	Memindahkan teks yang terpilih ke clipboard
Focus	Meletakkan kursor pada textbox yang bersangkutan
Paste	Menggantikan teks yang terpilih di textbox dengan konten yang ada di clipboard

TextBox (Cont.)

Method	Keterangan
SelectAll	Memilih semua teks yang ada di textbox
Undo	Membatalkan operasi edit terakhir
• Event umum pada textbox:	
Event	Keterangan
Click	Terjadi ketika user menekan textbox
Enter	Terjadi ketika textbox menerima focus
KeyDown	Terjadi ketika tombol keyboard ditekan pada saat textbox menerima focus
KeyPress	Terjadi ketika tombol keyboard yang memiliki kode ASCII ditekan pada saat textbox menerima focus
Leave	Terjadi ketika textbox kehilangan focus. Ini merupakan saat yang tepat untuk memeriksa isi textbox.
TextChanged (default)	Terjadi ketika nilai property Text berubah

TextBox (Cont.)

- Penggunaan kontrol textbox harus diminimalisir jika memungkinkan karena apabila kita mengizinkan user untuk mengetikkan sesuatu, pekerjaan kita sebagai programmer akan lebih sulit. Kita harus memvalidasi semua inputan user untuk memastikan informasi tersebut dapat bekerja pada koding kita.
- Disarankan untuk menggunakan kontrol lain yang bersifat ‘point and click’ untuk menggantikan textbox.

MIKROSKIL

Lesson Summary

Sekarang mahasiswa mampu:

- Memahami karakteristik dan penggunaan dari kontrol Button, Label dan TextBox





Latihan

MIKROSKIL

Latihan 1

Validasi Password

Masukkan password anda:

Validasi Masuk Keluar

Validasi Password

Masukkan password anda:

Validasi Masuk Keluar

Password salah! Input ulang!

Validasi Password

Masukkan password anda:

Validasi Masuk Keluar

Password benar!

Form2

Selamat datang di
aplikasi Visual C#
pertama anda!

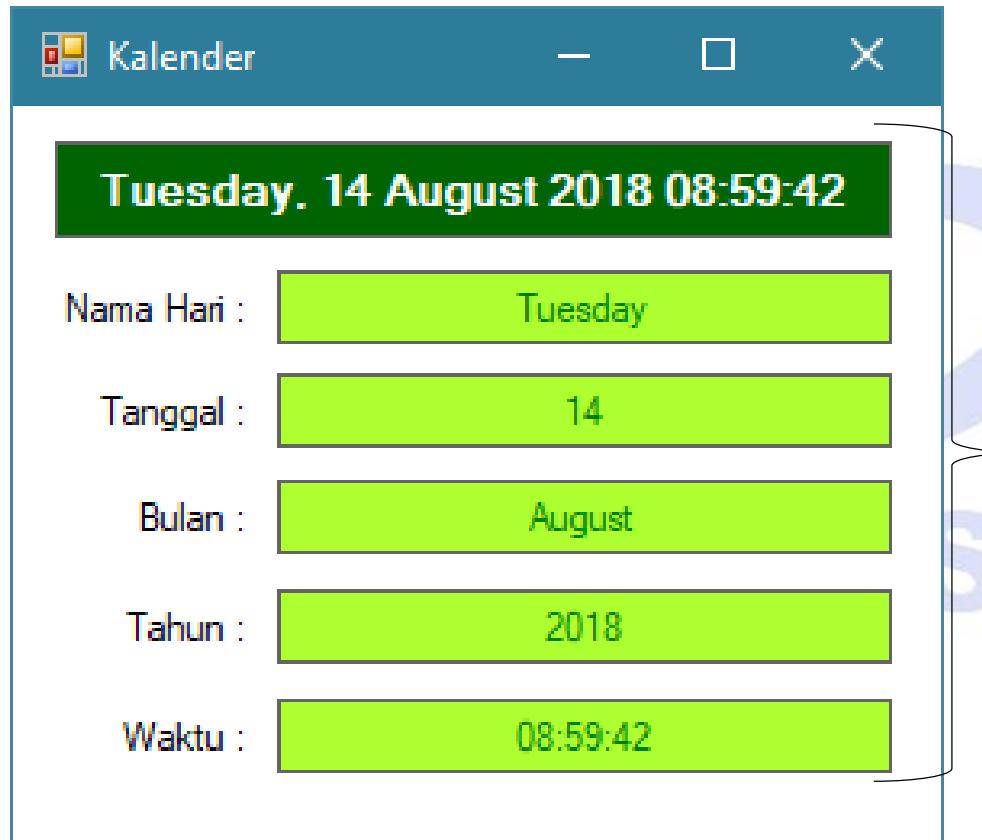
Tutup

Latihan 1 (Cont.)

Form1	Name	frmLatihan1a
	AcceptButton	btnValidasi
	CancelButton	btnKeluar
	Text	Validasi Password
Label1	Text	Masukkan password anda:
TextBox1	Name	txtPassword
	PasswordChar	*
Button1	Name	btnValidasi
	Text	&Validasi
Button2	Name	btnMasuk
	Text	&Masuk
Button3	Name	btnKeluar
		&Keluar

Label2	Name	lblKeterangan
	AutoSize	False
	BorderStyle	FixedSingle
	TextAlign	MiddleCenter
Form2	Name	frmLatihan1b
	ControlBox	False
Label1	AutoSize	False
	FontStyle	Bold
	FontSize	16
	Text	Selamat datang di aplikasi Visual C# pertama anda!
	TextAlign	MiddleCenter
Button1	Name	btnTutup
	Text	&Tutup

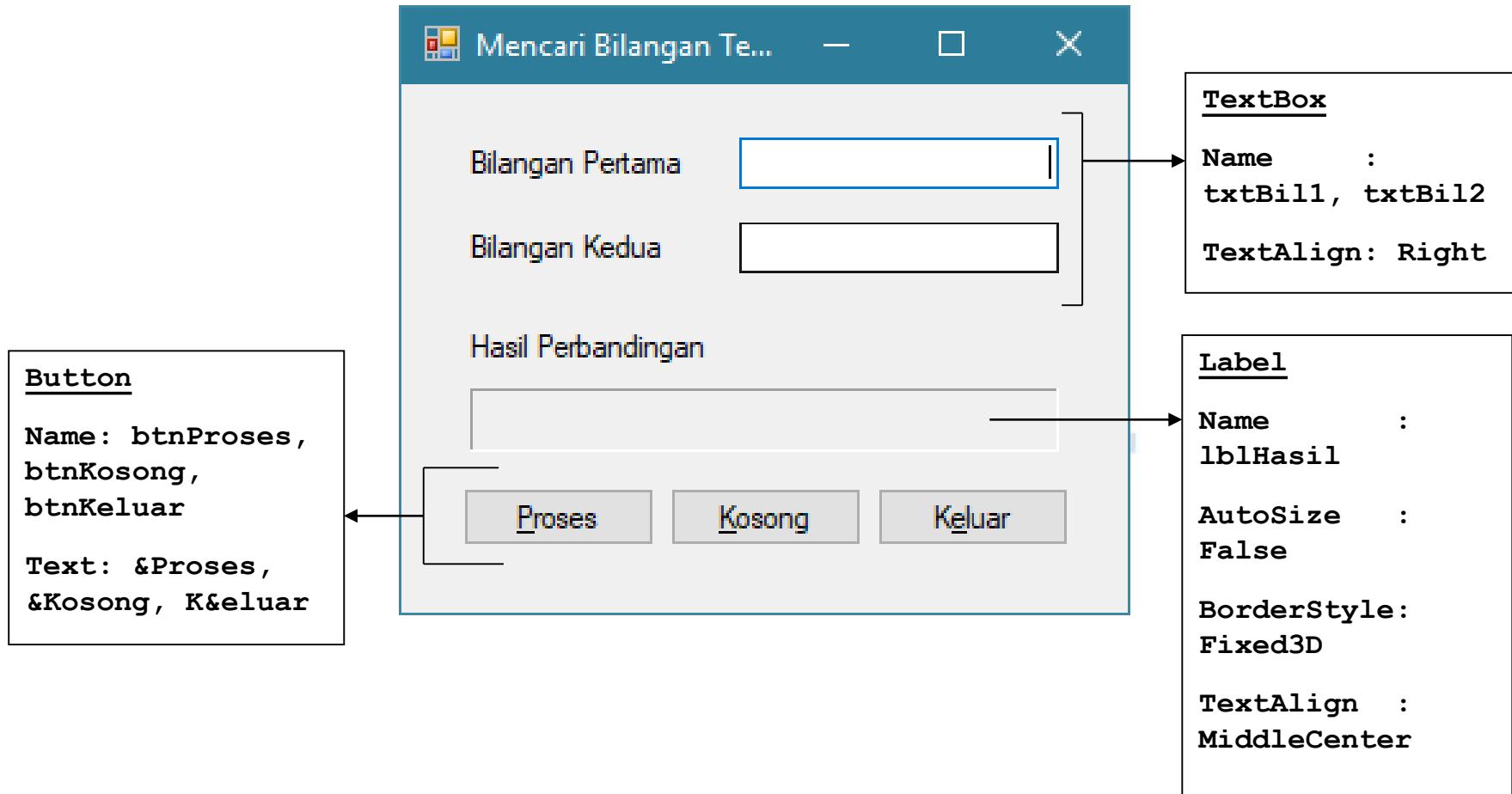
Latihan 2



Label

Name: lblTglLengkap,
lblHari, lblTanggal,
lblBulan, lblTahun, lblWaktu
AutoSize: False
BorderStyle: FixedSingle
 TextAlign: MiddleCenter
BackColor: DarkGreen
(lblTglLengkap), GreenYellow
ForeColor: MintCream
(lblTglLengkap), Green
Font: Bold, 10pt
(lblTglLengkap)

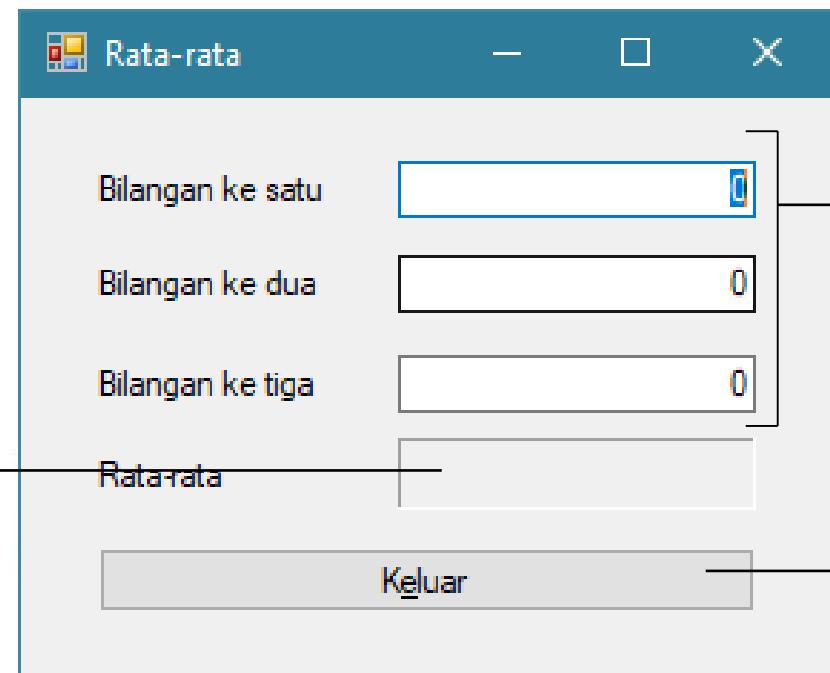
Latihan 3



Latihan 4

Label

Name : lblRata
AutoSize : False
BorderStyle: Fixed3D
TextAlign : MiddleRight



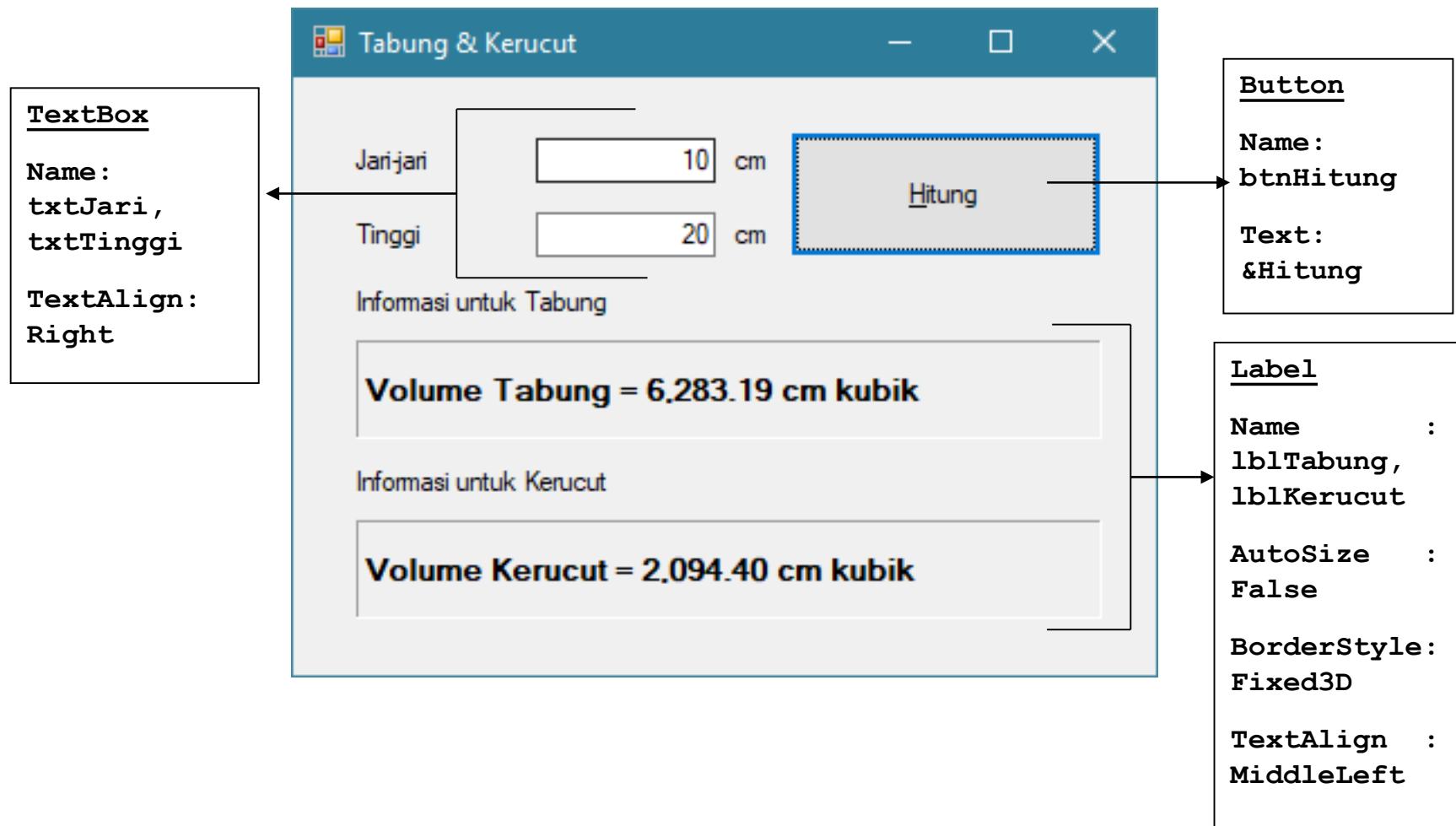
TextBox

Name : txtBill1, txtBill2, txtBill3
TextAlign: Right
Text : 0

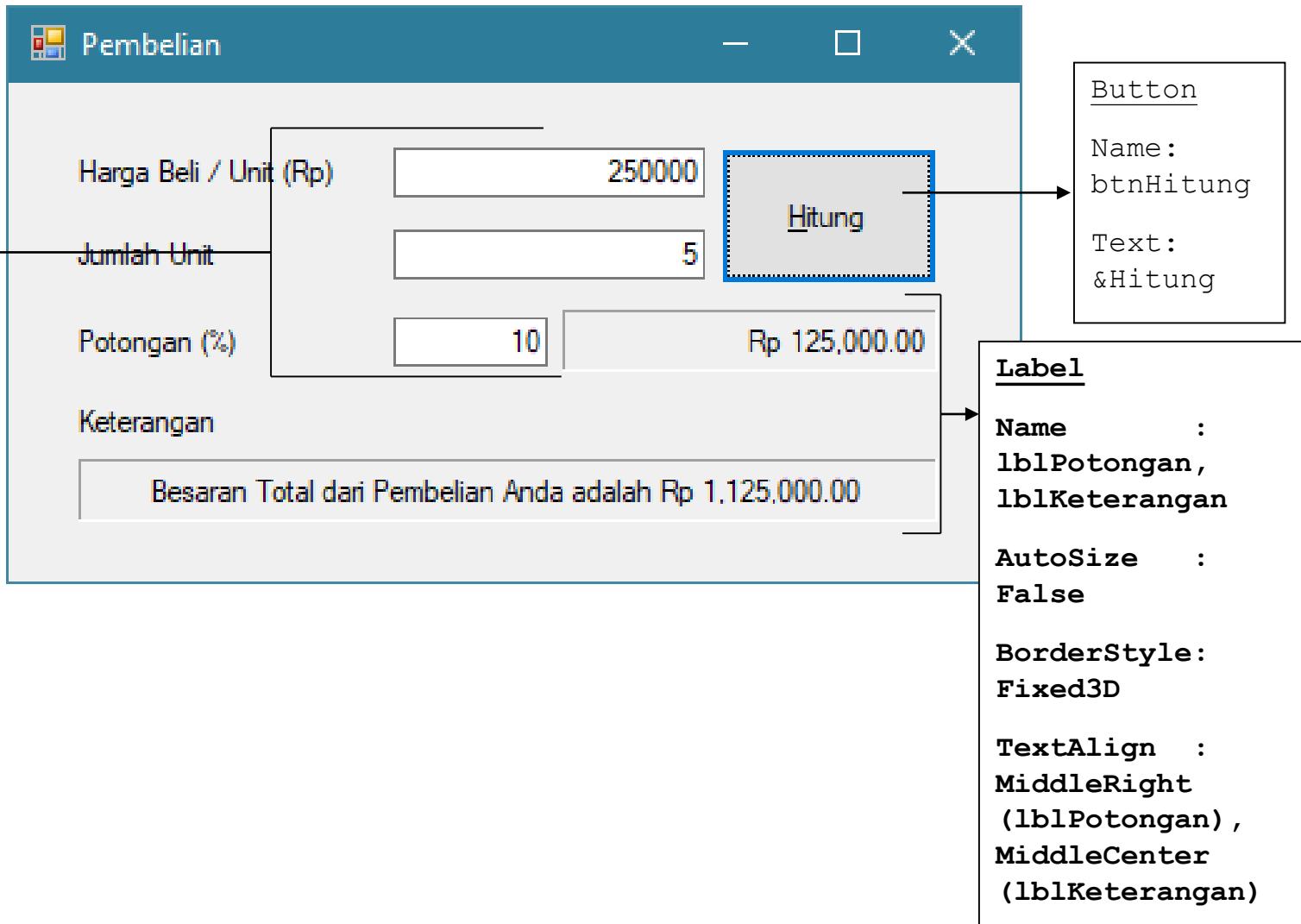
Button

Name: btnKeluar
Text: &Keluar

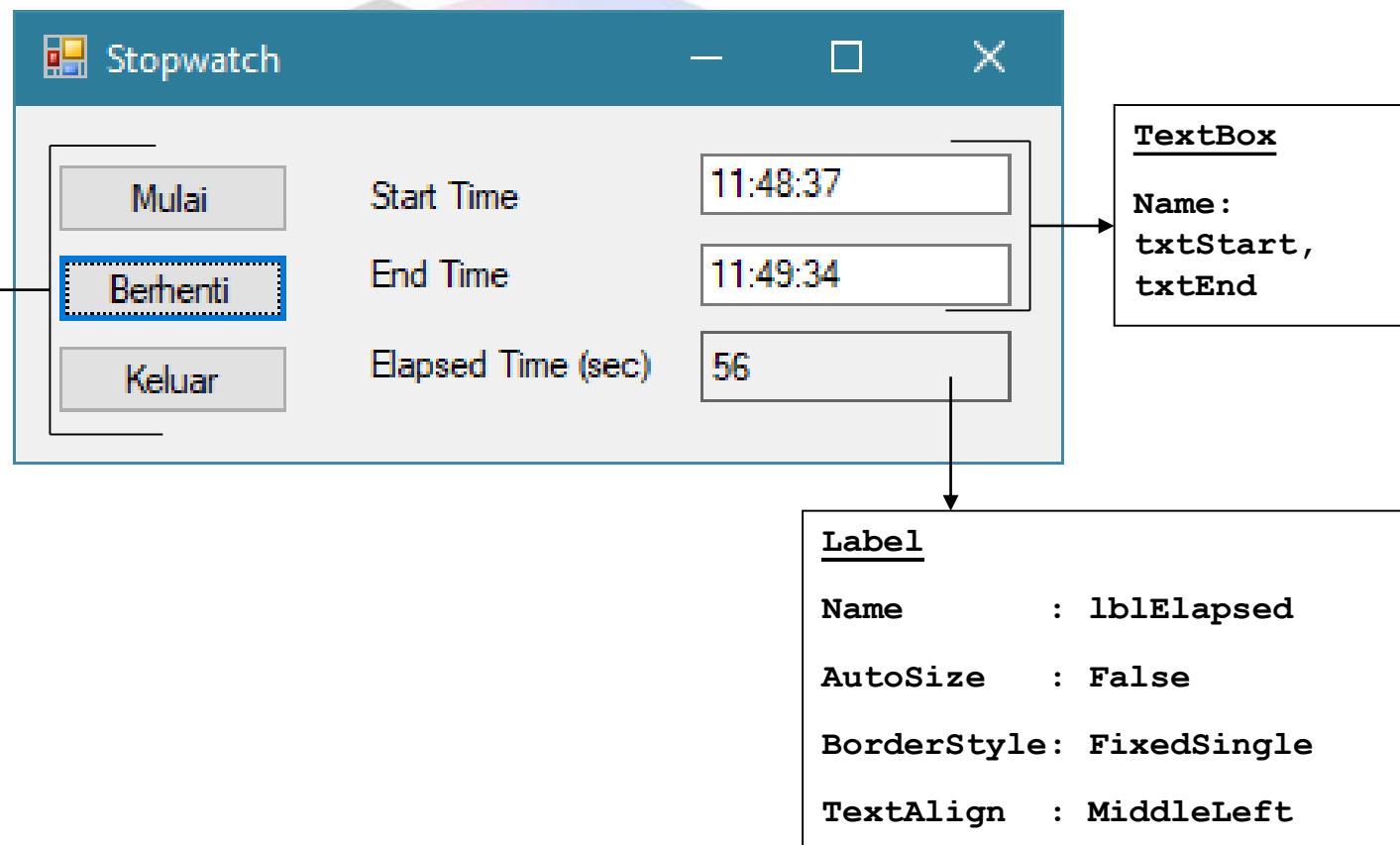
Latihan 5



Latihan 6



Latihan 7



GroupBox, Panel, CheckBox & RadioButton

MIKROSKIL

Lesson Overview

- Dalam memberikan pilihan inputan pada program, dapat menggunakan kontrol RadioButton maupun CheckBox. Kedua kontrol tersebut disebut sebagai kontrol ‘point and click’ yang memudahkan user untuk membuat pilihan dengan meng-klik menggunakan mouse sehingga mengurangi tingkat error pada aplikasi. Pilihan tersebut dapat dikelompokkan berdasarkan kriteria tertentu pada kontrol GroupBox maupun Panel.

MIKROSKIL

Objectives

Setelah mempelajari lesson ini, mahasiswa diharapkan mampu:

- Memahami karakteristik dan penggunaan kontrol GroupBox, Panel, CheckBox, dan RadioButton



GroupBox

- Digunakan untuk mengelompokkan sekumpulan kontrol yang saling berhubungan dalam aplikasi Visual C#.
- Kontrol-kontrol yang saling berhubungan harus dimasukkan ke dalam groupbox sehingga groupbox dan kontrol dapat digerakkan secara bersamaan.
- Sama seperti object form, kontrol groupbox merupakan kontrol container yang menampung kontrol-kontrol lain, sehingga property BackColor, ForeColor, dan Font akan diwariskan kepada kontrol. Hal yang sama berlaku untuk property Enabled dan Visible.
- Groupbox mempengaruhi cara kerja pilihan radiobutton.

GroupBox (Cont.)

- Radiobutton dalam kontrol groupbox bekerja sebagai suatu group yang terpisah (independen) dengan radiobutton di kontrol groupbox lain. Demikian pula dengan radiobutton dalam form juga dianggap sebagai satu group.
- Property umum pada groupbox:

Property	Keterangan
Name	Mengatur nama groupbox (GroupBox diawali dengan grp)
BackColor	Mengatur warna latar groupbox
Enabled	Jika diatur menjadi False, semua kontrol dalam groupbox kelihatan tetapi dalam kondisi disabled
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan
Text	Mengatur tulisan yang tampil pada judul groupbox
Visible	Jika bernilai False, groupbox (dan semua kontrol di dalamnya) menjadi tidak kelihatan

GroupBox (Cont.)

- Kontrol GroupBox memiliki beberapa method dan event, akan tetapi method dan event pada GroupBox jarang digunakan. Programmer lebih berfokus pada method dan event yang berhubungan dengan kontrol yang terdapat di dalam GroupBox.



Panel

- Fungsi dan perilakunya mirip dengan groupbox, yaitu digunakan untuk mengelompokkan kontrol.
- Perbedaan antara Panel dan GroupBox:
 - Panel memiliki BorderStyle, sedangkan GroupBox tidak.
 - GroupBox memiliki property Text untuk memberi judul (penjelasan isi atau fungsi GroupBox) dan tidak mengandung ScrollBar, sedangkan Panel dapat mengandung ScrollBar dan tidak dapat menampilkan Judul.

Panel (Cont.)

- Property umum pada panel:

Property	Keterangan
Name	Mengatur nama panel (Panel diawali dengan pnl)
AutoScroll	Mengatur apakah panel memunculkan scrollbar secara otomatis atau tidak
BackColor	Mengatur warna latar panel
BorderStyle	Mengatur tipe bingkai panel
Enabled	Jika diatur menjadi False, semua kontrol dalam panel kelihatan tetapi dalam kondisi disabled
Visible	Jika bernilai False, panel (dan semua kontrol di dalamnya) menjadi tidak kelihatan

Panel (Cont.)

- Kontrol Panel memiliki beberapa method dan event, akan tetapi method dan event pada Panel jarang digunakan. Programmer lebih berfokus pada method dan event yang berhubungan dengan kontrol yang terdapat di dalam Panel.



CheckBox

- Digunakan untuk membuat serangkaian pilihan dimana user dapat memilih 1 atau lebih dari pilihan-pilihan yang ada.
- Property umum pada checkbox:

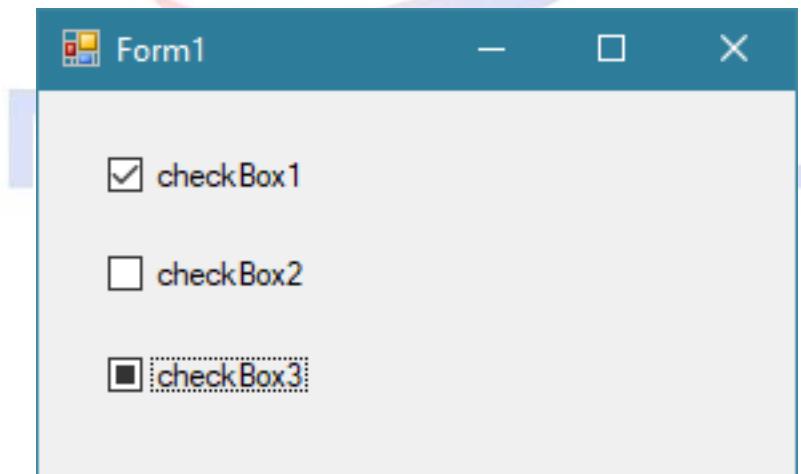
Property	Keterangan
Name	Mengatur nama checkbox (CheckBox diawali dengan chk)
AutoSize	Mengatur apakah ukuran kontrol berubah secara otomatis sesuai dengan isi kontrol atau tidak
BackColor	Mengatur warna latar checkbox
Checked	Mengatur apakah checkbox dalam kondisi terpilih (centang) atau tidak
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan

CheckBox (Cont.)

Property	Keterangan
Text	Mengatur tulisan yang ditampilkan di samping checkbox
TextAlign	Mengatur perataan tulisan pada checkbox
• Metode umum pada checkbox:	
Method	Keterangan
Focus	Mengatur focus pada checkbox
• Event umum pada checkbox:	
Event	Keterangan
CheckedChanged (default)	Terjadi ketika property Checked pada checkbox berubah (memunculkan / menghilangkan centang pada checkbox)

CheckBox (Cont.)

- Secara default, kontrol CheckBox hanya menerima 2 state (kondisi), yaitu checked dan unchecked.
- CheckBox dapat dikonfigurasi untuk menerima 3 state (kondisi), yaitu **checked**, **unchecked**, dan **indeterminate**. Caranya adalah dengan mengatur property **ThreeState** menjadi **True**.



RadioButton

- Digunakan untuk membuat serangkaian pilihan dimana user hanya dapat memilih 1 pilihan dari berbagai pilihan yang ada.
- Radiobutton yang diletakkan dalam form akan bekerja sebagai suatu group, dimana jika salah satu radiobutton terpilih, maka otomatis radiobutton lainnya dalam kondisi tidak terpilih.
- Untuk membuat pilihan radiobutton yang berbeda, dibutuhkan bantuan kontrol GroupBox ataupun Panel.
- Property umum pada radiobutton:

Property	Keterangan
Name	Mengatur nama radiobutton (RadioButton diawali dengan rdo)
AutoSize	Mengatur apakah ukuran kontrol berubah secara otomatis sesuai dengan isi kontrol atau tidak
BackColor	Mengatur warna latar radiobutton

RadioButton (Cont.)

Property	Keterangan
Checked	Mengatur apakah radiobutton dalam kondisi terpilih atau tidak
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan
Text	Mengatur tulisan yang ditampilkan di samping radiobutton
TextAlign	Mengatur perataan tulisan pada radiobutton
• Metode umum pada radiobutton:	
Method	Keterangan
Focus	Mengatur focus pada radiobutton

RadioButton (Cont.)

- Event umum pada radiobutton:

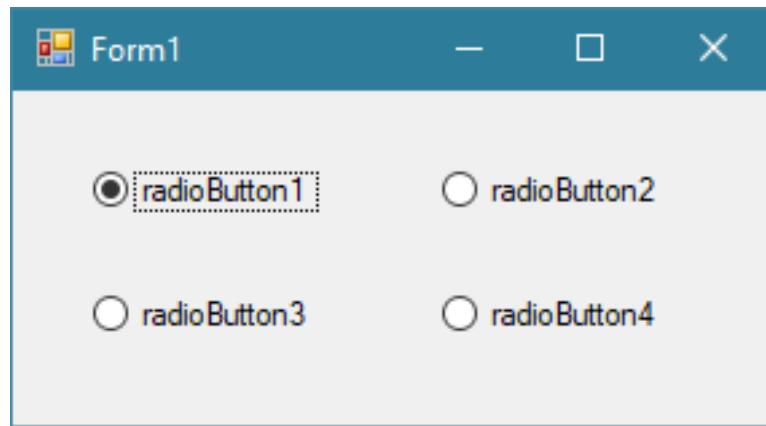
Event	Keterangan
CheckedChanged (default)	Terjadi ketika property Checked pada radiobutton berubah



Menangani Multiple Events

- Sebelumnya, setiap method akan menangani 1 event. Tetapi sekarang, dengan adanya kontrol seperti CheckBox dan RadioButton, akan lebih mudah apabila 1 buah method dapat menangani multiple events.
- Sebagai contoh, jika kita memiliki 4 RadioButton dalam sebuah group, ketika salah satu RadioButton dipilih, akan lebih mudah apabila terdapat 1 buah method untuk menentukan RadioButton mana yang dipilih, daripada harus membuat 4 buah method yang terpisah.

Menangani Multiple Events (Cont.)

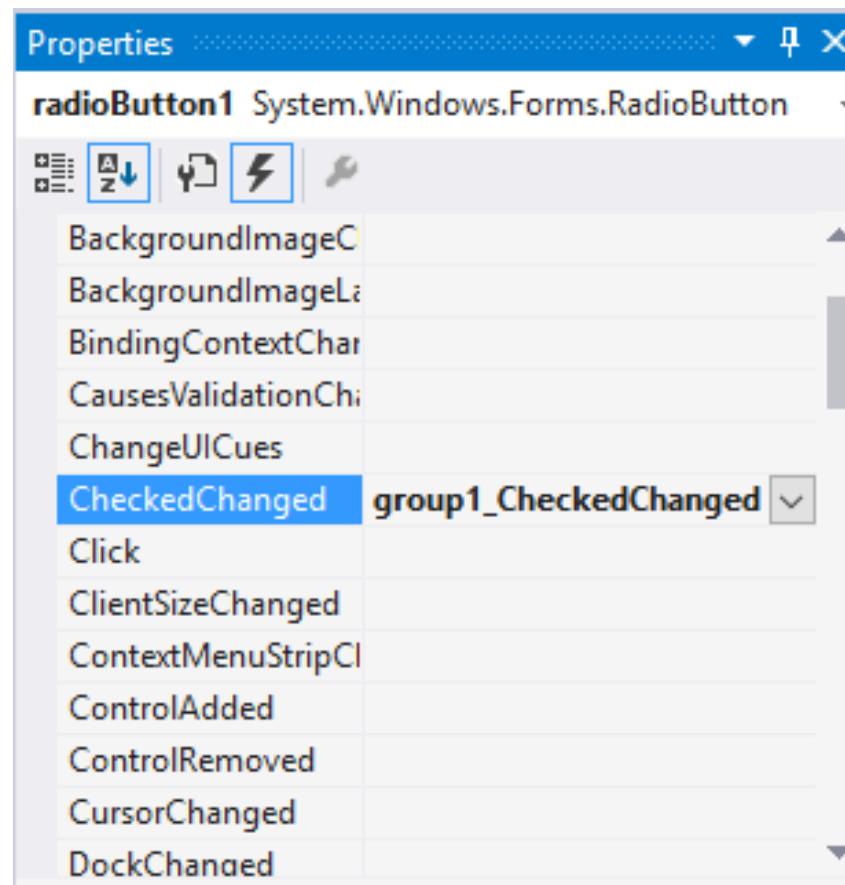


- Asumsikan kita memiliki 4 RadioButton yang saling berbagi event method **CheckedChanged**. Untuk melakukannya, pertama kita memilih radioButton1, kemudian buka Properties Window dan klik button Event. Cari event CheckedChanged.

Menangani Multiple Events (Cont.)

- Biasanya pada tahap ini, kita akan double klik pada event dan Visual C# akan membuat event method dengan nama radioButton1_CheckedChanged.
- Nama event method bersifat bebas. Kita dapat memberikan nama yang kita inginkan.
- Untuk menggunakan nama yang berbeda, kita mengetikkan nama pada isian di samping event CheckedChanged pada Properties Window.
- Nama yang digunakan sebaiknya dapat merepresentasikan 1 group RadioButton, bukan hanya 1 RadioButton.

Menangani Multiple Events (Cont.)



Menangani Multiple Events (Cont.)

- Setelah mengetikkan nama untuk event pada Properties Window dan menekan tombol Enter, maka pada code window akan memunculkan:

```
private void group1_CheckedChanged(object sender, EventArgs e)  
{  
}
```

- Method di atas hanya berlaku untuk radioButton1.
- Supaya kontrol lain dapat berbagi method ini, kita gunakan lagi Properties Window.
- Selanjutnya, pilih RadioButton yang lain. Buka Properties Window dan klik button Events, cari event CheckedChanged, kemudian pilih group1_CheckedChanged. Lakukan langkah yang sama untuk semua RadioButton.

Lesson Summary

Mahasiswa sekarang mampu:

- Memahami karakteristik dan penggunaan kontrol GroupBox, Panel, CheckBox, dan RadioButton





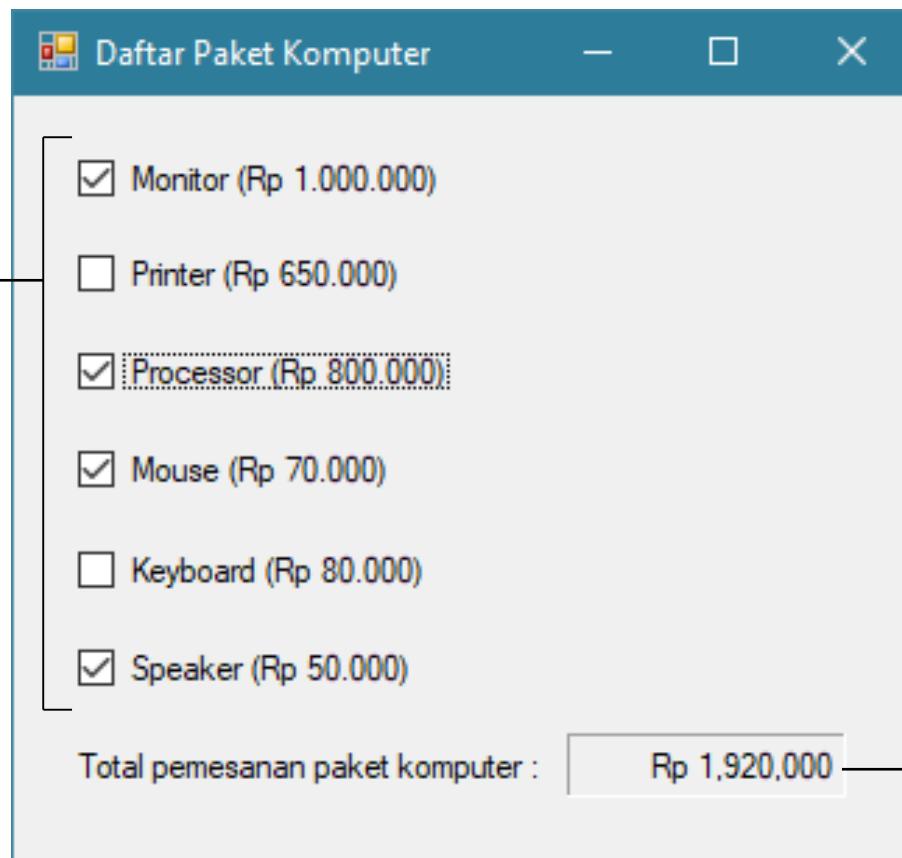
Latihan

MIKROSKIL

Latihan 1

CheckBox

Name :
chkMonitor,
chkPrinter,
chkProcessor,
chkMouse,
chkKeyboard,
chkSpeaker



Label

Name : lblTotal
AutoSize : False
BorderStyle: Fixed3D
 TextAlign : MiddleRight

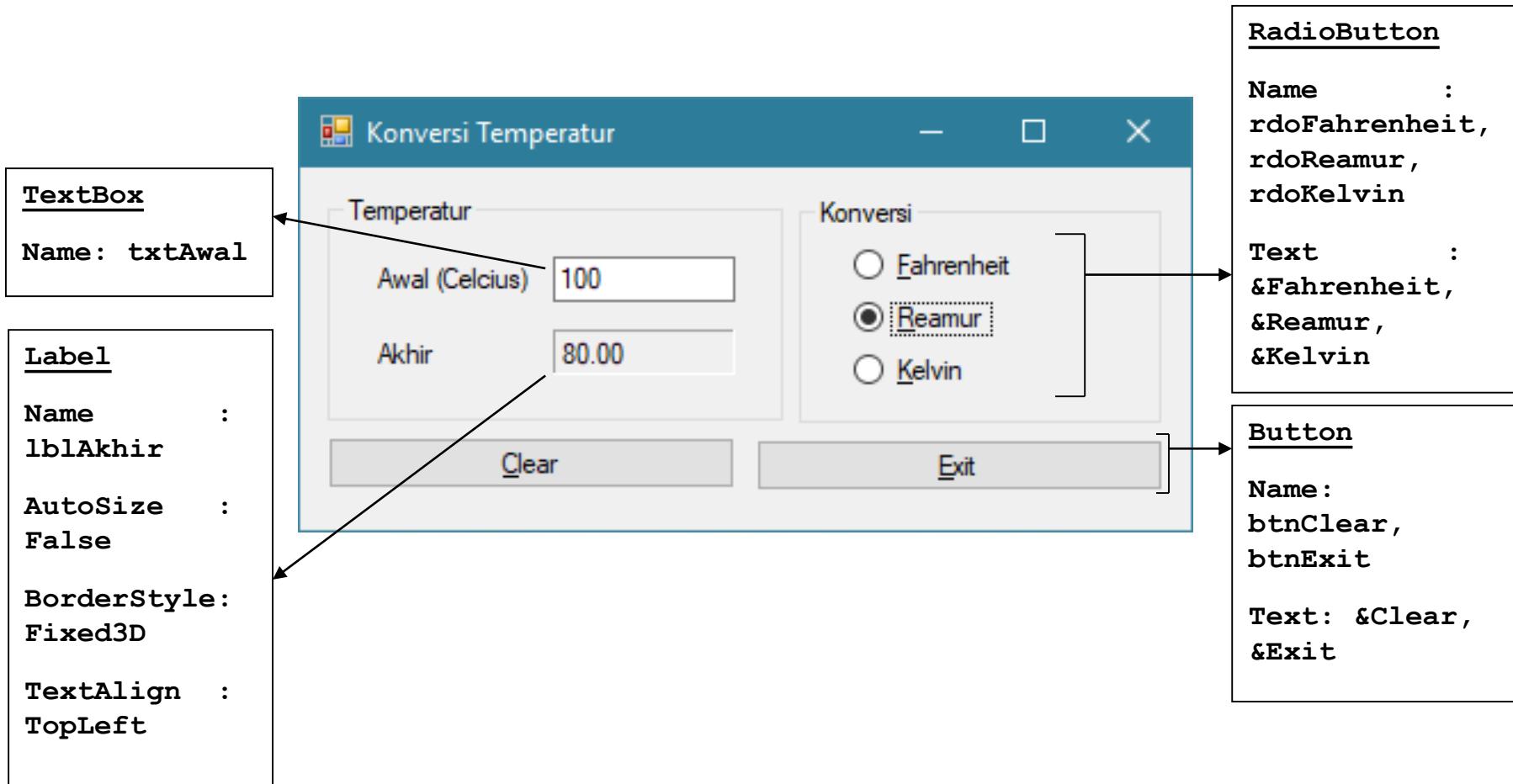
Latihan 2

Kursus Inggris

Paket

<input type="checkbox"/> Reading (Rp125.000)	Rp0
<input checked="" type="checkbox"/> Writing (Rp150.000)	Rp150.000
<input type="checkbox"/> Listening (Rp175.000)	Rp0
<input checked="" type="checkbox"/> Speaking (Rp200.000)	Rp200.000
Total	Rp350,000

Latihan 3



Latihan 4

Pembayaran

Masukkan pembelian :

Metode Pembayaran

Cash

Cash

Masukkan diskon (%) :

Nominal diskon :

Total :

Kredit

Kredit

Jumlah cicilan : kali

Besar Cicilan :

Hitung

Pembayaran

Masukkan pembelian :

Metode Pembayaran

Cash

Cash

Masukkan diskon (%) :

Nominal diskon :

Total :

Kredit

Kredit

Jumlah cicilan : kali

Besar Cicilan :

Hitung

Latihan 5

Pizza House

Ukuran

Small
 Medium
 Large

Topping

Extra Cheese Onions
 Mushrooms Green Peppers
 Black Olives Tomatoes

Pinggiran

Keju
 Sosis
 Nugget

Eat In Take Out

Pesan

Kosong **Keluar**

Pizza Order (TAKE OUT)

Ukuran : Small Pizza
Pinggiran : Keju

TOPPING

Extra Cheese
Mushrooms
Black Olives
Tomatoes



ListBox & ComboBox

MIKROSKIL

Lesson Overview

- Perbedaan ComboBox dan ListBox terdapat pada pilihan item. Pada ComboBox hanya satu item pilihan yang terlihat sebelum ComboBox diklik, sedangkan pada ListBox dapat menampilkan beberapa pilihan.



Objectives

Setelah mempelajari lesson ini, mahasiswa diharapkan mampu:

- Memahami karakteristik dan penggunaan kontrol ListBox dan ComboBox



ListBox

- Digunakan untuk menampilkan daftar dari sejumlah item atau pilihan menu dimana user dapat memilih 1 atau lebih item.
- Jika ListBox memiliki jumlah item yang melebihi ukurannya, secara otomatis akan menampilkan ScrollBar.
- Property umum pada listbox:

Property	Keterangan
Name	Mengatur nama listBox (ListBox diawali dengan Ist)
BackColor	Mengatur warna latar listBox
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan
Items	Mengatur item atau pilihan yang tampil pada listBox
SelectedIndex	Mengambil nilai indeks yang merupakan posisi item terpilih dalam listBox. Nilai indeks dimulai dari 0.

ListBox (Cont.)

Property	Keterangan
SelectedIndices	Mengambil nilai indeks yang merupakan posisi semua item terpilih dalam listbox. Item yang terpilih bisa lebih dari 1.
SelectedItem	Mengambil teks item terpilih dalam listbox
SelectedItems	Mengambil teks semua item terpilih dalam listbox. Item yang terpilih bisa lebih dari 1.
SelectionMode	Mengatur apakah user hanya dapat memilih 1 item atau boleh memilih lebih dari 1 item pada listbox
Sorted	Mengatur apakah item pada listbox diurutkan secara alfabetis
Text	Mengatur tulisan dari item terpilih pada listbox
TopIndex	Mengatur indeks dari item pertama yang terlihat di listbox

ListBox (Cont.)

- Metode umum pada listbox:

Method	Keterangan
ClearSelected	Menghilangkan seleksi semua item dalam listbox
FindString	Menemukan indeks dari item pertama dalam listbox yang dimulai dengan string yang dicari
GetSelected	Mengembalikan nilai apakah item tertentu dalam kondisi terseleksi atau tidak
SetSelected	Menyeleksi atau menghilangkan seleksi item dalam listbox

- Event umum pada listbox:

Event	Keterangan
SelectedIndexChanged (default)	Terjadi ketika property SelectedIndex berubah

ListBox (Cont.)

- **Items** pada ListBox memiliki property dan method sendiri untuk menambah atau menghapus item.
- Items pada ListBox dimulai dari indeks 0.
- Sehingga, item pertama pada ListBox dapat ditulis:

```
1stNama.Items[0]
```

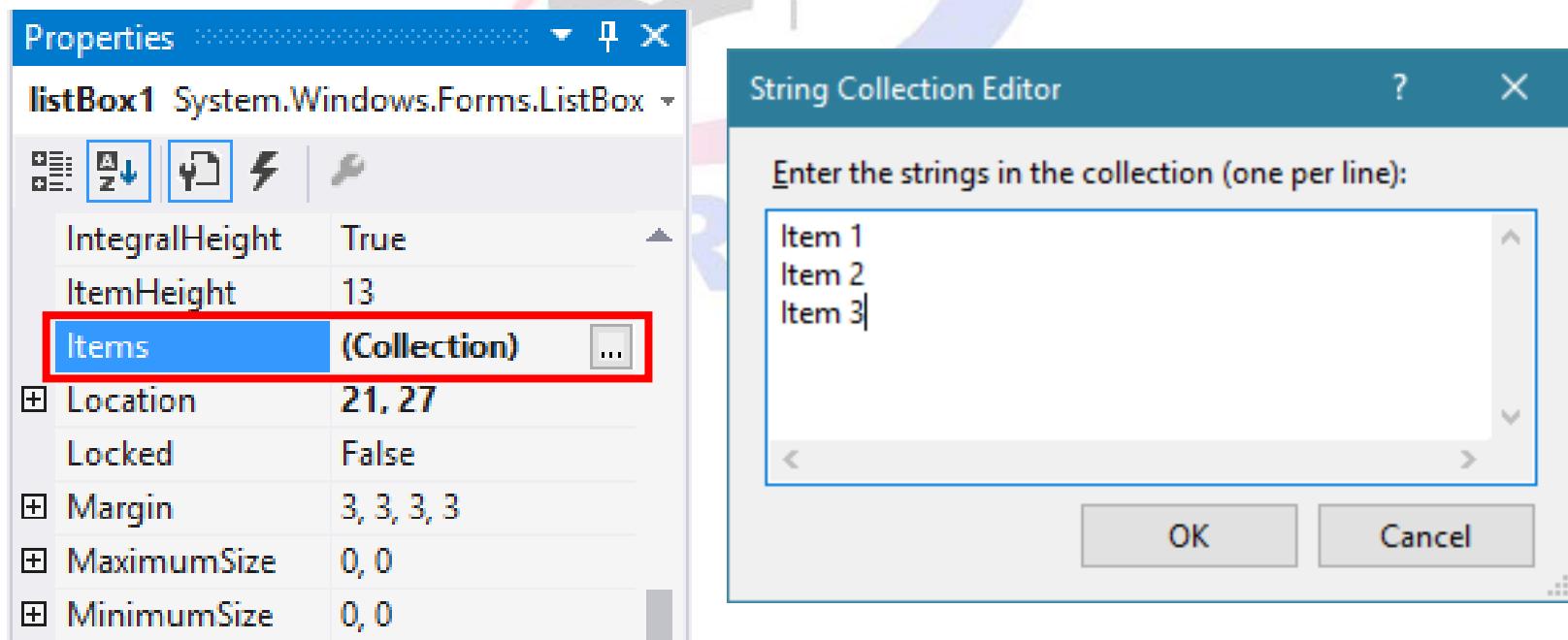
- Item terakhir pada ListBox dapat ditulis:

```
1stNama.Items[1stNama.Items.Count-1]
```

- **Count** (property Items) menentukan jumlah items pada ListBox.
- Pengurangan 1 dibutuhkan karena indeks dimulai dari 0.

ListBox (Cont.)

- Pada design mode, pilih property **Items**, kemudian klik tombol browse yang muncul di samping tulisan **(Collection)**. Kita dapat menambahkan item satu per satu, dengan catatan 1 item menempati 1 baris. Klik **OK** setelah selesai.



ListBox (Cont.)

- Pada tampilan code editor:

1. Untuk menambah item, gunakan method **Add**.

```
lstNama.Items.Add(ItemToAdd);
```

2. Untuk menghapus item, gunakan method **Remove** ataupun **RemoveAt**.

```
lstNama.Items.Remove(ItemToRemove);
```

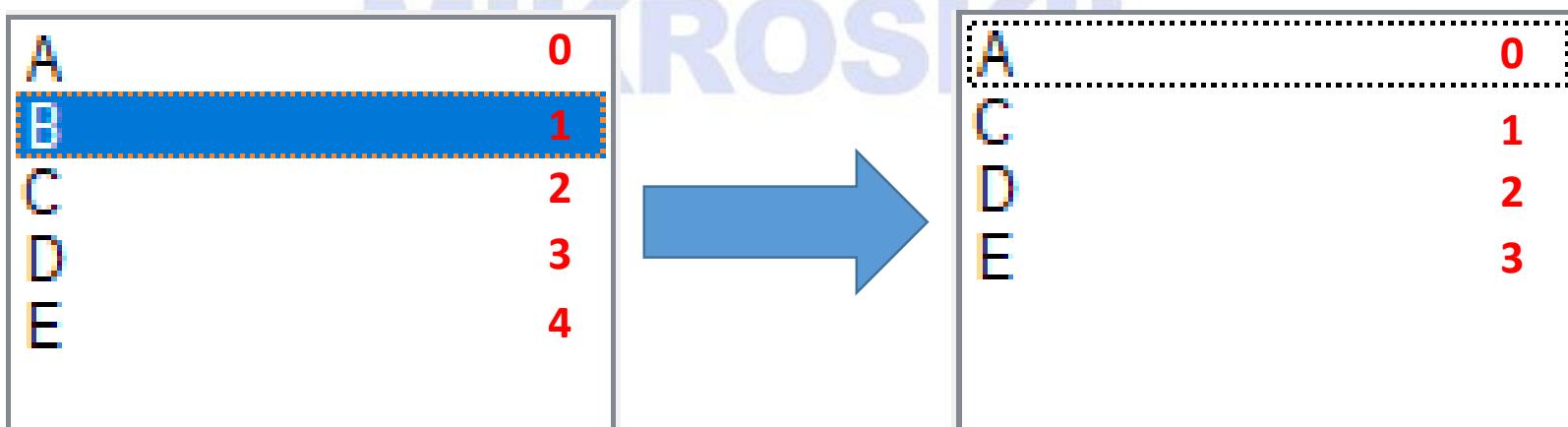
```
lstNama.Items.RemoveAt(IndexofItemToRemove);
```

3. Untuk menghapus semua item pada ListBox, gunakan method **Clear**.

```
lstNama.Items.Clear();
```

ListBox (Cont.)

- ListBox umumnya berisi tipe data string, walaupun tipe data yang lain juga memungkinkan.
- Sebagai catatan, ketika kita menghapus suatu item, maka index pada item-item berikutnya akan berubah karena proses penghapusan item tersebut. Contoh: apabila pada ListBox terdapat 5 item, dan kita menghapus item pada index 1, maka item-item berikutnya akan menempati index 1, 2, dan 3.



ListBox (Cont.)

- Property **SelectionMode** menentukan apakah user hanya dapat memilih 1 item saja atau dapat memilih lebih dari 1 item.
 - **SelectionMode.One** : user hanya dapat memilih 1 item saja.
 - **SelectionMode.MultiExtended** : user dapat memilih lebih dari 1 item dengan cara menekan tombol Shift + mouse click atau tombol Shift + tombol panah untuk menambah item. User juga dapat menekan tombol Ctrl + mouse click untuk select atau deselect item.
 - **SelectionMode.MultiSimple** : user dapat memilih lebih dari 1 item dengan cara mouse click atau menekan tombol space pada keyboard untuk select atau deselect item.

ListBox (Cont.)

- **SelectedItems** juga memiliki property tersendiri untuk menentukan item-item yang terpilih pada ListBox. Salah satu property adalah **Count** yang digunakan untuk mengetahui jumlah item yang terpilih. Nilai ini bersama dengan array SelectedIndices dapat mengidentifikasi serangkaian item yang terpilih.
- Contoh:

```
for (int i = 0; i <= lstNama.SelectedItems.Count - 1; i++)  
{  
    lblTampil.Text += lstNama.SelectedItems[i] + "\n";  
}
```

ComboBox

- Apabila kontrol ListBox mirip dengan sekumpulan CheckBox, maka kontrol ComboBox mirip dengan sekumpulan RadioButton.
- ComboBox memungkinkan user memilih 1 item dari daftar, dan dalam beberapa kasus, memungkinkan user untuk mengetikkan pilihan alternatif.
- Property pada ComboBox mirip dengan ListBox, hanya saja tidak ada property **SelectionMode** dan terdapat property **DropDownStyle**.
- Property umum pada combobox:

Property	Keterangan
Name	Mengatur nama combobox (ComboBox diawali dengan cbo)
BackColor	Mengatur warna latar combobox
DropDownStyle	Mengatur style dropdown dari combobox

ComboBox (Cont.)

Property	Keterangan
Font	Mengatur jenis, style, dan ukuran tulisan
ForeColor	Mengatur warna tulisan
Items	Mengatur item atau pilihan yang tampil pada combobox
MaxDropDownItems	Mengatur jumlah item yang tampil pada bagian dropdown combobox. Property IntegralHeight harus diatur jadi False.
SelectedIndex	Mengatur nilai indeks yang merupakan posisi item terpilih dalam combobox. Nilai indeks dimulai dari 0.
SelectedItem	Mengatur item terpilih dalam combobox
SelectedText	Mengatur teks yang terpilih pada bagian combobox yang dapat diedit
Sorted	Mengatur apakah items diurutkan secara alfabetis
Text	Mengatur nilai string yang ditampilkan di combobox

ComboBox (Cont.)

- Event umum pada combobox:

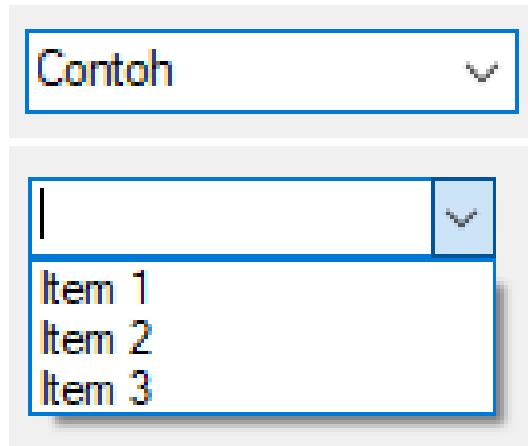
Event	Keterangan
KeyPress	Terjadi ketika tombol keyboard ditekan pada saat focus berada di kontrol combobox
SelectedIndexChanged	Terjadi ketika property SelectedIndex berubah (default)

- **Items** pada kontrol ComboBox mirip dengan items pada kontrol ListBox sehingga cara untuk menambah dan menghapus item pada ComboBox juga sama seperti pada ListBox.

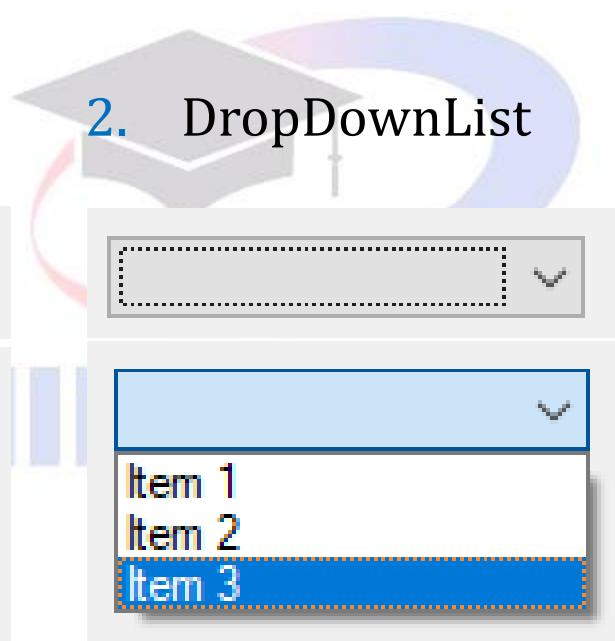
ComboBox (Cont.)

- Property DropDownListStyle pada ComboBox memiliki 3 pilihan nilai, yaitu:

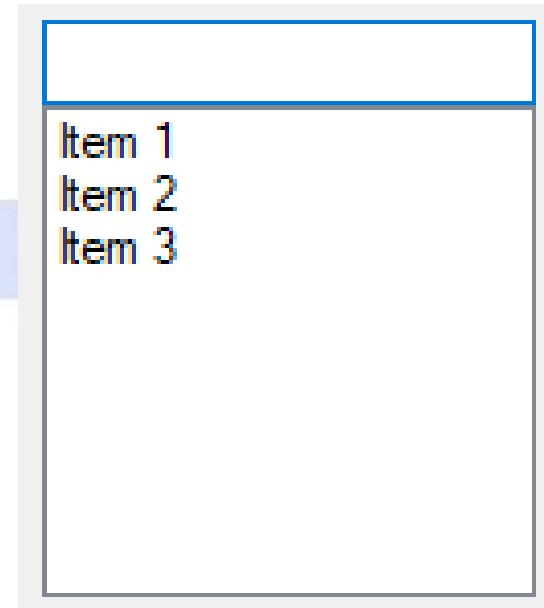
1. DropDown



2. DropDownList



3. Simple



Lesson Summary

Mahasiswa sekarang mampu:

- Memahami karakteristik dan penggunaan kontrol ListBox dan ComboBox





MIKROSKIL

Lesson Overview

- Kontrol MessageBox digunakan untuk menyajikan pesan kepada pengguna.



Objectives

Setelah mempelajari lesson ini, mahasiswa diharapkan mampu:

- Memahami karakteristik dan penggunaan dari kontrol MessageBox

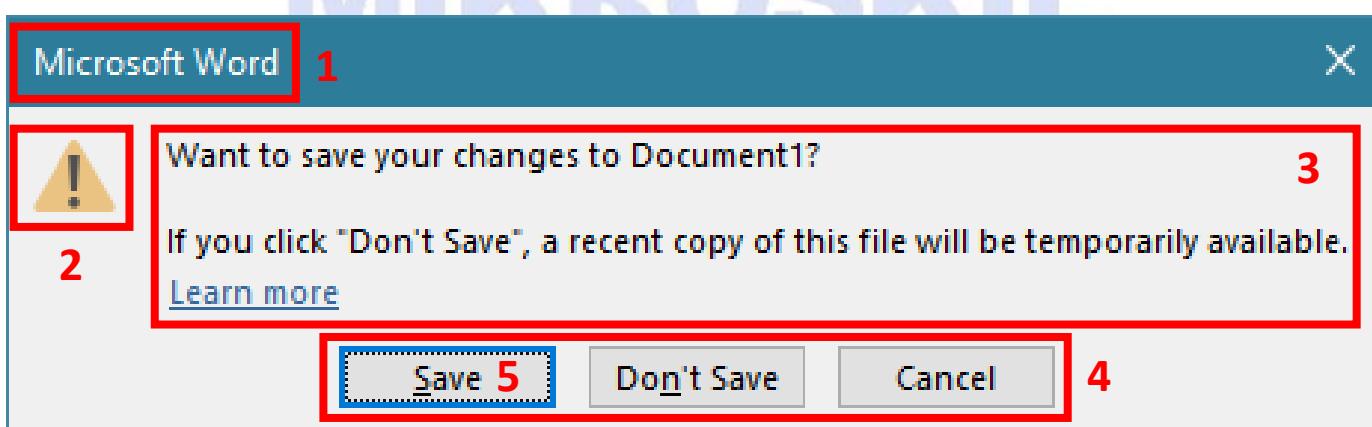


MessageBox

- Salah satu method yang paling banyak digunakan dalam Visual C# adalah objek MessageBox. Objek ini mengizinkan programmer untuk menampilkan pesan kepada user dan menerima feedback (respon) untuk informasi lebih lanjut.
- MessageBox dapat digunakan untuk menampilkan pesan error, menjelaskan masalah potensial, atau hanya untuk menampilkan hasil dari beberapa perhitungan.
- Objek MessageBox sangat berguna dengan kemampuan untuk menampilkan pesan, icon, maupun berbagai pilihan jenis tombol. User akan merespon dengan menekan salah satu tombol pada MessageBox.

MessageBox (Cont.)

- Untuk menggunakan MessageBox, programmer harus menentukan:
 - Caption yang berisi judul MessageBox
 - Icon yang sesuai
 - Text yang berisi pesan yang ingin disampaikan.
 - Button yang berisi pilihan tombol yang tampil pada MessageBox
 - Default Button untuk menentukan tombol mana yang dijadikan default



MessageBox (Cont.)

- Untuk menampilkan MessageBox, dilakukan melalui koding dengan menggunakan method **Show**.
- Ada beberapa cara untuk menggunakan method Show pada MessageBox:

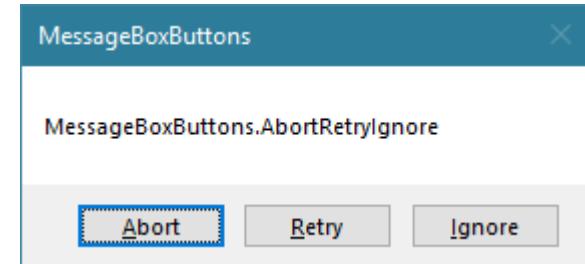
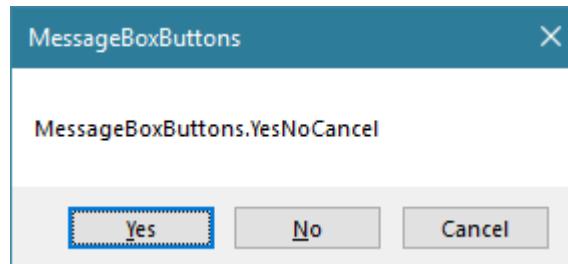
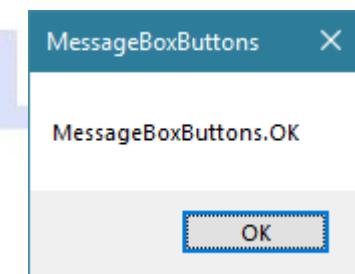
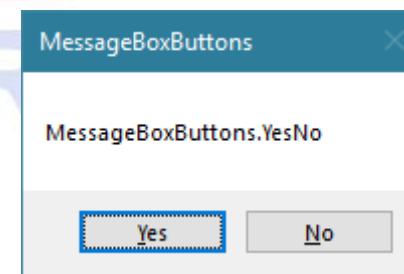
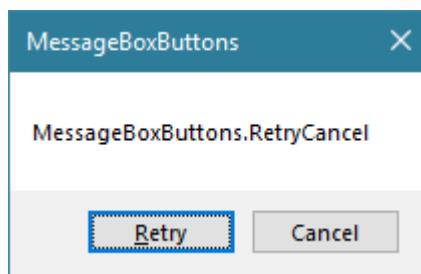
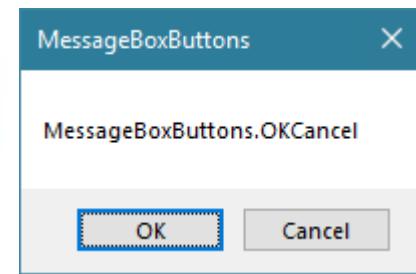
```
MessageBox.Show("text");  
MessageBox.Show("text", "caption");  
MessageBox.Show("text", "caption", Buttons);  
MessageBox.Show("text", "caption", Buttons, Icon);  
MessageBox.Show("text", "caption", Buttons, Icon, DefaultButton);
```

MessageBox (Cont.)

- Dalam implementasinya:
 - Jika parameter Default Button tidak diberikan, maka button pertama akan dijadikan sebagai default
 - Jika parameter Icon tidak diisi, tidak akan ada icon yang ditampilkan
 - Jika parameter Button tidak diisi, maka akan ditampilkan sebuah tombol OK.
 - Jika parameter Caption tidak diisi, maka tidak akan ada judul yang tampil
- Untuk parameter Text dan Caption, dapat diisi sendiri dengan tipe data String. Parameter yang lain ditentukan oleh konstanta yang sudah tersedia dalam Visual C#.

MessageBox (Cont.)

- Nilai yang dapat diisi dalam parameter MessageBoxButtons:
 1. `MessageBoxButtons.AbstRetryIgnore`
 2. `MessageBoxButtons.OK`
 3. `MessageBoxButtons.OKCancel`
 4. `MessageBoxButtons.RetryCancel`
 5. `MessageBoxButtons.YesNo`
 6. `MessageBoxButtons.YesNoCancel`



MessageBox (Cont.)

- Nilai yang dapat diisi dalam parameter MessageBoxIcon:
 - MessageBoxIcon.Asterisk
 - MessageBoxIcon.Information
 - MessageBoxIcon.Error
 - MessageBoxIcon.Hand
 - MessageBoxIcon.None
 - MessageBoxIcon.Stop
 - MessageBoxIcon.Exclamation
 - MessageBoxIcon.Warning
 - MessageBoxIcon.Question
 - Walaupun terdapat 8 nilai yang dapat diisi pada MessageBoxIcon, hanya ada 4 jenis icon yang akan tampil yaitu Information, Error, Exclamation, dan Question. Hal ini dikarenakan sistem operasi Windows saat ini hanya menyediakan 4 icon.
-
- The diagram illustrates the four icons available for MessageBoxIcon parameters. It features four icons arranged in a 2x2 grid. Top-left: A blue circle containing a white lowercase 'i'. Top-right: A yellow triangle pointing upwards containing a black exclamation mark '!' (Exclamation icon). Bottom-left: A red circle containing a large white 'X' (Error icon). Bottom-right: A blue circle containing a white question mark '?' (Question icon). To the right of each icon is a red numerical value or range: '1,2' for the 'i' icon, '7,8' for the '!' icon, '3,4,6' for the 'X' icon, and '9' for the '?' icon.

MessageBox (Cont.)

- Nilai yang dapat diisi dalam parameter MessageBoxDefaultButton:
 1. `MessageBoxDefaultButton.Button1`
 2. `MessageBoxDefaultButton.Button2`
 3. `MessageBoxDefaultButton.Button3`
- Default Button yang ditentukan bersifat relatif pada Button yang ditampilkan dari kiri ke kanan. Misalnya jika kita memiliki MessageBox dengan button Yes, No, dan Cancel, maka Button1 adalah Yes, Button2 adalah No, dan Button3 adalah Cancel.

MessageBox (Cont.)

- Ketika user memberikan respon terhadap MessageBox dengan menekan salah satu Button, maka untuk mengetahui button mana yang dipilih oleh user dapat menggunakan konstanta DialogResult.
- Nilai DialogResult yang tersedia sesuai dengan tombol yang ditampilkan pada MessageBox:
 1. `DialogResult.Abort`
 2. `DialogResult.Cancel`
 3. `DialogResult.Ignore`
 4. `DialogResult.No`
 5. `DialogResult.OK`
 6. `DialogResult.Retry`
 7. `DialogResult.Yes`

MessageBox (Cont.)

```
if (MessageBox.Show("Ini adalah contoh MessageBox",  
"Contoh MessageBox", MessageBoxButtons.OKCancel,  
MessageBoxIcon.Information,  
MessageBoxDefaultButton.Button1) == DialogResult.OK)  
{  
    //baris kode jika menekan tombol OK  
}  
else  
{  
    //baris kode jika menekan tombol Cancel  
};
```

Lesson Summary

Sekarang mahasiswa mampu:

- Memahami karakteristik dan penggunaan dari kontrol MessageBox





MIKROSKIL

Lesson Overview

- Kotak Dialog digunakan untuk berinteraksi dengan pengguna dan mengambil informasi. Kotak Dialog yang dibahas meliputi OpenFileDialog, SaveFileDialog, ColorDialog, FontDialog, Print Dialog, PrintPreviewDialog, dan PageSetupDialog. OpenFileDialog digunakan untuk memilih file yang akan dibuka. SaveFileDialog digunakan untuk memilih lokasi penyimpanan file. ColorDialog digunakan untuk memilih warna yang ingin dipakai. FontDialog digunakan untuk memilih font tulisan yang dipakai. PrintDialog digunakan untuk memilih printer, halaman yang diprint dan settingan terkait print lainnya. PrintPreviewDialog digunakan untuk menampilkan tampilan halaman yang akan diprint. PageSetupDialog digunakan untuk menampilkan pengaturan halaman dokumen yang mau diprint.

Objectives

Setelah mempelajari lesson ini, mahasiswa diharapkan mampu:

- Memahami karakteristik dan penggunaan dari kontrol DialogBox.



SaveFileDialog

- Dengan menggunakan SaveFileDialog, kita dapat menyimpan file dengan memilih lokasi penyimpanan file yang diinginkan.
- Apabila user menggunakan nama file yang sudah ada pada lokasi yang dipilih, maka file tersebut akan ditimpa dengan file baru.
- Sebagai catatan, kontrol SaveFileDialog tidak memiliki interface (antarmuka) yang bisa dilihat pada saat mendesain form. Setelah user memasukkan kontrol SaveFileDialog ke dalam form, kontrol SaveFileDialog akan tampil di tray area.

SaveFileDialog

- Property umum pada SaveFileDialog:

Property	Keterangan
Name	Mengatur nama savefiledialog (SaveFileDialog umumnya dinamai dengan dlgSave)
AddExtension	Menentukan apakah savefiledialog akan menambahkan extension secara otomatis pada nama file jika user tidak memasukkan extension
CheckFileExists	Menentukan apakah savefiledialog akan menampilkan peringatan jika nama file yang ingin disimpan sudah ada pada lokasi penyimpanan
CheckPathExists	Menentukan apakah savefiledialog akan menampilkan peringatan jika user memasukkan lokasi yang tidak tersedia
CreatePrompt	Menentukan apakah kotak dialog akan meminta persetujuan user untuk membuat file baru apabila user memasukkan file yang belum ada

SaveFileDialog (Cont.)

Property	Keterangan
DefaultExt	Menentukan extension default pada file
FileName	Menentukan string yang berisi nama file yang dipilih di dalam kotak dialog
Filter	Menentukan pilihan tipe file yang dapat disimpan ke dalam kotak dialog
FilterIndex	Menentukan indeks pilihan filter yang terpilih pada kotak dialog
InitialDirectory	Menentukan folder yang akan ditampilkan pertama kali ketika kotak dialog muncul
OverwritePrompt	Mengindikasikan apakah kotak dialog akan memberikan peringatan jika user memasukkan nama file yang sudah ada
Title	Menentukan judul dari kotak dialog

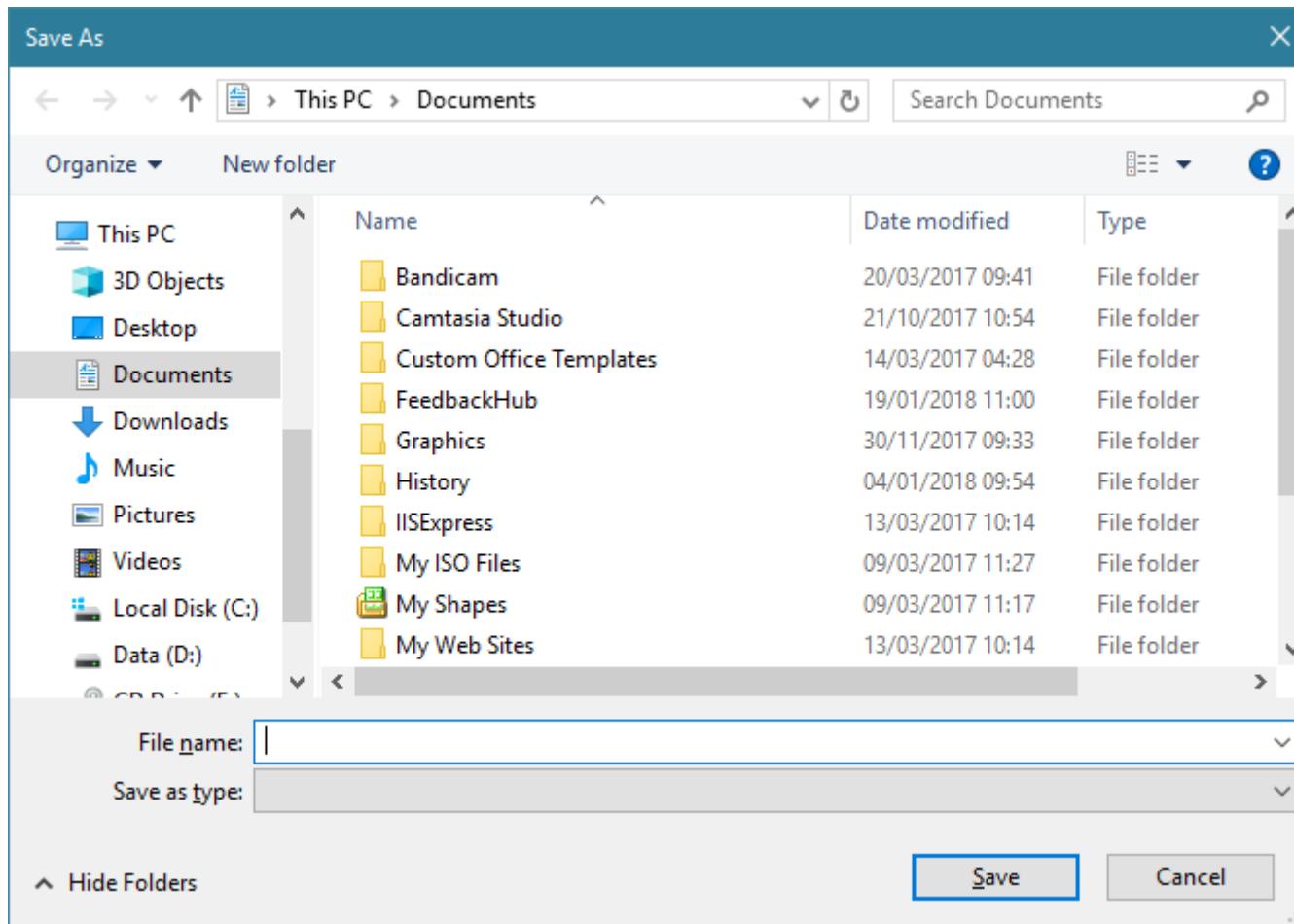
SaveFileDialog (Cont.)

- Method umum pada SaveFileDialog:

Method	Keterangan
ShowDialog	Menampilkan kotak dialog, mengembalikan nilai yang mengindikasikan button mana yang dipilih oleh user: OK atau Cancel.

- Untuk mengetahui button mana yang dipilih oleh user dapat menggunakan konstanta DialogResult.

SaveFileDialog (Cont.)



OpenFileDialog

- Pada sebagian aplikasi, programmer membutuhkan file name dari user. Namun, jika meminta user untuk mengetikkan lokasi dan nama file, rawan terjadi kesalahan sehingga dibutuhkan interface yang bersifat ‘point-and-click’ untuk mendapatkan nama file tersebut.
- Kontrol yang dapat digunakan adalah OpenFileDialog.
- Sebagai catatan, kontrol OpenFileDialog tidak memiliki interface (antarmuka) yang bisa dilihat pada saat mendesain form. Setelah user memasukkan kontrol OpenFileDialog ke dalam form, kontrol OpenFileDialog akan tampil di tray area.

OpenFileDialog (Cont.)

- Property umum pada OpenFileDialog:

Property	Keterangan
Name	Mengatur nama openfiledialog (OpenFileDialog umumnya dinamai dengan dlgOpen)
AddExtension	Menentukan apakah openfiledialog akan menambahkan extension secara otomatis pada nama file jika user tidak memasukkan extension
CheckFileExists	Menentukan apakah openfiledialog akan menampilkan peringatan jika nama file yang ingin dibuka belum ada pada lokasi penyimpanan
CheckPathExists	Menentukan apakah openfiledialog akan menampilkan peringatan jika user memasukkan lokasi yang tidak tersedia
DefaultExt	Menentukan extension default pada file

OpenFileDialog (Cont.)

Property	Keterangan
FileName	Menentukan string yang berisi nama file yang dipilih di dalam kotak dialog
Filter	Menentukan pilihan tipe file yang dapat dibuka ke dalam kotak dialog
FilterIndex	Menentukan indeks pilihan filter yang terpilih pada kotak dialog
InitialDirectory	Menentukan folder yang akan ditampilkan pertama kali ketika kotak dialog muncul
Title	Menentukan judul dari kotak dialog

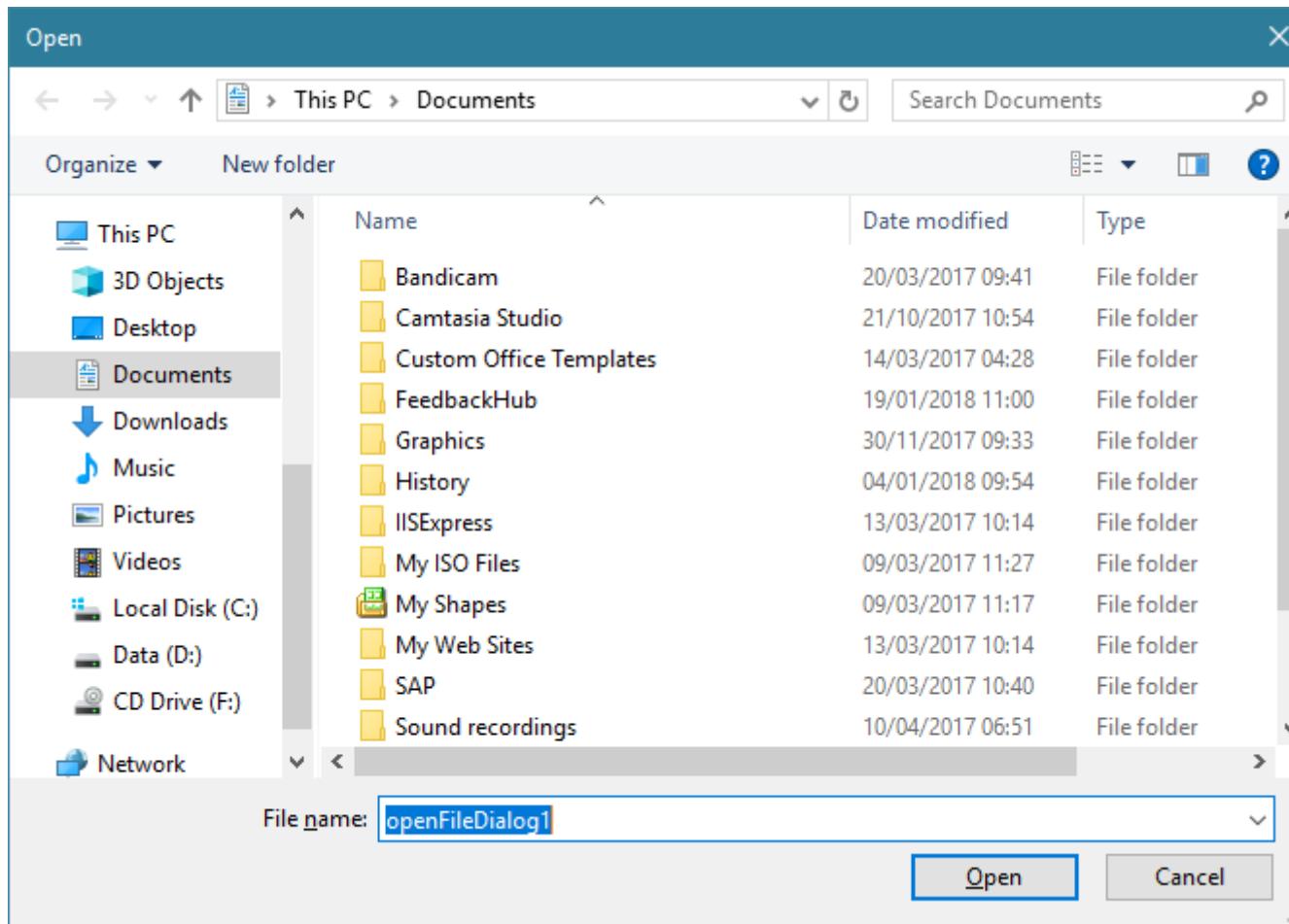
OpenFileDialog (Cont.)

- Method umum pada OpenFileDialog:

Method	Keterangan
ShowDialog	Menampilkan kotak dialog, mengembalikan nilai yang mengindikasikan button mana yang dipilih oleh user: OK atau Cancel.

- Untuk mengetahui button mana yang dipilih oleh user dapat menggunakan konstanta DialogResult.

OpenFileDialog (Cont.)



FontDialog

- FontDialog menyediakan fitur bagi user untuk mengganti tipe, style, ukuran, dan warna font.
- Sebagai catatan, kontrol FontDialog tidak memiliki interface (antarmuka) yang bisa dilihat pada saat mendesain form. Setelah user memasukkan kontrol FontDialog ke dalam form, kontrol FontDialog akan tampil di tray area.
- Property umum pada FontDialog:

Property	Keterangan
Name	Mengatur nama fontdialog (FontDialog umumnya dinamai dengan dlgFont)
Color	Mengindikasikan warna font
Font	Mengindikasikan tipe font
MaxSize	Menentukan ukuran maksimum font dalam satuan point yang dapat dipilih

FontDialog (Cont.)

Property	Keterangan
MinSize	Menentukan ukuran minimum font dalam satuan point yang dapat dipilih
ShowColor	Mengindikasikan apakah kotak dialog font menampilkan kotak pilihan warna. Nilai defaultnya adalah false.
ShowEffects	Mengindikasikan apakah kota dialog font mengizinkan user untuk mengatur efek (strikeout, underline)

MIKROSKIL

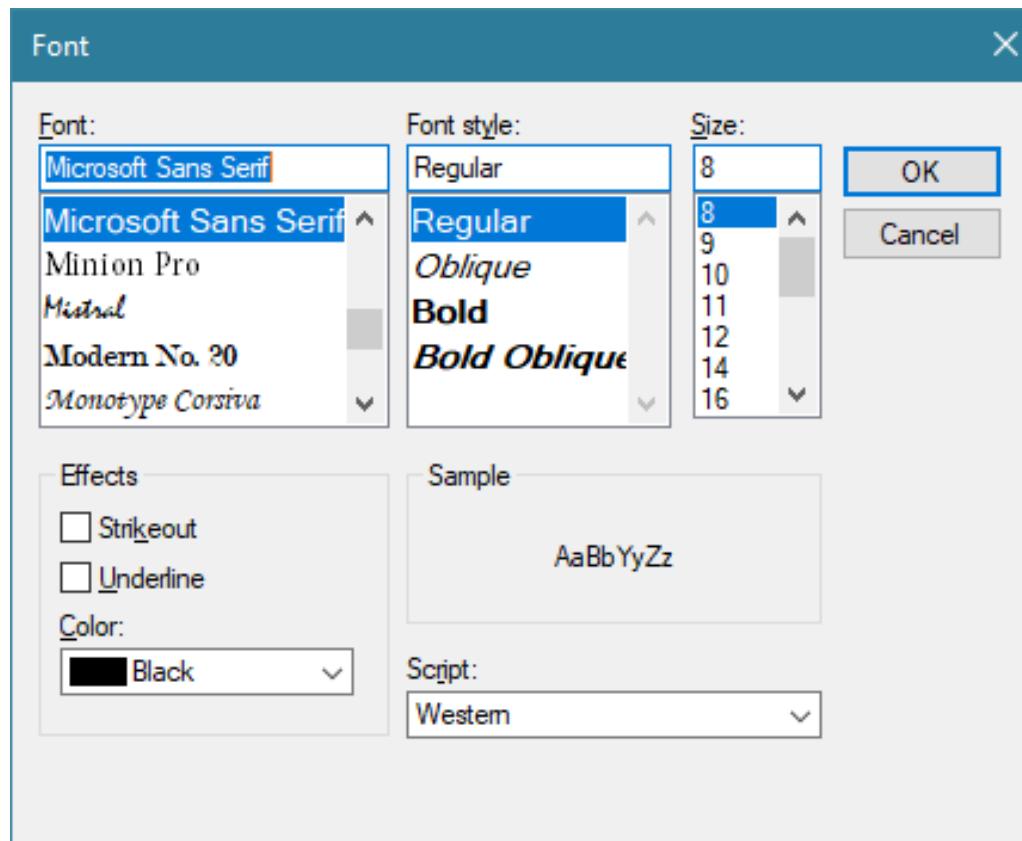
FontDialog (Cont.)

- Method umum pada FontDialog:

Method	Keterangan
ShowDialog	Menampilkan kotak dialog, mengembalikan nilai yang mengindikasikan button mana yang dipilih oleh user: OK atau Cancel.

- Untuk mengetahui button mana yang dipilih oleh user dapat menggunakan konstanta DialogResult.

FontDialog (Cont.)



ColorDialog

- ColorDialog mengizinkan user untuk memilih warna dari palet warna yang tersedia ataupun menambahkan warna custom ke palet tersebut.
- Warna yang terpilih dapat digunakan untuk mengatur property kontrol.
- Sebagai catatan, kontrol ColorDialog tidak memiliki interface (antarmuka) yang bisa dilihat pada saat mendesain form. Setelah user memasukkan kontrol ColorDialog ke dalam form, kontrol ColorDialog akan tampil di tray area.

ColorDialog

- Property umum pada ColorDialog:

Property	Keterangan
Name	Mengatur nama colordialog (ColorDialog umumnya dinamai dengan dlgColor)
AllowFullOpen	Mengatur nilai yang mengindikasikan apakah user dapat menggunakan color dialog box untuk menentukan custom color
AnyColor	Mengatur nilai yang mengindikasikan apakah kotak dialog menampilkan seluruh warna yang tersedia pada kumpulan warna dasar
Color	Menentukan warna yang terpilih
CustomColors	Mengatur kumpulan custom color yang tampil pada kotak dialog

ColorDialog (Cont.)

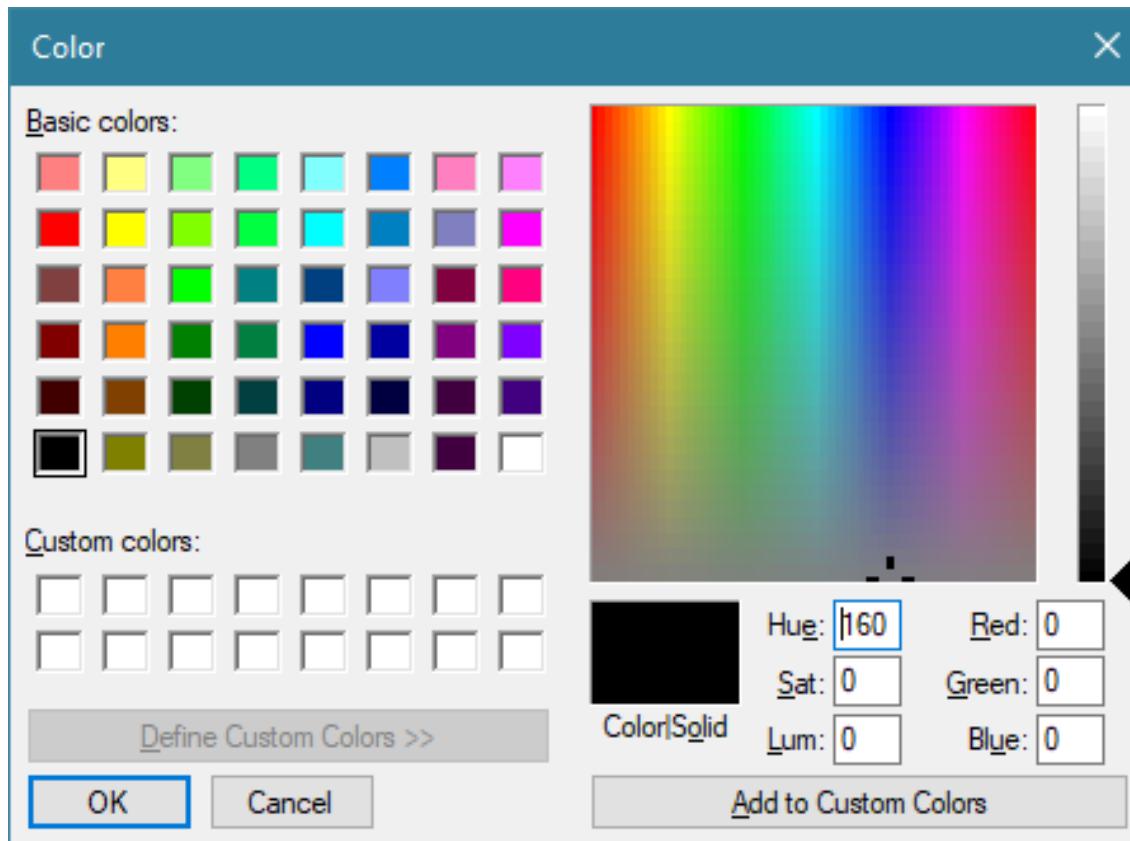
Property	Keterangan
FullOpen	Mengatur nilai yang mengindikasikan apakah kontrol digunakan untuk menambahkan custom color ketika kotak dialog dibuka
SolidColorOnly	Mengatur nilai yang mengindikasikan apakah kotak dialog akan membatasi user untuk memilih warna solid saja.

- Method umum pada ColorDialog:

Method	Keterangan
ShowDialog	Menampilkan kotak dialog, mengembalikan nilai yang mengindikasikan button mana yang dipilih oleh user: OK atau Cancel.

- Untuk mengetahui button mana yang dipilih oleh user dapat menggunakan konstanta DialogResult.

ColorDialog (Cont.)



PrintDialog

- Mengizinkan user untuk memilih printer yang ingin digunakan, memilih orientasi halaman, rentang halaman yang ingin diprint, dan jumlah rangkap yang ingin diprint.
- Sebagai catatan, kontrol PrintDialog tidak memiliki interface (antarmuka) yang bisa dilihat pada saat mendesain form. Setelah user memasukkan kontrol PrintDialog ke dalam form, kontrol PrintDialog akan tampil di tray area.
- Property umum pada PrintDialog:

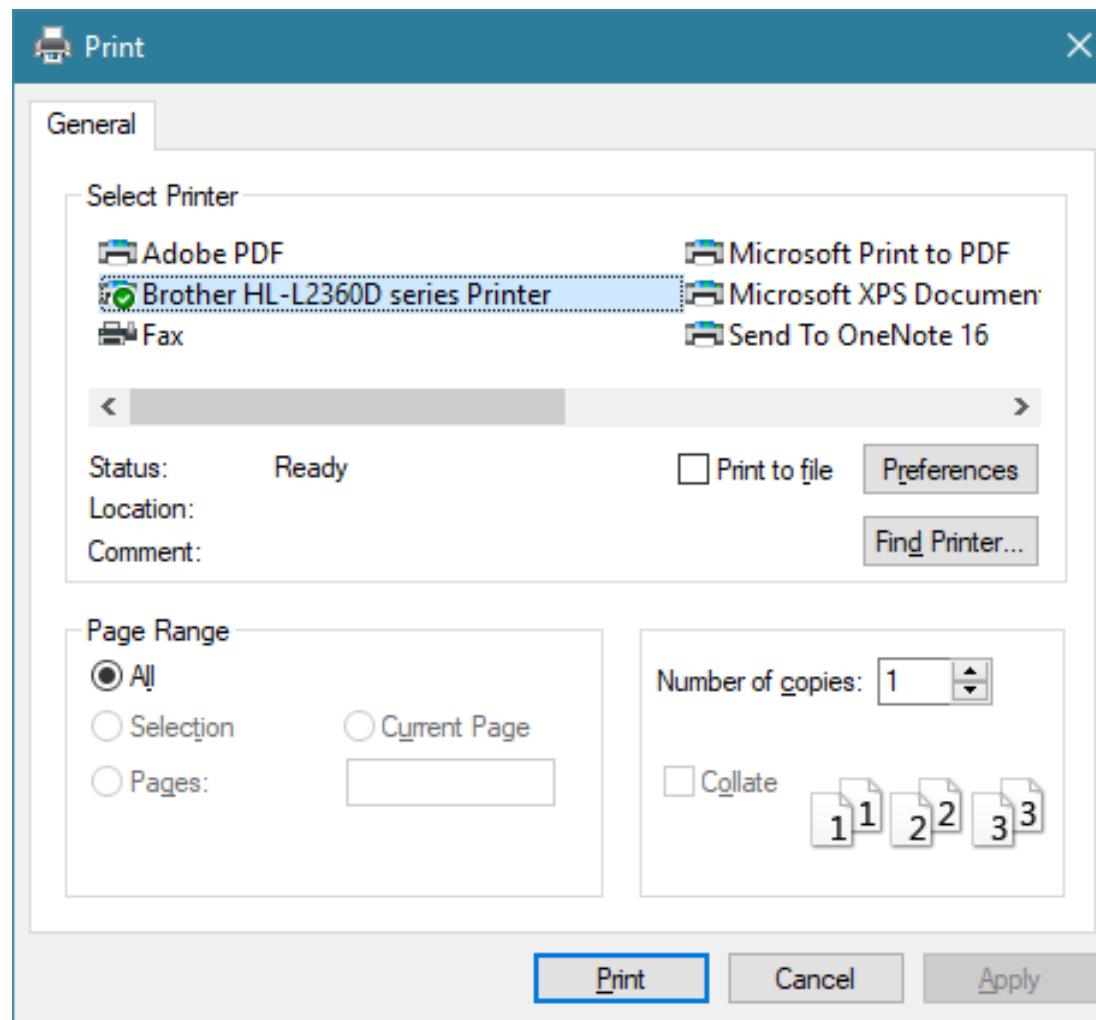
Property	Keterangan
Name	Mengatur nama printdialog (PrintDialog umumnya dinamai dengan dlgPrint)
AllowPrintToFile	Mengatur nilai yang mengindikasikan apakah checkbox print to file aktif atau tidak
AllowSelection	Mengatur nilai yang mengindikasikan apakah opsi From... To... Diaktifkan

PrintDialog (Cont.)

Property	Keterangan
AllowSomePages	Mengatur nilai yang mengindikasikan apakah tombol Pages option aktif atau tidak
Document	Sumber dokumen yang akan dprint
PrinterSettings	Mengatur PrinterSettings yang dimodifikasi kotak dialog
PrintToFile	Mengatur nilai yang mengindikasikan apakah checkbox print to file terpilih
• Metode umum pada PrintDialog:	

Method	Keterangan
ShowDialog	Menampilkan kotak dialog, mengembalikan nilai yang mengindikasikan button mana yang dipilih oleh user: OK atau Cancel.

PrintDialog (Cont.)



PrintPreviewDialog

- Mengizinkan user untuk melihat output print dalam bentuk mode preview dimana dapat melihat keseluruhan halaman, format halaman, serta zoom in dan zoom out dari halaman tersebut.
- Dokumen preview tersebut juga bisa langsung diprint melalui kontrol ini.
- Tools ini merupakan tools yang berguna bagi programmer ketika mengembangkan rutinitas print.
- Sebagai catatan, kontrol PrintPreviewDialog tidak memiliki interface (antarmuka) yang bisa dilihat pada saat mendesain form. Setelah user memasukkan kontrol PrintPreviewDialog ke dalam form, kontrol PrintPreviewDialog akan tampil di tray area.

PrintPreviewDialog (Cont.)

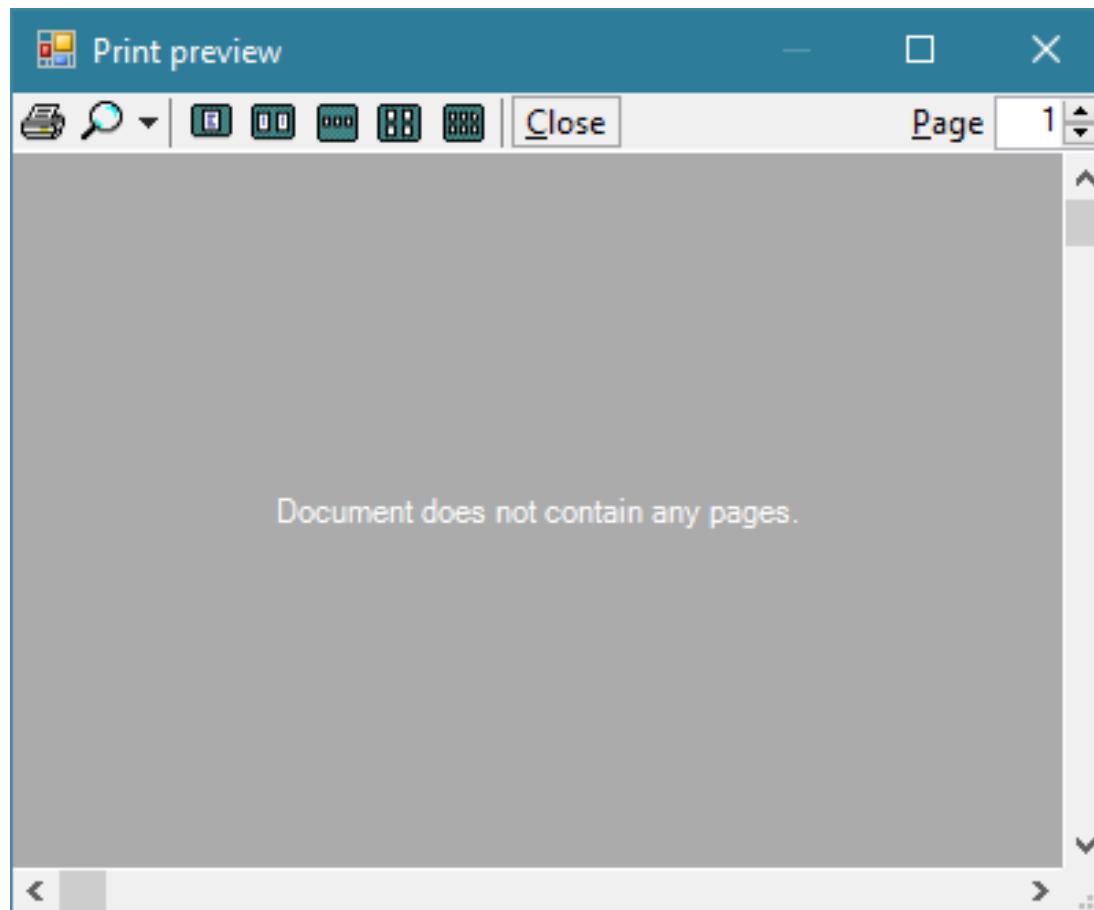
- Property umum pada PrintPreviewDialog:

Property	Keterangan
Name	Mengatur nama printpreviewdialog (PrintPreviewDialog umumnya dinamai dengan dlgPreview)
AcceptButton	Mengatur button mana yang akan terpilih jika user menekan tombol Enter
Document	Mengatur dokumen untuk dipreview
Text	Mengatur teks yang berhubungan dengan kontrol

- Method umum pada PrintPreviewDialog:

Method	Keterangan
ShowDialog	Menampilkan kotak dialog, mengembalikan nilai yang mengindikasikan button mana yang dipilih oleh user: OK atau Cancel.

PrintPreviewDialog (Cont.)



PageSetupDialog

- Mengizinkan user untuk mengatur parameter-parameter yang berhubungan dengan kegiatan print.
- User dapat mengatur bingkai, margin, header dan footer, maupun orientasi portrait atau landscape.
- Sebagai catatan, kontrol PageSetupDialog tidak memiliki interface (antarmuka) yang bisa dilihat pada saat mendesain form. Setelah user memasukkan kontrol PageSetupDialog ke dalam form, kontrol PageSetupDialog akan tampil di tray area.
- Property umum pada PageSetupDialog:

Property	Keterangan
Name	Mengatur nama pagesetupdialog (PageSetupDialog umumnya dinamai dengan dlgSetup)
AllowMargins	Mengatur nilai yang mengindikasikan apakah section margin pada kotak dialog aktif atau tidak

PageSetupDialog (Cont.)

Property	Keterangan
AllowOrientation	Mengatur nilai yang mengindikasikan apakah section orientation (portrait atau landscape) pada kotak dialog aktif atau tidak
AllowPaper	Mengatur nilai yang mengindikasikan apakah section paper (paper size dan paper source) pada kotak dialog aktif atau tidak
AllowPrinter	Mengatur nilai yang mengindikasikan apakah tombol printer aktif atau tidak
Document	Mengatur dokumen yang ingin diatur halamannya
MinMargins	Mengatur nilai minimum margin yang dapat dipilih oleh user dalam satuan seperseratus inci
PageSettings	Mengatur page settings yang dimodifikasi kotak dialog
PrinterSettings	Mengatur printer settings yang dimodifikasi kotak dialog

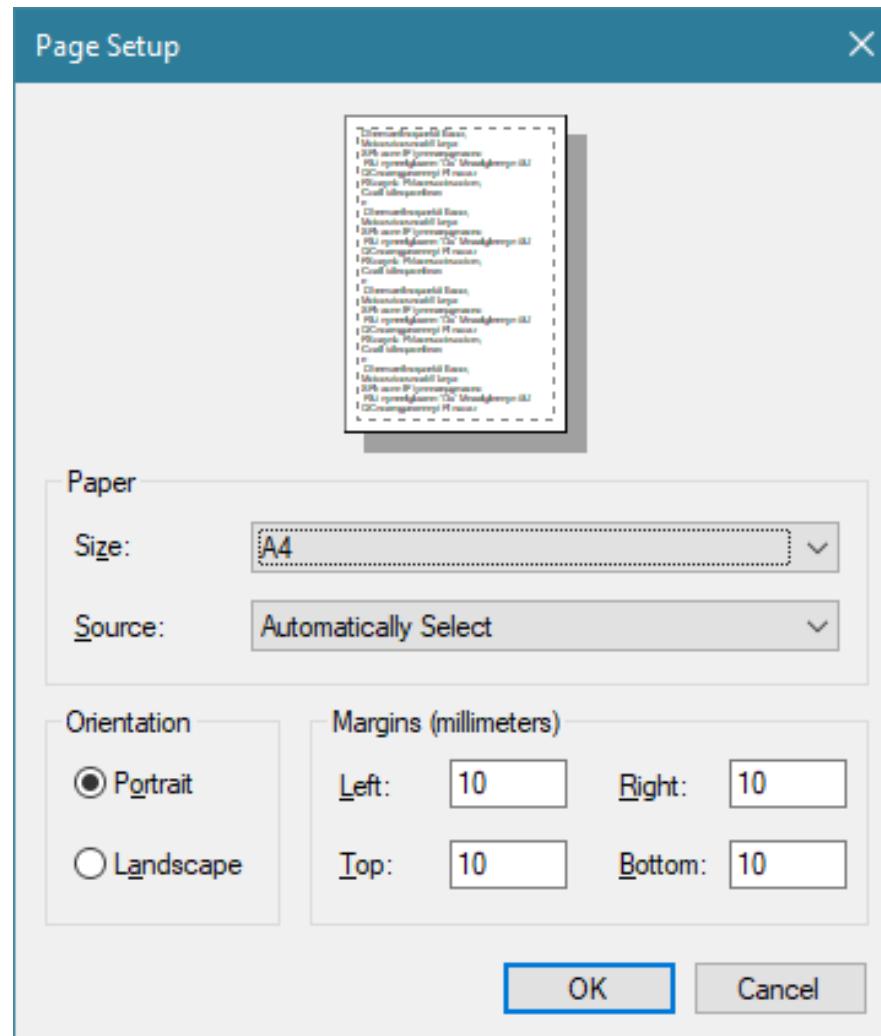
PageSetupDialog (Cont.)

- Method umum pada PageSetupDialog:

Method	Keterangan
ShowDialog	Menampilkan kotak dialog, mengembalikan nilai yang mengindikasikan button mana yang dipilih oleh user: OK atau Cancel.



PageSetupDialog (Cont.)



Lesson Summary

Sekarang mahasiswa mampu:

- Memahami karakteristik dan penggunaan dari kontrol DialogBox

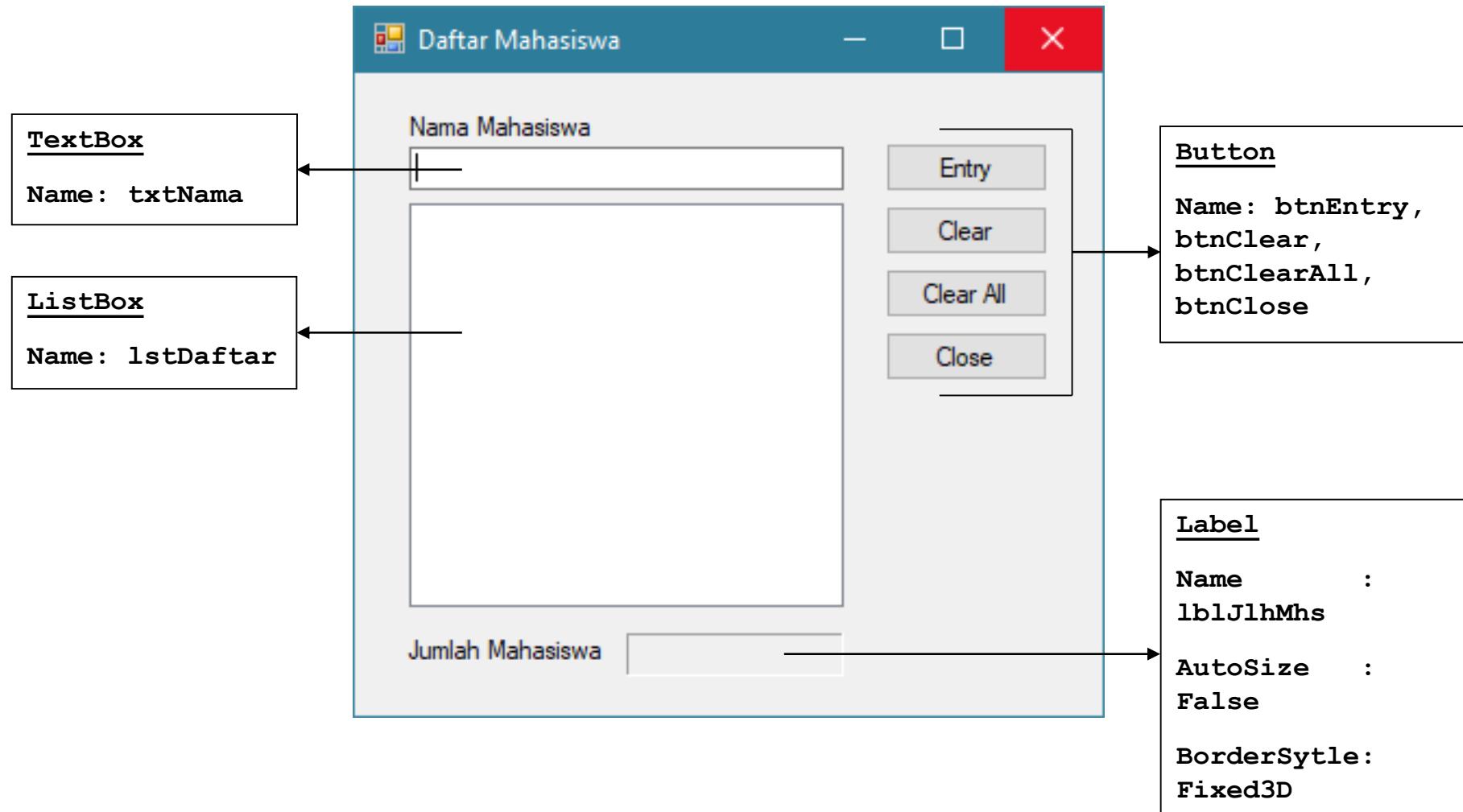




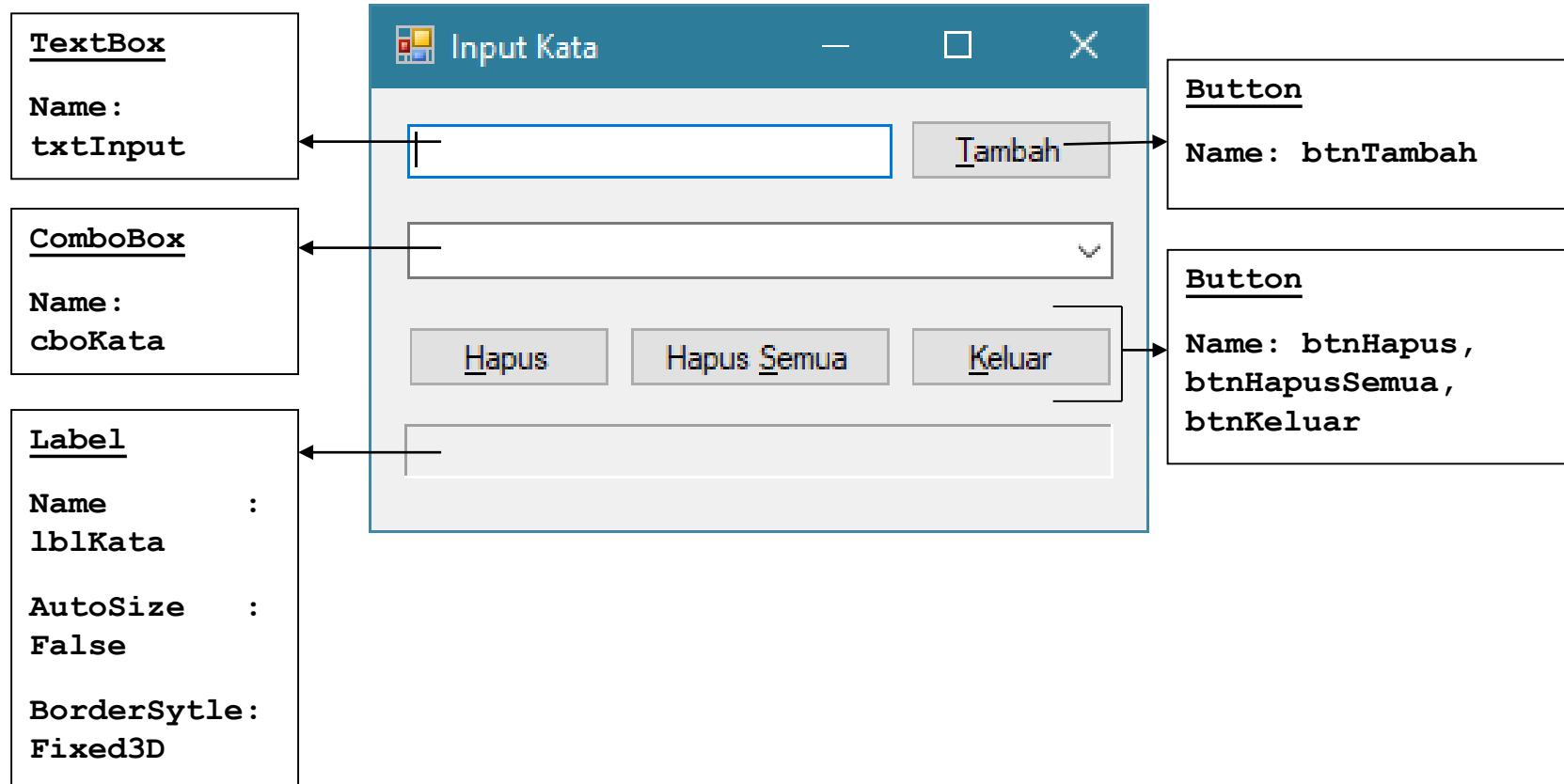
Latihan

MIKROSKIL

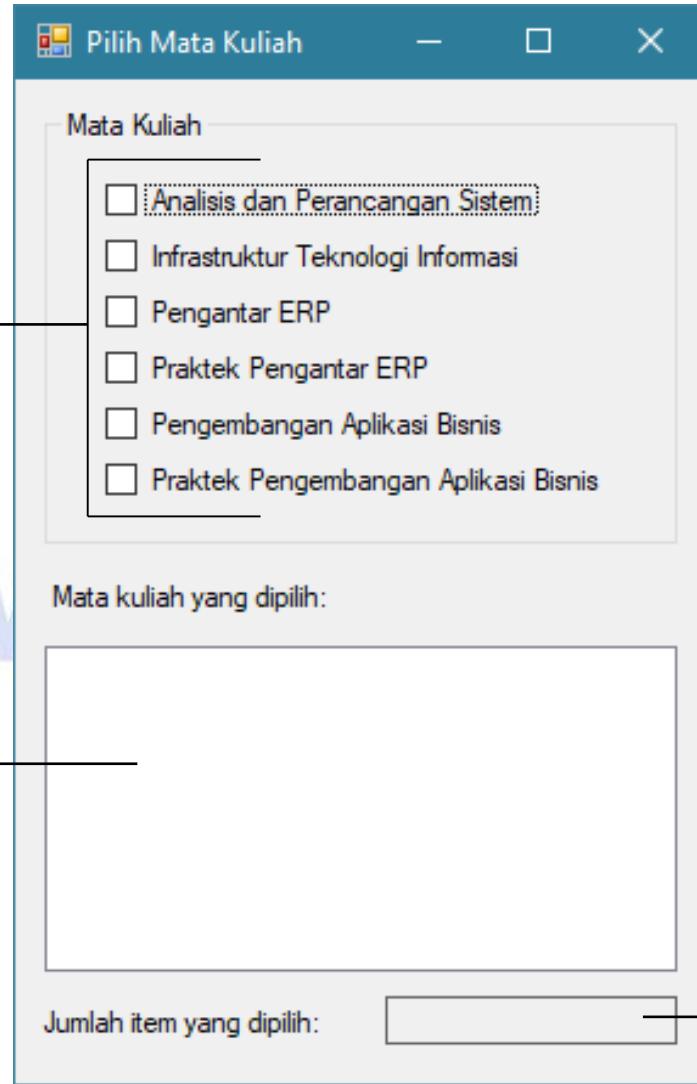
Latihan 1



Latihan 2



Latihan 3



CheckBox

Name: chkAPSI,
chkITI, chkERP,
chkPrakERP,
chkPAB,
chkPrakPAB

ListBox

Name: lstMataKuliah

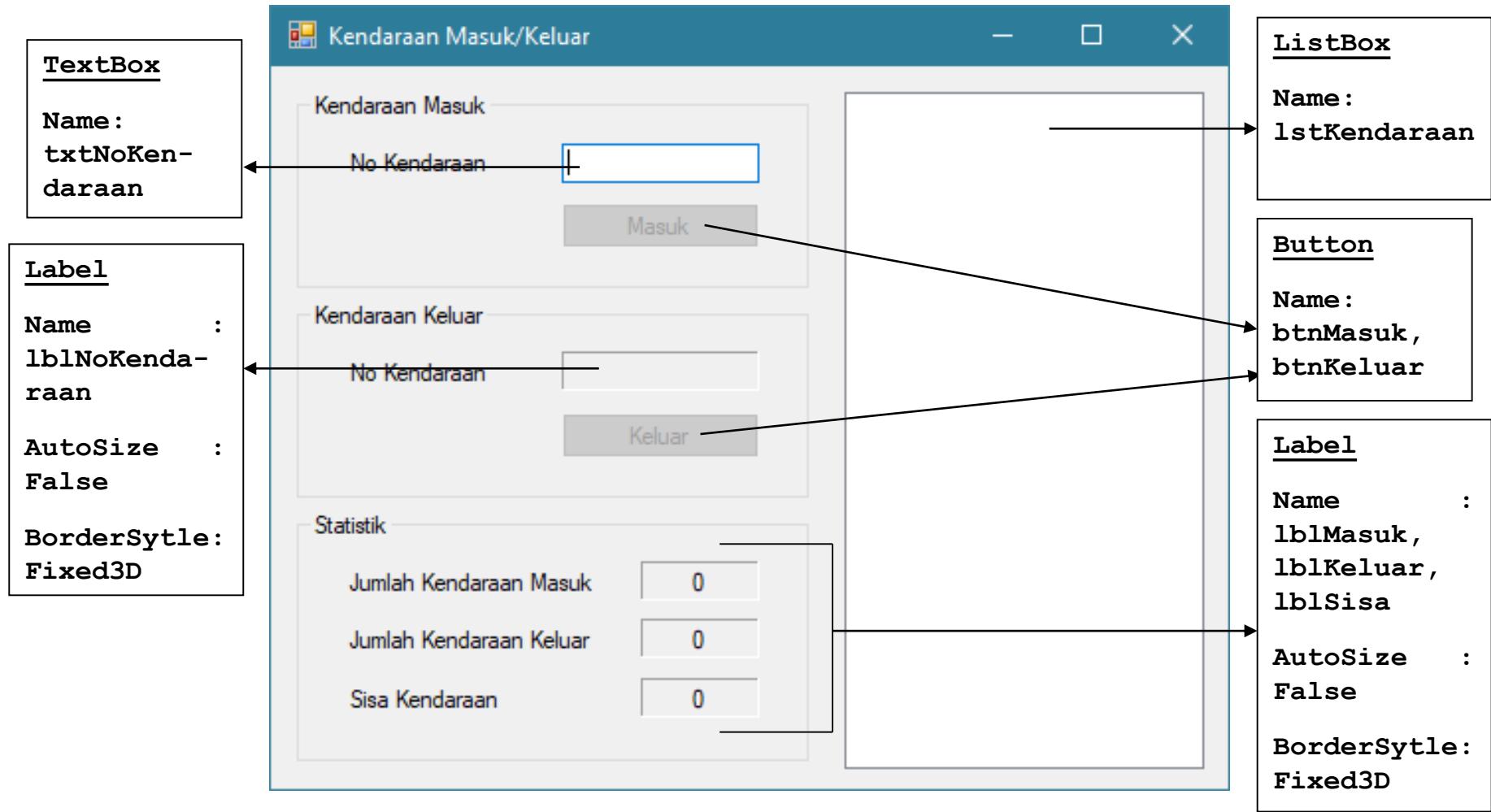
Label

Name :
lblJlhItem

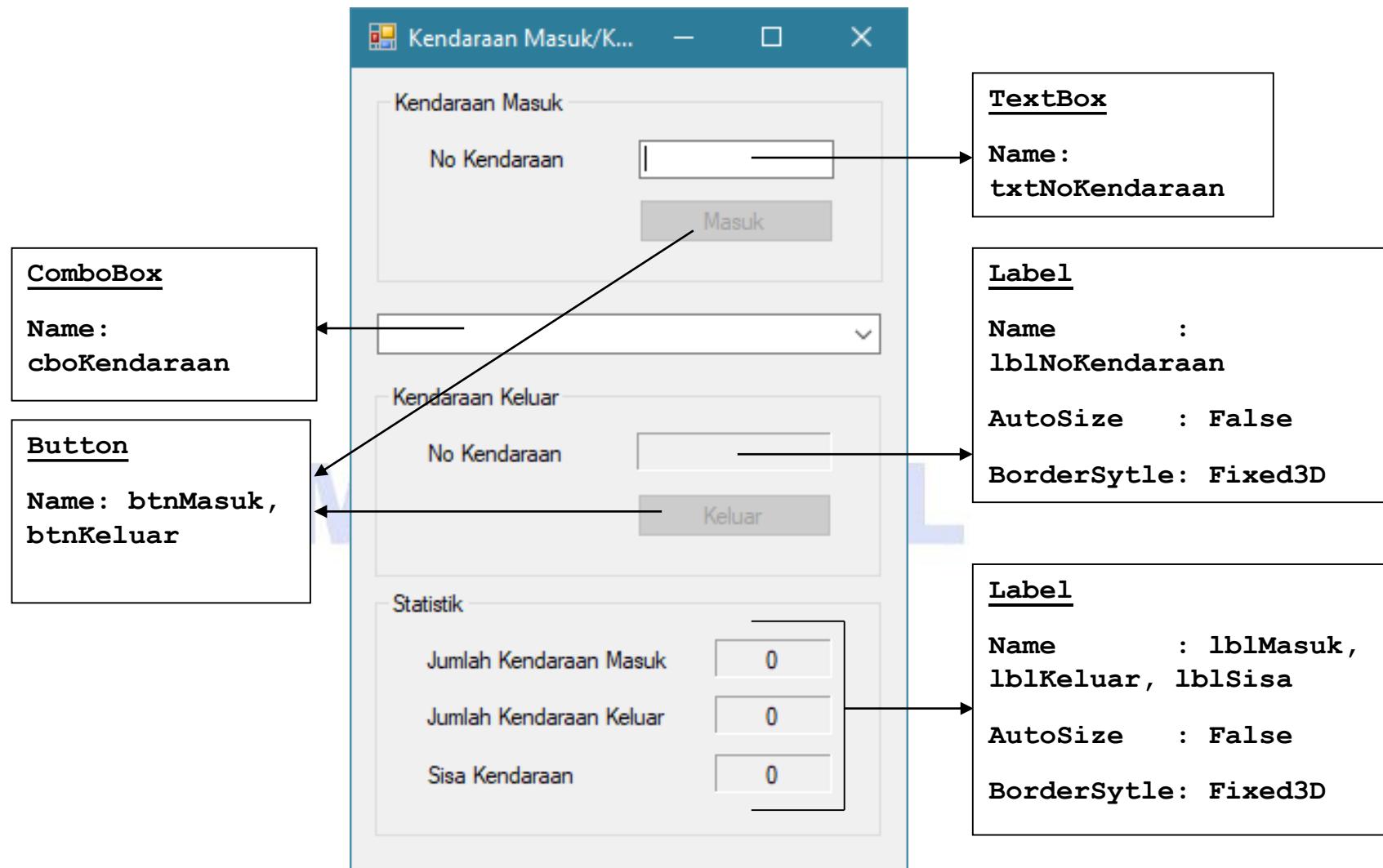
AutoSize : False

BorderStyle:
FixedSingle

Latihan 4



Latihan 5



Latihan 6

Penerbangan

Tiket Medan - Jakarta

Kelas

Bisnis Ekonomi

Penumpang

Nama

No KTP

Maskapai

Harga Tiket Rp0

Menu

Harga Menu Rp0

[Pesan](#)

[Kosong](#)

[Keluar](#)

Latihan 6 (Cont.)

Kelas	Maskapai	Harga Tiket	Menu	Harga Menu
Bisnis	Batik Air	Rp1.500.000	Opor Ayam	Rp70.000
			Ikan Asam Manis	Rp75.000
	Garuda Indonesia	Rp2.000.000	Steak Udang	Rp85.000
			Sphagetti	Rp80.000
Ekonomi	Batik Air	Rp1.375.000	Bihun Goreng	Rp40.000
			Kwetiau Goreng	Rp45.000
	Garuda Indonesia	Rp1.750.000	Nasi Goreng	Rp55.000
			Mie Goreng	Rp50.000

Unit Summary

Sekarang mahasiswa mampu:

- Membedakan Object, Property, Method, dan Event
- Mengetahui langkah-langkah pengembangan suatu aplikasi bisnis
- Memahami dasar-dasar pengoperasian Microsoft Visual Studio 2019
- Memahami jenis-jenis kontrol dan cara penamaan pada sebuah kontrol
- Memahami karakteristik dan penggunaan dari form
- Memahami karakteristik dan penggunaan dari kontrol Button, Label dan TextBox
- Memahami karakteristik dan penggunaan dari kontrol GroupBox, Panel, CheckBox, dan RadioButton
- Memahami karakteristik dan penggunaan dari kontrol ListBox dan ComboBox
- Memahami karakteristik dan penggunaan dari kontrol MessageBox
- Memahami karakteristik dan penggunaan dari kontrol DialogBox

Question & Answers