**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и управление»

Кафедра ИУ5. Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №7

**«**Работа с LINQ»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-25б |  | доцент каф. ИУ5 |
| Шкарин Е.А. |  | Гапанюк Ю.Е. |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2020 г.

# Постановка задачи:

Разработать программу, реализующую работу с LINQ to Objects. В качестве примера используйте проект «SimpleLINQ» из примера «Введение в LINQ».

1. Программа должна быть разработана в виде консольного приложения на языке C#. 2. Создайте класс «Сотрудник», содержащий поля:

• ID записи о сотруднике;

• Фамилия сотрудника;

• ID записи об отделе.

3. Создайте класс «Отдел», содержащий поля:

• ID записи об отделе;

• Наименование отдела.

4. Предполагая, что «Отдел» и «Сотрудник» связаны соотношением одинко-многим разработайте следующие запросы:

• Выведите список всех сотрудников и отделов, отсортированный по отделам.

• Выведите список всех сотрудников, у которых фамилия начинается с буквы «А».

• Выведите список всех отделов и количество сотрудников в каждом отделе.

• Выведите список отделов, в которых у всех сотрудников фамилия начинается с буквы «А».

• Выведите список отделов, в которых хотя бы у одного сотрудника фамилия начинается с буквы «А».

5. Создайте класс «Сотрудники отдела», содержащий поля:

• ID записи о сотруднике;

• ID записи об отделе. 6. Предполагая, что «Отдел» и «Сотрудник» связаны соотношением многоко-многим с использованием класса «Сотрудники отдела» разработайте следующие запросы:

• Выведите список всех отделов и список сотрудников в каждом отделе.

• Выведите список всех отделов и количество сотрудников в каждом отделе.

# Текст программы:

//Program.cs (делегаты)

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab7\_Linq

{

class Employee : Department

{

public Employee(int ID\_emp, string sur\_name, int ID\_dep, string dep\_name) : base(dep\_name)

{

ID\_employee = ID\_emp;

surName = sur\_name;

ID\_department\_empl = ID\_dep;

}

public int ID\_employee { get; set; }

public string surName { get; set; }

public int ID\_department\_empl { get; set; }

public override string ToString()

{

return $"{ID\_employee} {surName} {Name}";

}

class Department

{

public Department(string Name)

{

this.Name = Name;

}

public int ID\_department { get; set; }

public string Name { get; set; }

}

class Program

{

static void Main(string[] args)

{

#region uncorrect

var rnd = new Random();

var list\_employee = new List<Employee>()

{

new Employee(1, "Адлер", 9, "Отдел 9"),

new Employee(2, "Синицин", 8, "Отдел 8"),

new Employee(3, "Ковалев", 8, "Отдел 8"),

new Employee(4, "Масленников", 6, "Отдел 6"),

new Employee(5, "Петров", 6, "Отдел 6"),

new Employee(6, "Адхамов", 1, "Отдел 1"),

new Employee(7, "Алиев", 9, "Отдел 9"),

new Employee(8, "Аскеразде", 9, "Отдел 9"),

new Employee(9, "Попов", 1, "Отдел 1"),

new Employee(10, "Ильин", 7, "Отдел 7"),

new Employee(11, "Косов", 4, "Отдел 4"),

new Employee(12, "Ломов", 7, "Отдел 7")

};

var list\_department = new List<Department>();

Console.WriteLine("Группировка по отделу");

var group\_dep = list\_employee.GroupBy(item => item.ID\_department\_empl);

foreach (var group in group\_dep)

{

Console.WriteLine($"Номер отдела: {group.Key}");

foreach (var item in group)

{

Console.WriteLine(item.surName);

}

Console.WriteLine();

}

Console.WriteLine();

Console.WriteLine("Сотрудники у которых фамилия начинается с буквы А");

var surname\_A = list\_employee.Where(item => item.surName.StartsWith("А"));

foreach (var item in surname\_A)

{

Console.WriteLine(item);

}

Console.WriteLine();

Console.WriteLine("Количество сотрудников в отделе");

foreach (var group in group\_dep)

{

Console.WriteLine($"Номер отдела: {group.Key}");

Console.WriteLine(group.Count());

Console.WriteLine();

}

Console.WriteLine();

Console.WriteLine("Список отделов в которых есть сотрудники с Фамилией начинающейся на А");

foreach (var group in group\_dep)

{

if (group.Any(item => item.surName.StartsWith("А")))

{

Console.WriteLine($"Номер отдела: {group.Key}");

}

}

Console.WriteLine("Список отделов в которых все сотрудники с фамилией начинающейся на А");

foreach (var group in group\_dep)

{

if (group.All(item => item.surName.StartsWith("А")))

{

Console.WriteLine($"Номер отдела: {group.Key}");

}

}

#endregion

Console.ReadKey();

}

}

}

//department\_employees.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab7\_Linq

{

class department\_employees

{

public int ID\_departments { get; set; }

public int ID\_employees\_of\_department { get; set; }

public department\_employees(int depart)

{

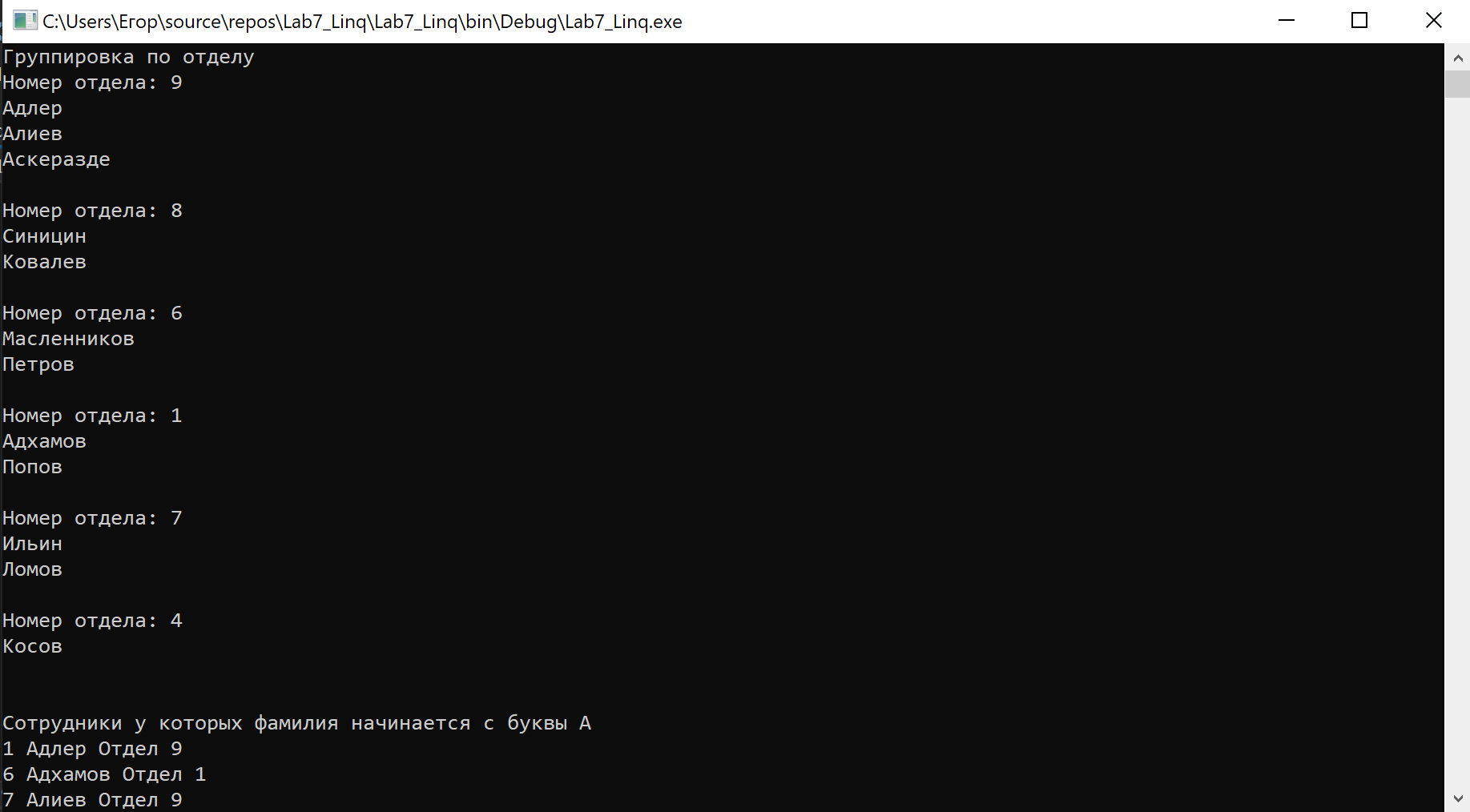
ID\_departments = depart;

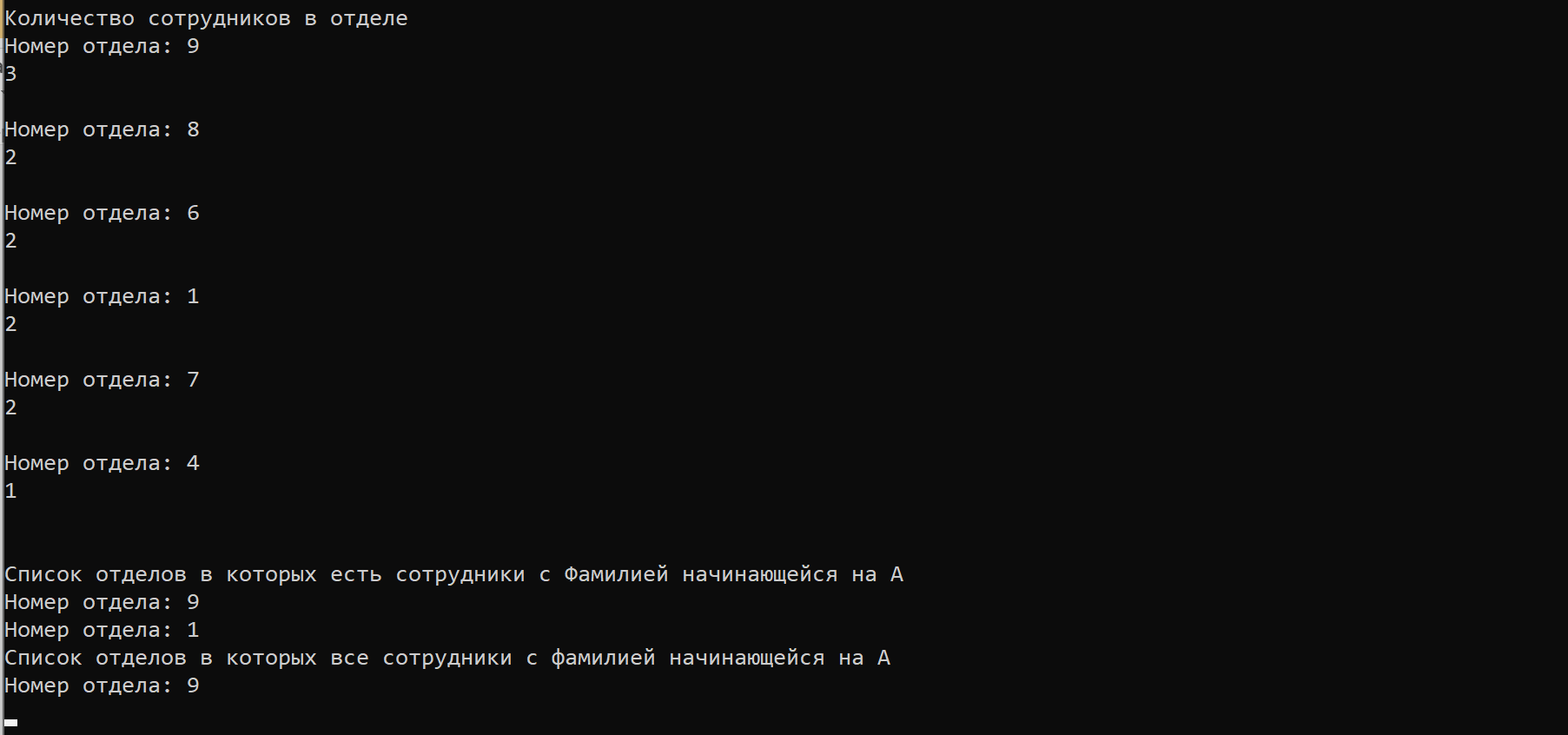
}

}

}

Тесты программы:





23.12.2020  
Шкарин Е.А