

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра АСУ

ОТЧЕТ

по лабораторной работе № 0

по дисциплине: “Инженерия программного обеспечения”

на тему: “Создание приложения .NET с использованием привязки данных
средствами Windows Forms”

Вариант №8

Выполнил:

студент группы ИС-186

Терещенко В.О.

Проверила:

Землянская С. Ю.

Донецк – 2021

Цель работы: закрепить знания по ООП и разработке GUI приложений, освоить методы привязки данных к элементам управления формы.

Задание к лабораторной работе:

Создать приложение Windows Forms, которое реализует задание по варианту.

Общие требования:

1. Построить класс (при необходимости – иерархию классов) для описания объекта, указанного в варианте. Среди методов обязательно наличие конструктора с параметрами, свойства – получение всех полей объекта.
2. Для ввода данных о новом объекте класса разработать модальную форму.
3. Контейнерный класс для работы с объектами должен содержать методы, обеспечивающие добавление и удаление записей, сохранение информации в файле и считывание из файла, отображение на экране. Для привязки данных класса к элементам управления использовать класс `BindingSource`.
4. Организовать перехват возможных исключений при помощи блоков `try...catch`; при проверке корректности ввода данных использовать компонент `ErrorProvider`;

Индивидуальное задание:

| № | Объект, поля | Обработка |
|---|---|---|
| 8 | Книги. Поля – автор, год издания, количество страниц. | Вывести все книги одного указанного автора, сформировать список уникальных авторов. |

Листинг программы:

Main.cs

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

```

using System.Windows.Forms;
using System.IO;
using System.Xml.Serialization;
using System.Reflection;

namespace Laboratory_work_0
{
    public partial class Main : Form
    {
        public BindingList<Book> Books { get; set; }

        public Main()
        {
            Books = GetBooks();
            InitializeComponent();
        }

        private BindingList<Book> GetBooks()
        {
            var list = new BindingList<Book>();
            list.Add(new Book("Test Author #1", 1995, 1024));
            list.Add(new Book("Test Author #2", 1996, 1025));
            list.Add(new Book("Test Author #3", 1997, 1026));
            list.Add(new Book("Test Author #4", 1998, 1027));
            list.Add(new Book("Test Author #5", 1999, 1028));

            return list;
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            var books = this.Books;
            dataGridView1.DataSource = books;
        }

        private void btn_addBook_Click(object sender, EventArgs e)
        {
            String bufAuthor = txtbx_author.Text;
            int bufYear = int.Parse(txtbx_year.Text);
            int bufNumOfPages = int.Parse(txtbx_numOfPages.Text);
            Books.Add(new Book(bufAuthor, bufYear, bufNumOfPages));
        }

        private void txtbx_author_TextChanged(object sender, EventArgs e)
        {
        }

        private void txtbx_year_TextChanged(object sender, EventArgs e)
        {
        }

        private void txtbx_numOfPages_TextChanged(object sender, EventArgs e)
        {
        }

        private void label1_Click(object sender, EventArgs e)
        {
        }

        private void btn_delete_Click(object sender, EventArgs e)

```

```

        {
            Books.RemoveAt(dataGridView1.SelectedRows[0].Index);
        }

private void btn_search_Click(object sender, EventArgs e)
{
    SearchByAuthor searchByAuthor = new SearchByAuthor(Books);
    searchByAuthor.Show();
}

private void btn_authors_Click(object sender, EventArgs e)
{
    UniqueAuthors uniqueAuthors = new UniqueAuthors(Books);
    uniqueAuthors.Show();
}

private void btn_saveToFile_Click(object sender, EventArgs e)
{
    var books = this.Books;
    XmlSerializer serializer = new XmlSerializer(books.GetType());
    using (StreamWriter writer = new StreamWriter("data.xml"))
    {
        serializer.Serialize(writer, books);
    }
}

private void btn_loadFromFile_Click(object sender, EventArgs e)
{
    Books = null;
    string path = Path.Combine(Environment.CurrentDirectory, "data.xml");
    XmlSerializer xmlSerializer = new XmlSerializer(typeof(BindingList<Book>));
    using (FileStream stream = File.Open(path, FileMode.Open))
    {
        Books = (BindingList<Book>)xmlSerializer.Deserialize(stream);
    }
    dataGridView1.DataSource = Books;
}
}
}

```

Book.cs

```

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

namespace Laboratory_work_0
{
    public class Book
    {
        public string Author { get; set; }
        public int ReleaseYear { get; set; }
        public int NumOfPages { get; set; }
        public Book() { }
        public Book(string author, int releaseYear, int numOfPages)
        {
            Author = author;
            ReleaseYear = releaseYear;
            NumOfPages = numOfPages;
        }
    }
}

```

SearchByAuthor.cs

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

namespace Laboratory_work_0
{
    public partial class SearchByAuthor : Form
    {
        public BindingList<Book> RecievedBooks { get; set; }
        public BindingList<Book> SearchedBooks { get; set; }

        public SearchByAuthor(BindingList<Book> books)
        {
            InitializeComponent();
            this.RecievedBooks = books;
        }

        private void btn_search_Click(object sender, EventArgs e)
        {
            String bufAuthor = txtbx_author.Text;
            BindingList<Book> searchedBooks = new BindingList<Book>();
            foreach(Book book in RecievedBooks)
            {
                if(book.Author.Equals(bufAuthor))
                {
                    searchedBooks.Add(book);
                }
            }
            dataGridView1.DataSource = searchedBooks;
        }

        private void txtbx_author_TextChanged(object sender, EventArgs e)
        {
        }

        private void SearchByAuthor_Load(object sender, EventArgs e)
        {
        }
    }
}
```

UniqueAuthors.cs

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

namespace Laboratory_work_0
{
    public partial class UniqueAuthors : Form
    {
        public BindingList<Book> RecievedBooks { get; set; }

        public UniqueAuthors(BindingList<Book> books)
        {
            InitializeComponent();
            this.RecievedBooks = books;
        }

        private void UniqueAuthors_Load(object sender, EventArgs e)
        {
            BindingList<string> allAuthors = new BindingList<string>();
            foreach (Book book in RecievedBooks)
            {
                allAuthors.Add(book.Author);
            }
            List<string> uniqueAuthors = new List<string>();
            uniqueAuthors.AddRange(allAuthors.Distinct());

            foreach (string author in uniqueAuthors)
            {
                lstbx_authors.Items.Add(author);
            }

        }

        private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
        }
    }
}
```

Результат работы программы:

Form1

Author

Year of release

Number of pages

Vadim

2021

199

add book

| | Author | ReleaseYear | NumOfPages |
|---|----------------|-------------|------------|
| ▶ | Test Author #1 | 1995 | 1024 |
| | Vadim | 123 | 123 |
| | Nastya | 1337 | 3 |
| | Oleg | 1889 | 200 |
| | Vadim | 2021 | 199 |
| * | | | |

delete

save to file

load from file

search

authors

UniqueAuthors

Test Author #1
Vadim
Nastya
Oleg

AddItem

Vadim

search

| | Author | ReleaseYear | NumOfPages |
|---|--------|-------------|------------|
| ▶ | Vadim | 123 | 123 |
| | Vadim | 2021 | 199 |
| * | | | |