



MODUL PEMROGRAMAN 2

Disusun oleh :

Radix Rascalía, S.T., M.T.

Shindy Arti, S.Pd., M.Eng.

Eko Abdul Ghofar, A.Md

Indah Cyithia Devi, A.Md

LP2M Politeknik Manufaktur Astra

JAKARTA



MODUL Mata Kuliah

Penulis :

Radix Rascalía, S.T., M.T.

Shindy Arti, S.Pd., M.Eng.

Eko Abdul Ghofar, A.Md

Indah Cyithia Devi, A.Md

Penerbit :

LP2M Politeknik Manufaktur Astra

Jl. Gaya Motor Raya No.8 Sunter II Jakarta 14330

Telepon: (021) 6519555 Fax: (021) 6519821

Email: sekretariat@polman.astra.ac.id

KATA PENGANTAR

Modul praktikum ini dibuat sebagai pedoman mahasiswa dalam melakukan kegiatan praktikum mata kuliah Pemrograman 2 di Politeknik Manufaktur Astra. Modul praktikum ini terdiri dari 12 topik praktikum yang membahas tentang dasar pemrograman C#, OOP, Windows Form, Kontrol Windows Form, koneksi program C# dengan basis data menggunakan .NET Framework, Store Procedure dan LINQ. Topik praktikum lainnya membahas mengenai pembuatan laporan windows form, asinkronous dan WPF, deployment aplikasi windows form dan versioning dan studi kasus. Modul praktikum ini diharapkan dapat membantu mahasiswa/i dalam melaksanakan praktikum dengan lebih baik, terarah, dan terencana. Pada setiap topik telah ditetapkan tujuan pelaksanaan praktikum dan semua kegiatan yang harus dilakukan oleh mahasiswa/i serta teori singkat untuk memperdalam pemahaman mahasiswa/i mengenai materi yang dibahas.

Penyusun menyakini bahwa dalam pembuatan Modul Praktikum mata kuliah Pemrograman 2 ini masih jauh dari sempurna. Oleh karena itu penyusun mengharapkan kritik dan saran yang membangun guna penyempurnaan modul praktikum ini dimasa yang akan datang. Akhir kata, penyusun mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung.

Jakarta, Maret 2022

Tim Penulis

DAFTAR ISI

KATA PENGANTAR	ii
DAFTAR ISI.....	iii
M01 Pengenalan Pemrograman 2	1
1. TUJUAN.....	1
2. DURASI WAKTU	1
3. DASAR TEORI.....	1
4. PERCOBAAN	5
5. LATIHAN	26

M03 Windows Form

1. TUJUAN

CPMK :

Sub-CPMK : Mahasiswa mampu mengimplementasikan windows form

2. DURASI WAKTU

2 pertemuan x 170 menit

3. DASAR TEORI

a. Apakah Windows Form

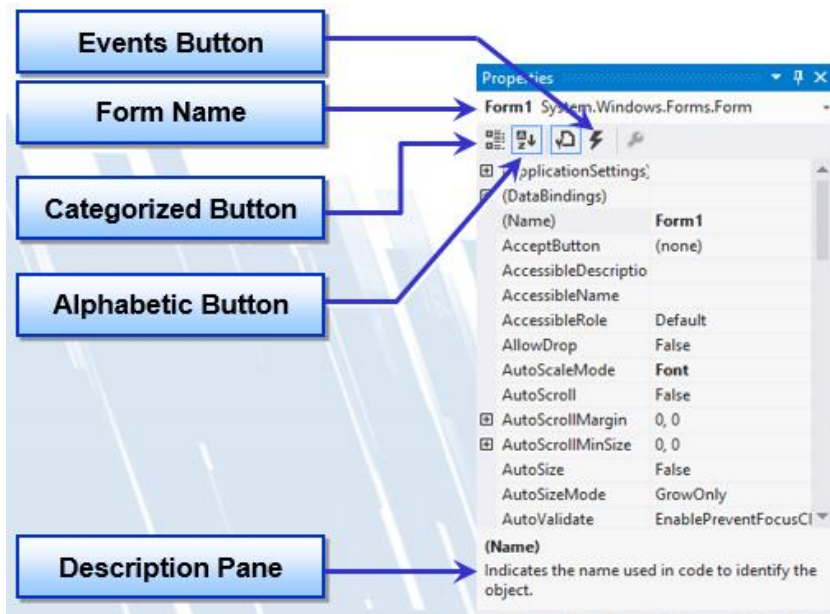
Windows Form Merupakan platform untuk pengembangan aplikasi Microsoft Windows, yang didasarkan pada . NET Framework. Window form juga dapat bertindak sebagai user interface local dalam arsitektur aplikasi multi-tier.

b. Karakteristik/sifat dari windows form

- Deployment : Aplikasi dapat diunduh, dipasang, dan dijalankan di komputer user
- Graphics : Termasuk GDI+ (Graphic Device Interface)
- Responsiveness : Memberikan kecepatan respon stabil untuk aplikasi
- Platform : Membutuhkan .NET Framework berjalan pada komputer klien.
- Model pemrograman : Berdasarkan client-side
- Keamanan : keamanan berbasis kode dan hak akses
- Aplikasi dapat mengakses ke local resource (file system, Windows registry, dll)

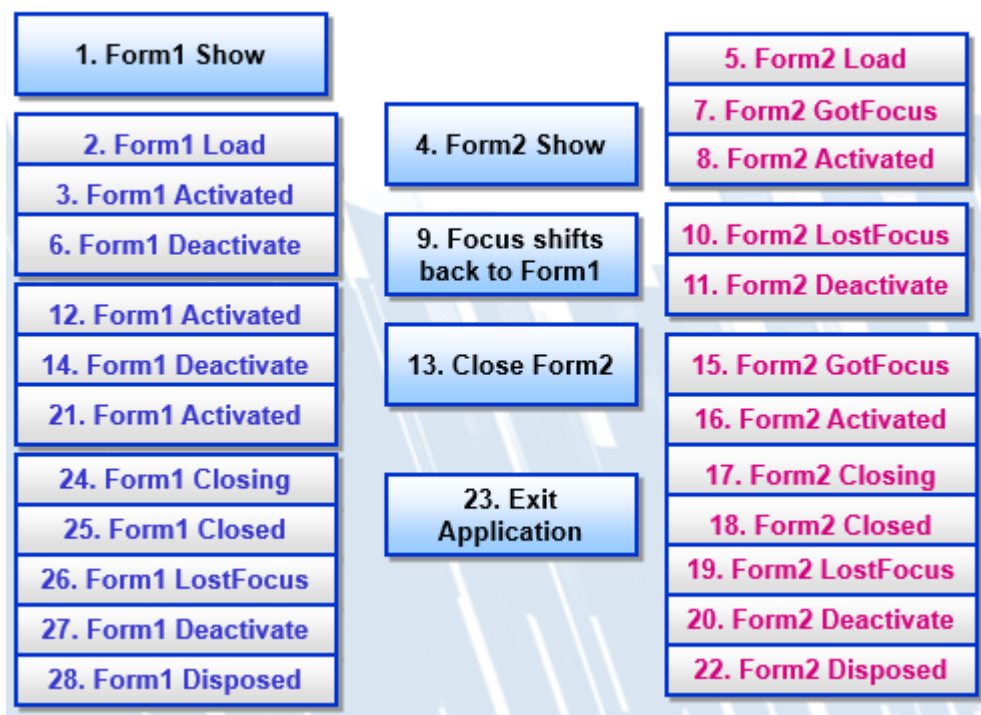
c. Form Properties

Form properties merupakan tools yang dapat digunakan untuk mengatur bagian-bagian yang dibutuhkan dalam merancang form seperti event button, form name, categorized button, alphabetic button, dan yang lainnya. Adapun gambaran untuk form properties pada visual studio dapat dilihat pada gambar berikut.



d. Siklus Hidup Windows Form (Windows Form Life Cycle)

Windows Form Life Cycle (WFLC) merupakan siklus hidup atau cara kerja form saat berinteraksi dengan form yang lainnya. Gambaran WFLC dapat dilihat seperti berikut.



Gambar di atas merupakan contoh untuk mendeskripsikan cara kerja antara 2 form. Gambaran tersebut juga dapat diimplementasikan kepada form yang lebih dari 2 form. Adapun penjelasannya sebagai berikut.

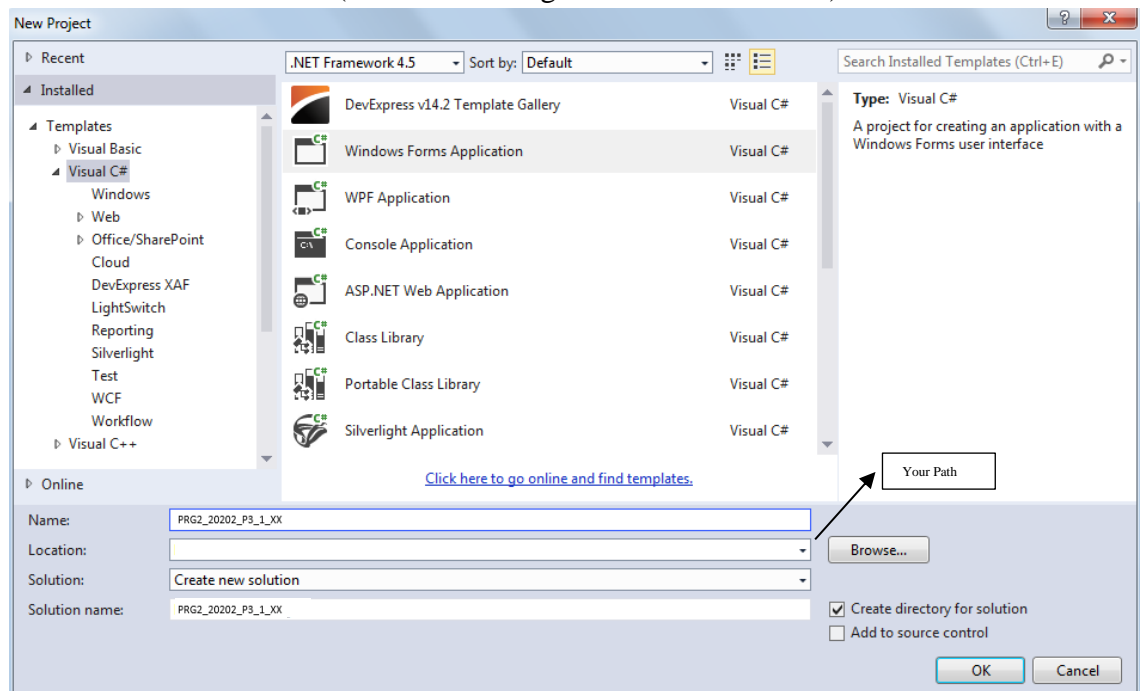
- 1) Form1 ditampilkan saat pertama kali aplikasi dijalankan.
- 2) Form1 melakukan load konten hingga menampilkan data dari form tersebut.
- 3) Form1 memiliki status aktif.
- 4) Kemudian Form2 ditampilkan melalui interaksi dari Form1.
- 5) Form2 melakukan load konten hingga menampilkan data dari form tersebut.
- 6) Form1 memiliki status tidak aktif.
- 7) Form2 melakukan proses fokus untuk beralih dari Form1.
- 8) Form2 memiliki status aktif.
- 9) Kemudian dari Form2 melakukan interaksi untuk menjalankan Form1.
- 10) Fokus kembali ke Form2.
- 11) Form2 memiliki status tidak aktif.
- 12) Form1 memiliki status aktif.
- 13) Mengeluarkan Form2.
- 14) Form1 memiliki status tidak aktif.
- 15) Fokus kembali ke Form2.
- 16) Form2 memiliki status aktif.
- 17) Form2 melakukan proses keluar dari program sehingga statusnya menjadi tidak ada.
- 18) Form2 keluar dari program.
- 19) Form2 sudah tidak memiliki fokus di program.
- 20) Form2 menjadi status tidak aktif.
- 21) Form1 memiliki status aktif.
- 22) Form2 memiliki status disposed (artinya form tersebut sudah tidak ada).
- 23) Proses keluar dari aplikasi.
- 24) Form1 melakukan proses keluar dari program.
- 25) Form1 keluar dari program.
- 26) Form1 sudah tidak memiliki fokus di program.
- 27) Form1 memiliki status tidak aktif.
- 28) Form2 memiliki status disposed (artinya form tersebut sudah tidak ada).

- e. Event dalam WFLC
 - Event adalah suatu aksi yang dapat menanggapi atau 'handle' dalam kode program.
 - Event dapat dihasilkan dari aksi yang dilakukan oleh pengguna, seperti klik mouse / menekan tombol, dengan kode program atau oleh sistem.
 - Setiap form dan control memiliki event yang diatur dalam kode program.
 - Event control : Properties disimbolkan dengan simbol petir.
- f. Event Handler pada form
 - Event Handler Adalah method yang dihasilkan dari suatu event.
 - Jika sebuah event dilakukan dan ada kode di dalam event handler tersebut, maka kode tersebut akan dijalankan sebagai reaksinya.
- g. Multiple Document Interface
 - Menampilkan banyak document pada waktu yang bersamaan.
 - Setiap dokumen ditampilkan di jendela yang sama

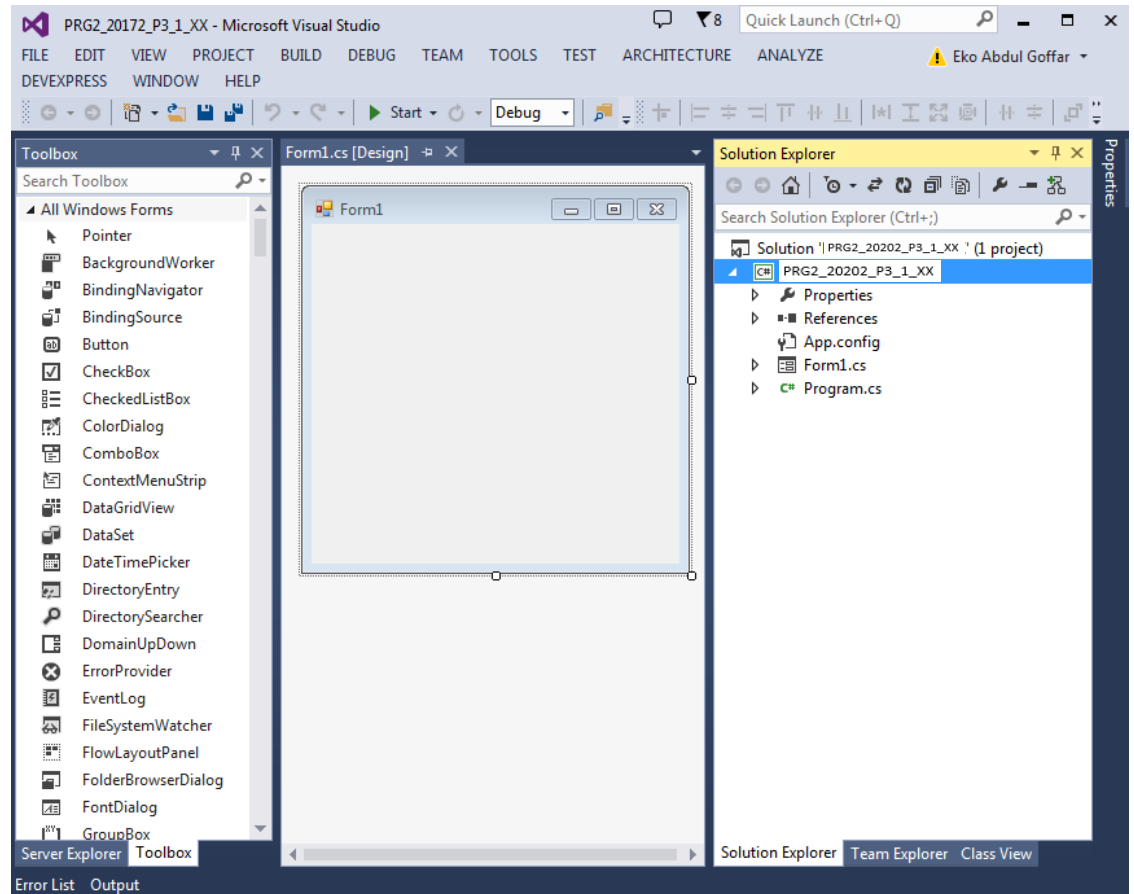
4. PERCOBAAN

Percobaan I: Membuat Aplikasi Window Sederhana

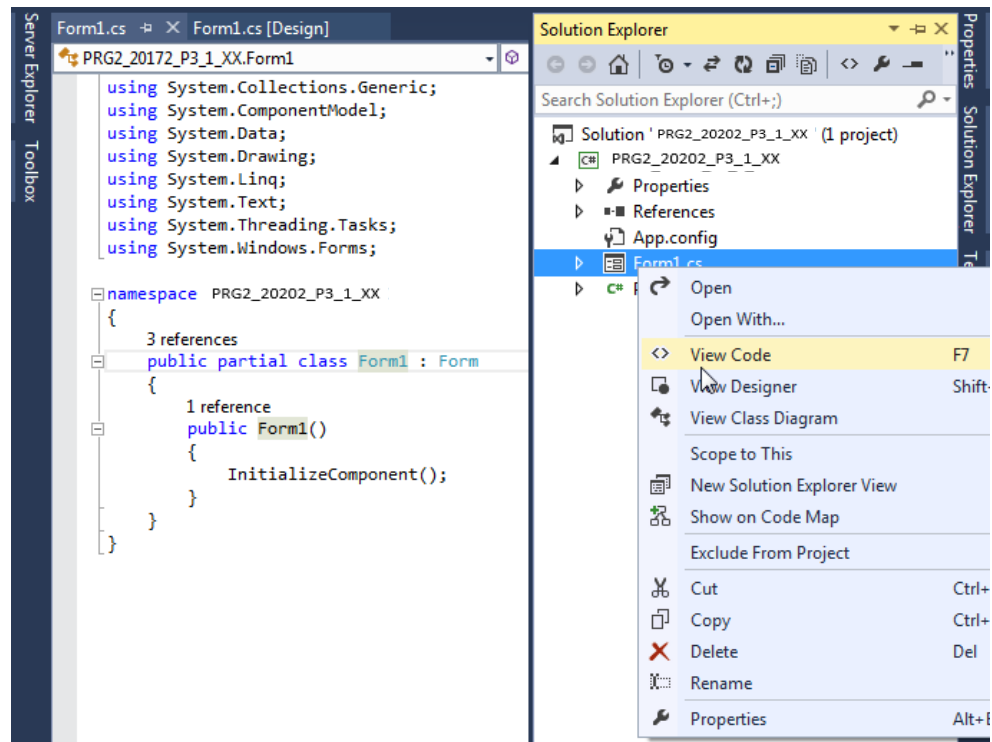
1. Buatlah sebuah project baru dengan cara klik pada File > New > Project. Pada Templates pilihlah Visual C# dan Windows Forms Application, beri nama PRG2_2022_P3_1_XX (XX adalah 2 digit terakhir NIM Anda) → klik OK.



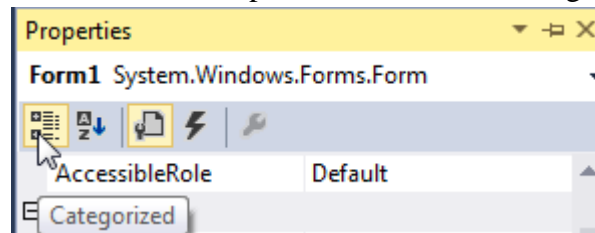
2. Perhatikan lingkungan pemrograman aplikasi windows di Visual Studio (VS), terdiri dari bagian utama untuk merancang form/mengkode program, solution explorer, Toolbox yang berisi koleksi kelas kendali / controls, dan Properties dari obyek kendali yang aktif pada saat tersebut.



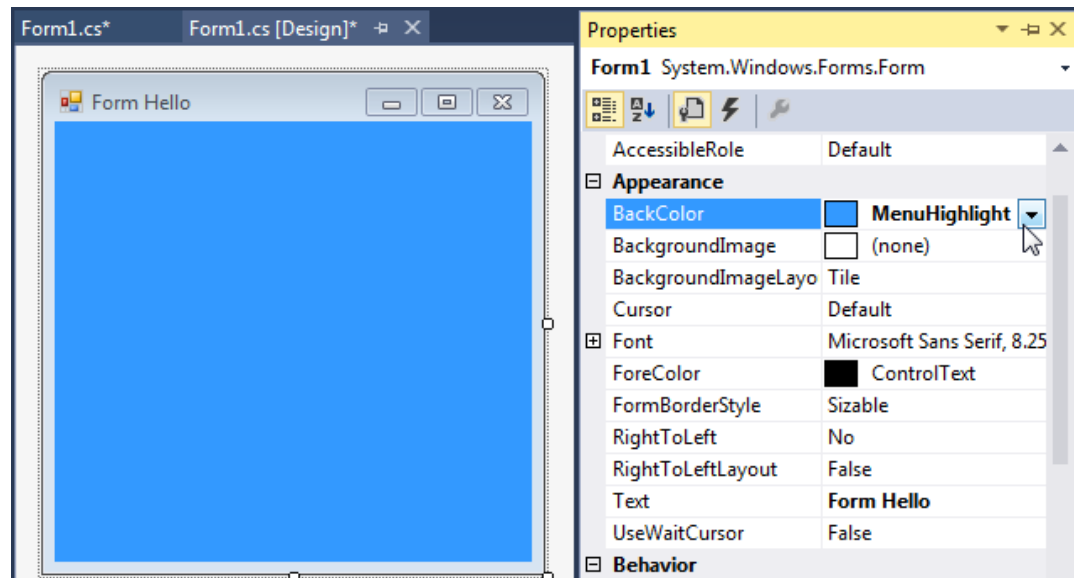
3. Secara otomatis VS akan menyediakan sebuah form yang diberi nama Form1. Pada Solution Explorer, klik kanan Form1.cs, pilih View Code, kemudian perhatikan kodenya.



4. Berdasarkan topic sebelumnya mengenai OOP, menurut Anda Form1 merupakan sebuah kelas atau Obyek? Jelaskan! Pada bagian mana di kode yang menunjukkan hal tersebut?
5. Pada Solution Explorer, klik kanan Form1.cs > View Designer
6. Perhatikan desain dari Form1 tersebut.
7. Pada window Properties, klik pada property Text, ganti Form1 dan isikan "Form Hello", tekan enter.
8. Perhatikan desain dari Form1 kembali.
9. Pada windows Properties, klik tombol Categorized



10. Cari kategori Appearance, klik BackColor, klik tombol panah, pilih tab System, pilih MenuHighlight



11. Ubahlah property size dari Form1 menjadi 300, 175
12. Pada windows Toolbox, All Windows Form, pilih kendali/*control* Label, klik dan tarik dan jatuhkan (*click & drag & drop*)di dalam Form1
13. Perhatikan windows property, pastikan windows tersebut adalah untuk obyek label1. Ubahlah properties dari obyek tersebut sebagai berikut:

Property	Value
(Name)	OutputLabel
BorderStyle	Fixed3D
Font	Trebuchet MS, 10pt, Bold
ForeColor	ActiveCaptionTex
Location	14,30
Size	264, 23
Text	(Hapus text yang ada dan biarkan kosong)
TextAlign	MiddleCenter

14. Pada windows Toolbox, All WindowsForm, pilih kendali/*control* Button, klik dan tarik dan jatuhkan (*click & drag & drop*)di dalam Form1
15. Perhatikan windows property, pastikan windows tersebut adalah untuk obyek button1.
Ubahlah properties dari obyek tersebut sebagai berikut:

Property	Value
(Name)	HelloButton
Location	57, 87
Size	75,25
Text	&Say Hello

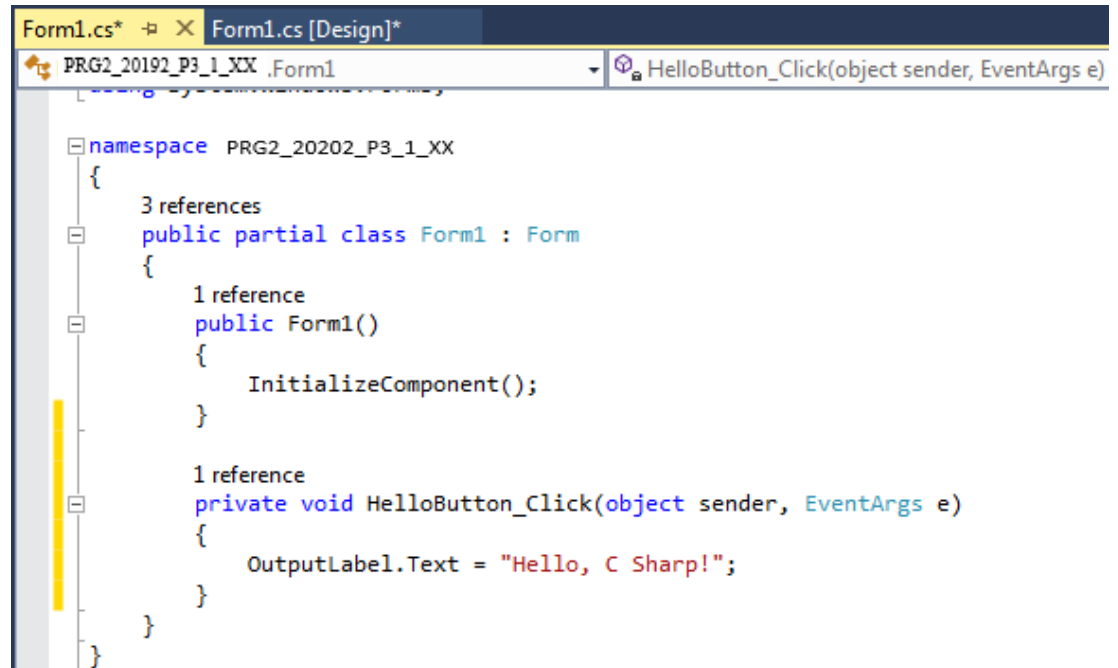
16. Pada windows Toolbox, All Windows Form, dobel klik kendali/control Button
17. Perhatikan windows property, pastikan windows tersebut adalah untuk obyek button1.
Ubahlah properties dari obyek tersebut sebagai berikut:

Property	Value
(Name)	ExitButton
Location	161, 87
Size	75,25
Text	E&xit

18. Untuk menambahkan kode Event Handler (perhatikan kejadian yang mungkin terjadi pada sebuah obyek disebut dengan Event, dan kode yang akan dijalankan untuk event tersebut disebut dengan Event Handler) jika HelloButton diklik, pada windows properties tekanlah toolbar button yang berlambang petir



19. Pastikan window properties sedang menunjukkan bekerja dengan HelloButton
20. Lakukan dobel klik pada event click. Secara otomatis VS akan menghasilkan kode HelloButton_click() event handler
21. Berdasarkan topic OOP sebelumnya, HelloButton_Click() ini merupakan sebuah ...?
22. Dalam event handler tersebut tambahkan kode sehingga berbentuk sbb:

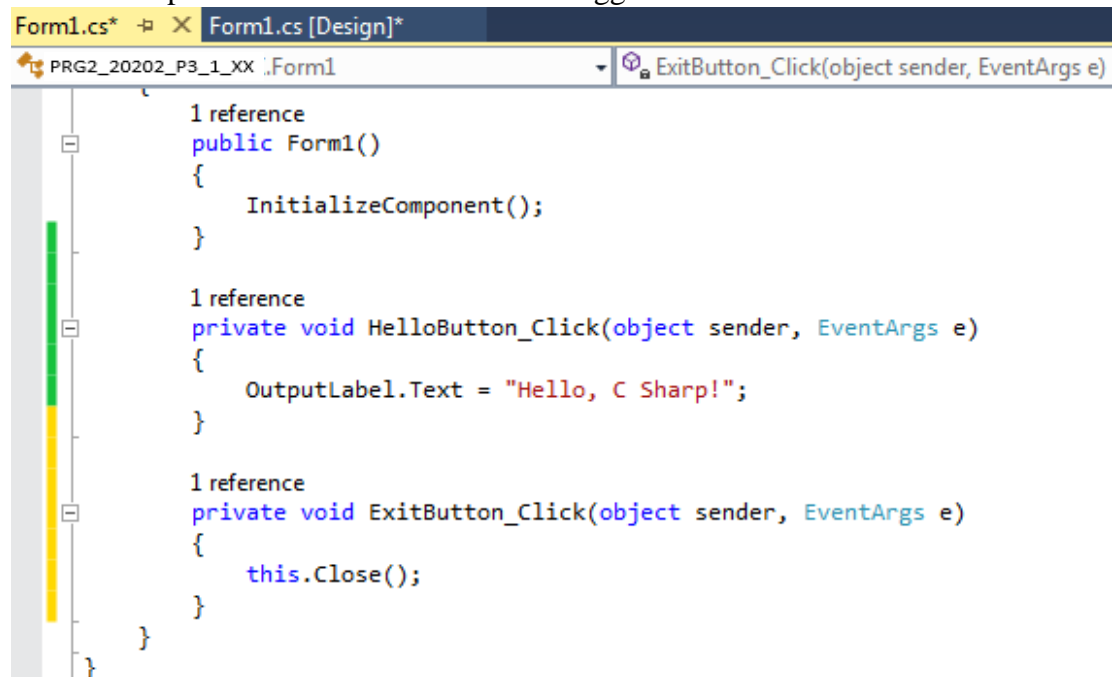


```
Form1.cs*  Form1.cs [Design]*
PRG2_20192_P3_1_XX .Form1
HelloButton_Click(object sender, EventArgs e)

namespace PRG2_20202_P3_1_XX
{
    3 references
    public partial class Form1 : Form
    {
        1 reference
        public Form1()
        {
            InitializeComponent();
        }

        1 reference
        private void HelloButton_Click(object sender, EventArgs e)
        {
            OutputLabel.Text = "Hello, C Sharp!";
        }
    }
}
```

23. Kembali ke desain Form1, lakukan double klik pada ExitButton
24. Maka dengan cara tersebut akan sama menghasilkan event handler ExitButton_Click()
25. Isikan kode pada event handler tersebut sehingga berbentuk sbb:



```
Form1.cs*  Form1.cs [Design]*
PRG2_20202_P3_1_XX .Form1
ExitButton_Click(object sender, EventArgs e)

    1 reference
    public Form1()
    {
        InitializeComponent();
    }

    1 reference
    private void HelloButton_Click(object sender, EventArgs e)
    {
        OutputLabel.Text = "Hello, C Sharp!";
    }

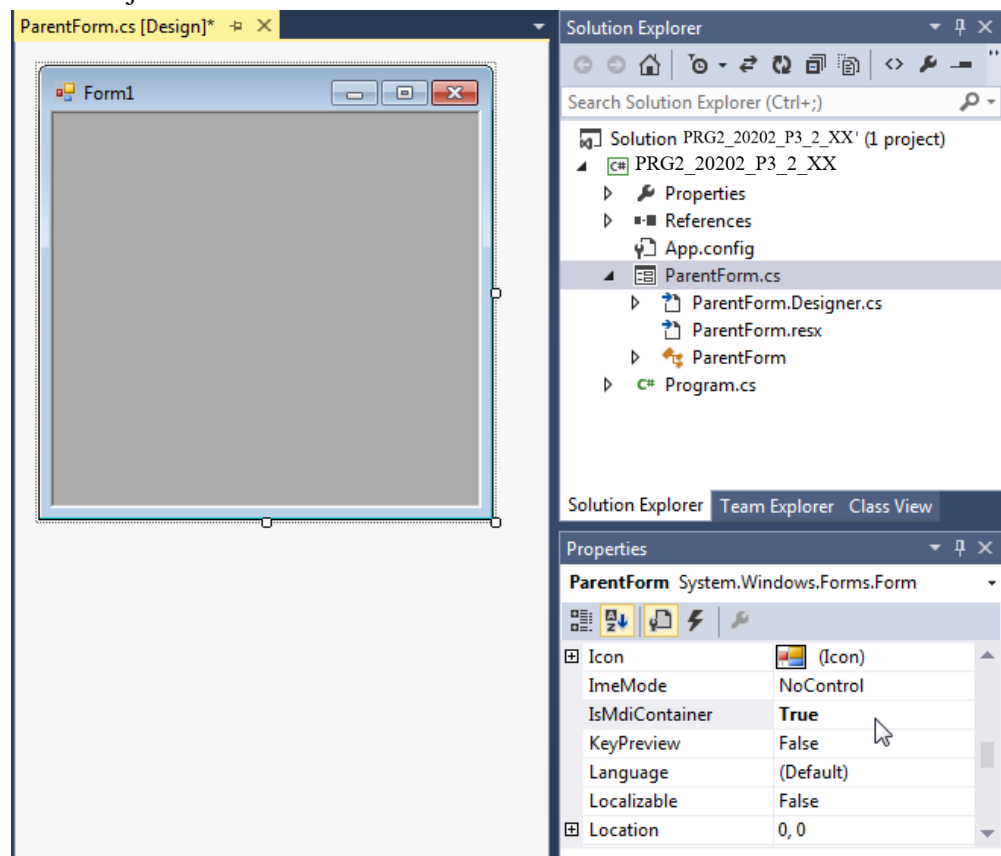
    1 reference
    private void ExitButton_Click(object sender, EventArgs e)
    {
        this.Close();
    }
}
```

26. Pilih menu Build >> Build Solution

27. Tekan F5 untuk menjalankan aplikasi
28. Klik button Say Hello. Perhatikan hasilnya.
29. Tekan Alt pada keyboard. Perhatikan text pada tombol.
30. Tekan Alt + X pada keyboard. Maka Anda akan keluar dari aplikasi.

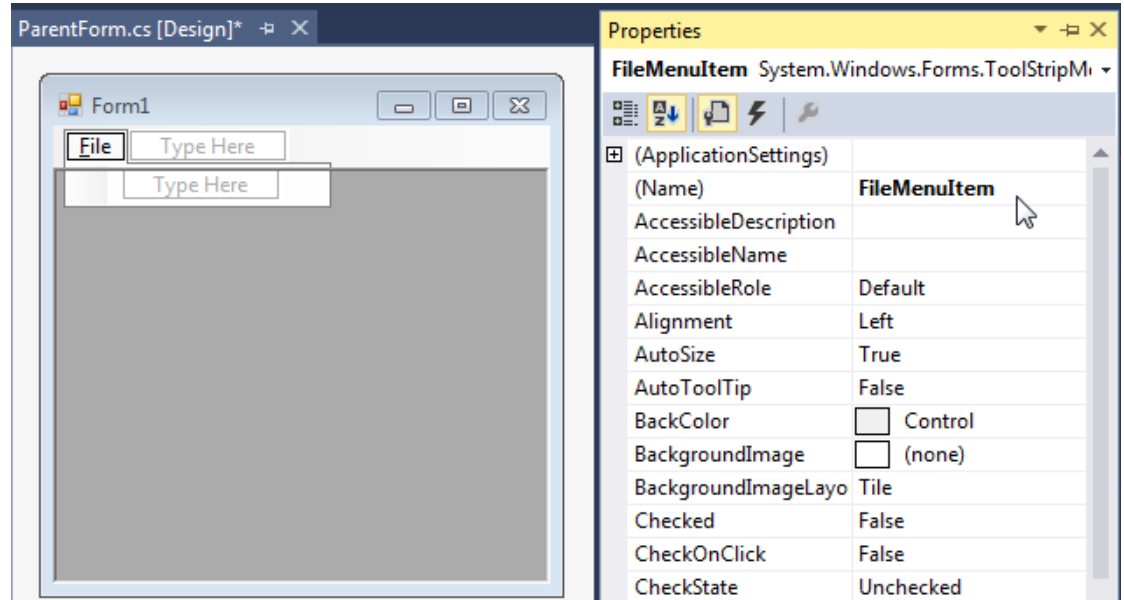
Percobaan II: Membuat aplikasi MDI dan mengatur perletakan kendali

1. Buatlah sebuah project baru dengan nama PRG2_20222_P3_2_XX (XX adalah 2 digit terakhir NIM Anda).
2. Pada Solution Explorer, klik kanan Form1.cs, rename menjadi ParentForm.cs
3. Ubahlah property Text form tersebut menjadi Parent Form
4. Ubahlah property menjadi IsMdiContainer menjadi True
5. Perhatikan di windows properties, jika property yang telah diubah akan memiliki jenis font tebal/*bold*.

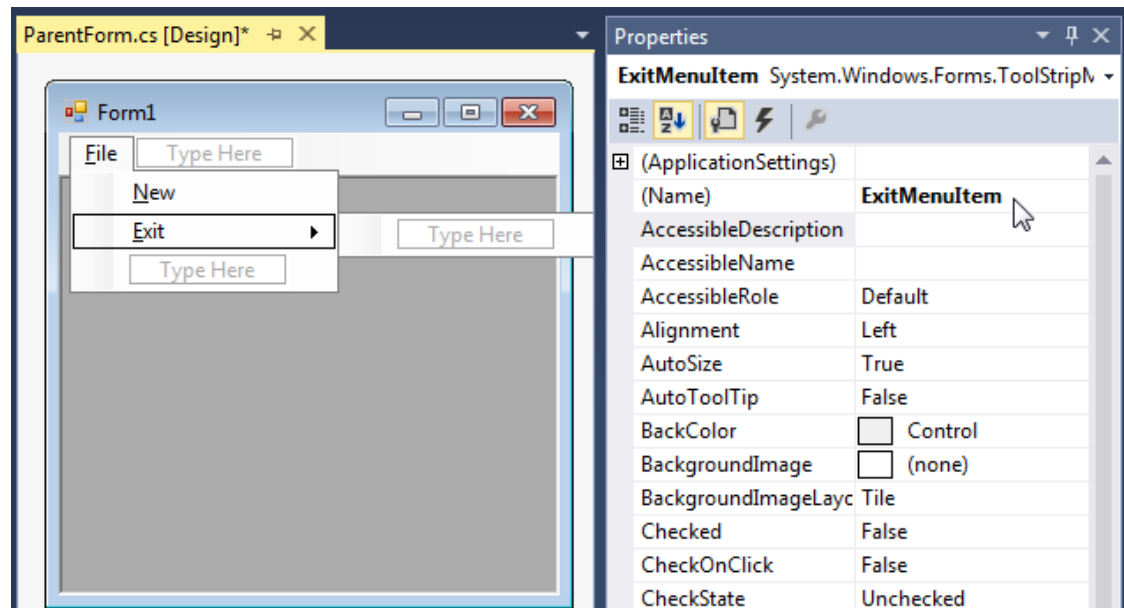


6. Dari Toolbox, Menus & Toolbars, tambahkan MenuStrip ke form tersebut dan beri nama MdiMenu

7. Pada bagian pertama MdiMenu yang bertuliskan type here, ketikkan &File. Ganti namanya menjadi File MenuItem



8. Di bawah menu File, ganti type here menjadi &New. Ganti namanya menjadi NewMenuItem
9. Di bawah menu File, ganti type here menjadi &New. Ganti namanya menjadi ExitMenuItem



10. Dobel klik control ExitMenuItem
11. Tambahkan kode pada event handler sehingga sbb:


```
ParentForm.cs* - ParentForm.cs [Design]*
PRG2_20202_P3_2_XX.ParentForm ExitMenuItem_Click(object sender, EventArgs e)
using System.Windows.Forms;

namespace PRG2_20202_P3_2_XX
{
    3 references
    public partial class ParentForm : Form
    {
        1 reference
        public ParentForm()
        {
            InitializeComponent();
        }

        1 reference
        private void ExitMenuItem_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

12. Pindah kembali ke design view
13. Pada menu di sebelah menu File, ganti type here menjadi &Windows. Ganti namanya menjadi WindowMenuItem.
14. Pada menu di bawah Window, ganti type here menjadi &Cascade. Ganti namanya menjadi WindowCascadeMenuItem.
15. Pada menu di bawah Cascade, ganti type here menjadi &Tile. Ganti namanya menjadi WindowTileMenuItem
16. Dobel klik pada menu Cascade. Tambahkan kode pada event handler.
17. Dobel klik pada menu Tile. Tambahkan kode pada event handler.
18. Kode akan berbentuk sbb:

```

ParentForm.cs
PRG2_20202_P3_2_XX,ParentForm
WindowTileMenuItem_Click(object sender, EventArgs e)

}

1 reference
private void WindowCascadeMenuItem_Click(object sender, EventArgs e)
{
    this.LayoutMdi(System.Windows.Forms.MdiLayout.Cascade);
}

1 reference
private void WindowTileMenuItem_Click(object sender, EventArgs e)
{
    this.LayoutMdi(System.Windows.Forms.MdiLayout.TileHorizontal);
}

}

```

19. Membuat Child Form sebagai berikut:
20. Pada VS, klik menu Project, Add Windows Form
21. Ganti property text form tersebut menjadi Child Form
22. Dari Toolbox, tambahkan kendali sebagai berikut:
23. Sebuah label dengan properties:

Property	Value
(Name)	OutputLabel
BorderStyle	Fixed3D
Font	Trebuchet MS, 10pt, Bold
ForeColor	ActiveCaption
Location	0,16
Size	292, 32
Text	(kosongkan)
TextAlign	MiddleCenter

24. Sebuah Button dengan properties:

Property	Value
(Name)	ExitButton
Location	32,56
Size	75,23
Text	E&xit

25. Sebuah button dengan properties:

Property	Value
(Name)	DisplayButton
Location	176,88
Size	75,23
Text	&Display

26. Sebuah Groupbox dengan properties:

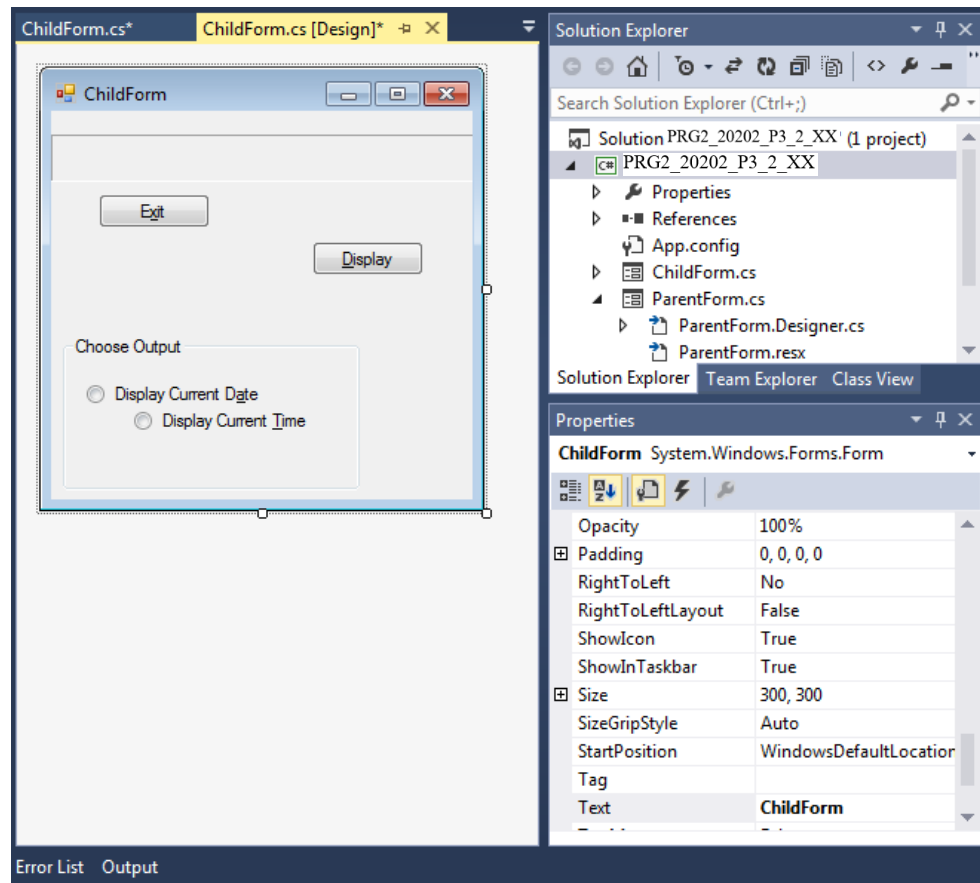
Property	Value
(Name)	OutputGroupBox
Location	8,152
Size	200,104
TabIndex	2
Text	Choose Output

27. Sebuah RadioButton dengan properties:

Property	Value
(Name)	DateOption
Location	16,30
Size	168,24
TabIndex	1
Text	Display Current D&ate

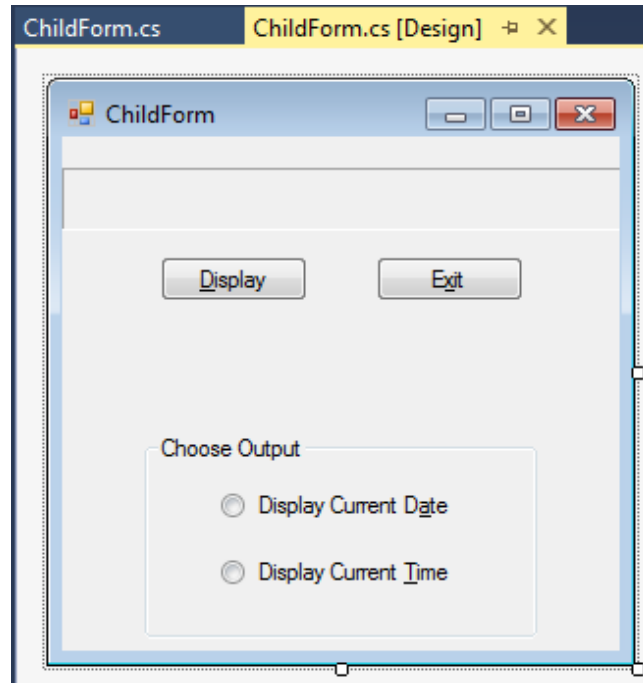
28. Sebuah RadioButton dengan properties:

Property	Value
(Name)	TimeOption
Location	48,48
Size	168,24
TabIndex	0
Text	Display Current &Time

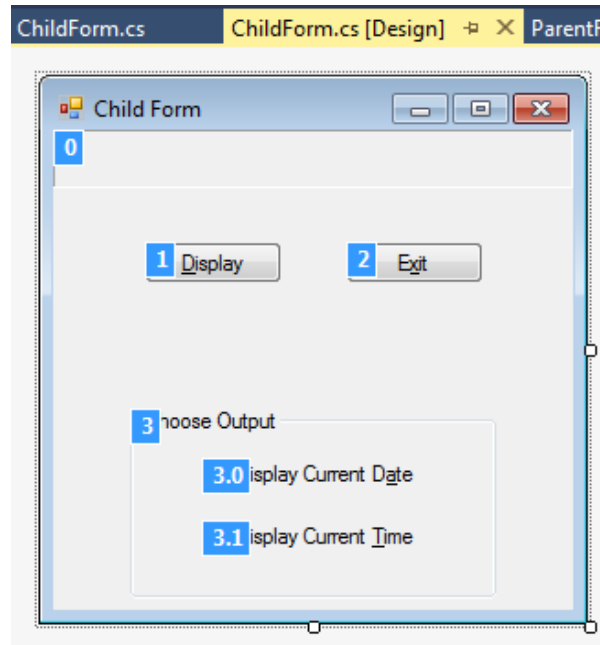


29. Mengatur tampilan kendali/control sebagai berikut:
30. Geserlah button di form, sehingga button Display berada di sebelah kiributton Exit.
31. Klik button Exit, tekan dan tahan keyboard CTRL, dan klik button Display. Kedua button akan terpilih dalam kondisi terpilih.
32. Pada VS, pilih menu Format, klik Align, dan klik Tops. Button Exit dan Display akan sejajar di bagian atasnya.
33. Dalam kondisi kedua button terpilih, akan muncul menu Format, klik Center in Form, dan klik Horizontally.
34. Klik Choose Output GroupBox
35. Pada menu Format, klik Center in Form, dan klik Horizontally
36. Klik Display Current Time, tekan dan tahan CTRL, klik Display Current Date.
37. Pada menu Format, klik Align, klik Lefts
38. Dalam kondisi kedua radio button terpilih, pada menu Format, klik vertical spacing, dan klik Increase. Lakukan dua kali.

39. Dalam kondisi kedua radio button terpilih, pada menu Format, klik Center in Form, klik Horizontally.
40. Dalam kondisi kedua radio button terpilih, pada menu Format, klik Center in Form, klik Vertically.



41. Mengatur urutan tab (Tab Order)
42. Pada menu View, klik Tab Order
43. Ubah urutan Tab dengan mengklik setiap kendali berdasarkan urutan: Label, button Display, button Exit, GroupBox Choose Output, Display Current Date, dan Display Current Time. Hasilnya sbb:



44. Pada menu View, klik Tab Order

45. Melakukan jangkar (anchor) dan dock pada kendali

46. Klik kendali Label, dan set property Dock menjadi Top

47. Klik kendali Exit, dan set property Anchor menjadi Top, Right

48. Klik kendali GroupBox, dan set property Anchor menjadi Bottom, Left, Right

49. Klik kendali Display Current Date dan set Anchor menjadi Top

50. Klik kendali Display Current Time dan set Anchor menjadi Bottom

51. Tambahkan sebuah variable outputText bertipe data string, kemudian dobel klik Exit, Display, Display Current Date, dan Display Current Time. Ketikkan kode sehingga sbb:



```
ChildForm.cs X ChildForm.cs [Design] ParentForm.cs
PRG2_20202_P3_2_XX.ChildForm
using System.Windows.Forms;

namespace PRG2_20202_P3_2_XX
{
    2 references
    public partial class ChildForm : Form
    {
        private string outputText = DateTime.Now.ToString("d");
        0 references
        public ChildForm()
        {
            InitializeComponent();
        }

        1 reference
        private void DisplayButton_Click(object sender, EventArgs e)
        {
            OutputLabel.Text = outputText;
        }

        1 reference
        private void ExitButton_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        1 reference
        private void DateOption_CheckedChanged(object sender, EventArgs e)
        {
            outputText = DateTime.Now.ToString("d");
        }

        1 reference
        private void TimeOption_CheckedChanged(object sender, EventArgs e)
        {
            outputText = DateTime.Now.ToString("t");
        }
    }
}
```

52. Menampilkan child form dalam parent form:

53. Lihatlah ParentForm dalam Design View

54. Dobel klik menu item New, dan isikan kode pada event handlernya sehingga sbb:

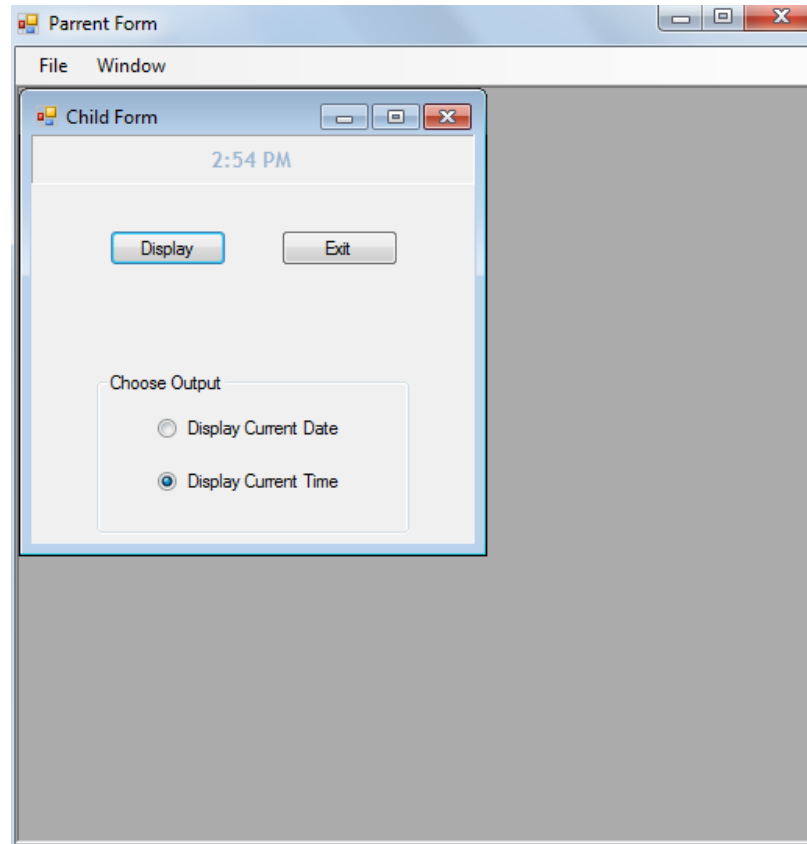
```
private void NewMenuItem_Click(object sender, EventArgs e)
{
    ChildForm newChild = new ChildForm();
    newChild.MdiParent = this;
    newChild.Show();
}
```

55. Build aplikasi dan jalankan

56. Apa yang terjadi? Mengapa demikian? Jelaskan dan perbaiki!

57. Setelah diperbaiki, jalankan kembali kode program. Ketika parent form muncul, pada menu File, klik New

58. Klik Display
59. Klik Display Current Time
60. Klik Display
61. Perhatikan hasilnya:



62. Maximize ukuran layar Child Form. Perhatikan efek dari dosk dan Anchor.
63. Klik menu New 3 kali
64. Maximize ukuran layar Parent Form
65. Klik menu Window Tile
66. Klik menu Window Cascade
67. Perhatikan hasilnya.

Percobaan III : Memakai control Groupbox, RadioButton, StatusBar, dan ToolStrip

1. Buatlah sebuah project baru dengan nama PRG2_20222_P3_3_XX (XX adalah 2 digit terakhir NIM Anda).
2. Tambahkan control pada form: 1 Label, 5 RadioButton, dan 1 GroupBox.
3. Atur properties control sesuai tabel berikut:

No.	Jenis Kontrol	Properti	Nilai Properti
1	Form	Text	PRG2_20192_P3_3_XX
		BackColor	White
		Size	470, 470
2	Label	Text	Pilih salah satu item berikut dan klik pada gambar yang sesuai!
		Location	12, 20
		Font	Tahoma, Bold, 9pt
3	GroupBox	Text	Nama Gambar
		Font	Micro Sans Serif, 9pt
		Size	175, 220
		Location	45, 80
4	RadioButton1	(Name)	rbMonitor
		Text	Monitor
5	RadioButton2	(Name)	rbKeyboard
		Text	Keyboard
6	RadioButton3	(Name)	rbMouse
		Text	Mouse
7	RadioButton4	(Name)	rbPrinter
		Text	Printer
8	RadioButton5	(Name)	rbScanner
		Text	Scanner

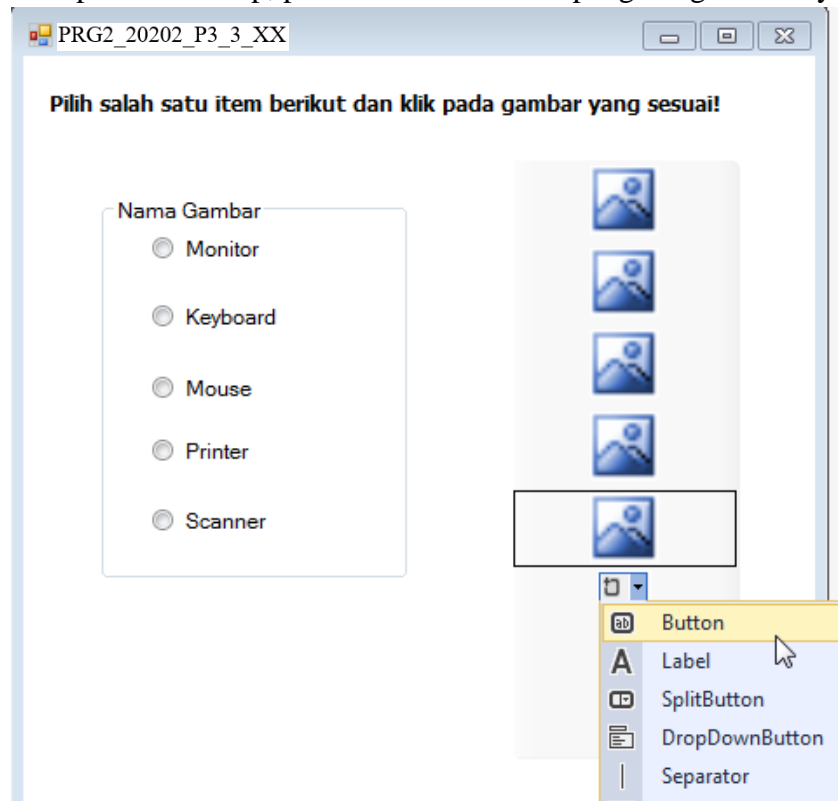
4. Tampilan form adalah seperti berikut:

The screenshot shows a Windows application window with the title bar 'PRG2_20202_P3_3_XX'. Inside the window, there is a label with the text 'Pilih salah satu item berikut dan klik pada gambar yang sesuai!'. Below this label is a group box titled 'Nama Gambar'. Inside the group box, there are five radio buttons arranged vertically, each with a corresponding label: 'Monitor', 'Keyboard', 'Mouse', 'Printer', and 'Scanner'. The 'Monitor' radio button is currently selected.

5. Tambahkan sebuah ToolStrip pada form, dan ubah propertiesnya sbb:
- (Name) : tsGambar
- AutoSize : false

Dock : None
GripStyle : Hidden
ImageScalingSize : 40, 40
Location : 281,60
LayoutStyle : VerticalStackWithOverflow
Size : 130, 343

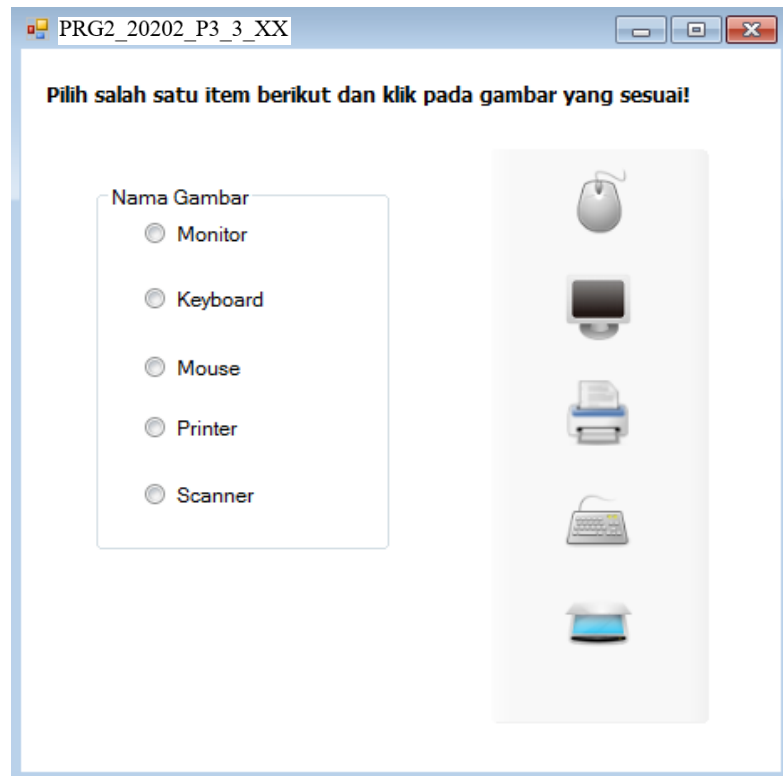
6. Klik pada ToolStrip, pilih Button. Lakukan pengulangan sebanyak 5 kali.



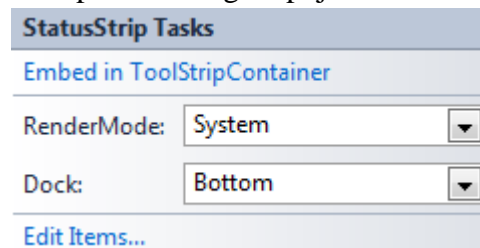
7. Copy folder icon yang telah disediakan, letakkan pada drive C:\
8. Lakukan pengaturan properties di setiap toolStripButton sbb:

No.	Jenis Kontrol	Properti	Nilai Properti
1	toolStripButton1	(Name)	tsbMouse
		AutoSize	false
		Image	Pilih Local resource → import → C:\icon → mouse.ico → OK
		Size	60, 60
		Tag	Mouse
		Text	Mouse
2	toolStripButton2	(Name)	tsbMonitor
		AutoSize	false
		Image	Pilih Local resource → import → C:\icon → monitor.ico → OK
		Size	60, 60
		Tag	Monitor
		Text	Monitor
3	toolStripButton3	(Name)	tsbPrinter
		AutoSize	false
		Image	Pilih Local resource → import → C:\icon → printer.ico → OK
		Size	60, 60
		Tag	Printer
		Text	Printer
4	toolStripButton4	(Name)	tsbKeyboard
		AutoSize	false
		Image	Pilih Local resource → import → C:\icon → keyboard.ico → OK
		Size	60, 60
		Tag	Keyboard
		Text	Keyboard
5	toolStripButton5	(Name)	tsbScanner
		AutoSize	false
		Image	Pilih Local resource → import → C:\icon → scanner.ico → OK
		Size	60, 60
		Tag	Scanner
		Text	Scanner

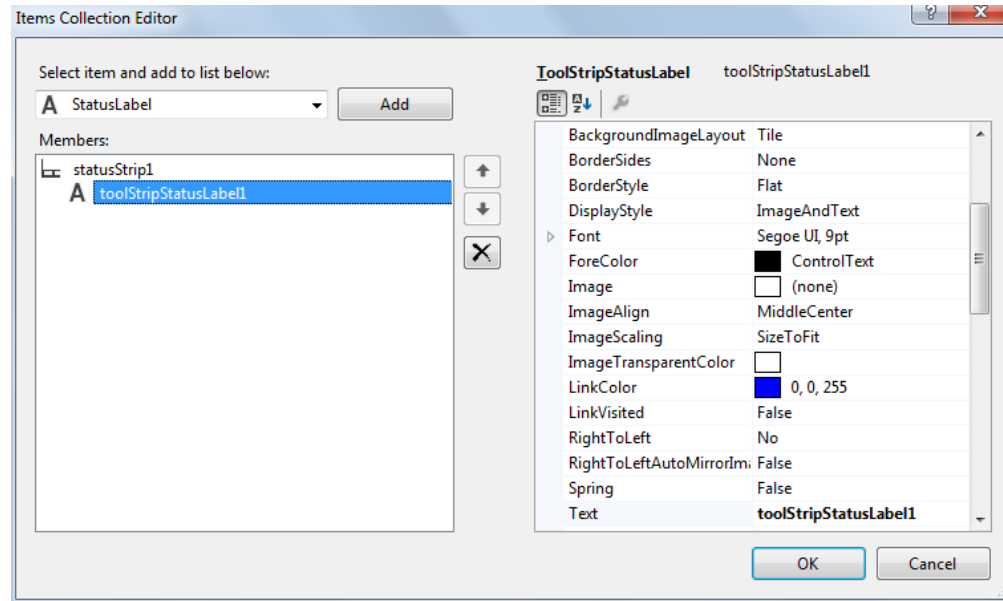
9. Tampilan form adalah sbb:



10. Tambahkan control StatusStrip pada form, pilih StatusLabel.
11. Klik panah di bagian pojok kanan atas dari StatusStrip → EditItems



12. Akan muncul kotak dialog berikut untuk mengatur item di dalam StatusStrip.



13. Atur properties toolStripStatusLabel1 sbb:

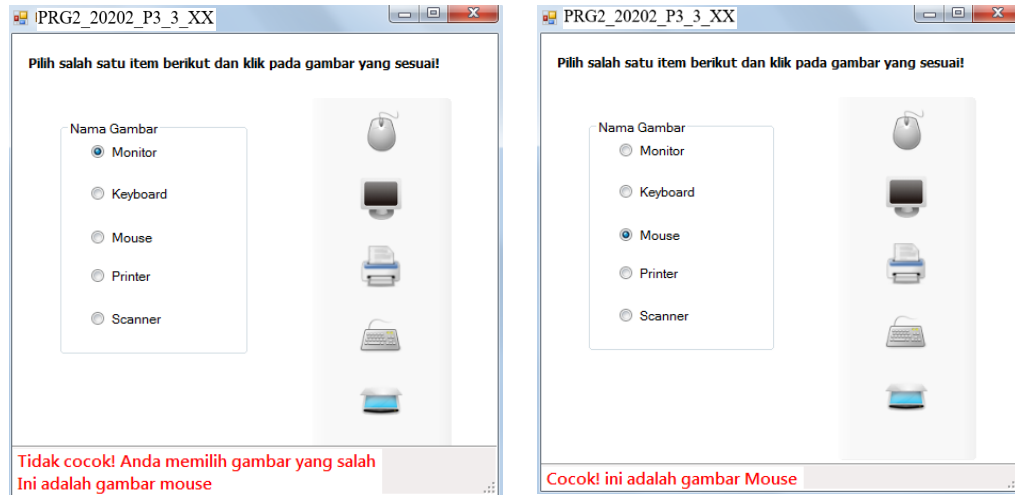
(Name) : tslKetGambar
 Font : Segoe UI, Semibold, 12pt
 ForeColor : Red
 TextAlign : MiddleLeft
 Text : (hapus dan biarkan kosong)

14. Dobel klik control tsbMouse, ketikkan kode program berikut:

```
private void tsbMouse_Click(object sender, EventArgs e)
{
    if (rbMouse.Checked == true)
    {
        tslKetGambar.Text = "Cocok! Ini adalah gambar Mouse";
    }
    else
    {
        tslKetGambar.Text = "Tidak cocok! Anda memilih gambar yang salah\nIni adalah gambar mouse";
    }
}
```

15. Ulangi kembali poin 14 untuk toolStrip lainnya.

16. Jalani program, coba pilih untuk setiap gambarnya. Sebagai contoh, klik toolStrip Mouse. Tampilannya sebagai berikut:



5. LATIHAN

Memakai control pilihan item: ComboBox, OptionButton, dan CheckBox

1. Buatlah sebuah project baru dengan nama PRG2_20222_P3_4_XX (XX adalah 2 digit terakhir NIM Anda).
2. Tambahkan control pada form: 4 Label, 1 TextBox, 1 ComboBox, 1 DateTimePicker, 2 GroupBox, 4 RadioButton, 8 CheckBox, dan 2 Button.
3. Atur layout form sebagai berikut (background bebas):

Astra Music School

FORM PENDAFTARAN

Nama :

Jenis Kelamin : -- Pilih Jenis Kelamin --

Tanggal Lahir : 08 January 2018

Pilihan Kelas

☐ Biola ☐ Piano

☐ Gitar ☐ Drum

☐ Saxophone ☐ Vokal

☐ Konduktor ☐ Komposer

Pilihan Jadwal

☐ Senin & Rabu, 14.00 - 16.00

☐ Selasa & Kamis, 15.00 - 17.00

☐ Sabtu & Minggu, 09.00 - 11.00

☐ Minggu, 13.00 - 17.00

4. Berikut event Click pada button Tampilkan, deklarasikan variabel local dalam event handlernya:

```
// deklarasi variabel lokal
string kelas = "";
string jadwal = "";
```

5. Atur penyeleksian kondisi pada pilihan Radio Button :

```

// penyeleksian kondisi pilihan RadioButton
// minggu
1 reference
private void radioButton4_CheckedChanged(object sender, EventArgs e)
{
    if (radioButton4.Checked)
    {
        jadwal = "Minggu, 13.00 - 17.00";
    }
}

// selasa & kamis
1 reference
private void radioButton2_CheckedChanged(object sender, EventArgs e)
{
    if (radioButton2.Checked)
    {
        jadwal = "Selasa & Kamis, 15.00 - 17.00";
    }
}

// jumat & sabtu
1 reference
private void radioButton3_CheckedChanged(object sender, EventArgs e)
{
    if (radioButton3.Checked)
    {
        jadwal = "Jumat & Sabtu, 09.00 - 11.00";
    }
}

// senin & rabu
1 reference
private void radioButton1_CheckedChanged(object sender, EventArgs e)
{
    if (radioButton1.Checked)
    {
        jadwal = "Senin & Rabu, 14.00 - 16.00";
    }
}

```

6. Atur penyeleksian kondisi pada pilihan ChecBox pada Button tampil:


```

private void btnTampil_Click(object sender, EventArgs e)
{
    // penyeleksian kondisi pilihan RadioButton
    if (cbBiola.Checked)
    {
        kelas += "Biola, ";
    }
    if (cbDrum.Checked)
    {
        kelas += "Drum, ";
    }
    if (cbGitar.Checked)
    {
        kelas += "Gitar, ";
    }
    if (cbKomposer.Checked)
    {
        kelas += "Komposer, ";
    }
    if (cbKonduktor.Checked)
    {
        kelas += "Konduktor, ";
    }
    if (cbPiano.Checked)
    {
        kelas += "Piano, ";
    }
    if (cbSaxophone.Checked)
    {
        kelas += "Saxophone, ";
    }
    if (cbVokal.Checked)
    {
        kelas += "Vokal, ";
    }
    // validasi peringatan jika kelas atau jadwal tidak dipilih
    if (kelas == "")
    {
        MessageBox.Show("Harus memilih salah satu dari pilihan kelas", "Warning!",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }
    if (jadwal == "")
    {
        MessageBox.Show("Harus memilih salah satu dari pilihan jadwal", "Warning!",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }

    //menampilkan data pada messageBox jika lolos dari validasi
    if(kelas != "" && jadwal != ""){
        MessageBox.Show
            ("Nama : "+ txtNama.Text +
            "\nJenis Kelamin : "+ cbJK.Text +
            "\nTanggal Lahir : "+ dtTglLahir.Text +
            "\nPilihan Kelas : "+ kelas +
            "\nPilihan Jadwal: "+ jadwal,
            "Informasi Pendaftaran",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
    }

    // mengosongkan isi variabel kelas setelah ditampilkan
    kelas = "";
}

```

7. Atur event Click pada button Tampilkan untuk menampilkan data
8. Atur event Click pada button Selesai untuk keluar dari Program.
9. Jalankan program, contoh tampilannya sbb:

Saat button Tampilkan diklik, maka akan muncul kotak dialog berikut:

10. Analisa Program
 - a. Jelaskan fungsi property MaxLength pada control TextBox.
 - b. Jelaskan fungsi property Items pada control ComboBox.
 - c. Jelaskan fungsi property Value, Format, dan CustomFormat pada control DateTimePicker.

- d. Jelaskan fungsi property Checked pada control RadioButton dan CheckBox.
- e. Kontrol ComboBox, RadioButton, dan CheckBox merupakan control yang dapat digunakan untuk menampilkan item pilihan, adakah perbedaan dari penggunaan ke-3 kontrol tersebut? Jelaskan jawaban Anda!
- f. Kenapa variable kelas harus dikosongkan setelah menampilkan data.

Catatan:

- 1. Tulislah jawaban anda dalam sebuah file dokumen, bernama PRG2_20222_P3_XX.docx (XX adalah 2 digit terakhir dari NIM Anda)
- 2. Buat folder PRG2_20222_P3_XX (XX adalah 2 digit terakhir dari NIM Anda)
- 3. Masukkan semua folder hasil kerja Anda ke dalam folder tersebut.
- 4. Kumpulkan file dan folder project tersebut setelah dikompresi dalam bentuk zip.