



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Прикладная информатика

О Т Ч Е Т
по лабораторной работе № 9

Дисциплина: Разработка приложений на языке C#

Название работы: Сериализация

Студент гр. ИУ6-72Б

01.10.2022

(Подпись, дата)

И.С. Марчук

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

А.М. Минитаева

(И.О. Фамилия)

Москва, 2022

Цель работы: изучение принципов сериализации и десериализации

Задание:

Требуется разработать программу, ведущую учёт заказов в магазине.

Классы:

- покупатель, с атрибутами: имя (string), адрес (string), скидка (double);
- товар: название (string) и цена (decimal);
- база данных товаров, хранящий ассоциативный массив («словарь») с информацией о товарах;
- orderLine с полями количество (int) и продукт (Product);
- order с полями номер заказа (int), клиент (Customer), скидка (decimal), общая стоимость (decimal) и строки заказа (List<OrderLine>).

Реализовать следующую логику основной программы:

1. Создаётся и заполняется база данных товаров (ассоциативный массив).
2. В консоли вводятся данные по конкретному покупателю, создаётся соответствующий объект.
3. Создаётся заказ для введённого ранее покупателя. Устанавливается скидка на заказ в соответствии со скидкой покупателя.
4. В цикле формируются необходимое количество строк заказа: вводятся коды товаров и количества их единиц.
5. Полная информация о заказе сохраняется в файле с заданным именем.

Создать методы, которые осуществляют сериализацию/десериализацию объекта типа База данных товаров. Формат выбрать самостоятельно.

Листинг программы:

Program.cs:

```
using System;

namespace lab9
{
    class Program
    {
        static void Main(string[] args)
        {
            // заполняем базу
            Base b = new Base();

            // сохраняем в файл
            b.write();
        }
    }
}
```

Base.cs:

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

using System.Text.Json;
using System.IO;
using System.Diagnostics;
using System.Xml.Linq;
using System.Xml.Serialization;
using System.Runtime.Serialization;

namespace lab9
{
    internal class Base
    {
        List<Order> orders = new List<Order>();
        List<Product> products = new List<Product>();
        public Base()
        {
            Console.WriteLine("Создание базы товаров: ");

            products.Add(new Product("Банан", 1));
            products.Add(new Product("Арбуз", 5));
            products.Add(new Product("Баклажан", 12));
            products.Add(new Product("Белый хлеб", 11));
            products.Add(new Product("Икра белая", 10));
            products.Add(new Product("Виноград", 6));
            Init();
        }
    }
}
```

```

    }

    public void Init()
    {
        int id = 0;
        Random random = new Random();

        int count = random.Next(2, 4);
        Console.WriteLine("Создание базы клиентов (" +
count+"шт.)");

        for (int i = 0; i < count; i++)
        {

            Client cl = new Client();

            var or = new Order(++id, cl);

            for (int j = 0; j < random.Next(1, 3); j++)
            {
                OrderLine ol = new OrderLine(random.Next(1,
10), products[random.Next(products.Count)]);
                or.AddOrderLine(ol);
            }

            orders.Add(or);
        }

    }

    // сохранение в файл
    public void write()
    {
        Console.WriteLine();

        // сериализация в джсон
        Console.WriteLine("Сохраняю в файл
lab9\\bin\\Debug\\net6.0\\data_json.txt");
        File.WriteAllText("data_json.txt",
JsonSerializer.Serialize(orders));
        Console.WriteLine("Сохранено");
    }
}
}

```

Client.cs:

```

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

```

```

namespace lab9
{
    internal class Client
    {
        public string name { get; set; }
        public string address { get; set; }
        public int discount { get; set; }

        public Client()
        {
            Console.WriteLine("    Создание нового клиента: ");
            Console.WriteLine("        Введите имя: ");
            this.name = Console.ReadLine();

            Console.WriteLine("        Введите Адрес: ");
            this.address = Console.ReadLine();

            Console.WriteLine("        Введите Скидку (0-99):");
            int balance = 0;
            while (!int.TryParse(Console.ReadLine(), out balance)
|| balance < 0 || balance > 99)
            {
                Console.WriteLine("Введено      не      корректное
число!");
            }
            this.discount = balance;

            Console.WriteLine("    Новый клиент создан: " + name
+ " Адрес: " + address + " Скидка: " + balance);
        }
    }
}

```

Order.cs:

```

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

namespace lab9
{
    internal class Order
    {
        public int id { get; set; }
        public Client client { get; set; }
        public decimal discond => client.discount;
        public decimal price => getPrice();
        public List<OrderLine> lines { get; }
    }
}

```

```

private decimal getPrice()
{
    decimal price = 0;
    foreach (OrderLine line in lines)
    {
        price += line.product.price * line.count;
    }
    return (price * (100 - discond)) / 100;
}
public Order(int id, Client client)
{
    Console.WriteLine("        Создание заказа id= "+ id+":
");

    this.id = id;
    this.client = client;
    lines = new List<OrderLine>();
}

public void AddOrderLine(OrderLine ol)
{
    lines.Add(ol);
}
}
}

```

OrderLine.cs:

```

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

namespace lab9
{
    internal class OrderLine
    {
        public int count { get; set; }
        public Product product { get; set; }

        public OrderLine(int count, Product product)
        {
            Console.WriteLine("        Пункт заказа: кол-во= " +
count + " Товар=" + product.name + ": ");
            this.count = count;
            this.product = product;
        }
    }
}

```

Product.cs:

```

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

namespace lab9
{
    internal class Product
    {
        public string name { get; }
        public decimal price { get; }

        public Product(string name, decimal price)
        {
            Console.WriteLine("    Создание товара: " + name + "
\tПо цене:" + price);

            this.name = name;
            this.price = price;
        }
    }
}

```

Работа программы показана на рисунке 1.

```

Шрифт
Выбрать Консоль отладки Microsoft Visual Studio
Создание базы товаров:
Создание товара: Банан      По цене:1
Создание товара: Арбуз     По цене:5
Создание товара: Баклажан   По цене:12
Создание товара: Белый хлеб По цене:11
Создание товара: Икра белая По цене:10
Создание товара: Виноград   По цене:6
Создание базы клиентов (2шт.)
Создание нового клиента:
Введите имя:
Ivan
Введите Адрес:
Moscow
Введите Скидку (0-99):99
Новый клиент создан: Ivan Адрес: Moscow Скидка: 99
Создание заказа id= 1:
Пункт заказа: кол-во= 2 Товар=Белый хлеб:
Пункт заказа: кол-во= 5 Товар=Арбуз:
Создание нового клиента:
Введите имя:
Marchuk
Введите Адрес:
Moscow2
Введите Скидку (0-99):10
Новый клиент создан: Marchuk Адрес: Moscow2 Скидка: 10
Создание заказа id= 2:
Пункт заказа: кол-во= 8 Товар=Икра белая:
Пункт заказа: кол-во= 3 Товар=Баклажан:
Сохраняю в файл lab9\bin\Debug\net6.0\data_json.txt
Сохранено

```

Рисунок 1 – Работа программы

Вывод: в процессе выполнения лабораторной работы были изучены средства сериализации и десериализации классов в C#.