**Министерство науки и высшего образования российской федерации**

**ФГАОУ ВО «Пермский государственный национальный исследовательский университет»**

Физический факультет

Технологии .NET Framework

Лабораторная работа №4 на тему

**«****Классы»**

Выполнил:

студент 2 курса физического факультета

группа ПМИ-2-2021

направление «01.03.02 Прикладная математика и информатика»

профиль «Инженерия программного обеспечения»

Макарова Полина Фёдоровна

Пермь, 2023 г.

1. Цель работы

Научиться:

• изучить структуру и принципы объявления классов, освоить технологию создания экземпляров классов (объектов). Задачи лабораторной работы:

• научиться объявлять классы;

• научиться создавать объекты классов;

• научиться работать с полями данных и методами классов.

1. Диаграмма классов

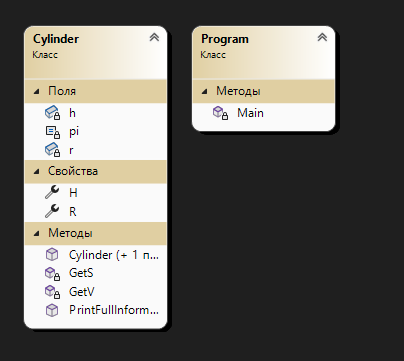


Рисунок – Диаграмма классов

1. Листинг программы

|  |
| --- |
| Листинг программы 1 и 2 инд.задания (уже модифицированная) |
| /\* Макарова Полина ПМИ-2  \* Вариант 10  \* Лабораторная работа 4  \*/  /\* 1. Спроектируйте класс (ы), наполните его (их) требуемой функциональностью,  \* продемонстрируйте работоспособность программы  \* Класс "Цилиндр".  \* Реализовать ввод и вывод полей данных, вычисление объёма, площади поверхности,  \* а так же вывод информации об объекте.  \*  \* 2. Модифицируйте вашу программу таким образом, чтобы она использовала информацию,  \* полученную в том числе и на лекциях. А именно (по возможности и целесообразности)  \* продемонстрировать: инкапсуляцию; применение полей, свойств, конструкторов;  \* включение, вложение, наследование классов; перегрузки методов/конструкторов;  \* возможность задание параметров «по умолчанию» и с клавиатуры;  \* задание параметров консоли и форматный вывод.  \*/  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  namespace Lab\_4  {  class Cylinder  {  // Поля данных: r - радиус, h - высота, pi - константа пи  private double r, h;  private const double pi = 3.141;  // Конструктор по умолчанию  public Cylinder()  {  r = 1;  h = 1;  }  // Конструктор  public Cylinder(double ra, double ha)  {  r = ra;  h = ha;  }  // Метод для установки и считывания значений полей  public double H  {  get { return this.h; }  set { this.h = value; }  }  public double R  {  get { return this.r; }  set { this.r = value; }  }  // Метод для вычисления объёма цилиндра  private double GetV()  {  return pi \* r \* r \* h;  }  // Метод для вычисления площади поверхности цилиндра  private double GetS()  {  return 2 \* pi \* r \* h + 2 \* pi \* r \* r;  }  // Метод для вывода полной информации об объекте в консоль  public void PrintFullInformation()  {  string str = "\n --------------------------------\n" +  " | |\n" +  " | Объект Цилиндр |\n" +  " | |\n" +  " --------------------------------\n";  Console.WriteLine(str);  Console.WriteLine(" Данные Цилиндра:\n\n Высота = {0:F3}\n Радиус = {1:F3}\n", h, r);  Console.WriteLine("\n Объём Цилиндра = {0:F3}\n", GetV());  Console.WriteLine(" Площадь поверхности Цилиндра = {0:F3}\n", GetS());  }  }  class Program  {  static void Main(string[] args)  {  // Параметры консоли  Console.Title = "Цилиндр";  Console.SetWindowSize(60, 30);  Console.BackgroundColor = ConsoleColor.White;  Console.ForegroundColor = ConsoleColor.Black;  Console.Clear();  string a;  double r, h;  // Создание объекта Цилиндр  Cylinder C;  C = new Cylinder();  Console.Write(" Введите 1 для ввода данных из консили\n" +  " или любую цифру для задания параметров по умолчанию\n ");  a = Console.ReadLine();  // Ввод данных из консоли  if (Convert.ToDouble(a) == 1)  {  Console.Write("\n Введите высоту Цилиндра = ");  a = Console.ReadLine();  C.H = (Convert.ToDouble(a));  Console.Write("\n Введите радиус Цилиндра = ");  a = Console.ReadLine();  C.R = (Convert.ToDouble(a));  }  // Вывод информации об объекте  C.PrintFullInformation();  Console.ReadKey();  }  }  } |

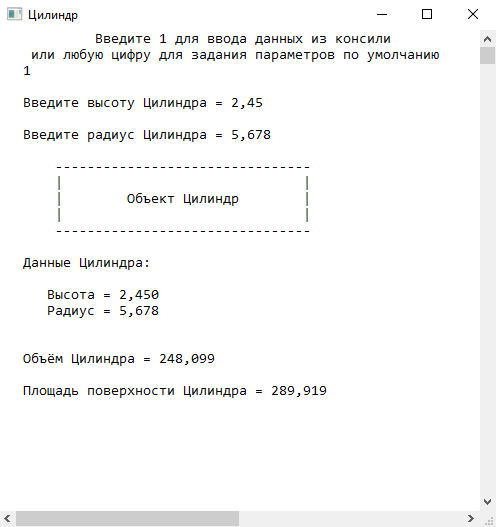


Рисунок 2 - Результат выполнения программы

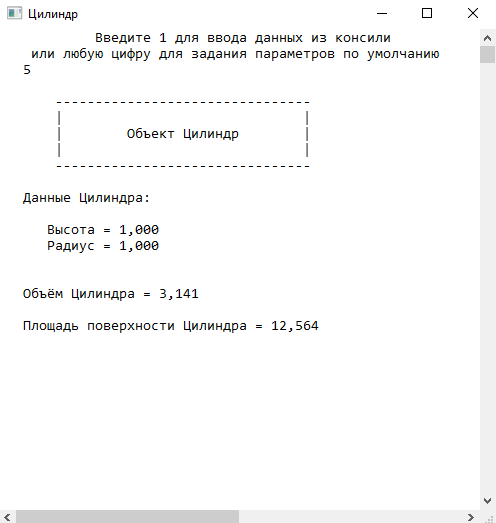


Рисунок 3 – Второй результат выполнения программы

1. Ответы на вопросы

***1. Что такое класс? К какому типу C# относится?***

Класс — это тип данных, объединяющий данные и методы их обработки, пользовательский шаблон, в соответствии с которым можно создавать объекты. Классы -это ссылочные типы C#.

***2. Какие элементы класса определяют состояние, а какие поведение класса?***

Состояние определяется такими элементами, как поля, константы, а поведение — функциями-членами класса.

***3. Что такое элементы класса? Какие группы элементов класса вы знаете?***

Элементы класса — это члены класса, которые составляют его тело; данные и функции, объявленные внутри класса. Можно разделить на 2 группы: данные и функции-члены.

***4. Какие типы элементов-данных вы знаете?***

Среди элементов-данных можно выделить поля и константы.

***5. Какие типы функций-членов класса вы знаете?***

Среди элементов-функций можно выделить методы, свойства, конструкторы, финализаторы, операции, индексаторы.

***6. Какие модификаторы доступности членов класса вы знаете?***

Существуют следующие модификаторы доступности членов классов: public, private, protected, internal или internal protected.

***7. Что такое защищенный член класса?***

У защищённого члена класса (protected) доступ осуществляется только из данного и производных классов.

***8. Что такое сигнатура?***

Сигнатура — это имя метода вкупе с количеством, типами и модификаторами его параметров.

***9. Какое ключевое слово используется для создания объекта класса?***

Для создания экземпляра класса используется ключевое слово new.

***10.Что такое конструктор класса? Их виды.***

Конструкторы — это метод класса, который не возвращает значения и имеет то же самое имя, что и класс. Если конструктор класса не определен программистом явно, то компилятор создаст конструктор по умолчанию. Различают 2 вида:

• конструкторы по умолчанию автоматически создаётся компилятором, если программист не описал собственный конструктор в классе; не имеет параметров;

• пользовательские конструкторы описывается программистом; если есть несколько пользовательских конструкторов в одном классе, то они различаются сигнатурой/

***11.Что такое перегрузка методов? Приведите пример.***

Перегрузка методов — возможность существования нескольких версий одного метода, но с разными сигнатурами (методы отличаются количеством и / или типом параметров). Чтобы перегрузить метод, просто объявляются методы с одинаковыми именами, но разными сигнатурами.

Пример:

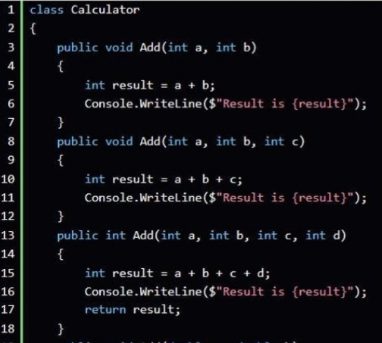


Рисунок 4 – Пример перегрузки метода

***12.Может ли один конструктор класса вызывать другой конструктор?***

Да, может.

***13.Что такое свойство класса? Для чего используется?***

Свойства — это наборы функций, которые могут быть доступны клиенту таким же способом, как общедоступные поля класса. Они предоставляют защищённый доступ к полям.

***14.Что такое полиморфизм и есть ли он в вашей программе?***

Полиморфизм — способность функции обрабатывать данные разных типов.

Его нет в моей программе.

***15.Что такое инкапсуляция и есть ли она в вашей программе?***

Инкапсуляция — это сокрытие элементов класса (например, метода или поля).

Она есть в моей программе.

Пример:

У меня встречаются инкапсулированные поля и методы класса.

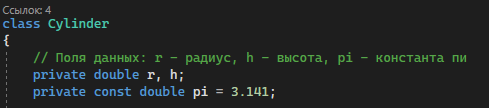


Рисунок 5 – Инкапсулированные поля класса

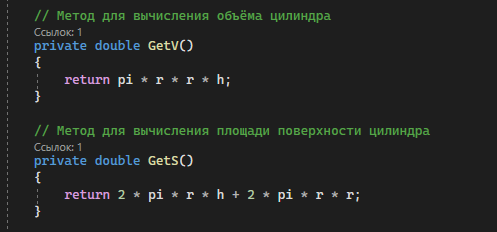


Рисунок 6 - Инкапсулированные методы класса