<Grid>

<ListBox x:Name="PartnersList" HorizontalContentAlignment="Stretch">

<ListBox.ItemTemplate>

<DataTemplate>

<Border BorderBrush="Gray" BorderThickness="1" Margin="5" Padding="10">

<StackPanel Orientation="Horizontal" HorizontalAlignment="Stretch">

<StackPanel Width="550">

<TextBlock FontWeight="Bold">

<Run Text="{Binding Тип\_партнера, Mode=OneWay}"/>

<Run Text=" | "/>

<Run Text="{Binding Наименование\_партнера, Mode=OneWay}"/>

</TextBlock>

<TextBlock Text="{Binding Директор}"/>

<TextBlock Text="{Binding Телефон\_партнера ,StringFormat=+7{0}}"/>

<TextBlock>

<Run Text="Рейтинг: "/>

<Run Text="{Binding Рейтинг , Mode=OneWay}"/>

</TextBlock>

</StackPanel>

<TextBlock Text="{Binding Discount, StringFormat={}{0}%}"

HorizontalAlignment="Right"

VerticalAlignment="Center"

Margin="0,0,10,0"

FontWeight="Bold"

Foreground="Black"/>

</StackPanel>

</Border>

</DataTemplate>

</ListBox.ItemTemplate>

</ListBox>

</Grid>

</Window>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

LoadPartners();

}

private void LoadPartners()

{

try

{

using (var db = new ekzEntities())

{

// Загружаем типы партнеров заранее

var partnerTypes = db.Тип\_партнера\_.ToDictionary(t => t.Код, t => t.Тип);

// Получаем данные партнеров с вычислением количества

var partners = db.Партнер\_

.AsEnumerable() // Переключаем на клиентскую обработку

.Select(p => new

{

p.Код,

p.Наименование\_партнера,

p.Директор,

p.Телефон\_партнера,

p.Рейтинг,

Тип\_партнера = partnerTypes.TryGetValue((int)p.Тип\_партнера, out var type) ? type : "Не указан",

TotalQuantity = db.Реализация\_продукции\_

.Where(pp => pp.Код\_партнера == p.Код)

.Sum(pp => (int?)pp.Количество\_продукции) ?? 0

})

.Select(p => new

{

p.Код,

p.Наименование\_партнера,

p.Директор,

p.Телефон\_партнера,

p.Рейтинг,

p.Тип\_партнера,

Discount = CalculateDiscount(p.TotalQuantity)

})

.ToList();

PartnersList.ItemsSource = partners;

}

}

catch (Exception ex)

{

MessageBox.Show("Произошла ошибка: " + ex.Message);

}

}

private double CalculateDiscount(int totalQuantity)

{

if (totalQuantity > 300000)

return 15;

if (totalQuantity > 50000)

return 10;

if (totalQuantity >= 10000)

return 5;

return 0;

}

}

}