

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

Исходное задание

8. В базе book.dat хранятся записи, содержащие: номер книги, наименование книги, автор, год издания, цена 1 экземпляра.

Обеспечить выдачу следующих справок:

- по автору книги найти все его книги;
- суммарную стоимость книг, выпущенных в каждом году;
- выдать список книг, изданных в указанном году;
- найти книги по фамилии автора, году издания.

Листинг программы

Book.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace v8_books
{
    // Класс для представления данных о книге
    public class Book
    {
        // Уникальный идентификатор книги
        public int Id { get; set; }

        // Название книги
        public string Title { get; set; }

        // Автор книги
        public string Author { get; set; }

        // Год издания книги
        public int Year { get; set; }

        // Цена книги
        public decimal Price { get; set; }

        // Конструктор класса для создания объекта с заданными параметрами
        public Book(int id, string title, string author, int year, decimal price)
        {
            // Инициализация свойств объекта
            Id = id;
            Title = title;
            Author = author;
            Year = year;
            Price = price;
        }
    }
}
```

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
}
```

Form1.cs

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

namespace v8_books
{
    public partial class Form1 : Form
    {
        // Список для хранения объектов класса Book
        private List<Book> books = new List<Book>();

        // Путь к файлу book.dat (относительный путь)
        private string filePath = "../book.dat";

        public Form1()
        {
            InitializeComponent();

            // Загрузка данных из файла при запуске приложения
            LoadData();
        }

        // Метод для загрузки данных из файла book.dat
        private void LoadData()
        {
            // Очистка списка books и DataGridView перед загрузкой новых данных
            books.Clear();
            dataGridView1.Rows.Clear();

            // Проверка существования файла book.dat
            if (File.Exists(filePath))
            {
                // Чтение всех строк из файла
                string[] lines = File.ReadAllLines(filePath);

                // Обработка каждой строки из файла
                foreach (string line in lines)
                {
                    // Разделение строки на части по запятой
                    string[] parts = line.Split(',');

                    // Проверка, что строка содержит необходимое количество частей (5)
                    if (parts.Length == 5)
                    {
                        // Попытка преобразования частей строки в соответствующие типы
                        if (int.TryParse(parts[0], out int id) &&
```

данных

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
        !string.IsNullOrEmpty(parts[1]) &&
        !string.IsNullOrEmpty(parts[2]) &&
        int.TryParse(parts[3], out int year) &&
        decimal.TryParse(parts[4], out decimal price))
    {
        // Получение названия книги и автора из частей строки
        string title = parts[1];
        string author = parts[2];

        // Создание объекта класса Book
        Book book = new Book(id, title, author, year, price);

        // Добавление книги в список books
        books.Add(book);

        // Добавление строки в DataGridView
        dataGridView1.Rows.Add(book.Id, book.Title, book.Author,
book.Year, book.Price);
    }
    else
    {
        // Вывод сообщения об ошибке, если данные в строке
некорректны
        MessageBox.Show($"Ошибка чтения строки: '{line}'. Данные
некорректны.", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    }
    else
    {
        // Вывод сообщения об ошибке, если в строке недостаточно данных
данных.", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    }
    }
    else
    {
        // Вывод сообщения об ошибке, если файл book.dat не найден
        MessageBox.Show("Файл book.dat не найден.", "Ошибка",
MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

// Метод для сохранения данных в файл book.dat
private void SaveData()
{
    // Создание нового списка строк для записи в файл
    List<string> lines = new List<string>();

    // Формирование строк для каждой книги в списке books
    foreach (Book book in books)
    {
        string line =
        $"{book.Id},{book.Title},{book.Author},{book.Year},{book.Price}";
        lines.Add(line);
    }

    // Запись списка строк в файл book.dat
```

©МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
File.WriteAllLines(filePath, lines);
}

// Обработчик события нажатия кнопки "Открыть"
private void button_open_Click(object sender, EventArgs e)
{
    // Вызов метода LoadData() для загрузки данных из файла
    LoadData();
}

// Обработчик события нажатия кнопки "Сохранить"
private void button_save_Click(object sender, EventArgs e)
{
    // Вызов метода SaveData() для сохранения данных в файл
    SaveData();
}

// Обработчик события нажатия кнопки "Добавить"
private void button_add_Click(object sender, EventArgs e)
{
    // Создание нового экземпляра формы AddBookForm
    AddBookForm addForm = new AddBookForm();

    // Отображение формы AddBookForm в режиме диалога
    if (addForm.ShowDialog() == DialogResult.OK)
    {
        // Получение объекта класса Book из формы AddBookForm
        Book book = addForm.Book;

        // Добавление новой книги в список books
        books.Add(book);

        // Добавление новой строки в DataGridView
        dataGridView1.Rows.Add(book.Id, book.Title, book.Author, book.Year,
book.Price);
    }
}

// Обработчик события нажатия кнопки "Изменить"
private void button_edit_Click(object sender, EventArgs e)
{
    // Проверка, что выбрана хотя бы одна ячейка в DataGridView
    if (dataGridView1.SelectedCells.Count > 0)
    {
        // Получение индекса выбранной строки
        int selectedRowIndex = dataGridView1.SelectedCells[0].RowIndex;

        // Получение объекта класса Book из списка books по индексу выбранной
строки
        Book book = books[selectedRowIndex];

        // Создание нового экземпляра формы AddBookForm и передача в нее объекта
класса Book
        AddBookForm editForm = new AddBookForm(book);

        // Отображение формы AddBookForm в режиме диалога
        if (editForm.ShowDialog() == DialogResult.OK)
        {

```

©МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
// Получение измененного объекта класса Book из формы AddBookForm
Book updatedBook = editForm.Book;

// Замена старого объекта Book в списке books на обновленный
books[selectedRowIndex] = updatedBook;

// Обновление данных в выбранной строке DataGridView
dataGridView1.Rows[selectedRowIndex].SetValues(updatedBook.Id,
updatedBook.Title, updatedBook.Author, updatedBook.Year, updatedBook.Price);
    }
}
else
{
    // Вывод сообщения об ошибке, если не выбрана ни одна ячейка в
DataGridView
    MessageBox.Show("Пожалуйста, выберите строку для редактирования.",
"Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Warning);
}
}

// Обработчик события нажатия кнопки "Удалить"
private void button_delete_Click(object sender, EventArgs e)
{
    // Проверка, что выбрана хотя бы одна ячейка в DataGridView
    if (dataGridView1.SelectedCells.Count > 0)
    {
        // Получение индекса выбранной строки
        int selectedRowIndex = dataGridView1.SelectedCells[0].RowIndex;

        // Удаление книги из списка books по индексу выбранной строки
        books.RemoveAt(selectedRowIndex);

        // Удаление строки из DataGridView
        dataGridView1.Rows.RemoveAt(selectedRowIndex);
    }
    else
    {
        // Вывод сообщения об ошибке, если не выбрана ни одна ячейка в
DataGridView
        MessageBox.Show("Пожалуйста, выберите строку для удаления.", "Ошибка",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }
}

// Обработчик события нажатия кнопки "Поиск по автору"
private void button_byAuthor_Click(object sender, EventArgs e)
{
    // Отображение диалогового окна для ввода автора
    string author = Microsoft.VisualBasic.Interaction.InputBox("Введите автора:",
"Поиск по автору");

    // Проверка, что введено непустое значение
    if (!string.IsNullOrEmpty(author))
    {
        // Получение списка книг, автор которых содержит введенное значение (без
учета регистра)
        List<Book> booksByAuthor = books.Where(b =>
b.Author.ToLower().Contains(author.ToLower())).ToList();
    }
}
```

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
// Проверка, что найдены книги по автору
if (booksByAuthor.Count > 0)
{
    // Вызов метода DisplayBooks для отображения найденных книг
    DisplayBooks(booksByAuthor, $"Книги автора {author}");
}
else
{
    // Вывод информационного сообщения, если книги по автору не найдены
    MessageBox.Show($"Книги автора '{author}' не найдены.", "Информация",
    MessageBoxButtons.OK, MessageBoxIcon.Information);
}
}

// Обработчик события нажатия кнопки "Поиск по году"
private void button_byYear_Click(object sender, EventArgs e)
{
    // Отображение диалогового окна для ввода года издания
    string yearInput = Microsoft.VisualBasic.Interaction.InputBox("Введите год
издания:", "Поиск по году");

    // Проверка корректности введенного года
    if (int.TryParse(yearInput, out int year))
    {
        // Получение списка книг, изданных в указанном году
        List<Book> booksByYear = books.Where(b => b.Year == year).ToList();

        // Проверка, что найдены книги по году издания
        if (booksByYear.Count > 0)
        {
            // Вызов метода DisplayBooks для отображения найденных книг
            DisplayBooks(booksByYear, $"Книги, изданные в {year} году");
        }
        else
        {
            // Вывод информационного сообщения, если книги по году издания не
            найдены
            MessageBox.Show($"Книги, изданные в {year} году, не найдены.",
            "Информация", MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
    }
    else
    {
        // Вывод сообщения об ошибке, если введенный год некорректен
        MessageBox.Show("Пожалуйста, введите корректный год.", "Ошибка",
        MessageBoxButtons.OK, MessageBoxIcon.Warning);
    }
}

// Обработчик события нажатия кнопки "Стоимость по годам"
private void button_totalByYear_Click(object sender, EventArgs e)
{
    // Группировка книг по году издания
    var groupedBooks = books.GroupBy(b => b.Year);

    // Создание StringBuilder для формирования строки результата
```

©МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
StringBuilder result = new StringBuilder();
result.AppendLine("Суммарная стоимость книг по годам:");

// Вычисление суммарной стоимости книг для каждого года и добавление в строку
результата
foreach (var group in groupedBooks)
{
    decimal totalPrice = group.Sum(b => b.Price);
    result.AppendLine($"{group.Key}: {totalPrice:C}");
}

// Отображение окна с результатом
MessageBox.Show(result.ToString(), "Суммарная стоимость книг по годам");
}

// Метод для отображения списка книг в диалоговом окне
private void DisplayBooks(List<Book> books, string title)
{
    // Создание StringBuilder для формирования строки результата
    StringBuilder result = new StringBuilder();
    result.AppendLine(title);

    // Добавление информации о каждой книге в строку результата
    foreach (Book book in books)
    {
        result.AppendLine($"{book.Id}: {book.Title} - {book.Author}
({book.Year}), {book.Price:C}");
    }

    // Отображение окна с результатом
    MessageBox.Show(result.ToString(), title);
}
}
}
```

AddBookForm.cs

```
using System;
using System.Windows.Forms;

namespace v8_books
{
    public partial class AddBookForm : Form
    {
        private Book book;

        public AddBookForm()
        {
            InitializeComponent();
            book = new Book(0, "", "", 0, 0);
        }

        public AddBookForm(Book existingBook)
        {
            InitializeComponent();
            book = existingBook;
            textBox_id.Text = existingBook.Id.ToString();
            textBox_title.Text = existingBook.Title;
        }
    }
}
```

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
        textBox_author.Text = existingBook.Author;
        textBox_year.Text = existingBook.Year.ToString();
        textBox_price.Text = existingBook.Price.ToString();
    }

    public Book Book
    {
        get { return book; }
    }

    private void button_ok_Click(object sender, EventArgs e)
    {
        if (ValidateInputs())
        {
            book.Id = int.Parse(textBox_id.Text);
            book.Title = textBox_title.Text;
            book.Author = textBox_author.Text;
            book.Year = int.Parse(textBox_year.Text);
            book.Price = decimal.Parse(textBox_price.Text);
            DialogResult = DialogResult.OK;
            Close();
        }
    }

    private void button_cancel_Click(object sender, EventArgs e)
    {
        DialogResult = DialogResult.Cancel;
        Close();
    }

    private bool ValidateInputs()
    {
        if (string.IsNullOrWhiteSpace(textBox_id.Text) ||
            string.IsNullOrWhiteSpace(textBox_title.Text) ||
            string.IsNullOrWhiteSpace(textBox_author.Text) ||
            string.IsNullOrWhiteSpace(textBox_year.Text) ||
            string.IsNullOrWhiteSpace(textBox_price.Text))
        {
            MessageBox.Show("Все поля должны быть заполнены.", "Ошибка",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
            return false;
        }

        if (!int.TryParse(textBox_id.Text, out int id) || id <= 0)
        {
            MessageBox.Show("Некорректный ID книги.", "Ошибка", MessageBoxButtons.OK,
                MessageBoxIcon.Error);
            return false;
        }

        if (!int.TryParse(textBox_year.Text, out int year) || year < 1000 || year >
            DateTime.Now.Year)
        {
            MessageBox.Show("Некорректный год издания.", "Ошибка",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
            return false;
        }
    }
}
```


Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

```
        if (!decimal.TryParse(textBox_price.Text, out decimal price) || price <= 0)
        {
            MessageBox.Show("Некорректная цена.", "Ошибка", MessageBoxButtons.OK,
            MessageBoxIcon.Error);
            return false;
        }
        return true;
    }
}
```

www.matburo.ru

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

Скриншоты с демонстрацией выполнения функций программы

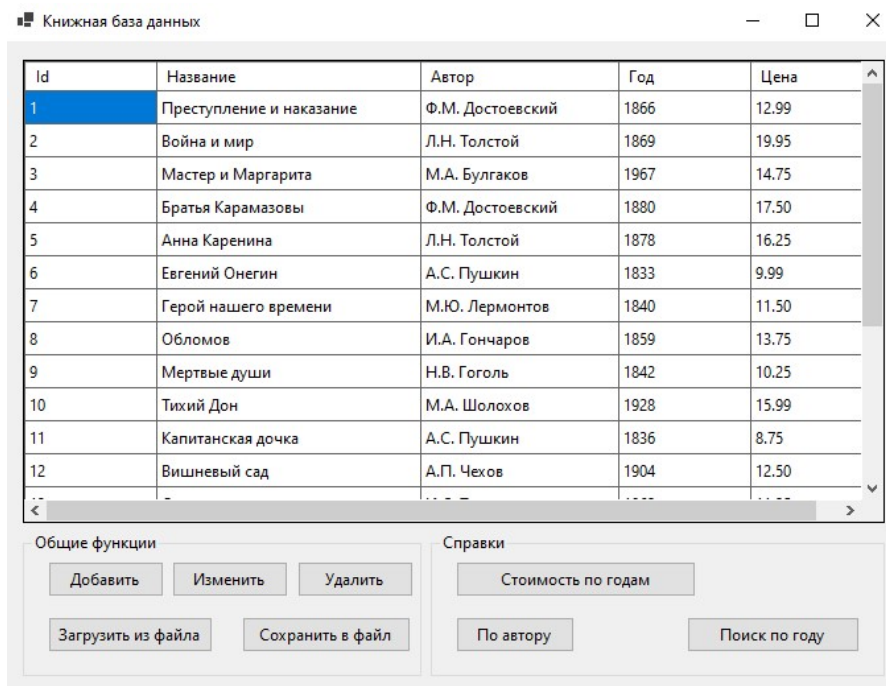
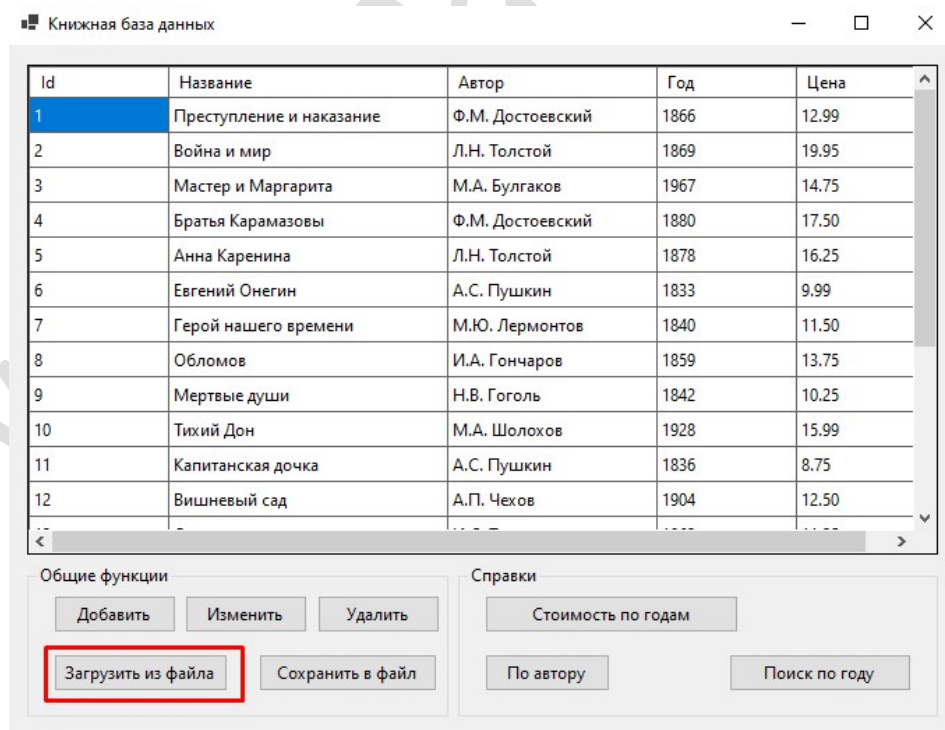


Рисунок 1 – Сразу после запуска



Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

Рисунок 2 – После загрузки из books.dat

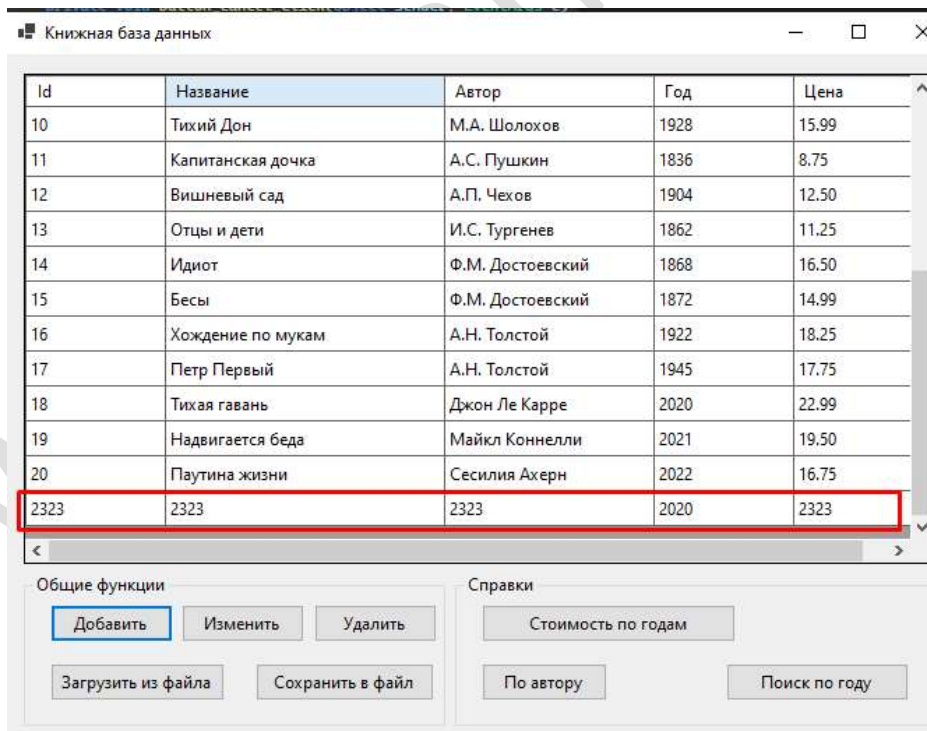
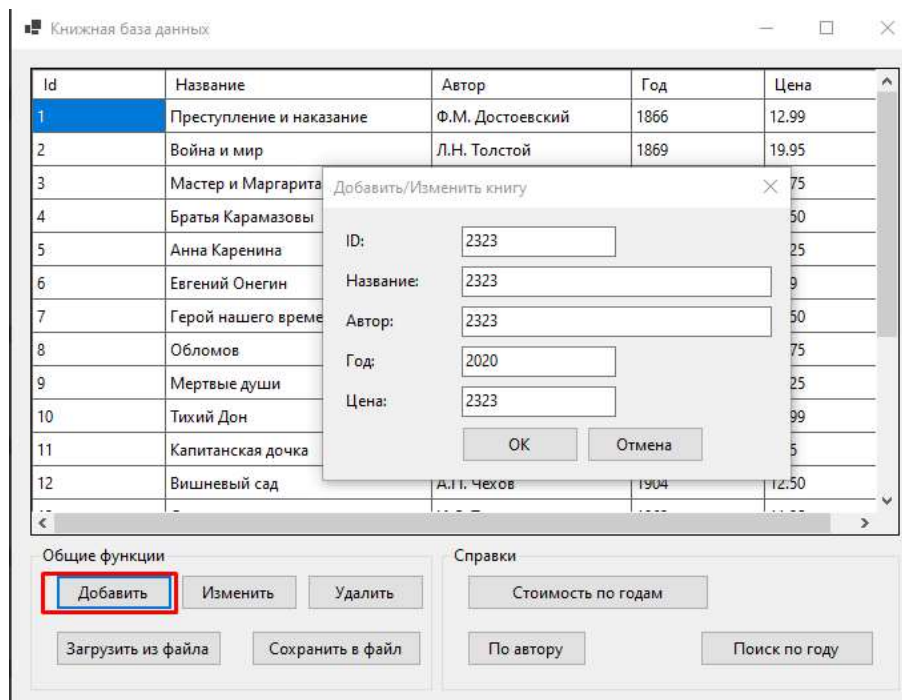


Рисунок 3 – Добавление книги

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

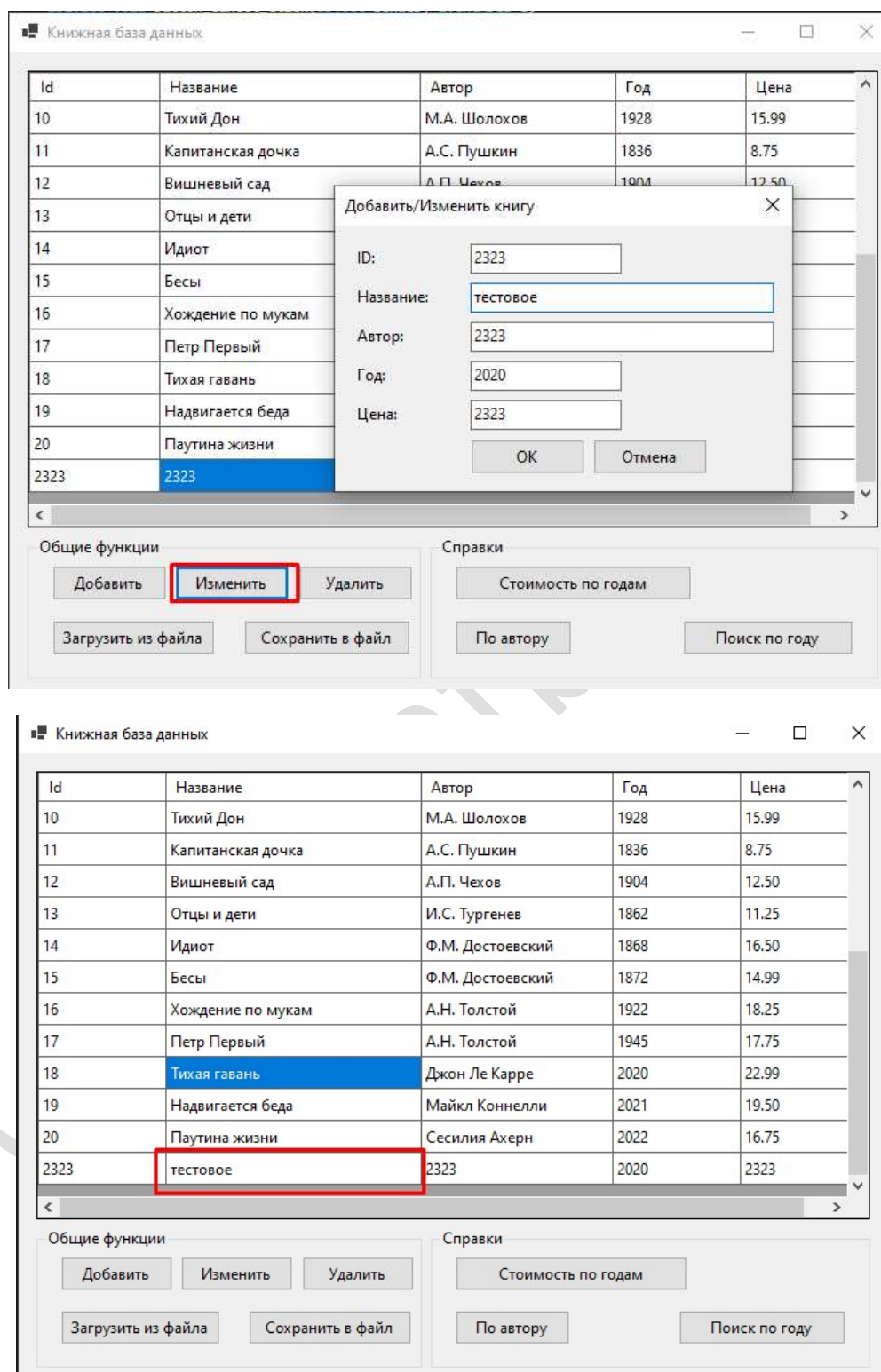


Рисунок 4 – Изменение книги

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

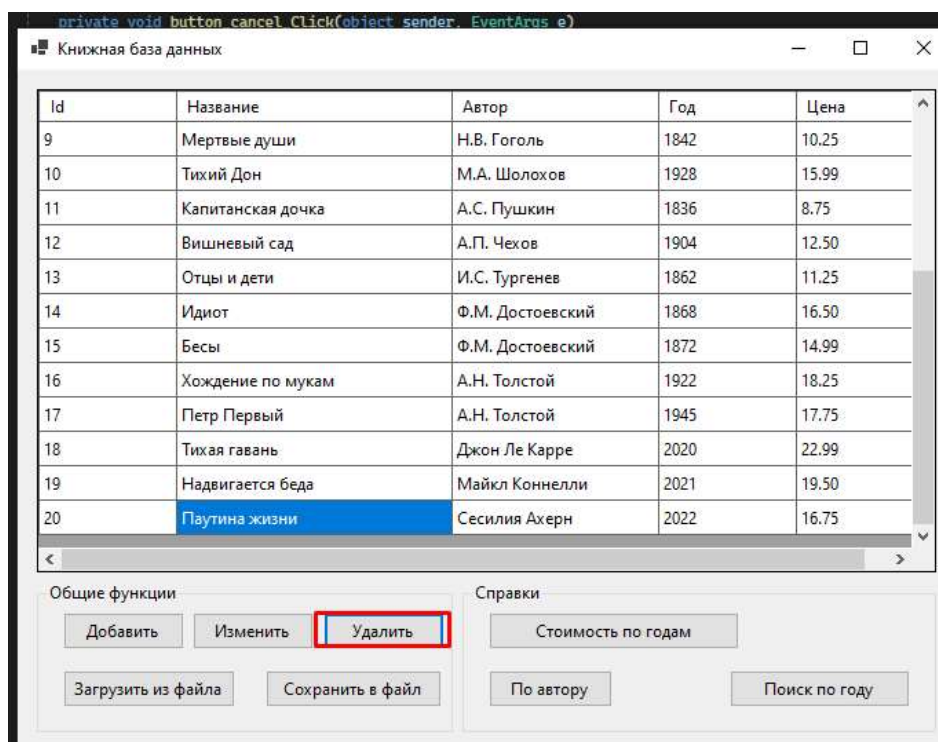


Рисунок 5 – Удаление книги

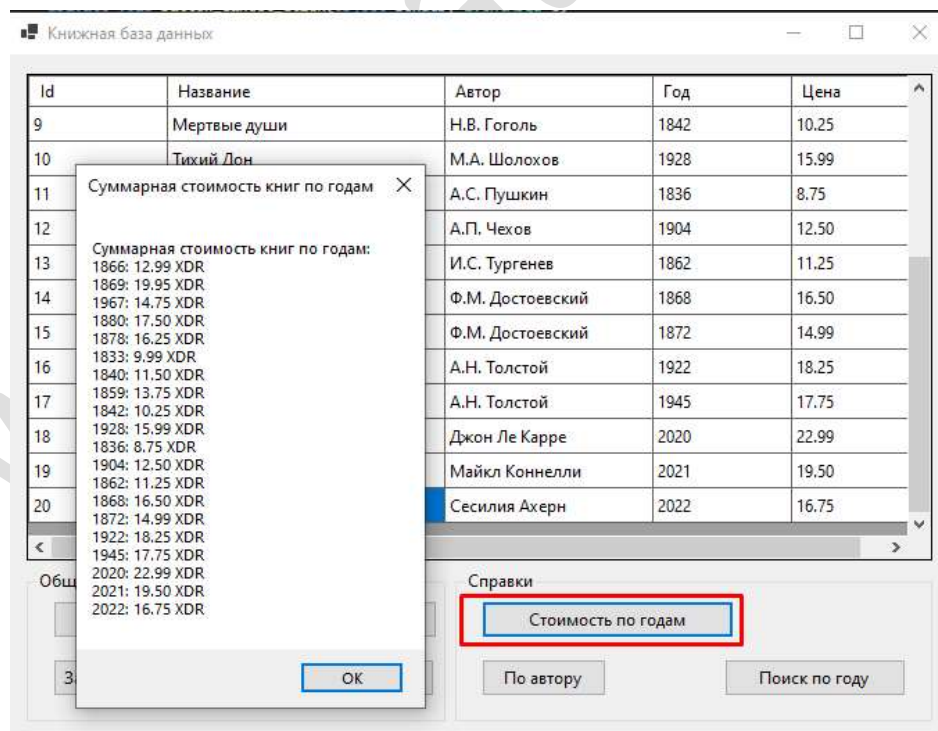


Рисунок 6 – Суммарная стоимость выпущенных книг в каждом году

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

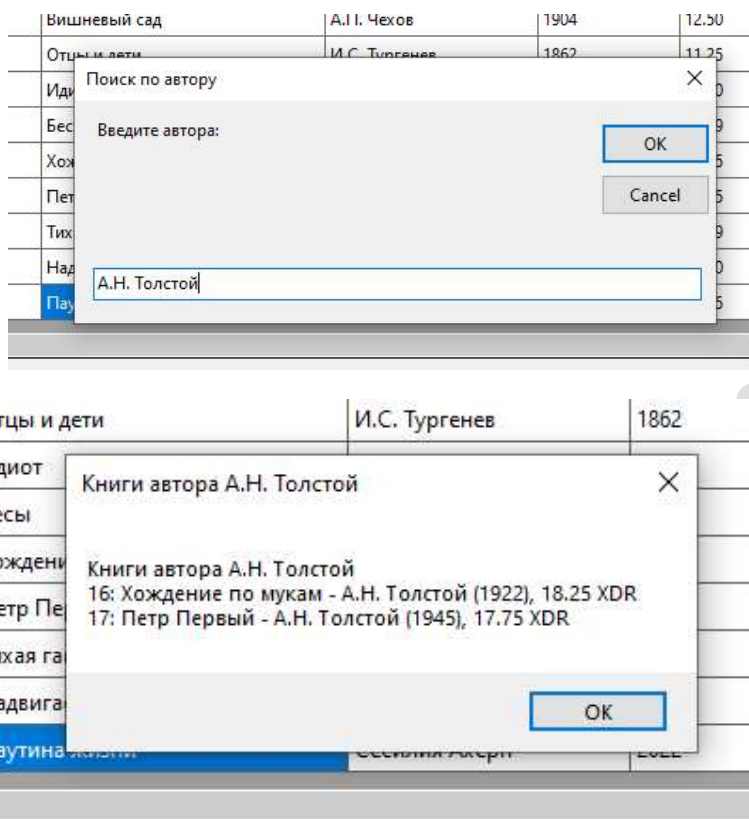
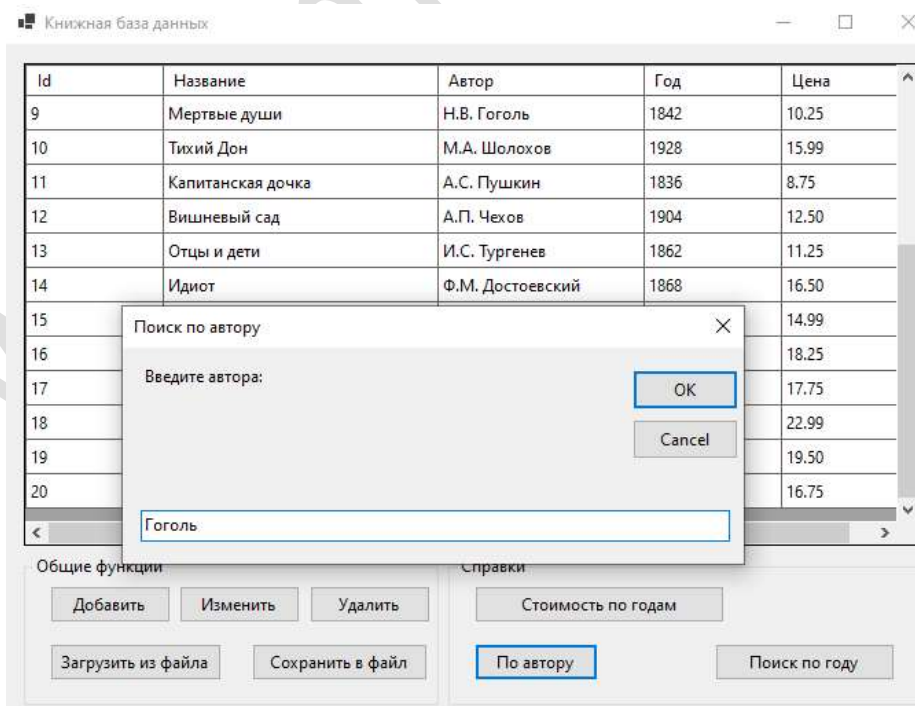


Рисунок 7 – Поиск книг по автору



Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

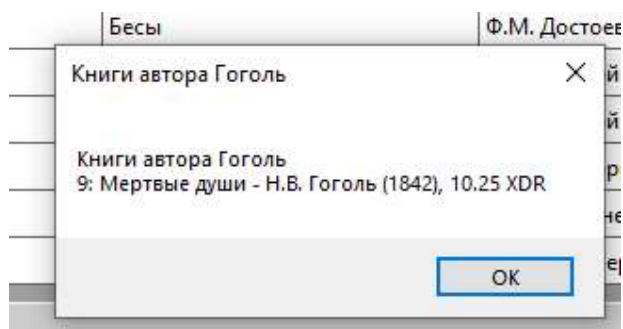


Рисунок 8 – Поиск книг по фамилии автора

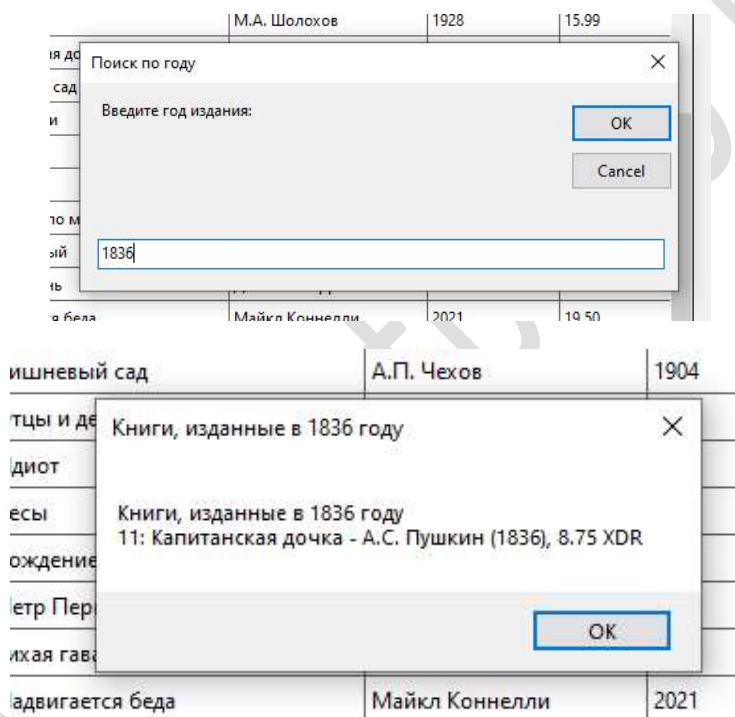


Рисунок 9 – Поиск книг по году издания

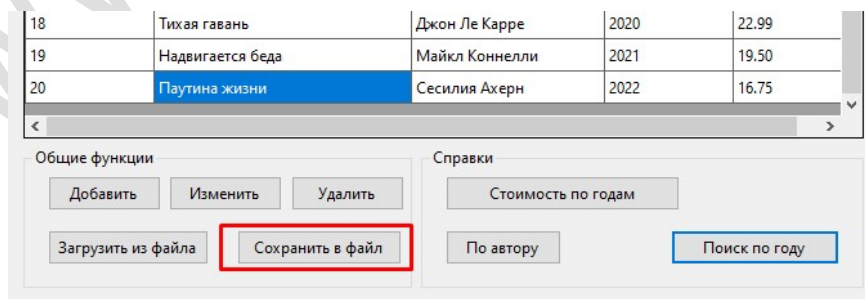


Рисунок 10 – Сохранения dataView в books.dat

Поможем вам с написанием программ: www.matburo.ru/sub_subject.php?p=pz

Содержимое файла book.dat

- 1, Преступление и наказание, Ф.М. Достоевский, 1866, 12.99
- 2, Война и мир, Л.Н. Толстой, 1869, 19.95
- 3, Мастер и Маргарита, М.А. Булгаков, 1967, 14.75
- 4, Братья Карамазовы, Ф.М. Достоевский, 1880, 17.50
- 5, Анна Каренина, Л.Н. Толстой, 1878, 16.25
- 6, Евгений Онегин, А.С. Пушкин, 1833, 9.99
- 7, Герой нашего времени, М.Ю. Лермонтов, 1840, 11.50
- 8, Обломов, И.А. Гончаров, 1859, 13.75
- 9, Мертвые души, Н.В. Гоголь, 1842, 10.25
- 10, Тихий Дон, М.А. Шолохов, 1928, 15.99
- 11, Капитанская дочка, А.С. Пушкин, 1836, 8.75
- 12, Вишневый сад, А.П. Чехов, 1904, 12.50
- 13, Отцы и дети, И.С. Тургенев, 1862, 11.25
- 14, Идиот, Ф.М. Достоевский, 1868, 16.50
- 15, Бесы, Ф.М. Достоевский, 1872, 14.99
- 16, Хождение по мукам, А.Н. Толстой, 1922, 18.25
- 17, Петр Первый, А.Н. Толстой, 1945, 17.75
- 18, Тихая гавань, Джон Ле Карре, 2020, 22.99
- 19, Надвигается беда, Майкл Коннелли, 2021, 19.50
- 20, Паутина жизни, Сесилия Ахерн, 2022, 16.75