Ростовская Елизавета, ПР23/2

Лабораторная работа №1

Тема: «Разработка структура проекта»

Цель: научиться разрабатывать структуру проекта, получить опыт разработки приложения для управления покупками в супермаркете C# с использованием Visual Studio.

Ход работы

1. Создайте базу данных «supermarket».

2. В базе данных «supermarket» создайте таблицу со следующей структурой:

CREATE TABLE products (

id INT PRIMARY KEY,

name NVARCHAR(100),

price DECIMAL(18, 2)

);

3. Добавьте данные в таблицу:

INSERT INTO products (id, name, price) VALUES (1, 'Яблоко', 20.00);

INSERT INTO products (id, name, price) VALUES (2, 'Банан', 15.00);

INSERT INTO products (id, name, price) VALUES (3, 'Молоко', 30.00);

4. Полученная таблица выглядит следующим образом:

5. Создайте новый проект SupermarketApp в Visual Studio. Переименуйте форму Form1.cs в SpisokProducts.cs. Добавьте все необходимые элементы на форму.

6. Создать выпадающий список из продуктов, которые находятся в базе данных «supermarket», в таблице «products» (id, name, price). Пользователь выбирает продукт из выпадающего списка и нажимает кнопку «Добавить». Продукт добавляется в ListBox. Можно добавить еще несколько продуктов или очистить ListBox с помощью кнопки «Очистить». После того как один продукт или несколько продуктов добавились в ListBox, пользователь нажимает на кнопку «Посчитать итог». Считается итоговая цена за всю покупку и выводится в TextBox.

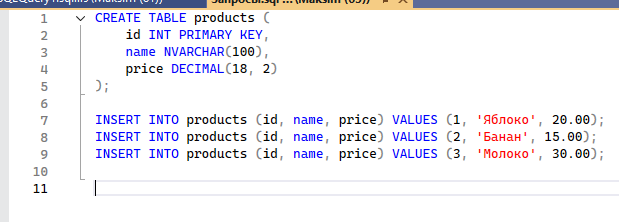


Рис. 1 – Запросы SQL

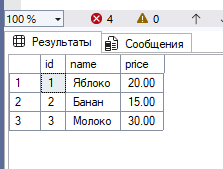


Рис. 2 – Таблица «Продукты»

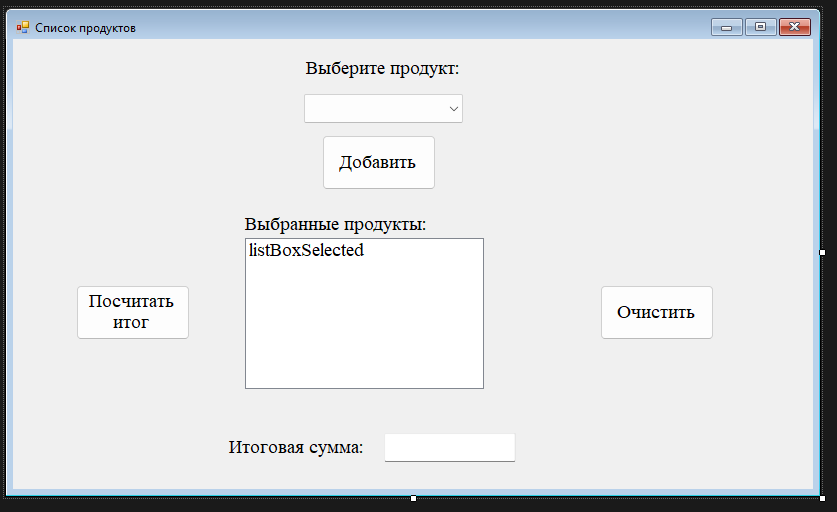


Рис. 3 – Макет формы «Список продуктов»

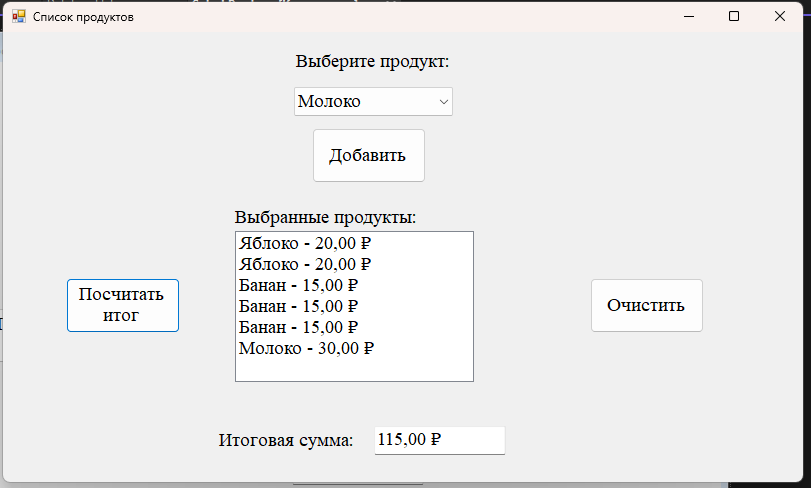


Рис 4. – Результат работы формы

Код формы «Список продуктов»:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Windows.Forms;

namespace PR1

{

public partial class SpisokProducts : Form

{

private List<Product> selectedProducts = new List<Product>();

private DatabaseHelper dbHelper = new DatabaseHelper();

public SpisokProducts()

{

InitializeComponent();

loadProducts();

}

private void buttonAdd\_Click(object sender, EventArgs e)

{

if (comboBoxProducts.SelectedItem != null)

{

Product selectedProduct = (Product)comboBoxProducts.SelectedItem;

selectedProducts.Add(selectedProduct);

listBoxSelected.Items.Add(selectedProduct.ToString());

}

}

private void loadProducts()

{

try

{

var products = dbHelper.GetProducts();

comboBoxProducts.DataSource = products;

comboBoxProducts.DisplayMember = "Name";

comboBoxProducts.ValueMember = "Id";

}

catch (Exception ex)

{

MessageBox.Show($"Ошибка загрузки продуктов: {ex.Message}", "Ошибка",

MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void buttonClear\_Click(object sender, EventArgs e)

{

selectedProducts.Clear();

listBoxSelected.Items.Clear();

textBoxTotal.Clear();

}

private void buttonCalculate\_Click(object sender, EventArgs e)

{

decimal total = selectedProducts.Sum(p => p.Price);

textBoxTotal.Text = total.ToString("C");

}

}

}