

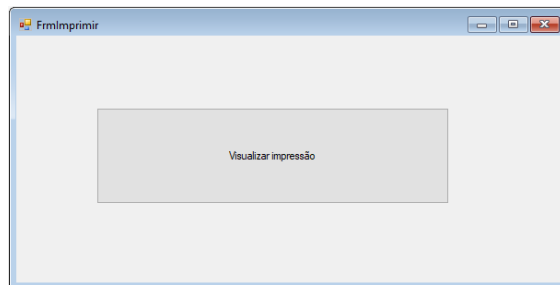
### Relatório

Acrescentar ao Projeto Agenda da Aula 07, já com as alterações da Aula 08, agora os comandos referentes a impressão.

No FrmPrincipal acrescentar a opção de Impressão conforme ilustrado abaixo.



Inserir um novo formulário com o nome FrmImprimir, com um botão de nome cmdVisualizar conforme ilustrado a seguir.



O relatório irá imprimir o nome e o telefone de todas as pessoas cadastradas na Agenda.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Drawing.Printing;

namespace ProjetoAgenda
{
    public partial class FrmImprimir : Form
    {
        public FrmImprimir()
        {
            InitializeComponent();

            ConexaoDados dados = new ConexaoDados();
            private int i;
            private void cmdVisualizar_Click(object sender, EventArgs e)
            {
                PrintDocument pd = new PrintDocument();
                pd.DocumentName = "Relatório";
                pd.BeginPrint += Pd_BeginPrint;
            }
        }
    }
}
```

## LTPII - Aula 09

```
pd.PrintPage += Imprimir;
//cria objeto preview
PrintPreviewDialog ppd = new PrintPreviewDialog();
ppd.Document = pd;
ppd.ShowDialog();
}

private void Pd_BeginPrint(object sender, PrintEventArgs e)
{
    //O que deve ocorrer antes de iniciar a impressão
    i = 0;
}

private void Imprimir(object sender, PrintPageEventArgs ev)
{
    //Configurações da página
    float linhaPorPagina = 0;
    float posicaoVertical = 0;
    float contador = 0;
    float margemEsqueda = 20;
    float margemSuperior = 20;
    float alturaFonte = 0;
    string linha = "";

    Font fonte = new Font("Arial", 14);
    alturaFonte = fonte.GetHeight(ev.Graphics);
    linhaPorPagina = Convert.ToInt32(ev.MarginBounds.Height / alturaFonte);

    //Título
    linha = "Lista Agenda";
    posicaoVertical = margemSuperior + contador * alturaFonte;

    ev.Graphics.DrawString(linha, fonte, Brushes.Black, margemEsqueda,
posicaoVertical);
    contador += 4;

    linha = "Nome";
    posicaoVertical = margemSuperior + contador * alturaFonte;
    ev.Graphics.DrawString(linha, fonte, Brushes.Black, margemEsqueda,
posicaoVertical);

    linha = "Telefone";
    ev.Graphics.DrawString(linha, fonte, Brushes.Black, margemEsqueda + 500,
posicaoVertical);

    contador += 1;
    linha = "-----";
    posicaoVertical = margemSuperior + contador * alturaFonte;
    ev.Graphics.DrawString(linha, fonte, Brushes.Black, margemEsqueda,
posicaoVertical);

    contador++;
    DataSet ds = dados.ListarDados();

    if (ds.Tables[0] != null)
    {
        while (i < ds.Tables[0].Rows.Count && contador < linhaPorPagina)
        {
            DataRow item = ds.Tables[0].Rows[i];

            linha = item["Nome"].ToString();
```

## LTPII - Aula 09

```
        posicaoVertical = margemSuperior + contador * alturaFonte;
        ev.Graphics.DrawString(linha, fonte, Brushes.Black, margemEsqueda,
posicaoVertical);

        linha = item["Telefone"].ToString();
        posicaoVertical = margemSuperior + contador * alturaFonte;
        ev.Graphics.DrawString(linha, fonte, Brushes.Black, margemEsqueda
+ 500, posicaoVertical);

        contador += 2;
        i++;
    }

}
else MessageBox.Show("Não existe Pessoa cadastrada!");

if (contador > linhaPorPagina)
{
    ev.HasMorePages = true;
}
else
{
    ev.HasMorePages = false;
}
}
}
}
```