Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

**Лабораторная работа №3**

**по дисциплине**

**«Технологии программирования»**

**ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ**

**Выполнил**:

ст. гр. ПРИ-120

Д. А. Грачев

**Принял**:

Данилов В. В.

Владимир, 2021

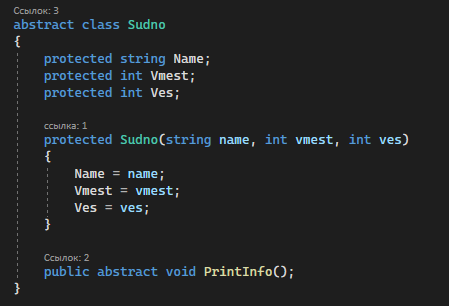
ЦЕЛЬ РАБОТЫ

Изучить основные концепции и особенности объектно-ориентированного программирования (ООП) в языке C#. Создать приложение, иллюстрирующее основные принципы ООП.

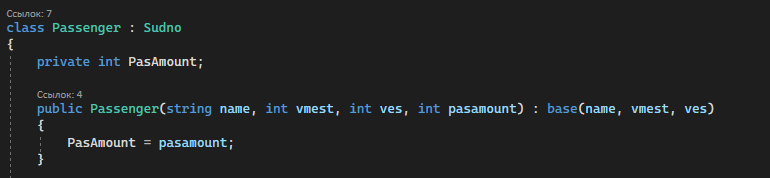
ВЫПОЛНЕНИЕ РАБОТЫ

**Предметная область:** «Морской грузовой порт» (примерные возможные объекты: порты, судна грузовые, пассажирские, речные объекты, грузы сотрудники, заказы на перевозку и т.д.);

1. Был создан абстрактный класс Sudno, описывающий основные данные о судне, а так же конструктор класса

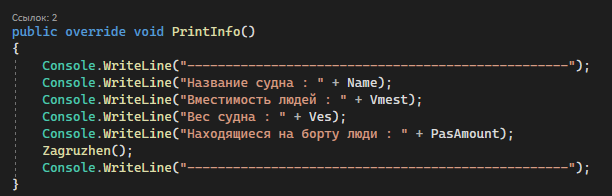


1. Был реализован один из принципов ООП(Объектно-ориентированного программирование) – Наследование; Был создан класс наследник Passenger

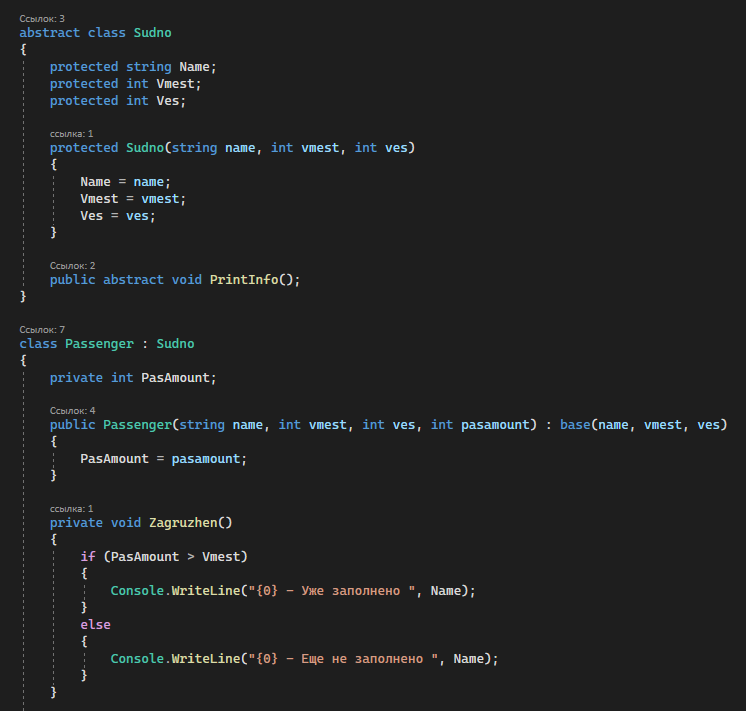


1. Так же был реализован другой принцип объектно-ориентированного программирования – Полиморфизм; Был переопределен метод абстрактного класса в классе наследнике

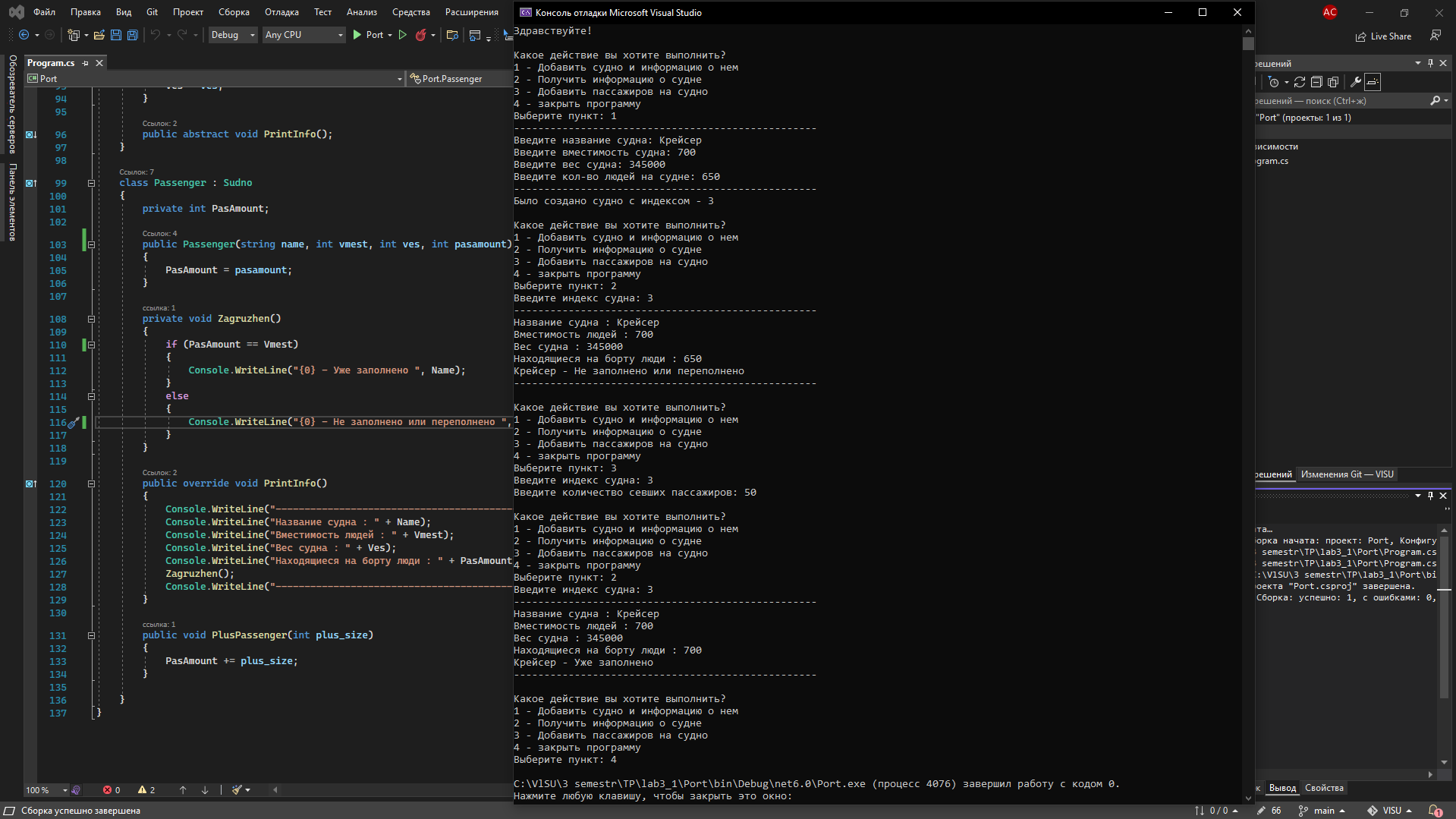




1. Был реализован третий принцип ООП – Инкапсуляция; Были созданы защищенные модификаторами доступа protected и private поля и методы класса



Скриншот работы программы представлен ниже



Листинг программы:

using System;

namespace Port

{

    class Program

    {

        static void Main(string[] args)

        {

            Passenger[] Spisok = new Passenger[7];

            Spisok[0] = new Passenger("Бригантина",520,1540000,330);

            Spisok[1] = new Passenger("Победа", 100, 120000, 100);

            Spisok[2] = new Passenger("Беда", 10, 15400, 0);

            byte LastInd = 3;

            Console.WriteLine("Здравствуйте!");

            while (true)

            {

                Console.WriteLine();

                Console.WriteLine("Какое действие вы хотите выполнить?");

                Console.WriteLine("1 - Добавить судно и информацию о нем");

                Console.WriteLine("2 - Получить информацию о судне");

                Console.WriteLine("3 - Добавить пассажиров на судно");

                Console.WriteLine("4 - закрыть программу");

                Console.Write("Выберите пункт: ");

                byte key = Convert.ToByte(Console.ReadLine());

                switch (key)

                {

                    case 1:

                        if (LastInd == 6)

                        {

                            Console.WriteLine("НЕ ОСТАЛОСЬ МЕСТА В ПОРТУ");

                            break;

                        }

                        Console.WriteLine("--------------------------------------------------");

                        Console.Write("Введите название судна: ");

                        string name = Console.ReadLine();

                        Console.Write("Введите вместимость судна: ");

                        int vmest = Convert.ToInt32(Console.ReadLine());

                        Console.Write("Введите вес судна: ");

                        int ves = Convert.ToInt32(Console.ReadLine());

                        Console.Write("Введите кол-во людей на судне: ");

                        int pasamount = Convert.ToInt32(Console.ReadLine());

                        Console.WriteLine("--------------------------------------------------");

                        Spisok[LastInd] = new Passenger(name, vmest, ves, pasamount);

                        Console.WriteLine("Было создано судно с индексом - " + LastInd);

                        LastInd++;

                        break;

                    case 2:

                        Console.Write("Введите индекс судна: ");

                        byte ind = Convert.ToByte(Console.ReadLine());

                        Spisok[ind].PrintInfo();

                        break;

                    case 3:

                        Console.Write("Введите индекс судна: ");

                        byte ind1 = Convert.ToByte(Console.ReadLine());

                        Console.Write("Введите количество севших пассажиров: ");

                        int plssize = Convert.ToInt32(Console.ReadLine());

                        Spisok[ind1].PlusPassenger(plssize);

                        break;

                    case 4:

                        Environment.Exit(0);

                        break;

                }

            }

        }

    }

    abstract class Sudno

    {

        protected string Name;

        protected int Vmest;

        protected int Ves;

        protected Sudno(string name, int vmest, int ves)

        {

            Name = name;

            Vmest = vmest;

            Ves = ves;

        }

        public abstract void PrintInfo();

    }

    class Passenger : Sudno

    {

        private int PasAmount;

        protected Passenger(string name, int vmest, int ves, int pasamount) : base(name, vmest, ves)

        {

            PasAmount = pasamount;

        }

        private void Zagruzhen()

        {

            if (PasAmount > Vmest)

            {

                Console.WriteLine("{0} - Уже заполнено ", Name);

            }

            else

            {

                Console.WriteLine("{0} - Еще не заполнено ", Name);

            }

        }

        public override void PrintInfo()

        {

            Console.WriteLine("--------------------------------------------------");

            Console.WriteLine("Название судна : " + Name);

            Console.WriteLine("Вместимость людей : " + Vmest);

            Console.WriteLine("Вес судна : " + Ves);

            Console.WriteLine("Находящиеся на борту люди : " + PasAmount);

            Zagruzhen();

            Console.WriteLine("--------------------------------------------------");

        }

        public void PlusPassenger(int plus\_size)

        {

            PasAmount += plus\_size;

        }

    }

}

ВЫВОД

В ходе выполнения лабораторной работы были изучены основные концепции и особенности объектно-ориентированного программирования (ООП) в языке C#. Создал приложение, иллюстрирующее основные принципы ООП.