

# Базовые компоненты интернет технологий

## ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7

Кочетков Михаил Дмитриевич  
Группа ИУ5-316

10 октября 2018 г.

## Задание

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

# Код

## Program.cs

```
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel.Design;
4  using System.Linq;
5
6  namespace Lab_7
7  {
8      class Program
9      {
10         static void Main(string[] args)
11         {
12             var departments = new List<Department>
13             {
14                 new Department(1, "Бухгалтерия"),
15                 new Department(2, "Маркетинговый отдел"),
16                 new Department(3, "Финансовый отдел"),
17                 new Department(4, "Отдел управления поставками"),
18                 new Department(5, "Отдел технической поддержки"),
19                 new Department(6, "Отдел разработки программного обеспечения"),
20                 new Department(7, "Отдел тестирования")
21             };
22
23             var employees = new List<Employee>
24             {
25                 new Employee(1, "Кочетков", 6),
26                 new Employee(2, "Белкина", 6),
27                 new Employee(3, "Дехтеров", 7),
28                 new Employee(4, "Крюков", 5),
29                 new Employee(5, "Хрюков", 1),
30                 new Employee(6, "Канюков", 3),
31                 new Employee(7, "Бадышев", 1),
32                 new Employee(8, "Антонова", 1),
33                 new Employee(9, "Симонов", 4),
34                 new Employee(10, "Сифонов", 3),
35                 new Employee(11, "Котов", 7),
36                 new Employee(12, "Датов", 5),
37                 new Employee(13, "Аверин", 2),
38                 new Employee(14, "Агатов", 2)
39             };
40
41             Console.WriteLine("Выведите список всех сотрудников и отделов, отсортированный
42                               ↪ по отделам");
43
44             var list1 =
45                 from employee in employees
46                 join department in departments on employee.DepartmentID equals
47                     ↪ department.ID
48                 orderby department.Name
```

```

47         select new
48         {
49             employee.Surname,
50             DepartmentName = department.Name
51         };
52
53     foreach (var item in list1)
54     {
55         Console.WriteLine(item);
56     }
57
58     Console.WriteLine();
59     Console.WriteLine("Выведите список всех сотрудников, у которых фамилия
    ↳ начинается с буквы «А»");
60
61     var list2 =
62         from employee in employees
63         where employee.Surname[0] == 'А'
64         select employee;
65
66     foreach (var item in list2)
67     {
68         Console.WriteLine(item);
69     }
70
71     Console.WriteLine();
72     Console.WriteLine("Выведите список всех отделов и количество сотрудников в
    ↳ каждом отделе");
73
74     var list3 =
75         from department in departments
76         join employee in employees on department.ID equals employee.DepartmentID
77         ↳ into employeesOnDepartment
78         select new
79         {
80             DepartmentID = department.ID,
81             DepartmentName = department.Name,
82             CountOfEmployees = employeesOnDepartment.Count()
83         };
84
85     foreach (var item in list3)
86     {
87         Console.WriteLine(item);
88     }
89
90     Console.WriteLine();
91     Console.WriteLine("Выведите список отделов, в которых у всех сотрудников
    ↳ фамилия начинается с буквы «А»");
92
93     var list4 =
94         from department in departments

```

```

95         join employee in employees on department.ID equals employee.DepartmentID
96         ↪ into employeesOnDepartment
97     where employeesOnDepartment.All(employee => employee.Surname[0] == 'A')
98     select department;
99
100 foreach (var item in list4)
101 {
102     Console.WriteLine(item);
103 }
104
105 Console.WriteLine();
106 Console.WriteLine("Выведите список отделов, " +
107     "в которых хотя бы у одного сотрудника фамилия начинается с
108     ↪ буквы «А»");
109
110 var list5 =
111     from department in departments
112     join employee in employees on department.ID equals employee.DepartmentID
113     ↪ into employeesOnDepartment
114     where employeesOnDepartment.Any(employee => employee.Surname[0] == 'A')
115     select department;
116
117 foreach (var item in list5)
118 {
119     Console.WriteLine(item);
120 }
121
122 Console.WriteLine();
123
124 var departmentEmployees = new List<DepartmentEmployees>
125 {
126     new DepartmentEmployees(5, 3),
127     new DepartmentEmployees(5, 1),
128     new DepartmentEmployees(5, 3),
129     new DepartmentEmployees(7, 5),
130     new DepartmentEmployees(3, 2),
131     new DepartmentEmployees(6, 7),
132     new DepartmentEmployees(6, 7),
133     new DepartmentEmployees(6, 7),
134     new DepartmentEmployees(7, 5),
135     new DepartmentEmployees(7, 2),
136     new DepartmentEmployees(5, 1),
137     new DepartmentEmployees(4, 6),
138     new DepartmentEmployees(4, 5),
139     new DepartmentEmployees(1, 4),
140     new DepartmentEmployees(2, 3),
141     new DepartmentEmployees(3, 3),
142     new DepartmentEmployees(11, 2),
143     new DepartmentEmployees(14, 5),
144     new DepartmentEmployees(12, 4),
145     new DepartmentEmployees(4, 7),

```

```

144         new DepartmentEmployees(3, 5),
145         new DepartmentEmployees(9, 2),
146         new DepartmentEmployees(9, 4),
147         new DepartmentEmployees(10, 6),
148         new DepartmentEmployees(12, 4),
149         new DepartmentEmployees(7, 2),
150         new DepartmentEmployees(1, 1),
151         new DepartmentEmployees(2, 1),
152         new DepartmentEmployees(12, 5)
153     };
154
155     Console.WriteLine("Выведите список всех отделов и список сотрудников в каждом
    ↳ отделе");
156
157     var list6 =
158         from departmentEmployee in departmentEmployees
159         group departmentEmployees by departmentEmployee.DepartmentID into deps
160         select new
161         {
162             Department = departments.Single(dep => dep.ID == deps.Key),
163             Employees = string.Join(", ", employees.FindAll(employee =>
    ↳ employee.DepartmentID == deps.Key))
164         };
165
166     foreach (var item in list6)
167     {
168         Console.WriteLine(item);
169     }
170
171     Console.WriteLine();
172     Console.WriteLine("Выведите список всех отделов и количество сотрудников в
    ↳ каждом отделе");
173
174     var list7 =
175         from departmentEmployee in departmentEmployees
176         group departmentEmployees by departmentEmployee.DepartmentID into deps
177         select new
178         {
179             Department = departments.Single(dep => dep.ID == deps.Key),
180             EmployeesCount = employees.FindAll(employee => employee.DepartmentID
    ↳ == deps.Key).Count
181         };
182
183     foreach (var item in list7)
184     {
185         Console.WriteLine(item);
186     }
187 }
188 }
189 }

```

## Employee.cs

```
1 namespace Lab_7
2 {
3     public class Employee
4     {
5         public int ID;
6         public string Surname;
7         public int DepartmentID;
8
9         public Employee(int id, string surname, int departmentId)
10        {
11            ID = id;
12            Surname = surname;
13            DepartmentID = departmentId;
14        }
15
16        public override string ToString()
17        {
18            return string.Format("{ Employee ID: {0}, Surname: {1}, DepartmentID: {2}
19            ↵ }", ID, Surname, DepartmentID);
20        }
21    }
```

## Department.cs

```
1 namespace Lab_7
2 {
3     public class Department
4     {
5         public int ID;
6         public string Name;
7
8         public Department(int id, string name)
9         {
10             ID = id;
11             Name = name;
12         }
13
14         public override string ToString()
15         {
16             return string.Format("{0} Department ID: {0}, Name: {1} }}", ID, Name);
17         }
18     }
19 }
```



## DepartmentEmployees.cs

```
1 namespace Lab_7
2 {
3     public class DepartmentEmployees
4     {
5         public int EmployeeID;
6         public int DepartmentID;
7
8         public DepartmentEmployees(int employeeId, int departmentId)
9         {
10             EmployeeID = employeeId;
11             DepartmentID = departmentId;
12         }
13
14         public override string ToString()
15         {
16             return string.Format("{ { DepartmentEmployees EmployeeID: {0}, DepartmentID:
17                                     ↪ {1} } }", EmployeeID, DepartmentID);
18         }
19     }
```

## Тесты

```
Выведите список всех сотрудников и отделов, отсортированный по отделам
{ Surname = Хрюков, DepartmentName = Бухгалтерия }
{ Surname = Бадышев, DepartmentName = Бухгалтерия }
{ Surname = Антонова, DepartmentName = Бухгалтерия }
{ Surname = Аверин, DepartmentName = Маркетинговый отдел }
{ Surname = Агатов, DepartmentName = Маркетинговый отдел }
{ Surname = Кочетков, DepartmentName = Отдел разработки программного обеспечения }
{ Surname = Белкина, DepartmentName = Отдел разработки программного обеспечения }
{ Surname = Дехтеров, DepartmentName = Отдел тестирования }
{ Surname = Котов, DepartmentName = Отдел тестирования }
{ Surname = Крюков, DepartmentName = Отдел технической поддержки }
{ Surname = Датов, DepartmentName = Отдел технической поддержки }
{ Surname = Симонов, DepartmentName = Отдел управления поставками }
{ Surname = Канюков, DepartmentName = Финансовый отдел }
{ Surname = Сифонов, DepartmentName = Финансовый отдел }
```

Рис. 1: Список всех сотрудников и отделов, отсортированный по отделам

```
Выведите список всех сотрудников, у которых фамилия начинается с буквы «А»
{ Employee ID: 8, Surname: Антонова, DepartmentID: 1 }
{ Employee ID: 13, Surname: Аверин, DepartmentID: 2 }
{ Employee ID: 14, Surname: Агатов, DepartmentID: 2 }
```

Рис. 2: Список всех сотрудников, у которых фамилия начинается с буквы «А»

```

Выведите список всех отделов и количество сотрудников в каждом отделе
{ DepartmentID = 1, DepartmentName = Бухгалтерия, CountOfEmployees = 3 }
{ DepartmentID = 2, DepartmentName = Маркетинговый отдел, CountOfEmployees = 2 }
{ DepartmentID = 3, DepartmentName = Финансовый отдел, CountOfEmployees = 2 }
{ DepartmentID = 4, DepartmentName = Отдел управления поставками, CountOfEmployees = 1 }
{ DepartmentID = 5, DepartmentName = Отдел технической поддержки, CountOfEmployees = 2 }
{ DepartmentID = 6, DepartmentName = Отдел разработки программного обеспечения, CountOfEmployees = 2 }
{ DepartmentID = 7, DepartmentName = Отдел тестирования, CountOfEmployees = 2 }

```

Рис. 3: Список всех отделов и количество сотрудников в каждом отделе

```

Выведите список отделов, в которых у всех сотрудников фамилия начинается с буквы «А»
{ Department ID: 2, Name: Маркетинговый отдел }

```

Рис. 4: 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»
{ Department ID: 1, Name: Бухгалтерия }
{ Department ID: 2, Name: Маркетинговый отдел }

```

Рис. 5: Список отделов, в которых хотя бы у одного сотрудника фамилия начинается с буквы «А»

```

Выведите список всех отделов и список сотрудников в каждом отделе
{ Department = { Department ID: 3, Name: Финансовый отдел }, Employees = { Employee ID: 6, Surname: Канюков, DepartmentID: 3 }, { Employee ID: 10, Surname: Сифонов, DepartmentID: 3 } }
{ Department = { Department ID: 1, Name: Бухгалтерия }, Employees = { Employee ID: 5, Surname: Хрюков, DepartmentID: 1 }, { Employee ID: 7, Surname: Бадмшев, DepartmentID: 1 }, { Employee ID: 8, Surname: Антонова, DepartmentID: 1 } }
{ Department = { Department ID: 5, Name: Отдел технической поддержки }, Employees = { Employee ID: 4, Surname: Крюков, DepartmentID: 5 }, { Employee ID: 12, Surname: Датов, DepartmentID: 5 } }
{ Department = { Department ID: 2, Name: Маркетинговый отдел }, Employees = { Employee ID: 13, Surname: Аверин, DepartmentID: 2 }, { Employee ID: 14, Surname: Арапов, DepartmentID: 2 } }
{ Department = { Department ID: 7, Name: Отдел тестирования }, Employees = { Employee ID: 3, Surname: Дехтеров, DepartmentID: 7 }, { Employee ID: 11, Surname: Котов, DepartmentID: 7 } }
{ Department = { Department ID: 6, Name: Отдел разработки программного обеспечения }, Employees = { Employee ID: 1, Surname: Кочетков, DepartmentID: 6 }, { Employee ID: 2, Surname: Белкина, DepartmentID: 6 } }
{ Department = { Department ID: 4, Name: Отдел управления поставками }, Employees = { Employee ID: 9, Surname: Симонов, DepartmentID: 4 } }

```

Рис. 6: Список всех отделов и список сотрудников в каждом отделе

```

Выведите список всех отделов и количество сотрудников в каждом отделе
{ Department = { Department ID: 3, Name: Финансовый отдел }, EmployeesCount = 2 }
{ Department = { Department ID: 1, Name: Бухгалтерия }, EmployeesCount = 3 }
{ Department = { Department ID: 5, Name: Отдел технической поддержки }, EmployeesCount = 2 }
{ Department = { Department ID: 2, Name: Маркетинговый отдел }, EmployeesCount = 2 }
{ Department = { Department ID: 7, Name: Отдел тестирования }, EmployeesCount = 2 }
{ Department = { Department ID: 6, Name: Отдел разработки программного обеспечения }, EmployeesCount = 2 }
{ Department = { Department ID: 4, Name: Отдел управления поставками }, EmployeesCount = 1 }

```

Рис. 7: Список всех отделов и количество сотрудников в каждом отделе