

```
using System;
using System.Windows.Forms;
//Referência às camadas
using Negocios;
using ObjetoTransferencia;

namespace Apresentacao
{
    public partial class FrmClienteConsultar : Form
    {
        public FrmClienteConsultar()
        {
            InitializeComponent();
            //Evita que o DataGridView gere as colunas automaticamente com o retorno do BD
            dgvPrincipal.AutoGenerateColumns = false;
        }

        //Atualiza a lista de clientes (Pesquisa ao BD)
        private void atualizaGridCliente()
        {
            NegociosCliente NegociosCliente = new NegociosCliente();

            ClienteColecao clienteColecao = NegociosCliente.ConsultaClienteNome (txtPesquisa.Text.Trim());

            dgvPrincipal.DataSource = null;
            dgvPrincipal.DataSource = clienteColecao;

            dgvPrincipal.Update();
            dgvPrincipal.Refresh();

            if (clienteColecao.Count <= 0)
            {
                MessageBox.Show("Não há registros para exibição!", "Consulta",
                    MessageBoxButtons.OK, MessageBoxIcon.Information);
                txtPesquisa.Clear();
                txtPesquisa.Focus();
            }
        }

        private void btnPesquisar_Click(object sender, EventArgs e)
        {
            atualizaGridCliente();
        }

        private void btnFechar_Click(object sender, EventArgs e)
        {
            Close();
        }

        private void Excluir_Click(object sender, EventArgs e)
        {
            //Verifica se há um registro selecionado
            if (dgvPrincipal.SelectedRows.Count <= 0)
            {

```

```
        MessageBox.Show("Não há registro selecionado!", "Excluir",  
            MessageBoxButtons.OK, MessageBoxIcon.Information);  
        return;  
    }  
  
    //Faz a pergunta ao usuário se realmente deseja excluir o registro  
    if (MessageBox.Show("Excluir  
registro?", "Excluir", MessageBoxButtons.YesNo, MessageBoxIcon.Question)  
        == DialogResult.Yes)  
    {  
        //Gera um novo objeto de acordo com a linha selecionada  
        //DataBoudItem pega todas as informações do grid (DataTable),  
        inclusive as ocultas  
        Cliente clienteClick = dgvPrincipal.SelectedRows[0].DataBoundItem  
            as Cliente;  
  
        NegociosCliente NegociosCliente = new NegociosCliente();  
  
        string retorno = NegociosCliente.ExcluirCliente(clienteClick);  
  
        try  
        {  
            int num = Convert.ToInt32(retorno);  
            MessageBox.Show("Registro excluído com sucesso!", "Excluir",  
                MessageBoxButtons.OK, MessageBoxIcon.Information);  
            atualizaGridCliente();  
        }  
        catch  
        {  
            MessageBox.Show("Erro ao excluir o registro!\n\n" + retorno,  
                "Excluir", MessageBoxButtons.OK, MessageBoxIcon.Information);  
        }  
    }  
}  
  
private void btnInserir_Click(object sender, EventArgs e)  
{  
    FrmClienteCadastrarAlterar frmClienteCadastrarAlterar = new  
        FrmClienteCadastrarAlterar("inserir", null);  
    frmClienteCadastrarAlterar.ShowDialog();  
  
    atualizaGridCliente();  
}  
  
private void btnAlterar_Click(object sender, EventArgs e)  
{  
    //Verifica se há um registro selecionado  
    if (dgvPrincipal.SelectedRows.Count <= 0)  
    {  
        MessageBox.Show("Selecione ao menos um registro para alteração!",  
            "Alterar", MessageBoxButtons.OK, MessageBoxIcon.Information);  
        return;  
    }  
  
    Cliente clienteClick = dgvPrincipal.SelectedRows[0].DataBoundItem as  
        Cliente;
```

```
FrmClienteCadastrarAlterar frmClienteCadastrarAlterar = new
    FrmClienteCadastrarAlterar("alterar", clienteClick);
frmClienteCadastrarAlterar.ShowDialog();

atualizaGridCliente();

txtPesquisa.Clear();
txtPesquisa.Focus();
    }
}
}
```