

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра ИБ

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Объектно-ориентированное программирование»
Тема: Использование типов-значений

Студент гр. 3363

Минко Д.А.

Преподаватель

Новакова Н.Е.

Санкт-Петербург
2024

Цель работы.

Цель данной лабораторной работы научиться создавать и использовать перечислимые типы (enum) и структуры в C# для представления и работы с данными, а также закрепить навыки объявления переменных, присвоения им значений и вывода результатов на консоль в рамках консольного приложения.

Ход работы.

1. Создание перечислимого типа (рис. 1).

В ходе работы была создана программа с использованием перечислимого типа (enum) для представления различных типов банковских счетов: Checking и Deposit. Были объявлены две переменные типа AccountType, которым присвоены соответствующие значения, и реализован вывод этих значений на консоль с помощью операторов Console.WriteLine. Программа успешно скомпилирована и сохранена.

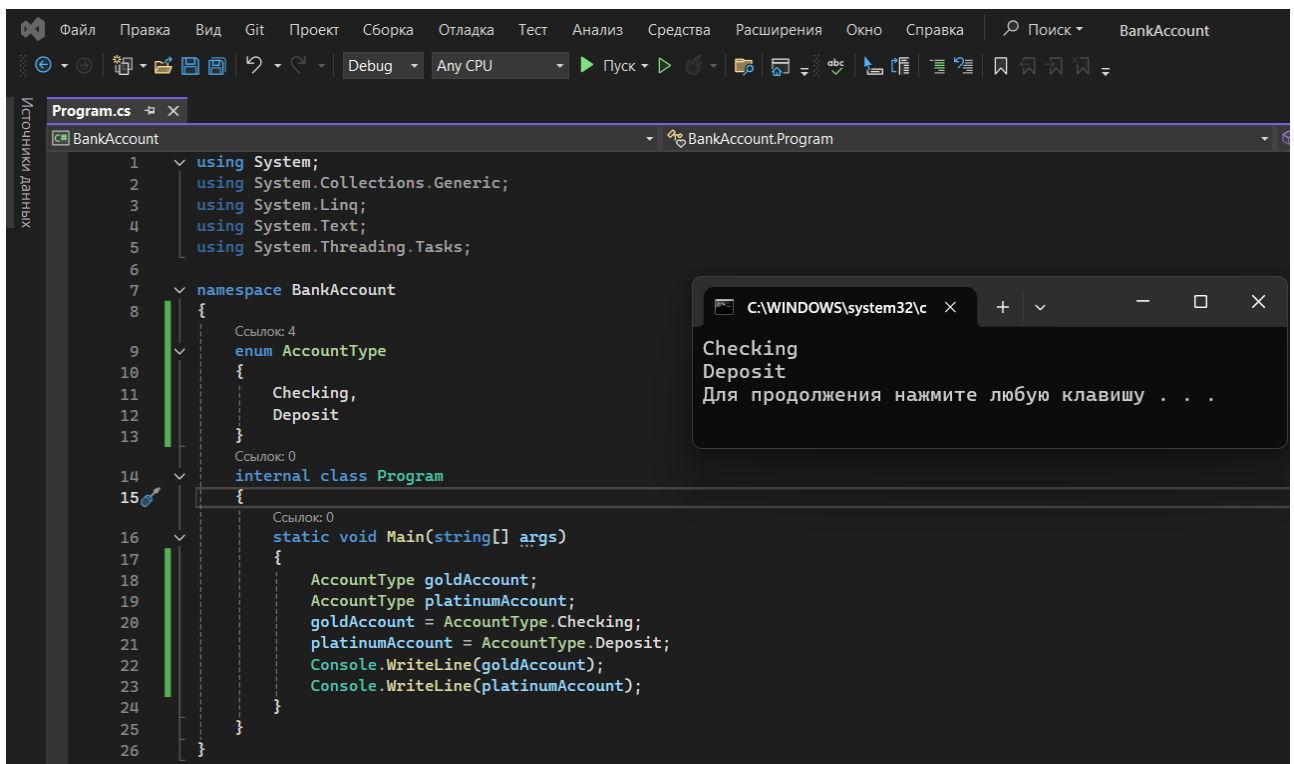


Рисунок 1 – Программа с перечислимым типом (enum)

2. Создание типа структура (рис. 2).

В ходе работы была создана программа с использованием структуры BankAccount для представления данных о банковском счете. В структуру были включены необходимые поля, а в методе Main объявлена переменная goldAccount, для которой заданы значения полей. Значения каждого элемента структуры были выведены на консоль с помощью операторов Console.WriteLine. Программа успешно скомпилирована и сохранена.

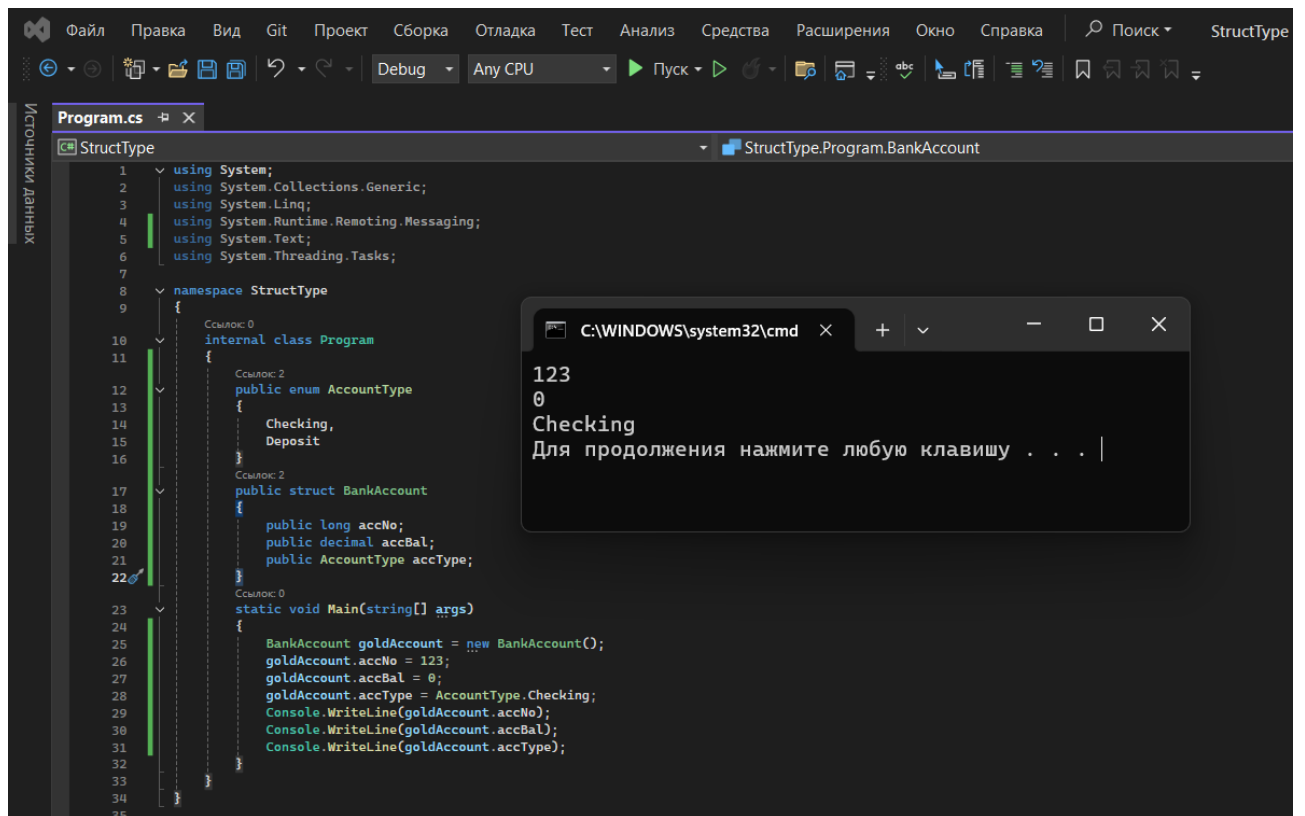


Рисунок 2 – Программа с типом Struct

Вывод.

В ходе выполнения лабораторной работы были изучены основы работы с перечислимыми типами (enum) и структурами в языке программирования C#. В первом упражнении была создана программа, включающая перечислимый тип для представления различных типов банковских счетов, таких как Checking и Deposit. Переменные типа enum были успешно объявлены, им были присвоены значения, которые затем выведены на консоль, что позволило на практике понять, как использовать перечисления для упрощения работы с фиксированными наборами данных.

Во втором упражнении была разработана структура BankAccount, которая использовалась для представления информации о банковском счете. Структура включала несколько полей, значения которых были присвоены и выведены на консоль. Это дало практическое представление о том, как структуры могут быть применены для хранения и обработки связанных данных.

ИСХОДНЫЙ КОД

Упражнение 1:

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

namespace BankAccount
{
    enum AccountType
    {
        Checking,
        Deposit
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            AccountType goldAccount;
            AccountType platinumAccount;
            goldAccount = AccountType.Checking;
            platinumAccount = AccountType.Deposit;
            Console.WriteLine(goldAccount);
            Console.WriteLine(platinumAccount);
        }
    }
}
```

Упражнение 2:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.Remoting.Messaging;
using System.Text;
using System.Threading.Tasks;

namespace StructType
{
    internal class Program
    {
        public enum AccountType
        {
            Checking,
            Deposit
        }
        public struct BankAccount
        {
            public long accNo;
            public decimal accBal;
        }
    }
}
```

```

        public AccountType accType;
    }
    static void Main(string[] args)
    {
        BankAccount goldAccount = new BankAccount();
        goldAccount.accNo = 123;
        goldAccount.accBal = 0;
        goldAccount.accType = AccountType.Checking;
        Console.WriteLine(goldAccount.accNo);
        Console.WriteLine(goldAccount.accBal);
        Console.WriteLine(goldAccount.accType);
    }
}

```