

**Московский государственный технический университет  
им. Н.Э.Баумана**

**Факультет ИУ**

**Кафедра ИУ5**

***Лабораторная работа №7 по курсу С#***

Группа: ИУ5-32

Миронов С.В.

## Лабораторная работа №7

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 один-ко-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 много-ко-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.

**Результат выполнения программы:**

```

1 Sales department
Abd
Athfd
Adeb
Aetun
Anton

2 Management department
Aeyfbsa
Shgjf
Kvdkgbk
Ae
Tslg
Oncgja
Lsfsa
Nssfhh
Kdwhfo

3 Shopping department
Ubdsk
ARE
Tsbab
Gssqg
Qdhwsqf
Ufjksaka
Ejsfs
Lsbafa
Psass
Nsgfa

(ID=1; name=Abd; depID=1)
(ID=2; name=Athfd; depID=1)
(ID=3; name=Adeb; depID=1)
(ID=4; name=Aetun; depID=1)
(ID=5; name=Anton; depID=1)
(ID=6; name=Aeyfbsa; depID=2)
(ID=9; name=Ae; depID=2)
(ID=16; name=ARE; depID=3)

1 Sales department количество:5
2 Management department количество:9
3 Shopping department количество:10

(ID=1; Title=Sales department)

(ID=1; Title=Sales department)
(ID=2; Title=Management department)
(ID=3; Title=Shopping department)

Sales department
5

Management department
9

Shopping department
10

Sales department
(ID=1; name=Abd; depID=1)
(ID=2; name=Athfd; depID=1)
(ID=3; name=Adeb; depID=1)
(ID=4; name=Aetun; depID=1)
(ID=5; name=Anton; depID=1)

Management department
(ID=6; name=Aeyfbsa; depID=2)
(ID=7; name=Shgjf; depID=2)
(ID=8; name=Kvdkgbk; depID=2)
(ID=9; name=Ae; depID=2)
(ID=10; name=Tslg; depID=2)
(ID=11; name=Oncgja; depID=2)
(ID=12; name=Lsfsa; depID=2)
(ID=13; name=Nssfhh; depID=2)
(ID=14; name=Kdwhfo; depID=2)

Shopping department
(ID=15; name=Ubdsk; depID=3)
(ID=16; name=ARE; depID=3)
(ID=17; name=Tsbab; depID=3)
(ID=18; name=Gssqg; depID=3)
(ID=19; name=Qdhwsqf; depID=3)
(ID=20; name=Ufjksaka; depID=3)
(ID=21; name=Ejsfs; depID=3)
(ID=22; name=Lsbafa; depID=3)
(ID=23; name=Psass; depID=3)
(ID=24; name=Nsgfa; depID=3)

```

## Код программы:

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

namespace lab7_csh
{
    class Program
    {
        static List<employee> E = new List<employee>()
        {
            new employee(1, "Abd", 1),
            new employee(2, "Athfd", 1),
            new employee(3, "Adeb", 1),
            new employee(4, "Aetun", 1),
            new employee(5, "Anton", 1),
            new employee(6, "Aeyfbsa", 2),
            new employee(7, "Shgjf", 2),
            new employee(8, "Kvdkgbk", 2),
            new employee(9, "Ae", 2),
            new employee(10, "Tslg", 2),
            new employee(11, "Oncgja", 2),
            new employee(12, "Lsfsa", 2),
            new employee(13, "Nssf", 2),
            new employee(14, "Kdwhfo", 2),
            new employee(15, "Ubdsk", 3),
            new employee(16, "ARE", 3),
            new employee(17, "Tsbab", 3),
            new employee(18, "Gssqg", 3),
            new employee(19, "Qdhwsgf", 3),
            new employee(20, "Ufjksaka", 3),
            new employee(21, "Ejsfs", 3),
            new employee(22, "Lsbafa", 3),
            new employee(23, "Psass", 3),
            new employee(24, "Nsgfa", 3)
        };

        static List<Department> D = new List<Department>()
        {
            new Department(1, "Sales department"),
            new Department(2, "Management department"),
            new Department(3, "Shopping department")
        };

        static List<Emp_Dep> ED = new List<Emp_Dep>()
        {
            new Emp_Dep(1,1),
            new Emp_Dep(2,1),
            new Emp_Dep(3,1),
            new Emp_Dep(4,1),
            new Emp_Dep(5,1),
            new Emp_Dep(6,2),
            new Emp_Dep(7,2),
            new Emp_Dep(8,2),
            new Emp_Dep(9,2),
            new Emp_Dep(10,2),
            new Emp_Dep(11,2),
            new Emp_Dep(12,2),
            new Emp_Dep(13,2),
            new Emp_Dep(14,2),
            new Emp_Dep(15,3),
            new Emp_Dep(16,3),
        }
    }
}
```

```

        new Emp_Dep(17,3),
        new Emp_Dep(18,3),
        new Emp_Dep(19,3),
        new Emp_Dep(20,3),
        new Emp_Dep(21,3),
        new Emp_Dep(22,3),
        new Emp_Dep(23,3),
        new Emp_Dep(24,3),
    };

    static void Main(string[] args)
    {
        var res1 = D.GroupJoin(E,
            d => d.departmentID,
            e => e.departmentID,
            (dep, emp) => new
            {
                ID=dep.departmentID,
                Title=dep.title,
                empl=emp.Select(em=>em.name)
            });
        foreach (var dep in res1)
        {
            Console.WriteLine(dep.ID + " " + dep.Title);
            foreach (string emp in dep.empl)
            {
                Console.WriteLine(emp);
            }
            Console.WriteLine();
        }

        var res2 = from e in E where e.name[0].Equals('A') select e;
        foreach (var x in res2) Console.WriteLine(x);
        Console.WriteLine();

        var res3 = from d in D
                    join e in E on d.departmentID equals e.departmentID into temp
                    select new { ID = d.departmentID, Title = d.title, count =
temp.Count() };
        foreach (var x in res3) Console.WriteLine(x.ID + " " + x.Title + "
количество:" + x.count);
        Console.WriteLine();

        var res4 = from d in D
                    join e in E on d.departmentID equals e.departmentID into temp
                    where temp.All(e => e.name[0].Equals('A'))
                    select d;
        foreach (var x in res4) Console.WriteLine(x);
        Console.WriteLine();

        var res5 = from d in D
                    join e in E on d.departmentID equals e.departmentID into temp
                    where temp.Any(e => e.name[0].Equals('A'))
                    select d;
        foreach (var x in res5) Console.WriteLine(x);
        Console.WriteLine();

        var res6 = from d in D
                    join ed in ED on d.departmentID equals ed.IDdep into temp
                    select new { Dep = d, Emp=temp};
        foreach (var x in res6)
        {
            Console.WriteLine(x.Dep.title);
            Console.WriteLine(x.Emp.Count());
        }
    }
}

```

```

        Console.WriteLine();
    }

    var res7 = from d in D
                join ed in ED on d.departmentID equals ed.IDdep into temp
                select new { Dep = d, Emp = temp };
    foreach (var x in res7)
    {
        Console.WriteLine(x.Dep.title);
        foreach (Emp_Dep emp in x.Emp)
        {
            var i = from e in E where e.ID == emp.IDemp select e;
            foreach(var y in i) Console.WriteLine(y);
        }
        Console.WriteLine();
    }
    Console.WriteLine();

    Console.ReadLine();
}
}
}

```

```

namespace lab7_csh
{
    class employee
    {
        public int ID;
        public string name;
        public int departmentID;

        public employee(int a, string b, int c)
        {
            ID = a; name = b; departmentID = c;
        }
        public override string ToString()
        {
            return "(ID=" + this.ID.ToString() + "; name=" + this.name + "; depID=" +
this.departmentID.ToString() + ")";
        }
    }
}

```

```

namespace lab7_csh
{
    class Department
    {
        public int departmentID;
        public string title;

        public Department(int a, string b)
        {
            departmentID = a; title = b;
        }

        public override string ToString()
        {
            return "(ID=" + this.departmentID.ToString() + "; Title=" + this.title + ")";
        }
    }
}

```

```
namespace lab7_csh
{
    class Emp_Dep
    {
        public int IDemp;
        public int IDdep;

        public Emp_Dep(int a, int b)
        {
            IDemp = a; IDdep = b;
        }
    }
}
```