

# Лабораторная работа № N

## Использование LINQ

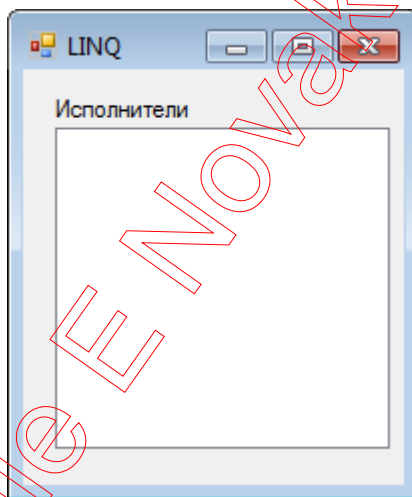
В лабораторной работе используется Microsoft Visual Studio 2010

### Подготовка к выполнению лабораторной работы

1. Откройте Visual Studio.
2. Создайте новый проект типа Windows Forms Application с названием OopLabs.Linq.
3. Добавьте в проект файл Artist.cs.
4. Изучите описание класса Album. Класс хранит информацию о музыкальном альбоме: имя исполнителя, название, дату выхода. Кроме того, определение класса включает статический метод Album.GetAlbums(), возвращающий список некоторых альбомов нескольких групп.

### Упражнение 1 – Вывод списка групп

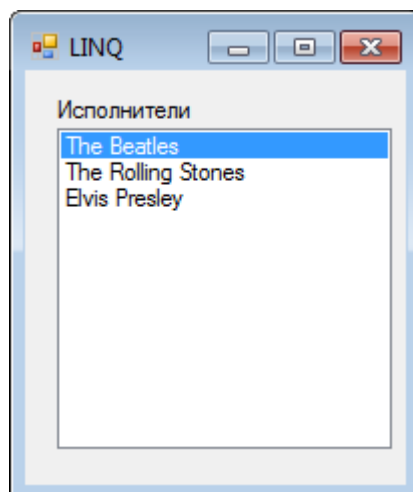
1. Разместите на главной форме надпись (объект Label) «Исполнители».
2. Ниже надписи разместите список (ListBox):



3. Добавьте обработчик события «Load» формы, выводящий в список группы, альбомы которых возвращаются функцией Album.GetAlbums(). Для этого присвойте параметру DataSource списка нужное значение. Используйте функции LINQ Select() для отображения коллекции, Distinct() для удаления дубликатов и ToList() для конвертации коллекции в список:

```
private void MainForm_Load(object sender, EventArgs e)
{
    ArtistsListBox.DataSource = Album.GetAlbums()
                                .Select(album => ...)
                                .Distinct()
                                .ToList();
}
```

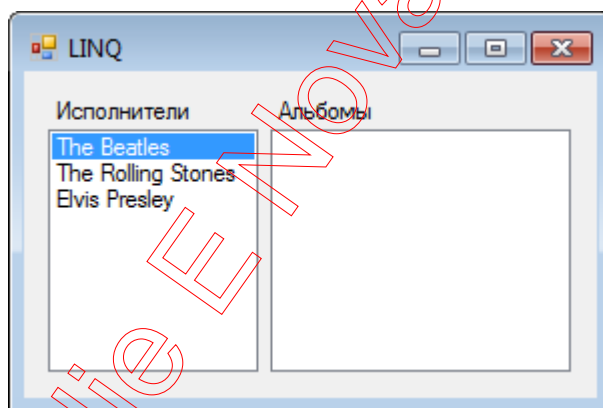
4. Откомпилируйте проект и запустите. Если все сделано правильно, то Вы увидите следующий результат:



5. Откорректируйте ошибки и сохраните результат.
6. Покажите результат преподавателю. Полученный листинг сохраните для отчета.

## Упражнение 2 – Вывод альбомов для групп

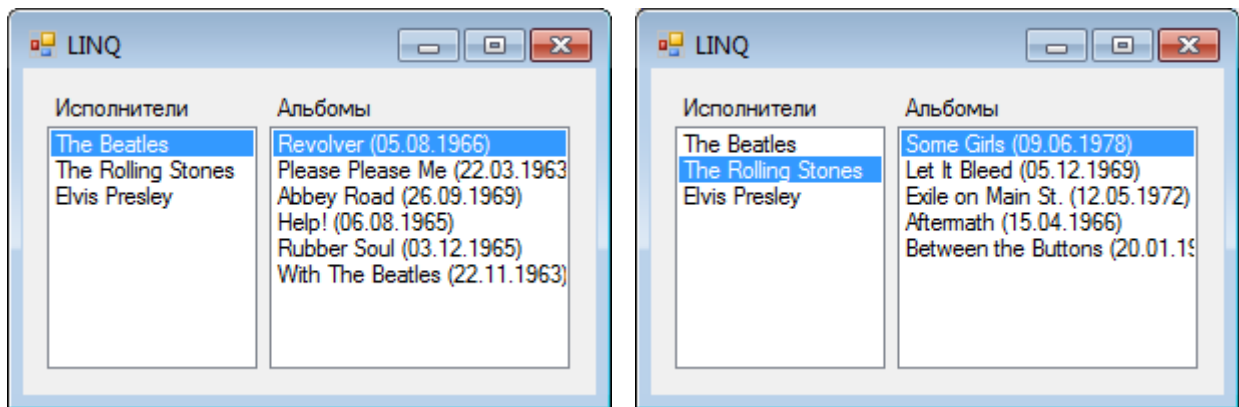
1. На форме, правее созданных ранее элементов разместите надпись «Альбомы» и еще один список:



2. Добавьте обработчик события «SelectedIndexChanged» списка исполнителей, выводящий в список альбомов альбомы выбранного исполнителя. Используйте функцию LINQ Where для фильтрации коллекции альбомов. Для получения выделенного пункта списка исполнителей используйте свойство объекта ListBox SelectedItem:

```
private void ArtistsListBox_SelectedIndexChanged(object sender, EventArgs e)
{
    AlbumsListBox.DataSource = Album.GetAlbums()
        .Where(album => ...)
        .ToList();
}
```

3. Откомпилируйте проект и запустите. Если все сделано правильно, то Вы увидите следующий результат:



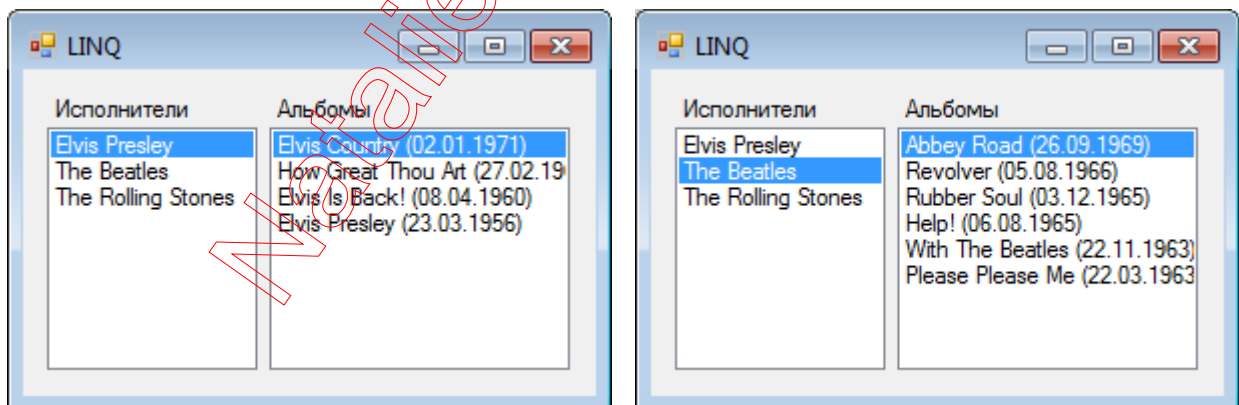
4. Откорректируйте ошибки и сохраните результат.
5. Покажите результат преподавателю. Полученный листинг сохраните для отчета.

### Упражнение 3 – Сортировка списков исполнителей и альбомов

1. Измените созданный ранее обработчик события «SelectedIndexChanged» списка исполнителей так, чтобы альбомы сортировались по дате выхода от более поздних к более-ранним. Используйте для этого функцию `OrderByDescending()`:

```
AlbumsListBox.DataSource = Album.GetAlbums()
    .Where(album => ...)
    .OrderByDescending(album => ...)
    .ToList();
```

2. Измените созданный ранее обработчик события «Load» формы так, чтобы исполнители выводились в список в алфавитном порядке. Используйте для этого функцию `LINQ OrderBy()`.
3. Откомпилируйте проект и запустите. Если все сделано правильно, то Вы увидите следующий результат:



4. Откорректируйте ошибки и сохраните результат.
5. Покажите результат преподавателю. Полученный листинг сохраните для отчета.