Отчет по лабораторной работе № 7 по курсу “Базовые компоненты интернет-технологий”

ИСПОЛНИТЕЛЬ:

студент группы ИУ5-33 _____

(подпись)

Желанкина А.С.

"__"_____2017 г.

Москва, МГТУ - 2017

Описание задания

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.

Текст программы

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;

namespace Lab7
{
    class Program
    {
        /// <summary>
        /// Класс данных о сотруднике
        /// </summary>
        public class Employee
        {
            /// <summary>
            /// Ключ сотрудника
            /// </summary>
            public int id;

            /// <summary>
            /// Фамилия сотрудника
            /// </summary>
            public string surname;

            /// <summary>
            /// ID записи об отделе
            /// </summary>
            public int id_department;

            /// <summary>
            /// Конструктор
            /// </summary>
            public Employee(int i, string s, int i_d)
            {
                id = i;
                surname = s;
                id_department = i_d;
            }

            /// <summary>
            /// Приведение к строке
            /// </summary>
            public override string ToString()
            {
                return "(id = " + id.ToString() + "; surname = " + surname +
                    "; id_department = " + id_department.ToString() + ")";
            }
        }

        /// <summary>
        /// Класс данных об отделе
        /// </summary>
        public class Department
        {
            /// <summary>
            /// Ключ отдела
            /// </summary>
            public int id_department;

            /// <summary>
```

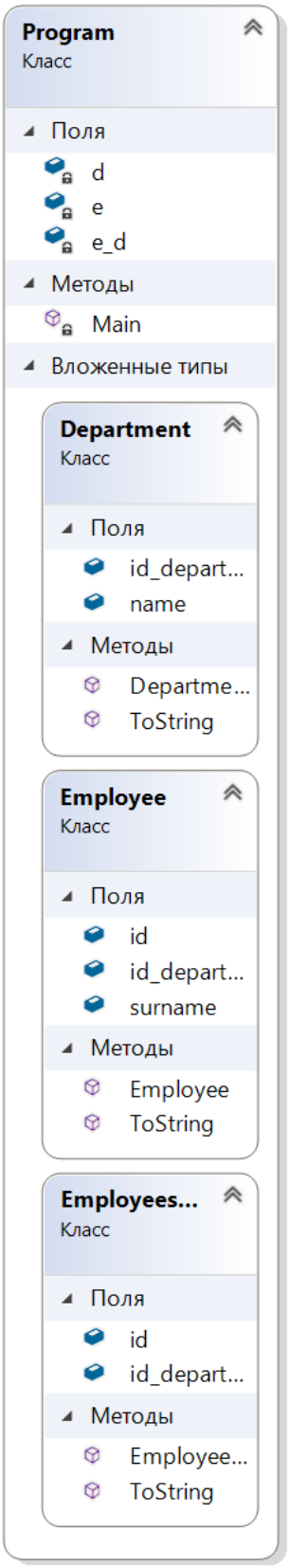
```
            /// Название отдела
            /// </summary>
            public string name;

            /// <summary>
            /// Конструктор
            /// </summary>
            public Department(int i, string n)
            {
                id_department = i;
                name = n;
            }

            /// <summary>
            /// Приведение к строке
            /// </summary>
            public override string ToString()
            {
                return "(id_department = " + id_department.ToString() + "; name = " +
name + ")";
            }
        }

        /// <summary>
        /// Класс сотрудники отдела
        /// </summary>
        public class Employees_of_department
        {
            /// <summary>
            /// ID сотрудника
            /// </summary>
            public int id;

            /// <summary>
            /// ID отдела
            /// </summary>
            public int id_department;

            /// <summary>
            /// Конструктор
            /// </summary>
            public Employees_of_department(int i, int i_d)
            {
                id = i;
                id_department = i_d;
            }

            /// <summary>
            /// Приведение к строке
            /// </summary>
            public override string ToString()
            {
                return "(id = " + id.ToString() + "; id_department = " +
id_department.ToString() + ")";
            }
        }

        //Пример данных
        static List<Employee> e = new List<Employee>()
            {
                new Employee(1, "Dior", 11),
                new Employee(2, "Musk", 12),
                new Employee(3, "Warhol", 13),
                new Employee(5, "King", 15),
                new Employee(6, "Chanel", 11),
```

```
                new Employee(7, "Adam", 13),
                new Employee(4, "Adorée", 17),
                new Employee(8, "Akerman", 17),
                new Employee(9, "Picasso", 13)
            };

        static List<Department> d = new List<Department>()
            {
                new Department(13, "art"),
                new Department(15, "writing"),
                new Department(11, "fashion"),
                new Department(17, "cinema"),
                new Department(12, "technologies")
            };

        static List<Employees_of_department> e_d = new List<Employees_of_department>()
            {
                new Employees_of_department(1, 11),
                new Employees_of_department(2, 12),
                new Employees_of_department(3, 14),
                new Employees_of_department(4, 17),
                new Employees_of_department(5, 15),
                new Employees_of_department(6, 11),
                new Employees_of_department(7, 13),
                new Employees_of_department(8, 17),
                new Employees_of_department(9, 13)
            };

        static void Main(string[] args)
        {
            Console.WriteLine("List of all employees and departments, sorted by
department: ");
            var q1 = from x in e
                     orderby x.id_department descending, x.id ascending
                     select x;
            foreach (var x in q1)
                Console.WriteLine(x);

            Console.WriteLine("A list of all employees whose surname starts with the
letter 'A': ");
            var q2 = from x in e
                     where x.surname[0] is 'A'
                     orderby x.surname ascending, x.id descending
                     select x;
            foreach (var x in q2)
                Console.WriteLine(x);

            Console.WriteLine("List of all departments and number of employees in each
department: ");
            var q3 = from x in d
                     join y in e on x.id_department equals y.id_department into temp
                     from t in temp
                     select new { v1 = x.name, v2 = x.id_department, cnt = temp.Count()
};
            q3 = q3.Distinct();
            foreach (var x in q3)
                Console.WriteLine(x);

            Console.WriteLine("A list of departments in which all employees start with
the letter 'A': ");
            var q4_1 = from x in e
                       join y in q2 on x.id_department equals y.id_department into temp
                       from t in temp
                       select new { v1 = x.id_department, cnt = temp.Count() };
            q4_1 = q4_1.Distinct();
```

```
            var q4 = from x in q3
                     from y in q4_1
                     where (x.cnt == y.cnt) && (x.v2 == y.v1)
                     select new { v1 = x.v1 };
            q4 = q4.Distinct();
            foreach (var x in q4)
                Console.WriteLine(x);

            Console.WriteLine("List of departments in which at least one employee " +
                "has a surname beginning with the letter 'A': ");
            var q5_1 = from x in e
                     where x.surname[0] is 'A'
                     select new { v1 = x.id_department };
            q5_1 = q5_1.Distinct();
            var q5 = from x in d
                     from y in q5_1
                     where x.id_department == y.v1
                       select new { v1 = x.name };
            q5 = q5.Distinct();
            foreach (var x in q5)
                Console.WriteLine(x);

            //II

            Console.WriteLine("A list of all departments and a list of employees in each
department: ");
            var q6_1 = from x in e
                     join l in e_d on x.id equals l.id into temp
                     from t1 in temp
                     join y in d on t1.id_department equals y.id_department into temp2
                     from t2 in temp2
                     select new { id = x.id_department, name = t2.name };
            q6_1 = q6_1.Distinct();
            foreach (var x in q6_1)
                Console.WriteLine(x);
            var q6_2 = from x in e
                       join l in e_d on x.id equals l.id into temp
                       from t1 in temp
                       join y in e on t1.id equals y.id into temp2
                       from t2 in temp2
                       select new { id = x.id, surname = t2.surname };
            q6_2 = q6_2.Distinct();
            foreach (var x in q6_2)
                Console.WriteLine(x);

            Console.WriteLine("List of all departments and number of employees in each
department: ");
             var q7_1 = from x in e_d
                        join y in e on x.id_department equals y.id_department into temp
                        from t in temp
                        select new { number = temp.Count(), id = t.id_department };
            q7_1 = q7_1.Distinct();
            var q7_2 = from x in e
                       join ed in e_d on x.id equals ed.id into temp
                       from t1 in temp
                       join y in d on t1.id_department equals y.id_department into temp2
                      from t2 in temp2
                     select new { name = t2.name, id = t2.id_department };
            q7_2 = q7_2.Distinct();
            var q7 = from x in q7_1
                     from y in q7_2
                     where x.id == y.id
                     select new { name = y.name, number = x.number };
            q7 = q7.Distinct();
            foreach (var x in q7)
```

```
                Console.WriteLine(x);

            Console.ReadKey();
        }
    }
}
```

Диаграмма классов



Результаты выполнения