



**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И
НАУКИ**

ГОРОДА МОСКВЫ

**Государственное бюджетное
профессиональное**

**образовательное учреждение города
Москвы**

«Колледж малого бизнеса № 4»

(ГБПОУ КМБ № 4)

**Отчёт по Домашнему заданию (Основы Алгоритмизации
и программирования).**

Специальность: 09.02.07 Информационные системы и программирование

Форма обучения: очная

Студент(ка): Геронок Артём Игоревич

Группа: ИПО-22.24

Руководитель: Рыбаков Александр Сергеевич

Отчётная работа защищена с оценкой «____» _____

Москва, 2026 г.

ЗАДАНИЕ:

Основные понятия объектно-ориентированного программирования.

1. Инкапсуляция

Практика:

1 Используя , напишите программный код следующих заданий.

1.1. Создайте класс Student. Определите в этом классе строковое поле name (имя).

1.2. Создайте класс Car. Определите в этом классе целочисленное поле year (год выпуска).

1.3. Реализуйте класс Point. Определите в этом классе целочисленное поле x.

1.4. Создайте класс Person с целочисленным полем age (возраст). Реализуйте в этом классе метод Print для печати поля age в консольном окне.

1.5. Разработайте класс Table с целочисленными полями rows (строки), cols (столбцы). Реализуйте в этом классе метод Display для отображения полей rows, cols в консольном окне.

1.6. Объявите класс Manager с полями age (целочисленный тип, возраст) и name (строковый тип, имя). Создайте метод GetAge для получения значения поля age и метод GetName для получения значения поля name.

1.7. Создайте класс Point3D с целочисленными полями x, y, z. Реализуйте метод Show для отобра-

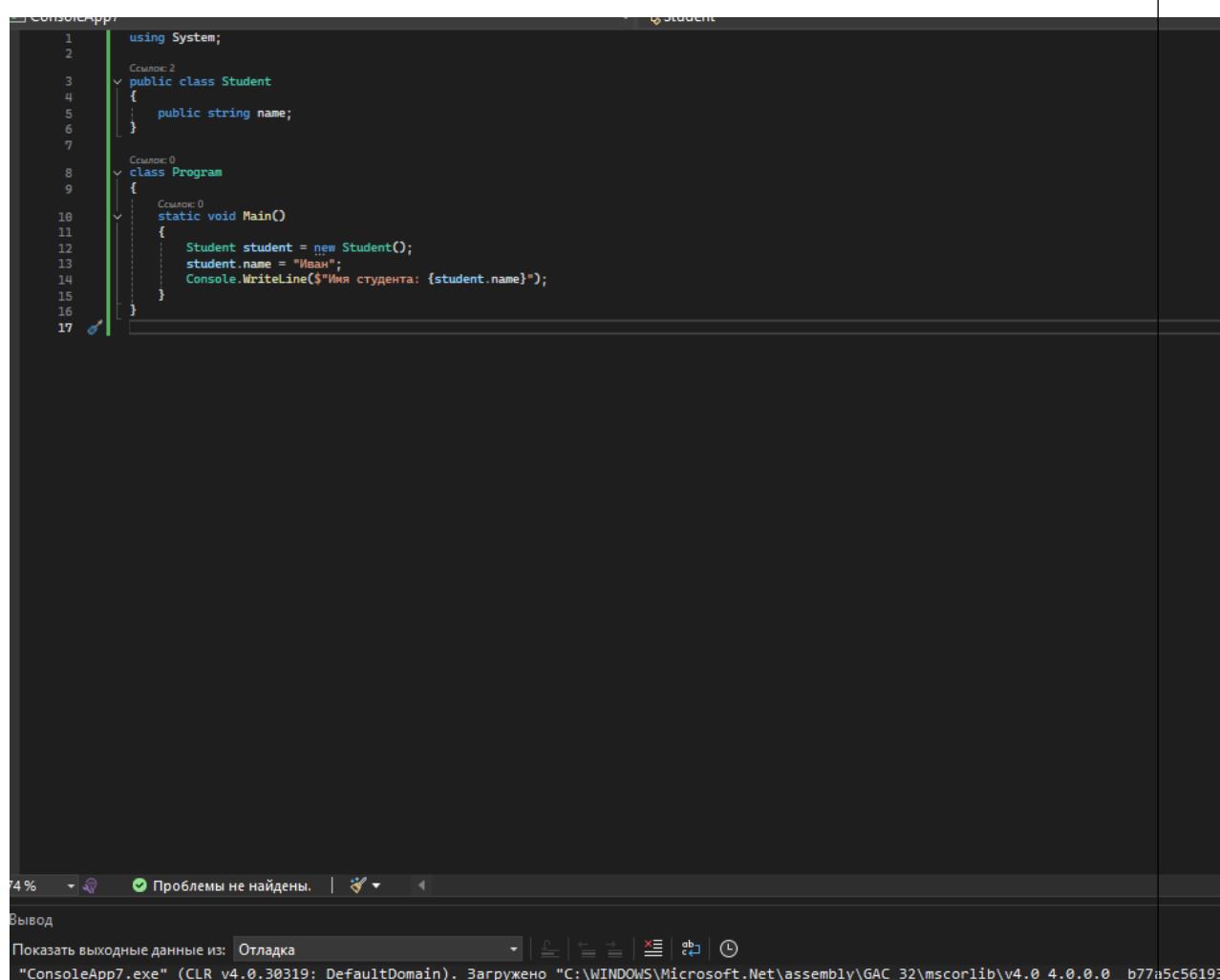
жения полей x, y, z в консольном окне.

1.8. Объявите класс Shop с строковым полем name (имя). В этом классе реализуйте метод GetName

для

получения значения поля name и метод SetName для установления нового значения полю name.

1.1



```
1  using System;
2
3  Символ: 2
4  public class Student
5  {
6      public string name;
7  }
8
9  Символ: 0
10 public class Program
11 {
12     Символ: 0
13     static void Main()
14     {
15         Student student = new Student();
16         student.name = "Иван";
17         Console.WriteLine($"Имя студента: {student.name}");
18     }
19 }
```

1.2

```

1  using System;
2
3  public class Car
4  {
5      public int year;
6  }
7
8  class Program
9  {
10     static void Main()
11     {
12         Car car = new Car();
13         car.year = 2020;
14         Console.WriteLine($"Год выпуска автомобиля: {car.year}");
15     }
16 }

```

Год выпуска автомобиля: 2020

C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 18128) завершил работу с кодом 0 (0x0).

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".

Нажмите любую клавишу, чтобы закрыть это окно:

1.3

```

1  using System;
2
3  public class Point
4  {
5      public int x;
6  }
7
8  class Program
9  {
10     static void Main()
11     {
12         Point point = new Point();
13         point.x = 10;
14         Console.WriteLine($"Координата X: {point.x}");
15     }
16 }

```

Координата X: 10

C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 3124) завершил работу с кодом 0 (0x0).

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".

Нажмите любую клавишу, чтобы закрыть это окно:

1.4

Отчет по дисциплине по Основы алгоритмизации и программирования

```
Program.cs
1  using System;
2
3  public class Person
4  {
5      public string name;
6      public int age;
7
8      public void Print()
9      {
10         Console.WriteLine($"Возраст: {age}");
11     }
12 }
13
14 class Program
15 {
16     static void Main()
17     {
18         Person person = new Person();
19         person.age = 25;
20         person.Print();
21     }
22 }
```

```
Консоль отладки Microsoft Vi
Возраст: 25

C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 24428) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

1.5

```
Program.cs
1  using System;
2
3  public class Table
4  {
5      public int rows;
6      public int cols;
7
8      public void Display()
9      {
10         Console.WriteLine($"Строки: {rows}, Столбцы: {cols}");
11     }
12 }
13
14 class Program
15 {
16     static void Main()
17     {
18         Table table = new Table();
19         table.rows = 5;
20         table.cols = 3;
21         table.Display();
22     }
23 }
```

```
Консоль отладки Microsoft Vi
Строки: 5, Столбцы: 3

C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 24608) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

1.6

```

1 Program.cs  ✘
2 ConsoleApp7
3
4 using System;
5
6 public class Manager
7 {
8     public int age;
9     public string name;
10
11     public int GetAge()
12     {
13         return age;
14     }
15
16     public string GetName()
17     {
18         return name;
19     }
20 }
21
22 class Program
23 {
24     static void Main()
25     {
26         Manager manager = new Manager();
27         manager.age = 35;
28         manager.name = "Алексей";
29         Console.WriteLine($"Name: {manager.GetName()}, Возраст: {manager.GetAge()}");
30     }
31 }

```

Консоль отладки Microsoft Visual Studio
Имя: Алексей, Возраст: 35
C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 2384) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

1.7

```

1 Program.cs  ✘
2 ConsoleApp7
3
4 using System;
5
6 public class Point3D
7 {
8     public int x;
9     public int y;
10    public int z;
11
12    public void Show()
13    {
14        Console.WriteLine($"{x}, {y}, {z}");
15    }
16 }
17
18 class Program
19 {
20     static void Main()
21     {
22         Point3D point = new Point3D();
23         point.x = 5;
24         point.y = 10;
25         point.z = 15;
26         point.Show();
27     }
28 }

```

Консоль отладки Microsoft Visual Studio
X: 5, Y: 10, Z: 15
C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 21696) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

1.8

Отчет по дисциплине по Основы алгоритмизации и программирования

```
using System;
class Shop
{
    public string name;
    public string GetName()
    {
        return name;
    }
    public void SetName(string newName)
    {
        name = newName;
    }
}
class Program
{
    static void Main()
    {
        Shop shop = new Shop();
        shop.SetName("Продукты");
        Console.WriteLine($"Название магазина: {shop.GetName()}");
    }
}
```

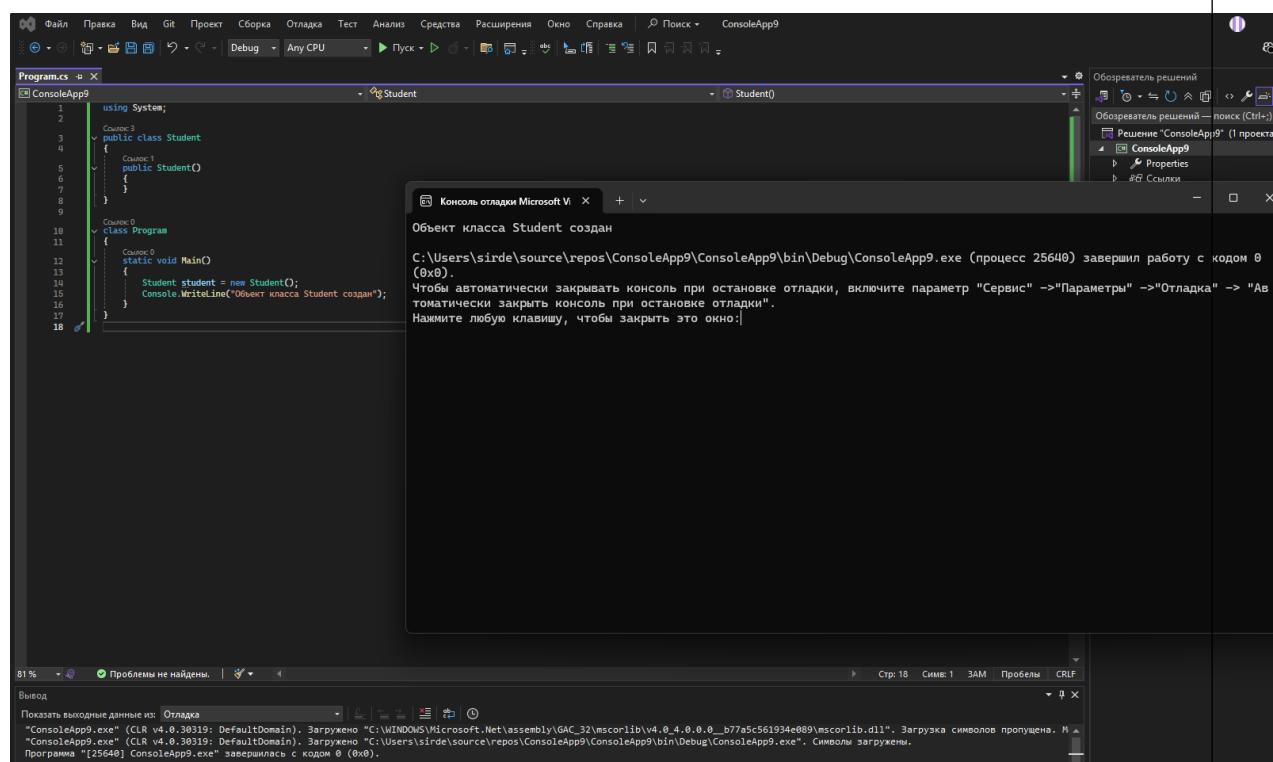
Консоль отладки Microsoft VI
Название магазина: Продукты
C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 26832) завершил работу с кодом 0 (0x0). Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки". Нажмите любую клавишу, чтобы закрыть это окно.

Вывод
Показать выходные данные из: Отладка
"ConsoleApp7.exe" (CLR v4.0.30319: DefaultDomain). Загружено "C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\mscorw.dll". Загружено "C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7.exe" (CLR v4.0.30319: DefaultDomain). Загружено "C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7.exe" завершилась с кодом 0 (0x0).

2. Конструкторы класса

Конструкторы, инициализаторы и деструкторы

2.1.



The screenshot shows the Microsoft Visual Studio IDE interface. On the left, the code editor displays `Program.cs` with the following C# code:

```

1  using System;
2
3  public class Student
4  {
5      public Student()
6      {
7          Console.WriteLine("Конструктор Student создан");
8      }
9
10 }
11
12 class Program
13 {
14     static void Main()
15     {
16         Student student = new Student();
17         Console.WriteLine("Объект класса Student создан");
18     }
}

```

In the center, the Output window shows the message "Объект класса Student создан". Below it, the Task List window shows the command "ConsoleApp9" was run. At the bottom, the Command Prompt window shows the application's output:

```

Показать выходные данные из: Отладка
"ConsoleApp9.exe" (CLR v4.0.30319; DefaultDomain). Загружено "C:\WINDOWS\Microsoft.NET\assembly\GAC_32\mscorlib\v4.0_4.0.0_0_b77a5c561934e089\mscorlib.dll". Загрузка символов пропущена. Модуль был загружен из памяти.
"ConsoleApp9.exe" (CLR v4.0.30319; DefaultDomain). Загружено "C:\Users\sirde\source\repos\ConsoleApp9\ConsoleApp9\bin\Debug\ConsoleApp9.exe". Символы загружены.
Программа "[25640] ConsoleApp9.exe" завершилась с кодом 0 (0x0).

```

2.2.

Отчет по дисциплине по Основы алгоритмизации и программирования

The screenshot shows the Microsoft Visual Studio IDE interface. On the left, the code editor displays `Program.cs` with the following C# code:

```
1  using System;
2
3  class Child
4  {
5      public Child()
6      {
7          Console.WriteLine("Создано один экземпляр класса Child");
8      }
9  }
10
11 class Program
12 {
13     static void Main()
14     {
15         Child child1 = new Child();
16         Child child2 = new Child();
17
18         Console.WriteLine("Создано два экземпляра класса Child");
19     }
}
```

To the right of the code editor is a terminal window titled "Консоль отладки Microsoft Visual Studio". It shows the output of the application's execution:

```
Создано два экземпляра класса Child
C:\Users\sirde\source\repos\ConsoleApp9\ConsoleApp9\bin\Debug\ConsoleApp9.exe (процесс 5456) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

The status bar at the bottom indicates "Стр: 19 Симв: 2 ЗАМ Проблемы CRLF".

2.3.

The screenshot shows the Microsoft Visual Studio IDE interface. On the left, the code editor displays `Program.cs` with the following C# code:

```
1  using System;
2
3  class Car
4  {
5      public int year;
6
7      public Car(int yearValue)
8      {
9          year = yearValue;
10     }
11 }
12
13 class Program
14 {
15     static void Main()
16     {
17         Car car = new Car(2022);
18         Console.WriteLine($"Год выпуска автомобиля: {car.year}");
19     }
20 }
```

To the right of the code editor is a terminal window titled "Консоль отладки Microsoft Visual Studio". It shows the output of the application's execution:

```
Год выпуска автомобиля: 2022
C:\Users\sirde\source\repos\ConsoleApp9\ConsoleApp9\bin\Debug\ConsoleApp9.exe (процесс 26060) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

The status bar at the bottom indicates "Стр: 20 Симв: 2 ЗАМ Проблемы CRLF".

2.4. Создайте класс Person с целочисленным полем age (возраст). Реализуйте в этом классе метод Print для печати поля age в консольном окне.

```

ConsoleApp9
1 using System;
2
3 namespace ConsoleApp9
4 {
5     public class Car
6     {
7         public string name;
8         public string color;
9
10        public Car(string nameValue, string colorValue)
11        {
12            name = nameValue;
13            color = colorValue;
14        }
15    }
16
17    class Program
18    {
19        static void Main()
20        {
21            Car lada = new Car("LADA VESTA", "black");
22            Car bmw = new Car("BMW X5", "white");
23
24            Console.WriteLine($"Автомобиль: {lada.name}, цвет: {lada.color}");
25            Console.WriteLine($"Автомобиль: {bmw.name}, цвет: {bmw.color}");
26        }
27    }
}

```

Консоль отладки Microsoft Visual Studio

Автомобиль: LADA VESTA, цвет: black
Автомобиль: BMW X5, цвет: white

C:\Users\sirde\source\repos\ConsoleApp9\ConsoleApp9\b1n\Debug\ConsoleApp9.exe (процесс 22056) завершил работу с кодом (0x0).

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

2.5. Разработайте класс Table с целочисленными полями rows (строки), cols (столбцы). Реализуйте в этом классе метод Display для отображения полей rows, cols в консольном окне.

```
ConsoleApp9
1  using System;
2
3  public class Product
4  {
5      protected string name;
6
7      public Product(Product other)
8      {
9          name = other.name;
10     }
11
12     public Product(string nameValue)
13     {
14         name = nameValue;
15     }
16
17     public void ShowName()
18     {
19         Console.WriteLine($"Название продукта: {name}");
20     }
21
22     class Program
23     {
24         static void Main()
25         {
26             Product original = new Product("Молоко");
27             Product copy = new Product(original);
28
29             original.ShowName();
30             copy.ShowName();
31         }
32     }
33 }
```

Консоль отладки Microsoft Visual Studio

Название продукта: Молоко
Название продукта: Молоко

C:\Users\sirde\source\repos\ConsoleApp9\ConsoleApp9\bin\Debug\ConsoleApp9.exe (процесс 6856) завершил работу с кодом 0x0.

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" ->"Автоматически закрывать консоль при остановке отладки".

Нажмите любую клавишу, чтобы закрыть это окно:

2.6. Объявите класс Manager с полями name (строковый тип, ленный тип, возраст) и age (целочисленный тип). Создайте метод GetAge для получения значения поля age и метод GetName для получения значения поля name.

ConsoleApp9

```
1  using System;
2
3  public class Person
4  {
5      private int age;
6
7      public Person()
8      {
9          age = 18;
10     }
11
12     public void Print()
13     {
14         Console.WriteLine($"Возраст: {age}");
15     }
16 }
17
18 class Program
19 {
20     static void Main()
21     {
22         Person person = new Person();
23         person.Print();
24     }
25 }
```

Консоль отладки Microsoft Visual Studio

Возраст: 18

C:\Users\sirde\source\repos\ConsoleApp9\ConsoleApp9\bin\Debug\ConsoleApp9.exe (процесс 26352) завершил работу с кодом 0 (0x0).

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".

Нажмите любую клавишу, чтобы закрыть это окно:

2.7

Program.cs

```
1 using System;
2
3 public class Manager
4 {
5     private int age;
6     private string name;
7
8     public Manager(int ageValue, string nameValue)
9     {
10         age = ageValue;
11         name = nameValue;
12     }
13
14     public Manager(Manager other)
15     {
16         age = other.age;
17         name = other.name;
18     }
19
20     public void Print()
21     {
22         Console.WriteLine($"Имя: {name}, Возраст: {age}");
23     }
24
25
26     class Program
27     {
28         static void Main()
29         {
30             Manager firstManager = new Manager(20, "Дамир");
31             Manager secondManager = new Manager(18, "Сара");
32
33             Console.WriteLine("Первый менеджер:");
34             firstManager.Print();
35
36             Console.WriteLine("Второй менеджер:");
37             secondManager.Print();
38         }
39     }
40 
```

Консоль отладки Microsoft Visual Studio

```
Первый менеджер:
Имя: Дамир, Возраст: 20
Второй менеджер:
Имя: Сара, Возраст: 18
```

C:\Users\sirde\source\repos\ConsoleApp9\ConsoleApp9\bin\Debug\ConsoleApp9.exe (процесс 7848) завершил работу с ошибкой 0x0.

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".

Нажмите любую клавишу, чтобы закрыть это окно:

Проблемы не найдены.

3 Свойства классов

Практика:

3 Используя, напишите программный код следующих заданий.

3.1. Создайте класс Student. Определите в этом классе строковое автосвойство Name (имя).

3.2. Создайте класс Child с открытыми стандартным конструктором и целочисленным автосвойством Age (возраст).

Создайте

экземпляра класса Child с инициализацией автосвойства Age.

3.3. Создайте класс Car. Определите в этом классе открытое целочисленное

стандартное
свойство Year, закрытое целочисленное поле year (год выпуска) и общедоступный стандартный конструктор. В
свойстве Year доступ к полю year осуществляется без
ограничений, а изменение значения поля year допустимо на
положительные числа.

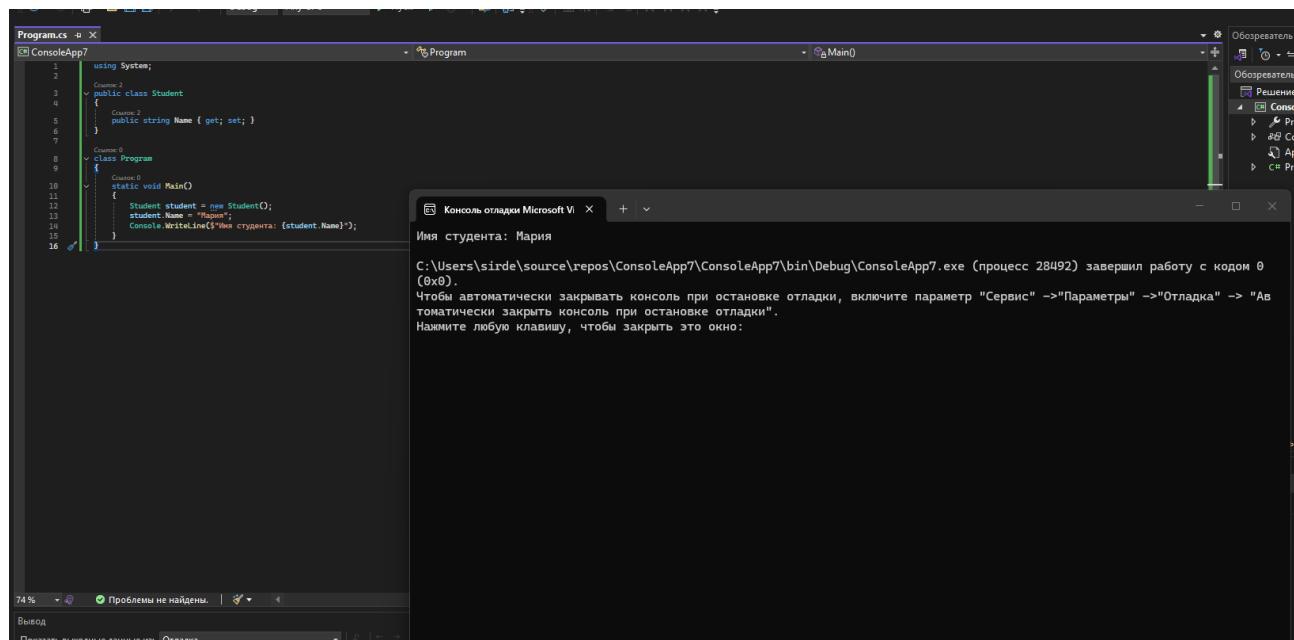
3.4. Определите класс Car с открытыми строковыми
автосвойствами Name (имя) и Color (цвет) и с открытым
стандартным конструктором. Используйте объектную
инициализацию

создании экземпляра класса Car. В создаваемом объекте
с свойству Name присвойте значение “KIA SOUL”, а

свойству Color – значение «green».

3.5. Создайте класс Product с защищенным строковым полем name. В этом классе реализуйте открытое свойство Name с ограниченным доступом. В свойстве Name невозможно изменять значение поля name, но можно получить доступ на чтение к полю name. Кроме того, создайте открытый стандартный конструктор с присвоением полю name значения «Рамиль».

3.1



```
1  using System;
2
3  public class Student
4  {
5      protected string Name { get; set; }
6  }
7
8  class Program
9  {
10     static void Main()
11     {
12         Student student = new Student();
13         student.Name = "Мария";
14         Console.WriteLine($"Имя студента: {student.Name}");
15     }
16 }
```

Консоль отладки Microsoft Visual Studio
Имя студента: Мария
C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 28492) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

3.2

```

Program.cs  ✘
ConsoleApp7  ✘ Program  ✘ Main()
1  using System;
2
3  public class Child
4  {
5      public int Age { get; set; }
6
7  }
8
9  class Program
10 {
11     static void Main()
12     {
13         Child child = new Child { Age = 7 };
14         Console.WriteLine($"Возраст ребенка: {child.Age}");
15     }
16 }

```

Консоль отладки Microsoft Vi ✘ +

Возраст ребенка: 7

C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 20144) завершил работу с кодом 0 (0x0).

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".

Нажмите любую клавишу, чтобы закрыть это окно:

3.3

```

Program.cs  ✘
ConsoleApp7  ✘ Program  ✘ Main()
1  using System;
2
3  public class Car
4  {
5      private int year;
6
7      public int Year
8      {
9          get { return year; }
10         set
11         {
12             if (value > 0)
13                 year = value;
14         }
15     }
16
17     class Program
18     {
19         static void Main()
20         {
21             Car car = new Car();
22             car.Year = -5;
23             Console.WriteLine($"Год выпуска: {car.Year}");
24         }
25     }
26 }

```

Консоль отладки Microsoft Vi ✘ +

Год выпуска: 2022

Год выпуска после попытки установить отрицательное значение: 2022

C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 4920) завершил работу с кодом 0 (0x0).

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".

Нажмите любую клавишу, чтобы закрыть это окно:

3.4

Отчет по дисциплине по Основы алгоритмизации и программирования

```
ConsoleApp7.cs
using System;
namespace ConsoleApp7
{
    public class Car
    {
        public string Name { get; set; }
        public string Color { get; set; }
    }
    class Program
    {
        static void Main()
        {
            Car car = new Car { Name = "KIA SOUL", Color = "green" };
            Console.WriteLine($"Автомобиль: {car.Name}, Цвет: {car.Color}");
        }
    }
}
```

Консоль отладки Microsoft Visual Studio
Автомобиль: KIA SOUL, Цвет: green
C:\Users\cirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 28144) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

3.5

```
Program.cs
using System;
namespace ConsoleApp7
{
    public class Product
    {
        protected string name;
        public string Name
        {
            get { return name; }
            private set { }
        }
    }
    public Product()
    {
        name = "Рамиль";
    }
    class Program
    {
        static void Main()
        {
            Product product = new Product();
            Console.WriteLine($"Название продукта: {product.Name}");
        }
    }
}
```

Консоль отладки Microsoft Visual Studio
Название продукта: Рамиль
C:\Users\cirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 28504) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

4 Наследование

Практика:

4. Используя, напишите программный код следующих заданий.

4.1. Создайте два класса Person и Student. Класс Student является дочерним, а класс Person – базовым. Установите корректное отношение наследования между классами Person и Student.

4.2. Создайте три класса Animal, Cat, Dog. Класс Animal является родительским для классов Cat, Dog. Установите правильные отношения наследования между этими тремя классами.

4.3. Создайте два класса Entity, Product. Entity является базовым классом, а Product – производным. Установите корректное

отношение

ния между двумя классами Entity, Product.

4.4. Пусть класс Dishes является родительским, а Cup – дочерним классом. Установить корректное отношение наследования между двумя классами Dishes, Cup.

4.5. Создайте три класса Entity, Staff, Manager. Класс Manager наследуется от класса Staff, а класс Staff от класса Entity. Установите правильные отношения наследования между этими классами.

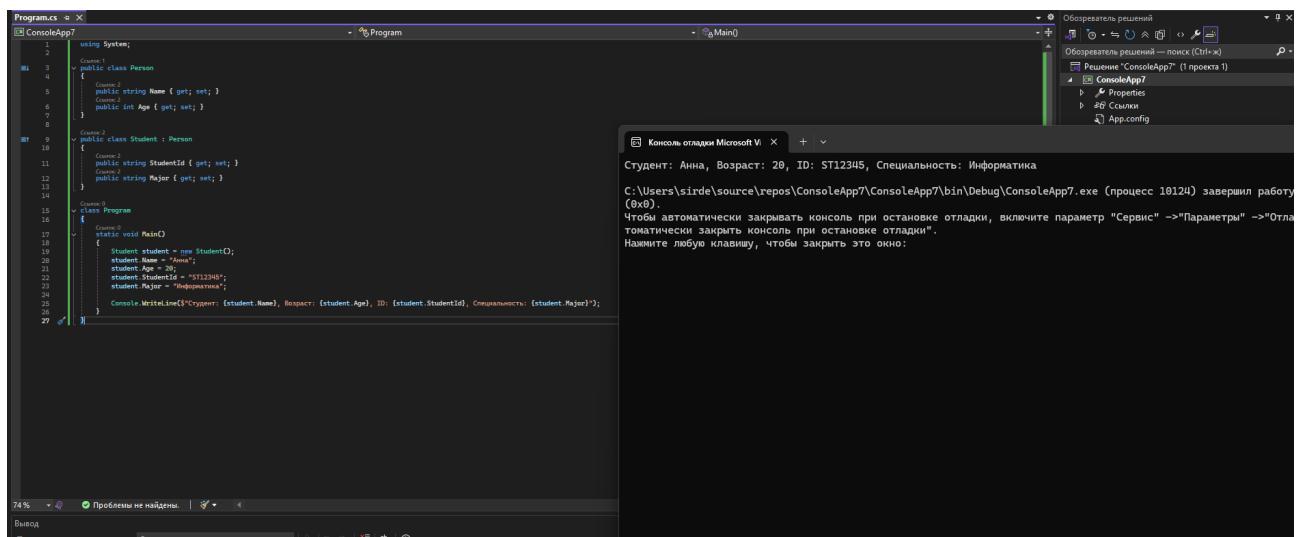
4.6. Реализуйте два класса Animal, Predator, где Animal – родительский класс, а Predator – дочерний. Создайте целочисленное поле age. Доступ к полю age возможен в контекстах классов Animal, Predator.

4.7. Пусть класс Transport является базовым, а SpaceShuttle – дочерний

класс.

строковое поле name в рамках классов Transport, SpaceShuttle.

4.1

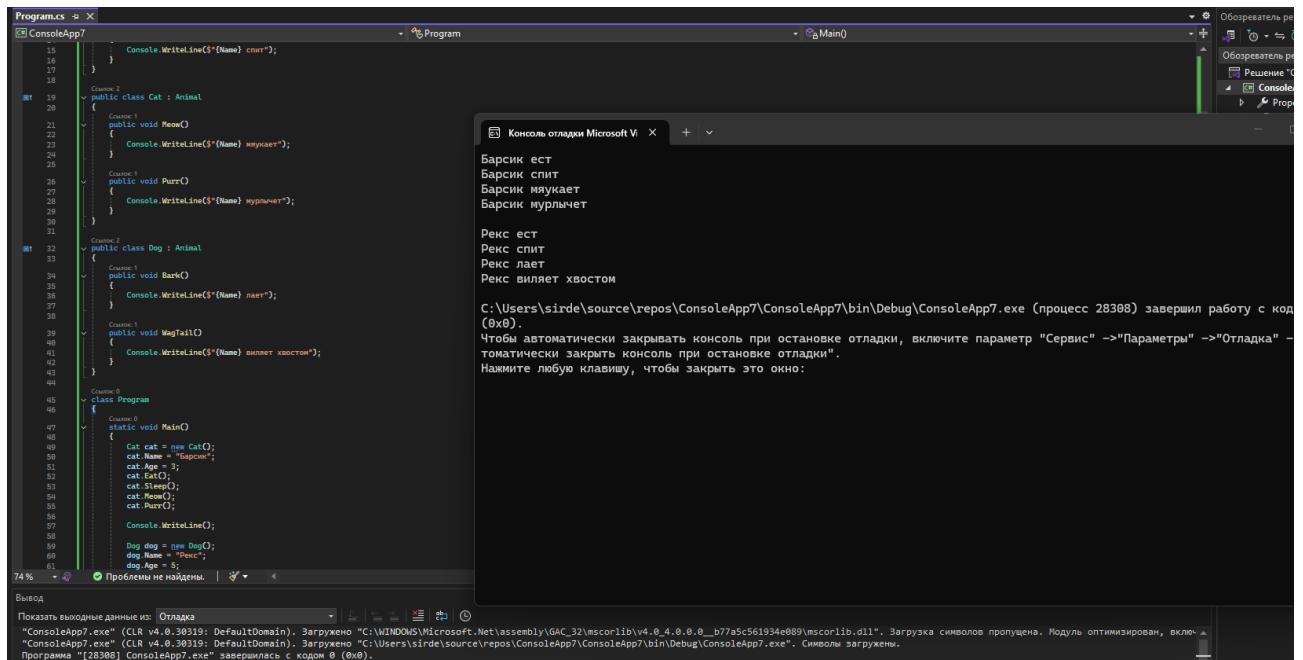


```

using System;
class Person
{
    public string Name { get; set; }
    public int Age { get; set; }
}
class Student : Person
{
    public string StudentId { get; set; }
    public string Major { get; set; }
}
class Program
{
    static void Main()
    {
        Student student = new Student();
        student.Name = "Анна";
        student.Age = 20;
        student.StudentId = "S12345";
        student.Major = "Информатика";
        Console.WriteLine($"Студент: {student.Name}, Возраст: {student.Age}, ID: {student.StudentId}, Специальность: {student.Major}");
    }
}

```

4.2



```

public class Animal
{
    public void Meow()
    {
        Console.WriteLine($"{Name} мяукает");
    }
    public void Purp()
    {
        Console.WriteLine($"{Name} мурлычет");
    }
}
public class Cat : Animal
{
    public void Bark()
    {
        Console.WriteLine($"{Name} мяу");
    }
}
public class Dog : Animal
{
    public void WagTail()
    {
        Console.WriteLine($"{Name} хвостом");
    }
}
class Program
{
    static void Main()
    {
        Cat cat = new Cat();
        cat.Name = "Барсик";
        cat.Age = 3;
        cat.Bark();
        cat.Sleep();
        cat.Meow();
        cat.Purp();
        Dog dog = new Dog();
        dog.Name = "Рекс";
        dog.Age = 5;
        dog.WagTail();
        Console.WriteLine();
    }
}

```

4.3

```

1 // Код для Entity
2 public class Entity
3 {
4     public int Id { get; set; }
5     public DateTime CreatedAt { get; set; }
6     public Entity()
7     {
8         CreatedAt = DateTime.Now;
9     }
10 }
11
12 // Код для Product : Entity
13 public class Product : Entity
14 {
15     public string Name { get; set; }
16     public decimal Price { get; set; }
17     public int Quantity { get; set; }
18
19     public void DisplayProductInfo()
20     {
21         Console.WriteLine($"ID: {Id}, Цена: {Price}, Количество: {Quantity}");
22     }
23 }
24
25 // Код для Program
26 class Program
27 {
28     static void Main()
29     {
30         Product product = new Product();
31         product.Id = 101;
32         product.Name = "Ноутбук";
33         product.Price = 75000;
34         product.Quantity = 10;
35
36         product.DisplayProductInfo();
37     }
38 }
39 
```

Консоль отладки Microsoft V... +

ID: 101, Создано: 14.02.2026 22:28:24
Товар: Ноутбук, Цена: 75 000,00 ?, Количество: 10
C:\Users\sirde\source\repos\ConsoleApp7\ConsoleApp7\bin\Debug\ConsoleApp7.exe (процесс 24412) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно!

4.4

```

1 // Код для Dishes
2 public class Dishes
3 {
4     public Material Material { get; set; }
5     public string Color { get; set; }
6     public double Weight { get; set; }
7
8     public void Wash()
9     {
10         Console.WriteLine("Посуда моется");
11     }
12
13     public void Dry()
14     {
15         Console.WriteLine("Посуда сушится");
16     }
17 }
18
19 // Код для Cup : Dishes
20 public class Cup : Dishes
21 {
22     public double Volume { get; set; }
23     public bool HasHandle { get; set; }
24
25     public void Fill()
26     {
27         Console.WriteLine("Чашка наполняется");
28     }
29
30     public void Drink()
31     {
32         Console.WriteLine("Пью из чашки");
33     }
34 }
35
36 // Код для Program
37 class Program
38 {
39     static void Main()
40     {
41         Cup cup = new Cup();
42         cup.Material = "Керамика";
43     }
44 }
45 
```

Консоль отладки Microsoft V... +

Посуда моется
Посуда сушится
Чашка наполняется
Пью из чашки
Чашка: материал – Керамика, цвет – Белый, объем – 250мл
C:\Users\sirde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 3936) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно!

4.5

Отчет по дисциплине по Основы алгоритмизации и программирования

The screenshot shows the Microsoft Visual Studio interface with the following details:

- Code Editor:** Displays the `Program.cs` file for a project named `ConsoleApp8`. The code defines a `Manager` class with properties `Name`, `TeamSize`, and `Department`, and methods `Work()`, `ManageTeam()`, and `ConductMeeting()`. It also contains a `Main()` method that creates a `Manager` object and prints its details.
- Output Window:** Shows the command-line output of the application, which includes the entity ID (1001), name (Алексей Петров), team size (8), department (Разработка), and a message about the application finishing work.
- Solution Explorer:** Shows the solution structure with the `ConsoleApp8` project selected.

4.6

The screenshot shows the Microsoft Visual Studio interface with the following details:

- Code Editor:** Displays the `Program.cs` file for a project named `ConsoleApp8`. The code defines an `Animal` class with a protected `age` field and a `Predator` class that inherits from it. The `Predator` class has methods `SetAge`, `ShowAge`, and `Hunt`. It also contains a `Main()` method that creates a `Predator` object and prints its age.
- Output Window:** Shows the command-line output of the application, which includes the animal's age (5), a message about the predator hunting, and a message about the predator's age being increased to 6.
- Solution Explorer:** Shows the solution structure with the `ConsoleApp8` project selected.

4.7

5 Полиморфизм

5. Используя , напишите программный код следующих заданий.

5.1. Создайте класс Strategy с виртуальным методом

Display. Метод Display распечатывает на консольном окне

сообщение «Strategy».

5.2. Создайте класс Weather с виртуальным методом Show.

Метод Show отображает на консольном окне сообщение

«My

Weather».

Создайте

пляр класса Weather и для созданного экземпляра класса

вызовите метод Show.

5.3.

Создайте два класса Strategy и ConservativeStrategy. Класс Strategy является базовым, а ConservativeStrategy – дочерним. Реализуйте виртуальный метод Display в классе Strategy и переопределите его в

дочернем классе ConservativeStrategy.

5.4.

Пусть Animal является базовым классом, а Cat – дочерним.

В классе Animal создайте закрытое строковое поле type, а в

классе Cat закрытое целочисленное поле age. Для классов

Animal, Cat реализуйте стандартные конструкторы. В

стандартном конструкторе класса Animal полю type присваивается значение “My Type”, а в стандартном

конструкторе класса Cat поле age инициализируется значением 5. Создайте виртуальным метод Print в классе Animal.

Методом

Print

в классе Animal распечатывается значение поля type, а в классе Cat распечатывается значение поля age.

Создайте по одному объекту классов Animal, Cat. Для каждого объекта вызовите метод Print.

5.5. Пусть Entity является абстрактным классом. В классе

Entity реализуйте абстрактный метод Display.

5.6.

Пусть Entity является абстрактным базовым классом, а Product – производным классом. Абстрактный метод

Display объявлен в классе Entity, а в классе Product этот

метод переопределён для отображения в консольном

окне сообщения «My Product». Создайте объект класса Pr

oduct и для этого объекта вызовите метод Display.

5.7. Создайте интерфейс IPrintable. В этом интерфейсе

определите метод Display без параметров.

Затем

определите класс ConsolePrinting с реализацией методов

интерфейса IPrintable для печати сообщения «My Conso
le»

на консольном окне. Создайте объект класса ConsolePri
nting

и вызовите для этого объекта метод Display.

5.1

Отчет по дисциплине по Основы алгоритмизации и программирования

```
ConsoleApp8
1  using System;
2
3  class Program
4  {
5      public void Display()
6      {
7          Console.WriteLine("Strategy");
8      }
9  }
10
11 class Concrete : Program
12 {
13     public override void Display()
14     {
15         Console.WriteLine("Concrete");
16     }
17 }
18
```

```
Консоль отладки Microsoft Vi
Strategy
```

C:\Users\sirde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 18992) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

5.2

```
ConsoleApp8
1  using System;
2
3  class Weather
4  {
5      public void Show()
6      {
7          Console.WriteLine("My Weather");
8      }
9  }
10
11 class Program
12 {
13     public static void Main()
14     {
15         Weather weather = new Weather();
16         weather.Show();
17     }
18 }
```

```
Консоль отладки Microsoft Vi
My Weather
```

C:\Users\sirde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 19956) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

81% Стр: 18 Симв: 1 Проблемы не найдены.

Вывод Показать выходные данные из: Отладка "ConsoleApp8.exe" (CoreCLR: cirhost). Загружено "C:\Program Files\dotnet\shared\Microsoft.NETCore.App\8.0.22\System.Security.Principal.Windows.dll". Загрузка символов пропущена. Модуль оптимизирован, включен параметр

5.3

```

using System;
namespace ConsoleApp8
{
    public class Strategy
    {
        public virtual void Display()
        {
            Console.WriteLine("Strategy");
        }
    }

    public class ConservativeStrategy : Strategy
    {
        public override void Display()
        {
            Console.WriteLine("ConservativeStrategy");
        }
    }

    class Program
    {
        static void Main()
        {
            Strategy strategy = new Strategy();
            strategy.Display();

            ConservativeStrategy conservative = new ConservativeStrategy();
            conservative.Display();

            Strategy polyStrategy = new PolyStrategy();
            polyStrategy.Display();
        }
    }
}

```

Консоль отладки Microsoft Vi

Strategy
ConservativeStrategy
ConservativeStrategy

C:\Users\srde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 27256) завершил работу
одом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -
томатически закрыть консоль при остановке отладки.
Нажмите любую клавишу, чтобы закрыть это окно!

5.4

```

using System;
namespace ConsoleApp8
{
    public class Animal
    {
        private string type;

        public Animal()
        {
            type = "My Type";
        }

        public virtual void Print()
        {
            Console.WriteLine($"Тип животного: {type}");
        }
    }

    public class Cat : Animal
    {
        private int age;

        public Cat()
        {
            age = 5;
        }

        public override void Print()
        {
            Console.WriteLine($"Возраст кота: {age}");
        }
    }

    class Program
    {
        static void Main()
        {
            Animal animal = new Animal();
            animal.Print();

            Cat cat = new Cat();
            cat.Print();

            Animal animalCat = new Cat();
            animalCat.Print();
        }
    }
}

```

Консоль отладки Microsoft Vi

Тип животного: My Type
Возраст кота: 5
Возраст кота: 5

C:\Users\srde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 7520) завершил работу
одом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -
томатически закрыть консоль при остановке отладки.
Нажмите любую клавишу, чтобы закрыть это окно!

5.5

Отчет по дисциплине по Основы алгоритмизации и программирования

```
ConsoleApp8.cs
1  using System;
2
3  public abstract class Entity
4  {
5      public abstract void Display();
6  }
7
8  class Program
9  {
10     static void Main()
11     {
12         Console.WriteLine("Abstract class Entity contains abstract method Display()");
13     }
14 }
```

```
Консоль отладки Microsoft Vi
Абстрактный класс Entity содержит абстрактный метод Display
C:\Users\sirde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 28200) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

5.6

```
ConsoleApp8.cs
1  using System;
2
3  public abstract class Entity
4  {
5      public abstract void Display();
6  }
7
8  public class Product : Entity
9  {
10     public override void Display()
11     {
12         Console.WriteLine("My Product");
13     }
14 }
15
16 class Program
17 {
18     static void Main()
19     {
20         Product product = new Product();
21         product.Display();
22     }
23 }
```

```
Консоль отладки Microsoft Vi
My Product
C:\Users\sirde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 15556) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

5.7

```
Program.cs
1  using System;
2
3  public interface IPrintable
4  {
5      void Display();
6  }
7
8  public class ConsolePrinting : IPrintable
9  {
10     public void Display()
11     {
12         Console.WriteLine("My Console");
13     }
14 }
15
16 class Program
17 {
18     static void Main()
19     {
20         ConsolePrinting consolePrinting = new ConsolePrinting();
21         consolePrinting.Display();
22     }
23 }
```

```
Консоль отладки Microsoft Vi
My Console
C:\Users\sirde\source\repos\ConsoleApp8\ConsoleApp8\bin\Debug\net8.0\ConsoleApp8.exe (процесс 6048) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

Выход
Показать выходные данные из: Отладка
"ConsoleApp8.exe" (CoreCLR: clrhost). Загружено "C:\Program Files\dotnet\shared\Microsoft.NETCore.App\8.0.22\System.Security.Principal.Windows.dll". Загрузка символов пропущена. Модуль оптимизирован, включен параметр отладки
"ConsoleApp8.exe" (CoreCLR: clrhost). Загружено "C:\Program Files\dotnet\shared\Microsoft.NETCore.App\8.0.22\System.Security.Claims.dll". Загрузка символов пропущена. Модуль оптимизирован, включен параметр отладки
"ConsoleApp8.exe" (CoreCLR: clrhost). Загружено "C:\Program Files\dotnet\shared\Microsoft.NETCore.App\8.0.22\System.Runtime.Loader.dll". Загрузка символов пропущена. Модуль оптимизирован, включен параметр отладки
"ConsoleApp8.exe" (CoreCLR: clrhost). Загружено "C:\Program Files\dotnet\shared\Microsoft.NETCore.App\8.0.22\System.Text.Encoding.Extensions.dll". Загрузка символов пропущена. Модуль оптимизирован, включен параметр отладки

Официальная литература:

1 Microsoft, Learn. Документация по C#. Вложенные типы. —

2024

— URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/classes-and-structs/nested-types> (дата обр. 27.02.2024).

2 Microsoft, Learn. Документация по C#. Конструкторы.

— 2024

— URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/classes-and-structs/constructors> (дата обр. 27.02.2024).

3 Microsoft, Learn. Документация по C#. Наследование.

— 2024

— URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/fundamentals/object-oriented/inheritance> (дата обр. 27.02.2024).

4 Microsoft, Learn. Документация по C#. Полиморфизм.

— 2024

— URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/fundamentals/object-oriented/polymorphism> (дата обр. 27.02.2024).

5 Microsoft, Learn. Документация по C#. Свойства. —

2024 —

URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/properties> (дата обр. 27.02.2024).

6 Microsoft, Learn. Документация по C#. Стандартные блоки про-

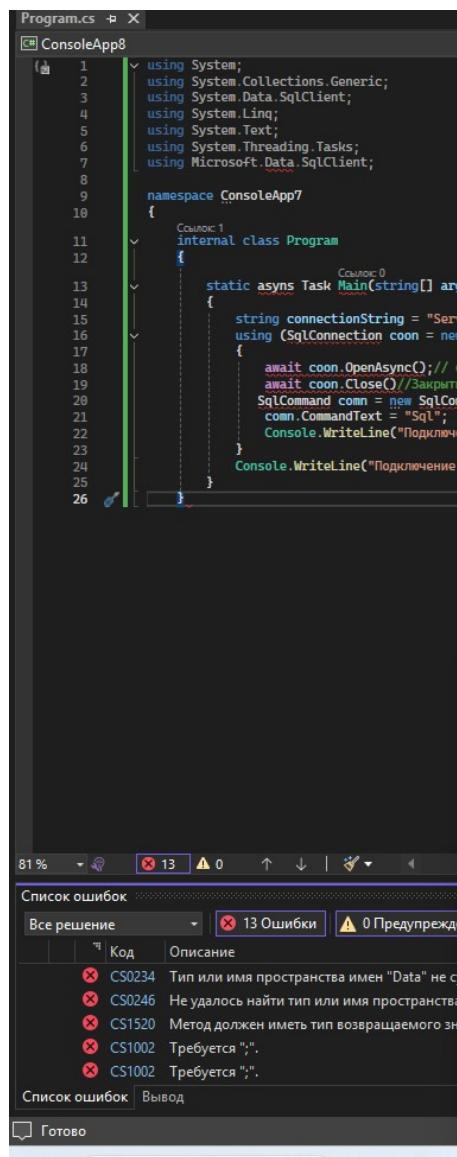
грамммы. — 2024 — URL: <https://docs.microsoft.com>

/ ru - ru / dotnet / csharp / tour -
of -
csharp / program -
building -
27.02.2024).

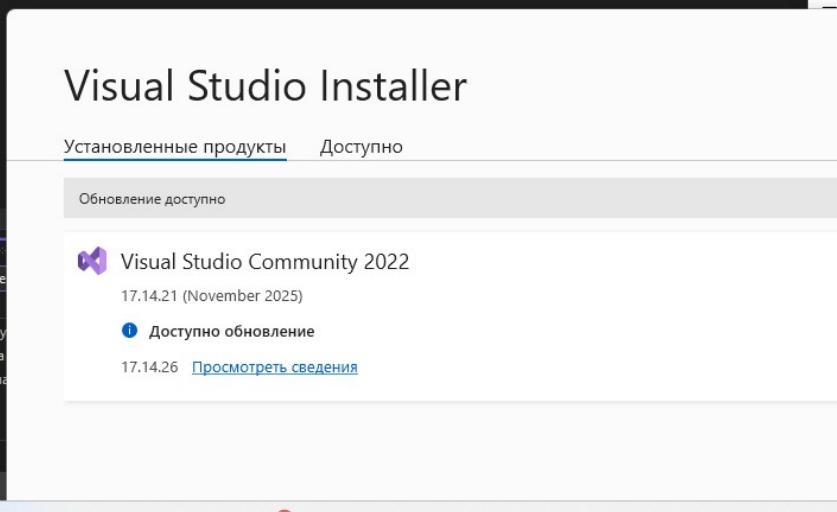
7 Microsoft, Learn. Документация по C#. Типы и члены.
— 2024
— URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/types> (дата обр.
27.02.2024).

Практическая работа подключение к MySQL server .docx

Отчет по дисциплине по Основы алгоритмизации и программирования



The screenshot shows the Visual Studio IDE interface. On the left, the code editor displays `Program.cs` with C# code for connecting to a local database and executing a SQL query. The code includes imports for System, System.Collections.Generic, System.Data.SqlClient, System.Linq, System.Text, System.Threading.Tasks, and Microsoft.Data.SqlClient. It defines a static asynchronous `Main` method that creates a `SqlConnection`, opens it, executes a query, and then closes it. The output window at the bottom shows 13 errors, mostly related to syntax and type issues.



The screenshot shows the Visual Studio Installer window. It lists "Установленные продукты" (Installed products) under the "Доступно" (Available) tab. It shows that Visual Studio Community 2022 version 17.14.21 (November 2025) is installed and has an available update to version 17.14.26. A link to "Просмотреть сведения" (View details) is provided.