

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;
using System.Collections.ObjectModel;

namespace SistemaPontoVenda
{
    /// <summary>
    /// Interaction logic for jRealizarVenda.xaml
    /// </summary>
    public partial class jRealizarVenda : Window
    {
        //Declarar uma lista de Itens.
        ObservableCollection<ItensPedido>
listaItens = new
ObservableCollection<ItensPedido>();

        //Um contador para o número do pedido
        private int contador = 1;

        public jRealizarVenda()
        {
            InitializeComponent();
        }
    }
}

```

```

        private void Window_Loaded(object sender,
RoutedEventArgs e)
        {
            //Preenche a combobox com os ID dos
clientes
            clsCliente cliente = new clsCliente();
            cboClientes.ItemsSource =
cliente.GetClientes();

            //Preenche a combobox com a descrição dos
produtos
            Produto produto = new Produto();
            cboProduto.ItemsSource =
produto.RetornarTodosProdutos();
        }

        private void btnAbrirPedido_Click(object
sender, RoutedEventArgs e)
        {
            //Usa o método GetUltimoPedido da Classe
Pedido para ir ao banco e trazer o último pedido.
            Pedido pedido = new Pedido();
            txtPedido.Text =
(pedido.GetUltimoPedido() + 1).ToString();
        }

        private void btnAdicionarItem_Click(object
sender, RoutedEventArgs e)
        {
            Produto produto = new Produto();

            //Pegar o ID do produto selecionado na
cboProduto
            produto =
(Produto)cboProduto.SelectedItem;

```

```

        ItensPedido item = new ItensPedido();
        item.ItemNum = contador;
        if (txtQtidade.Text == string.Empty)
        {
            MessageBox.Show("Por favor!, digite
a quantidade.");
            txtQtidade.Focus();
        }
        else
        {
            item.Qtidade =
Convert.ToInt16(txtQtidade.Text);
            item.PrecoVenda = Convert.ToDecimal(
txtPrecoVenda.Text);
            if (txtPedido.Text == string.Empty)
            {
                MessageBox.Show("Por favor!,
informe o número do pedido.");
                txtPedido.Focus();
            }
            else
            {
                item.PedidoID = Convert.ToInt16(
txtPedido.Text);
            }
            item.ProdutoID = produto.ProdutoID;

            listaItens.Add(item);
            dataItens.ItemsSource = listaItens;

            //Incrementa o contador
            contador += 1;
        }
    }

```

```

        private void
cboProduto_SelectionChanged(object sender,
SelectionChangedEventArgs e)
    {
        Produto produto = new Produto();

        produto =
(Produto)cboProduto.SelectedItem;
        txtPrecoVenda.Text =
produto.PrecoUnitario.ToString();
    }

    private void btnCalcularPedido_Click(object
sender, RoutedEventArgs e)
    {
        Decimal valorTotalPedido = 0;
        foreach (ItensPedido item in
listaItens)
        {
            valorTotalPedido += item.PrecoVenda
* item.Qtdade;
        }
        tbValorTotal.Text =
valorTotalPedido.ToString();
    }

    private void btnFecharPedido_Click(object
sender, RoutedEventArgs e)
    {
        Pedido pedido = new Pedido();
        pedido.PedidoID =
Convert.ToInt16(txtPedido.Text);
        pedido.Data =
Convert.ToDateTime(datePicker1.SelectedDate);
    }

```

```

        pedido.Valor =
Convert.ToDecimal(tbValorTotal.Text);

        //Instancia um cliente para poder
recuperar o seu ID da combobox, onde está o cliente
selecionado
        clsCliente cliente = new clsCliente();
        cliente =
(clsCliente)cboClientes.SelectedItem;

        pedido.ClienteID = cliente.ClienteID;

        try
        {
            //Inserir o pedido no banco de
dados.
            pedido.InserirPedido(pedido);

            //Instancia um itempedido para que
possa ser inserido.
            ItensPedido newItem = new
ItensPedido();
            //Percorre toda a lista de itens da
datagrid.
            foreach (ItensPedido item in
listaItens)
            {
                //Atualizar as propriedades do
novo item com as propriedades do item lido da
lista.

                newItem.ItemNum = item.ItemNum;
                newItem.Qtidade = item.Qtidade;
                newItem.PrecoVenda =
item.PrecoVenda;

```

```

        newItem.PedidoID =
item.PedidoID;
        newItem.ProdutoID =
item.ProdutoID;
        //Chama o método inserirItem
para gravar os dados do item no banco de dados.

newItem.InserirItensPedido(newItem);

        //Chama o método
DarBaixaNoEstoque para atualizar a quantidade
estocada na tabela Produto
        Produto produto = new
Produto();

produto.DarBaixaNoProduto(newItem.ProdutoID,
newItem.Qtidade);
    }
    MessageBox.Show("Operação realizada
com sucesso.");
    txtPedido.Focus();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Erro na operação."
+ ex.Message );
    }
}

private void btnExcluirItem_Click(object
sender, RoutedEventArgs e)
{
    if (dataItens.SelectedItem != null)
    {

```

```
listaItens.Remove((ItensPedido)dataItens.SelectedItem);
    }
    else
    {
        MessageBox.Show("Atenção!, você precisa selecionar um item para poder excluir.");
    }
}
}
```