

Aulas 06 e 07 – Atividades Práticas com SDI, Modal e MDI

Este documento contém instruções passo a passo, esboço dos formulários e códigos de exemplo para as atividades práticas das Aulas 06 (múltiplos formulários – SDI e Modal) e 07 (MDI).

PARTE 1 — Aula 06: SDI e Modal

Objetivo: abrir outros formulários a partir do principal, entender a diferença entre Show() e ShowDialog(), e capturar retorno de um formulário modal.

1) Exemplo Base – Form Principal abrindo SDI e Modal

Crie um projeto WinForms e adicione TRÊS Forms: frmPrincipal (startup), frmJanela e frmModal.

No frmPrincipal, adicione dois botões: btnAbrirJanela (abre SDI) e btnAbrirModal (abre modal).

```
// frmPrincipal.cs
private void btnAbrirJanela_Click(object sender, EventArgs e)
{
    var janela = new frmJanela();
    janela.Show(); // SDI: abre janela independente
}

private void btnAbrirModal_Click(object sender, EventArgs e)
{
    using (var dlg = new frmModal())
    {
        dlg.StartPosition = FormStartPosition.CenterParent;
        dlg.ShowDialog(this); // Modal: bloqueia até fechar
    }
}
```

2) Atividade 1 – Form de Ajuda (Modal)

Crie um frmAjuda com um Label de texto explicativo e um Button 'Fechar'. Abra-o como modal a partir do principal.

```
// frmPrincipal.cs (abrir ajuda)
private void mnuAjuda_Click(object sender, EventArgs e)
{
    using (var ajuda = new frmAjuda())
    {
        ajuda.ShowDialog(this);
    }
}
```

```
}
```

```
// frmAjuda.cs (botão fechar)
private void btnFechar_Click(object sender, EventArgs e)
{
    this.Close();
}
```

3) Atividade 2 – Form de Configurações (Modal) com Retorno

Crie um frmConfiguracoes com um GroupBox contendo dois RadioButtons: rdbTemaClaro e rdbTemaEscuro, e dois botões: btnOk e btnCancel. A ideia é que o formulário modal 'retorne' a escolha do usuário.

No frmConfiguracoes, exponha uma propriedade somente-leitura para o tema escolhido e configure o DialogResult nos botões:

```
// frmConfiguracoes.cs
public partial class frmConfiguracoes : Form
{
    public string TemaSelecionado { get; private set; } = "Claro";

    public frmConfiguracoes()
    {
        InitializeComponent();
    }

    private void btnOk_Click(object sender, EventArgs e)
    {
        TemaSelecionado = rdbTemaEscuro.Checked ? "Escuro" : "Claro";
        this.DialogResult = DialogResult.OK;    // sinaliza retorno de sucesso
        this.Close();
    }

    private void btnCancel_Click(object sender, EventArgs e)
    {
        this.DialogResult = DialogResult.Cancel;
        this.Close();
    }
}
```

No frmPrincipal, abra o modal, verifique o DialogResult e aplique a configuração:

```
// frmPrincipal.cs (abrir configurações modal e ler retorno)
private void mnuConfiguracoes_Click(object sender, EventArgs e)
{
    using (var cfg = new frmConfiguracoes())
    {
        var dr = cfg.ShowDialog(this);
    }
}
```

```

        if (dr == DialogResult.OK)
        {
            // Lê a propriedade exposta pelo modal
            if (cfg.TemaSelecionado == "Escuro")
            {
                this.BackColor = System.Drawing.Color.FromArgb(45,45,48);
                this.ForeColor = System.Drawing.Color.WhiteSmoke;
            }
            else
            {
                this.BackColor = System.Drawing.SystemColors.Control;
                this.ForeColor = System.Drawing.Color.Black;
            }
            MessageBox.Show($"Tema aplicado: {cfg.TemaSelecionado}",
                "Configurações",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
        else
        {
            MessageBox.Show("Configurações canceladas.", "Configurações",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
    }
}

```

Obs.: É a combinação de `DialogResult.OK` (no modal) e `ShowDialog()` (no chamador) que permite saber se o usuário confirmou ou cancelou. O dado 'retornado' é exposto por propriedade pública no form modal.

4) Extra – Abrir via MenuStrip

Adicione um MenuStrip no frmPrincipal com menus: Arquivo → Abrir Janela, Abrir Modal; Ajuda → Sobre.

```

// Handlers do MenuStrip
private void mnuAbrirJanela_Click(object sender, EventArgs e)
{
    var janela = new frmJanela();
    janela.Show();
}

private void mnuAbrirModal_Click(object sender, EventArgs e)
{
    using (var dlg = new frmModal())
    {
        dlg.ShowDialog(this);
    }
}

```

PARTE 2 — Aula 07: MDI (Multiple Document Interface)

Objetivo: criar uma janela pai (MDI) que contém múltiplas janelas-filhas (documentos).

1) Exemplo Base – Form MDI Pai e Filhos

No frmMDIPrincipal, defina `IsMdiContainer = true`. Adicione um MenuStrip com Arquivo → Novo, Fechar todos, Salvar; Janela → Cascata, Lado a Lado, Organizar Ícones; e Ajuda → Sobre.

```
// frmMDIPrincipal.cs (abrir filho com TextBox multilinha)
private int _contador = 1;

private void mnuNovo_Click(object sender, EventArgs e)
{
    var filho = new frmDocumento(); // Form com TextBox multilinha: txtDoc
    (Dock=Fill)
    filho.MdiParent = this;
    filho.Text = $"Documento {_contador++}";
    filho.Show();
}
```

2) Organização das Janelas (Cascata, Lado a Lado, Ícones)

```
// frmMDIPrincipal.cs
private void mnuCascata_Click(object sender, EventArgs e)
{
    this.LayoutMdi(MdiLayout.Cascade);
}

private void mnuLadoALado_Click(object sender, EventArgs e)
{
    this.LayoutMdi(MdiLayout.TileVertical);
}

private void mnuOrganizarIcones_Click(object sender, EventArgs e)
{
    this.LayoutMdi(MdiLayout.ArrangeIcons);
}
```

3) Fechar Todos os Filhos

```
// frmMDIPrincipal.cs
private void mnuFecharTodos_Click(object sender, EventArgs e)
{
    foreach (Form f in this.MdiChildren)
        f.Close();
}
```

4) Extra – Salvar Conteúdo do Documento .TXT

No frmDocumento (filho), inclua um TextBox multilinha `txtDoc`. No MDI principal, implemente salvar o filho ativo:

```
// frmMDIPrincipal.cs
using System.IO;

private void mnuSalvar_Click(object sender, EventArgs e)
{
    var ativo = this.ActiveMdiChild as frmDocumento;
    if (ativo == null)
    {
        MessageBox.Show("Nenhum documento ativo.", "Salvar",
            MessageBoxButtons.OK, MessageBoxIcon.Warning);
        return;
    }

    using (var dlg = new SaveFileDialog())
    {
        dlg.Filter = "Arquivo de texto (*.txt)|*.txt";
        if (dlg.ShowDialog(this) == DialogResult.OK)
        {
            File.WriteAllText(dlg.FileName, ativo.ConteudoTexto);
            MessageBox.Show("Documento salvo!", "Salvar",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
    }
}
```

Exponha a propriedade ConteudoTexto no frmDocumento:

```
// frmDocumento.cs
public string ConteudoTexto => txtDoc.Text;
```

Esboços de Interfaces (Wireframes)

frmPrincipal (Aula 06): dois botões ou MenuStrip para abrir Janela (SDI) e Modal; menu Configurações e Ajuda.

frmConfiguracoes (Modal): GroupBox 'Tema' com RadioButtons 'Claro' e 'Escuro', botões OK/Cancelar.

frmMDIPrincipal (Aula 07): janela com MenuStrip (Arquivo, Janela, Ajuda) e área MDI; documentos filho com TextBox multiline.