

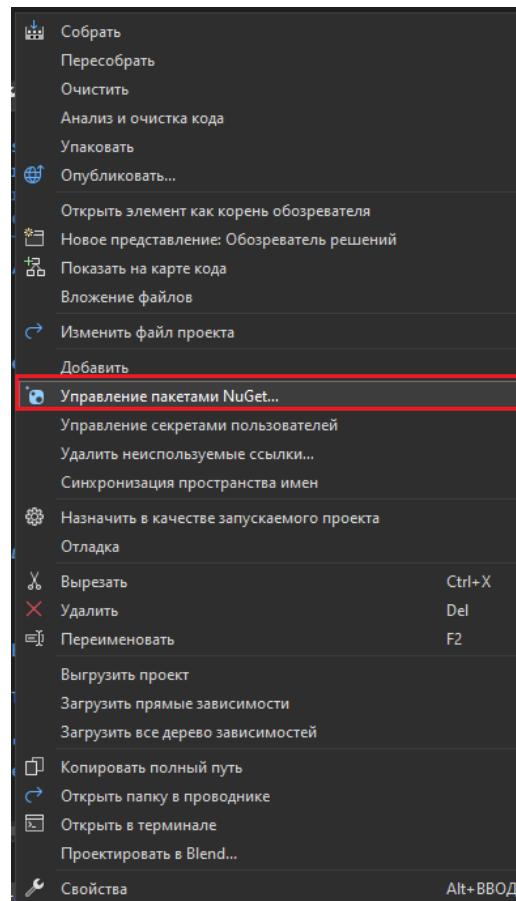
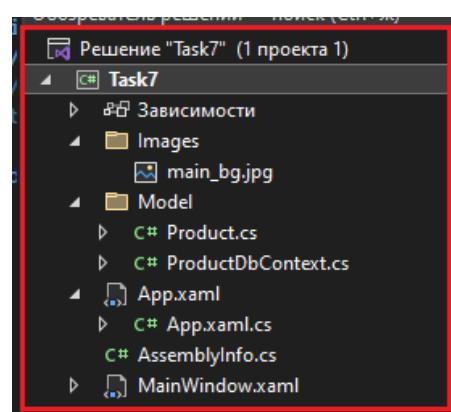
## Практическая работа №7 | Использование EntityFrameworkCore и SQLite

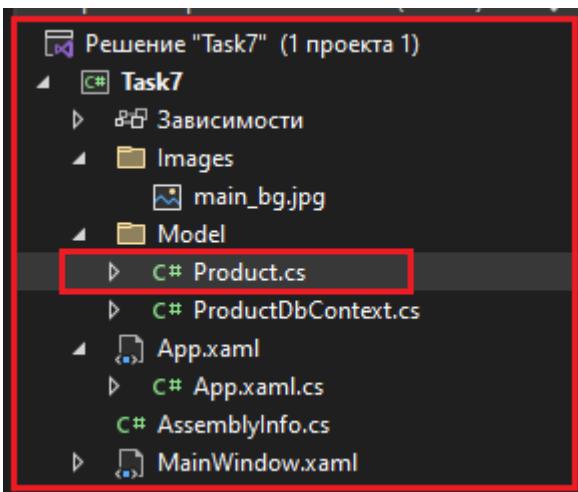
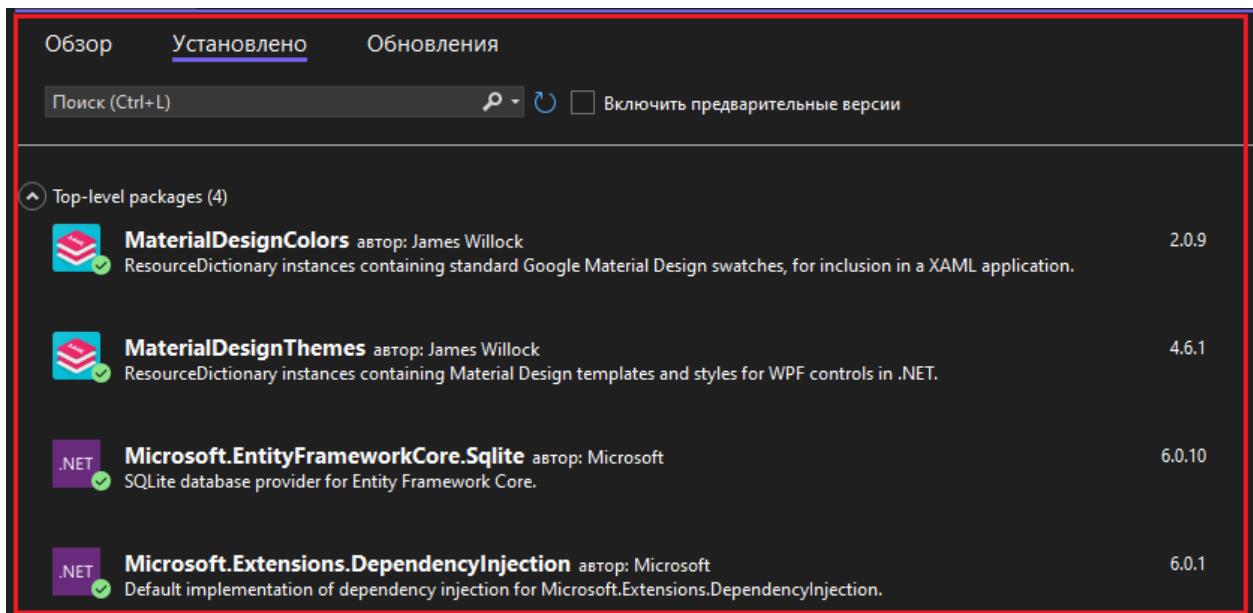
The screenshot shows a Windows application window titled "Практическая работа №7 | Использование EntityFrameworkCore и SQLite". Inside, there is a table listing three products:

#	Название товара	Описание товара
1	Видеокарта KFA2 GeForce RTX 3070 CORE	(LHR) [37NSL6MD2KCK] [PCI-E 4.0, 8 ГБ GDDR6, 256 бит, DisplayPort x3, HDMI, GPU 1500 МГц]
2	Видеокарта GIGABYTE GeForce RTX 3090 TI GAMING	[GV-N309TGAMING-24GD] [PCI-E, 24 ГБ GDDR6X, 384 бит, DisplayPort x3, HDMI, GPU 1560 МГц]
3	Видеокарта Palit GeForce RTX 3060 DUAL OC	(LHR) [NE63060T19K9-190AD] [PCI-E 4.0, 12 ГБ GDDR6, 192 бит, DisplayPort x3, HDMI, GPU 1320 МГц]

Below the table are two modal dialogs:

- Добавить новый товар**: A dialog with fields for Название товара, Описание товара, Цена товара, Кол-во товара, and a green "Добавить" button.
- Редактировать товар**: A dialog with fields for Название товара, Описание товара, Цена товара, Кол-во товара, and a green "Редактировать" button.





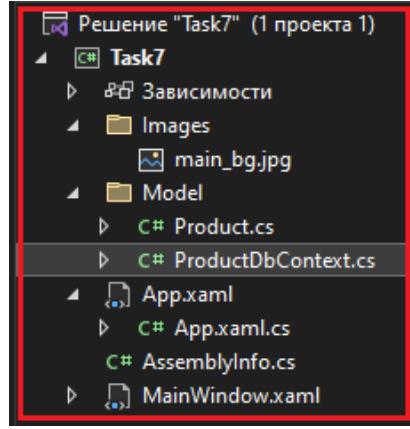
```
public class Product
{
    public int ProductID { get; set; }

    public string ProductName { get; set; }

    public string ProductDescription { get; set; }

    public double ProductPrice { get; set; }

    public int ProductUnit { get; set; }
}
```

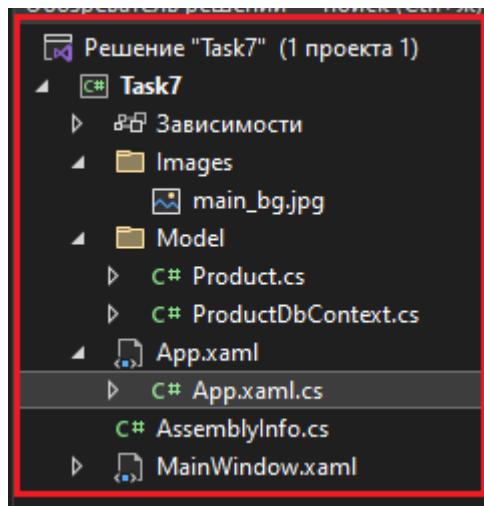


```
public class ProductDbContext : DbContext
{
    #region Конструктор
    Ссылок: 0
    public ProductDbContext(DbContextOptions<ProductDbContext> options)
        :base(options)
    {
        Database.EnsureCreated();
    }
    #endregion

    #region Public свойства
    Ссылок: 3
    public DbSet<Product> Products { get; set; }
    #endregion

    #region Методы
    Ссылок: 0
    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {
        modelBuilder.Entity<Product>().HasData(GetProducts());
        base.OnModelCreating(modelBuilder);
    }
}
```

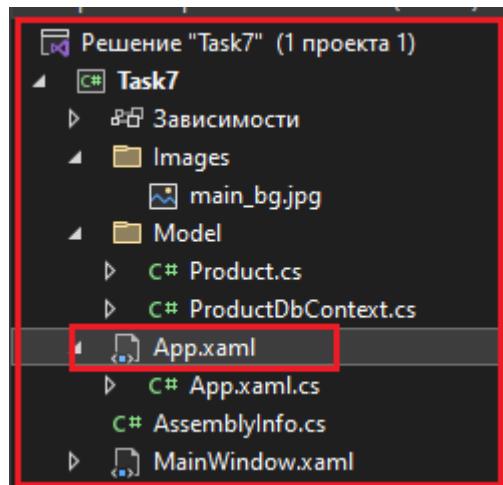
```
Ссылок: 1
private Product[] GetProducts() => new Product[]
{
    new Product{ProductID = 1,
                ProductName = "Видеокарта KFA2 GeForce RTX 3070 CORE",
                ProductDescription = "(LHR) [37NSL6MD2KCK] [PCI-E 4.0, 8 ГБ GDDR6, 256 бит]",
                ProductPrice = 49799,
                ProductUnit = 20},
    new Product{ProductID = 2,
                ProductName = "Видеокарта GIGABYTE GeForce RTX 3090 TI GAMING",
                ProductDescription = "[GV-N309TGAMING-24GD] [PCI-E, 24 ГБ GDDR6X, 384 бит]",
                ProductPrice = 99999,
                ProductUnit = 18},
    new Product{ProductID = 3,
                ProductName = "Видеокарта Palit GeForce RTX 3060 DUAL OC",
                ProductDescription = "(LHR) [NE63060T19K9-190AD] [PCI-E 4.0, 12 ГБ GDDR6, 192 бит]",
                ProductPrice = 38999,
                ProductUnit = 40},
    new Product{ProductID = 4,
                ProductName = "Видеокарта GIGABYTE GeForce RTX 3070 Ti GAMING OC",
                ProductDescription = "[GV-N307TGAMING OC-8GD] [PCI-E 4.0, 8 ГБ GDDR6X, 256 бит]",
                ProductPrice = 70999,
                ProductUnit = 52}
};
```



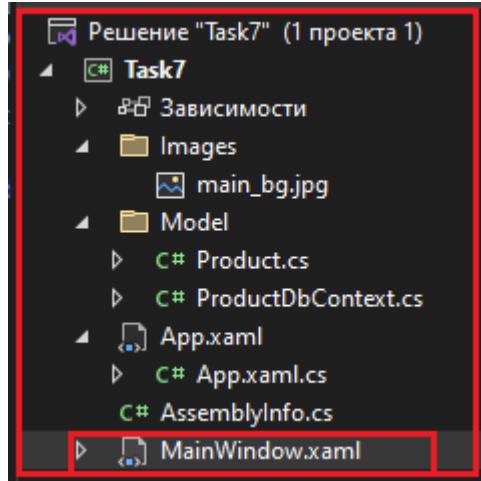
```
Ссылок: 4
public partial class App : Application
{
    private readonly ServiceProvider serviceProvider;

    Ссылок: 1
    public App()
    {
        ServiceCollection services = new ServiceCollection();
        services.AddDbContext<ProductDbContext>(p =>
        {
            p.UseSqlite("Data Source = Task7.db");
        });
        services.AddSingleton<MainWindow>();
        serviceProvider = services.BuildServiceProvider();
    }

    Ссылок: 1
    private void OnStartUp(object s, StartupEventArgs e)
    {
        var mainWindow = serviceProvider.GetService<MainWindow>();
        mainWindow.Show();
    }
}
```



```
<Application x:Class="Task7.App"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:local="clr-namespace:Task7"
    xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes"
    Startup="OnStartUp">
</Application>
```



```
<Window x:Class="Task7.MainWindow"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:local="clr-namespace:Task7"
    xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes"
    Height="500"
    Width="1200"
    WindowStyle="None"
    WindowStartupLocation="CenterScreen"
    ResizeMode="NoResize"
    Background="Transparent"
    AllowsTransparency="True"
    FontFamily="Consolas">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="45"/>
            <RowDefinition/>
            <RowDefinition/>
        </Grid.RowDefinitions>
        <Image Source="Images/main_bg.jpg"
            Grid.RowSpan="3"
            Stretch="UniformToFill"/>
        <Label FontSize="18"
            HorizontalAlignment="Center"
            VerticalAlignment="Center"
            Margin="5"
            Grid.Row="0"
            Content="Практическая работа №7 | Использование EntityFrameworkCore и SQLite"
            Foreground="White"/>
    </Grid>
</Window>
```

```
<DataGrid x:Name="ProductDG"
    Grid.Row="1"
    Margin="5"
    AutoGenerateColumns="False"
    IsReadOnly="True"
    CanUserSortColumns="False"
    Background="#LightGray"
    RowBackground="White"
    AlternatingRowBackground="#FFF0F0"
    GridLinesVisibility="Horizontal"
    ColumnHeaderHeight="50"
    CanUserAddRows="False"
    CanUserDeleteRows="False"
    CanUserResizeRows="False"
    Effect="{StaticResource MaterialDesignElevationShadow16}">
<DataGrid.Columns>
    <DataGridTextColumn Width="Auto"
        Header="#""
        Binding="{Binding ProductID}"/>
    <DataGridTextColumn Width="Auto"
        Header="Название товара"
        Binding="{Binding ProductName}"/>
    <DataGridTextColumn Width="Auto"
        Header="Описание товара"
        Binding="{Binding ProductDescription}"/>
    <DataGridTextColumn Width="Auto"
        Header="Цена товара"
        Binding="{Binding ProductPrice}"/>
    <DataGridTextColumn Width="Auto"
        Header="Кол-во товара"
        Binding="{Binding ProductUnit}"/>
```

```
<DataGridTemplateColumn Header="Редактировать товар">
    <DataGridTemplateColumn.CellTemplate>
        <DataTemplate>
            <Button Click="BtnSelectProductToEdit"
                Foreground="White">
                <StackPanel Orientation="Horizontal">
                    <materialDesign:PackIcon Kind="EventEdit"
                        Margin="0 2 3 0"/>
                    <TextBlock Text="Редактировать"/>
                </StackPanel>
            </Button>
        </DataTemplate>
    </DataGridTemplateColumn.CellTemplate>
</DataGridTemplateColumn>

<DataGridTemplateColumn Header="Удалить товар">
    <DataGridTemplateColumn.CellTemplate>
        <DataTemplate>
            <Button Click="BtnDeleteProduct"
                Foreground="White">
                <StackPanel Orientation="Horizontal">
                    <materialDesign:PackIcon Kind="Remove"
                        Margin="0 0 3 0"/>
                    <TextBlock Text="Удалить"/>
                </StackPanel>
            </Button>
        </DataTemplate>
    </DataGridTemplateColumn.CellTemplate>
</DataGridTemplateColumn>
</DataGrid.Columns>
</DataGrid>
```

```
<Grid Grid.Row="2">
    <Grid.ColumnDefinitions>
        <ColumnDefinition/>
        <ColumnDefinition/>
    </Grid.ColumnDefinitions>

    <Border Grid.Column="0"
            Margin="5"
            BorderThickness="1"
            BorderBrush="#White">
        <StackPanel Margin="5">
            <Label Content="Добавить новый товар"
                  FontWeight="Bold"
                  Foreground="#White"
                  HorizontalAlignment="Center"
                  VerticalAlignment="Center" Margin="5"/>

            <Grid Name="NewProductGrid">
                <Grid.RowDefinitions>
                    <RowDefinition/>
                    <RowDefinition/>
                    <RowDefinition/>
                    <RowDefinition/>
                    <RowDefinition/>
                </Grid.RowDefinitions>

                <Grid.ColumnDefinitions>
                    <ColumnDefinition/>
                    <ColumnDefinition/>
                </Grid.ColumnDefinitions>
            </Grid>
        </StackPanel>
    </Border>
</Grid>
```

```
<Label Grid.Row="0"
       Grid.Column="0"
       Content="Название товара"
       Foreground="#White"/>
<TextBox Grid.Row="0"
         Grid.Column="1"
         Text="{Binding ProductName}"
         Foreground="#White"
         TextWrapping="Wrap"/>

<Label Grid.Row="1"
       Grid.Column="0"
       Content="Описание товара"
       Foreground="#White"/>
<TextBox Grid.Row="1"
         Grid.Column="2"
         Text="{Binding ProductDescription}"
         Foreground="#White"
         TextWrapping="Wrap"/>

<Label Grid.Row="2"
       Grid.Column="0"
       Content="Цена товара"
       Foreground="#White"/>
<TextBox Grid.Row="2"
         Grid.Column="3"
         Text="{Binding ProductPrice}"
         Foreground="#White"/>
```

```
<Label Grid.Row="3"
       Grid.Column="0"
       Content="Кол-во товара"
       Foreground="White"/>
<TextBox Grid.Row="3"
         Grid.Column="4"
         Text="{Binding ProductUnit}"
         Foreground="White"/>

<Button Grid.Row="4"
        Grid.ColumnSpan="2"
        Width="150"
        Margin="5"
        Click="BtnAddItem"
        HorizontalAlignment="Center"
        VerticalAlignment="Center"
        Foreground="White"
        Style="{StaticResource MaterialDesignRaisedButton}"
        materialDesign:ButtonProgressAssist.Value="-1"
        materialDesign:ButtonProgressAssist.IsIndicatorVisible="True"
        materialDesign:ButtonProgressAssist.IsIndeterminate="True"
        IsEnabled="{Binding DataContext.ControlsEnabled,
                    RelativeSource={RelativeSource FindAncestor,
                    AncestorType=Window}}">
    <StackPanel Orientation="Horizontal">
        <materialDesign:PackIcon Kind="DatabaseAdd"
                                  Margin="0 2 3 0"/>
        <TextBlock Text="Добавить"/>
    </StackPanel>
</Button>
</Grid>
</StackPanel>
</Border>
```

```
<Border Grid.Column="1"
        Margin="5"
        BorderThickness="1"
        BorderBrush="White">
    <StackPanel Margin="5">
        <Label Content="Редактировать товар"
              FontWeight="Bold"
              HorizontalAlignment="Center"
              VerticalAlignment="Center"
              Margin="5"
              Foreground="White"/>

        <Grid Name="UpdateProductGrid">
            <Grid.RowDefinitions>
                <RowDefinition/>
                <RowDefinition/>
                <RowDefinition/>
                <RowDefinition/>
                <RowDefinition/>
            </Grid.RowDefinitions>

            <Grid.ColumnDefinitions>
                <ColumnDefinition/>
                <ColumnDefinition/>
            </Grid.ColumnDefinitions>
        </Grid>
    </StackPanel>
</Border>
```

```
<Label Grid.Row="0"
       Grid.Column="0"
       Content="Название товара"
       Foreground="White"/>
<TextBox Grid.Row="0"
         Grid.Column="1"
         Text="{Binding ProductName}"
         Foreground="White"
         TextWrapping="Wrap"/>

<Label Grid.Row="1"
       Grid.Column="0"
       Content="Описание товара"
       Foreground="White"/>
<TextBox Grid.Row="1"
         Grid.Column="2"
         Text="{Binding ProductDescription}"
         Foreground="White"
         TextWrapping="Wrap"/>

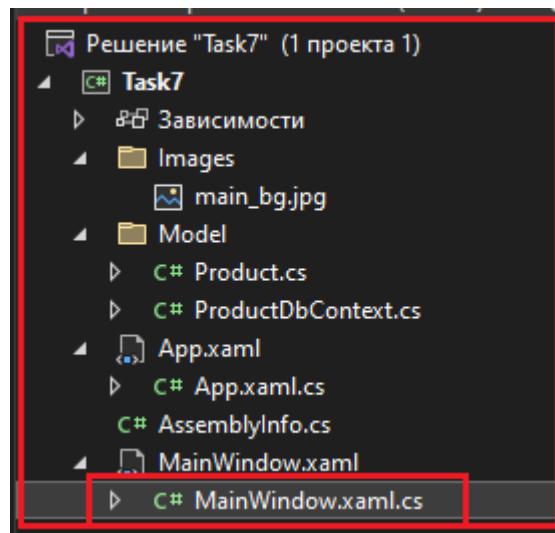
<Label Grid.Row="2"
       Grid.Column="0"
       Content="Цена товара"
       Foreground="White"/>
<TextBox Grid.Row="2"
         Grid.Column="3"
         Text="{Binding ProductPrice}"
         Foreground="White"/>
```

```
<TextBox Grid.Row="2"
         Grid.Column="3"
         Text="{Binding ProductPrice}"
         Foreground="White"/>

<Label Grid.Row="3"
       Grid.Column="0"
       Content="Кол-во товара"
       Foreground="White"/>
<TextBox Grid.Row="3"
         Grid.Column="4"
         Text="{Binding ProductUnit}"
         Foreground="White"/>

<Button Grid.Row="4"
        Grid.Column="2"
        Width="150"
        Click="BtnEditItem"
        Margin="5"
        HorizontalAlignment="Center"
        VerticalAlignment="Center"
        Foreground="White"
        Style="{StaticResource MaterialDesignRaisedButton}"
        materialDesign:ButtonProgressAssist.Value="-1"
        materialDesign:ButtonProgressAssist.IsIndicatorVisible="True"
        materialDesign:ButtonProgressAssist.IsIndeterminate="True"
        IsEnabled="{Binding DataContext.ControlsEnabled,
                    RelativeSource={RelativeSource FindAncestor,
                    AncestorType=Window}}">
```

```
<StackPanel Orientation="Horizontal">
    <materialDesign:PackIcon Kind="EventEdit"
        Margin="0 2 3 0"/>
    <TextBlock Text="Редактировать"/>
</StackPanel>
</Button>
</Grid>
</StackPanel>
</Border>
</Grid>
</Window>
```



```
ProductDbContext context;
Product NewProduct = new Product();
Product selectedProduct = new Product();
Ссылок: 0
public MainWindow(ProductDbContext context)
{
    InitializeComponent();
    this.context = context;
    InitializeComponent();
    GetProducts();
    NewProductGrid.DataContext = NewProduct;
}
Ссылок: 4
private void GetProducts()
{
    ProductDG.ItemsSource = context.Products.ToList();
}
Ссылок: 1
private void BtnSelectProductToEdit(object sender, RoutedEventArgs e)
{
    selectedProduct = (sender as FrameworkElement).DataContext as Product;
    UpdateProductGrid.DataContext = selectedProduct;
}
```

```
private void BtnDeleteProduct(object sender, RoutedEventArgs e)
{
    var productToDelete = (sender as FrameworkElement).DataContext as Product;
    context.Products.Remove(productToDelete);
    context.SaveChanges();
    GetProducts();
}

Ссылка 1
private void BtnAddItem(object sender, RoutedEventArgs e)
{
    context.Products.Add(NewProduct);
    context.SaveChanges();
    GetProducts();
    NewProduct = new Product();
    NewProductGrid.DataContext = NewProduct;
}

Ссылка 1
private void BtnEditItem(object sender, RoutedEventArgs e)
{
    context.Update(selectedProduct);
    context.SaveChanges();
    GetProducts();
}
```