

PROGRAMAÇÃO COM ACESSO A BANCO DE DADOS

C# .NET: Listar e Buscar



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE

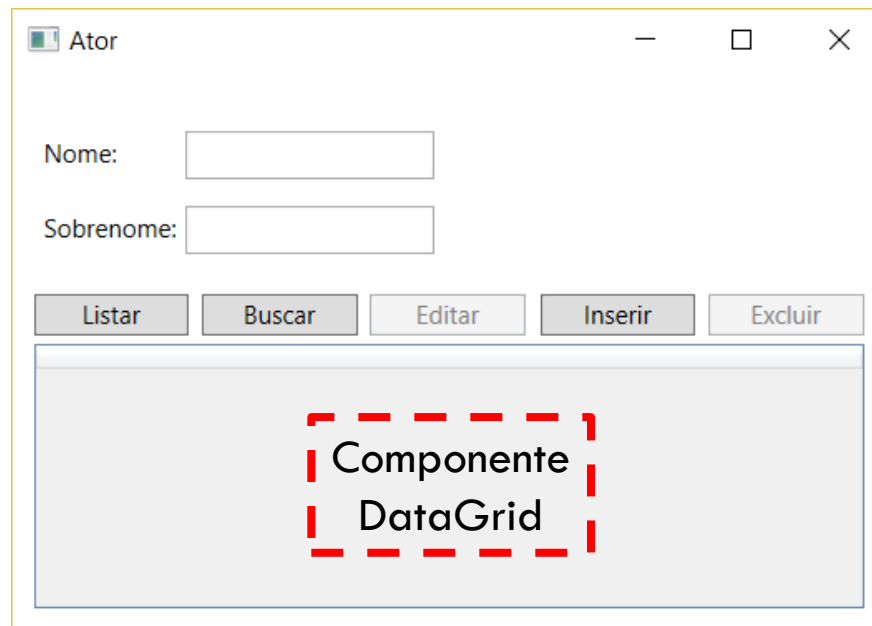
eliezio.soares@ifrn.edu.br | <https://docente.ifrn.edu.br/elieziosoares>

Msc. Eliezio Soares

GERENCIAMENTO DE ATOR

INTERFACE (ATOR.XAML)

- Vamos melhorar a interface da tela de gerenciamento de atores adicionando alguns componentes:



- Nesta prática implementaremos as funções listar e buscar.

GERENCIAMENTO DE ATOR

INTERFACE (ATOR.XAML)

- O código xaml do componente DataGrid deve ficar da seguinte forma:

```
<DataGrid x:Name="dgAtor" Margin="10,131,11.8,10" ItemsSource="{Binding}"/>
```

- A propriedade ItemsSource obtém ou define uma coleção usada para gerar o conteúdo.
- O valor “{Binding}” permite a ligação dinâmica do conteúdo.

GERENCIAMENTO DE ATOR

INTERFACE (ATOR.XAML)

- Os botões de Edição e exclusão tiveram a propriedade “IsEnabled” alterada para false.
 - Ator.xaml

```
<Button x:Name="editarAtorBT" Content="Editar" Click="editarAtorBT_Click"  
... isEnabled="False"/>
```

- Propriedades (visual):



LISTAR ATOR

DAL (DATA ACCESS LAYER)

1. Conectar ao banco de dados LocadoraDVD;
2. Listar todos os atores (select *);
3. Preencher e retornar um DataTable (ADO .NET);

LISTAR ATOR

DAL (DATA ACCESS LAYER)

```
public DataTable listarAtor()
{
    DataTable dt = new DataTable();
    try
    {
        using (pgsqlConnection = new NpgsqlConnection(connectionString))
        {
            pgsqlConnection.Open();
            string cmd = "Select * from ator order by ator_id;";

            using (NpgsqlDataAdapter Adpt = new NpgsqlDataAdapter(cmd, pgsqlConnection))
            {
                Adpt.Fill(dt);
            }
        }
    }
    catch (Exception ex)
    {
        throw;
    }
    finally
    {
        pgsqlConnection.Close();
    }
    return dt;
}
```

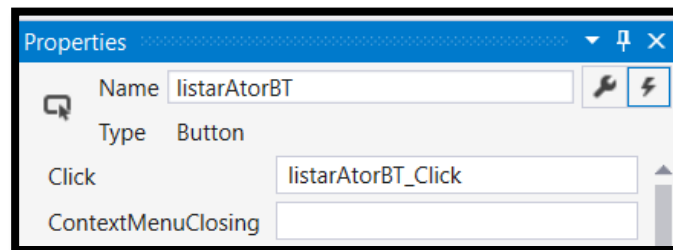
NpgsqlDataAdapter:
um adaptador da biblioteca Npgsql para a ADO .NET.

LISTAR ATOR

MANIPULAÇÃO

1. O botão “Listar” deve ser ligado ao seu manipulador.
 - Com um duplo clique sobre o botão o Visual Studio gera um método no arquivo Ator.xaml.cs;
 - Pode ser feito manualmente, acrescentando a propriedade “Click”.

```
<Button x:Name="listarAtorBT" Content="Listar" ... Click="listarAtorBT_Click"/>
```



LISTAR ATOR

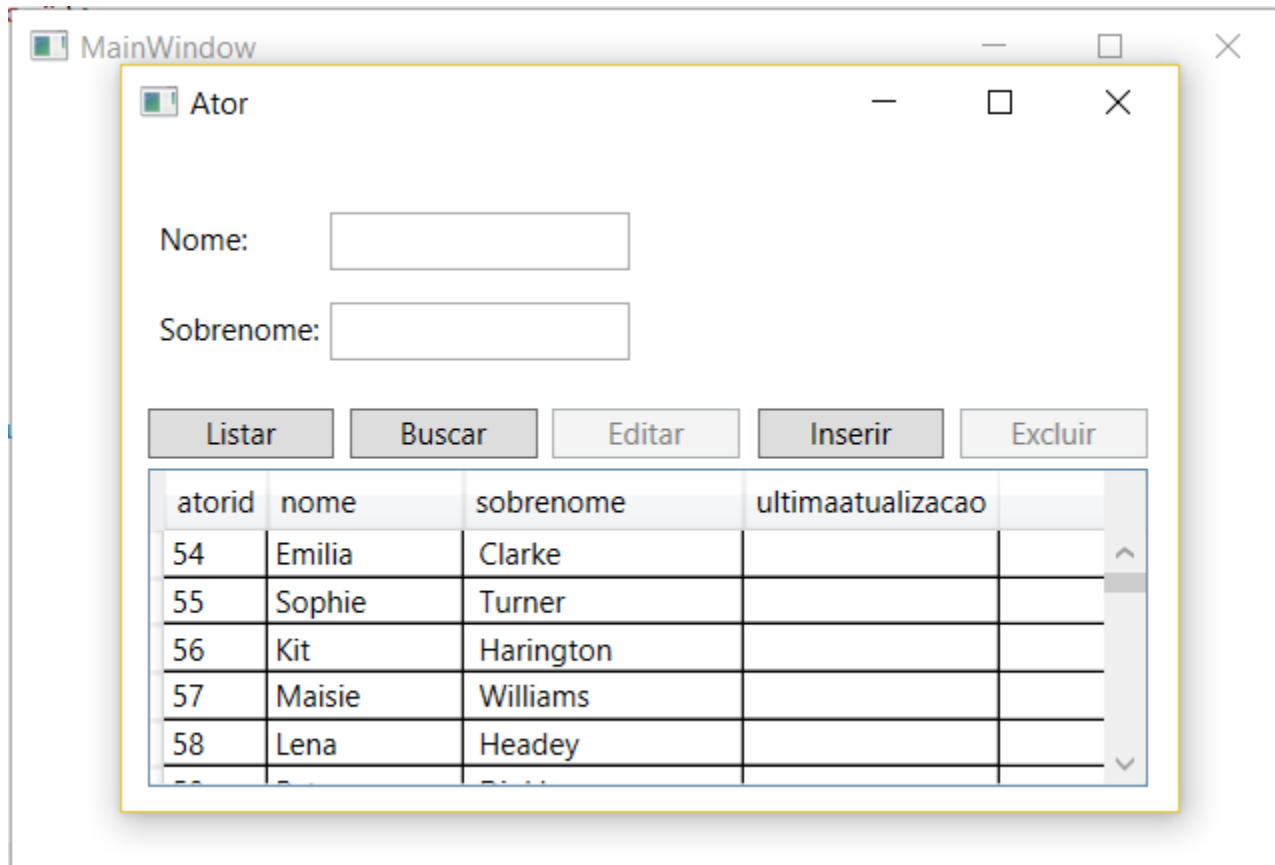
ATOR.XAML.CS

1. Executar o método `listarAtor()` da classe DAL;
 - O método retorna um `DataTable`;
2. Alterar a propriedade `DataContext` do `DataGrid` Ator (`dgAtor`);
 - Definimos o `DataTable` recebido da DAL como fonte de dados para o componente `DataGrid`.

```
private void listarAtorBT_Click(object sender, RoutedEventArgs e)
{
    try
    {
        DataTable dt = dao.listarAtor();
        dgAtor.DataContext = dt.DefaultView;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Erro: " + ex.Message);
        return;
    }
}
```


LISTAR ATOR

EXECUÇÃO



BUSCAR ATOR

DAL (DATA ACCESS LAYER)

1. Receber um nome e sobrenome;
2. Conectar ao banco de dados LocadoraDVD;
3. Selecionar as ocorrências de atores cujos nomes OU sobrenomes contenham os valores recebidos;
4. Preencher e retornar um DataTable (ADO .NET);



BUSCAR ATOR

DAL (DATA ACCESS LAYER)

```
public DataTable buscarAtor(String nome, String sobrenome)
{
    DataTable dt = new DataTable();
    try
    {
        using (pgsqlConnection = new NpgsqlConnection(connectionString))
        {
            pgsqlConnection.Open();
            string cmd;
            if(nome != String.Empty && sobrenome != String.Empty)
                cmd = String.Format("Select * from ator WHERE nome like '{0}%' OR sobrenome like '{1}%' order by ator_id;", nome, sobrenome);
            else if(nome != String.Empty)
                cmd = String.Format("Select * from ator WHERE nome like '{0}%' order by ator_id;", nome);
            else
                cmd = String.Format("Select * from ator WHERE sobrenome like '{0}%' order by ator_id;", sobrenome);

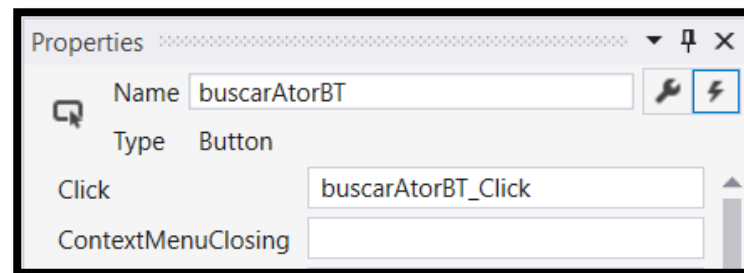
            using (NpgsqlDataAdapter Adpt = new NpgsqlDataAdapter(cmd, pgsqlConnection))
            {
                Adpt.Fill(dt);
            }
        }
    }
    catch (Exception ex)
    {
        throw;
    }
    finally
    {
        pgsqlConnection.Close();
    }
    return dt;
}
```

BUSCAR ATOR

MANIPULAÇÃO

1. O botão “Buscar” deve ser ligado ao seu manipulador.
 - Com um duplo clique sobre o botão o Visual Studio gera um método no arquivo Ator.xaml.cs;
 - Pode ser feito manualmente, acrescentando a propriedade “Click”.

```
<Button x:Name="buscarAtorBT" Content="Buscar" HorizontalAlignment="Left" ...  
Click="buscarAtorBT_Click"/>
```



BUSCAR ATOR

ATOR.XAML.CS

1. Executar o método `buscarAtor(<String>, <String>)` da classe DAL;
 - O método retorna um `DataTable`;
2. Alterar a propriedade `DataContext` do `DataGrid` Ator (`dgAtor`);
 - Definimos o `DataTable` recebido da DAL como fonte de dados para o componente `DataGrid`.

```
private void buscarAtorBT_Click(object sender, RoutedEventArgs e)
{
    try
    {
        [1] DataTable dt = dao.buscarAtor(nomeTB.Text, sobrenomeTB.Text);
        dgAtor.DataContext = dt.DefaultView; [2]
    }
    catch (Exception ex)
    {
        MessageBox.Show("Erro: " + ex.Message);
        return;
    }
}
```

DÚVIDAS?



REFERÊNCIAS BIBLIOGRÁFICAS

Npgsql: Getting Started. Disponível em:
<<http://www.npgsql.org/doc/index.html#getting-started>>. Acesso
em 24 Jan. 2017.

Classe DataGridView: .NET Framework (current version) . Disponível em:
<[https://msdn.microsoft.com/pt-br/library/system.windows.controls.datagrid\(v=vs.110\).aspx](https://msdn.microsoft.com/pt-br/library/system.windows.controls.datagrid(v=vs.110).aspx)> Acesso
em 31 Jan. 2017.