**17 Сборки, библиотеки, атрибуты, директивы**

Задание №1. Создать библиотеку с двумя классами:

треугольник (методы ввода сторон, проверки на существование, вычисления периметра, вычисления площади, определения вида треугольника (разносторонний, равнобедренный, равносторонний));

прямоугольник (методы ввода сторон, вычисления периметра, вычисления площади).

Листинг программы:

public class Triangle

{

int a;

int b;

int c;

public Triangle(int a, int b, int c)

{

this.a = a;

this.b = b;

this.c = c;

}

static bool TriangleExists(int a, int b, int c)

{

if (a + b > c && b + c > a && c + a > b && a > 0 && b > 0 && c > 0)

{

return true;

}

else

{

return false;

}

}

static int Perim(int a, int b, int c)

{

int res = a + b + c;

return res; }

static double Square(int a, int b, int c)

{

double p2 = (a + b + c) / 2;

double S = Math.Sqrt(p2 \* (p2 - a) \* (p2 - b) \* (p2 - c));

return S;

}

static string Type(int a, int b, int c)

{

if (a == b && a == c)

return "Equilateral";

else if (a != b && a != c && b != c)

return "Versatile";

else return "Isosceles";

}

}

public class Rectangle

{

int a;

int b;

public Rectangle(int a, int b)

{

this.a = a;

this.b = b;

}

static int Perim(int a, int b)

{

int res = 2 \* (a + b);

return res;

}

static double Square(int a, int b)

{

double S = a \* b;

return S; }

}

Таблица 17.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| a, b, c | Square, Type, Perim, TriangleExists, |

Источник: собственная разработка

Анализ результата разработки библиотеки представлен на рисунке 17.1.

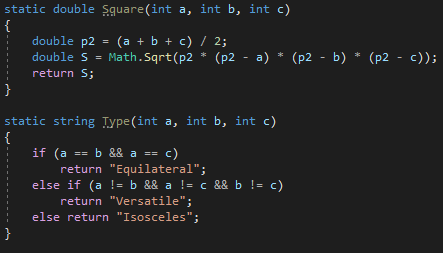


Рисунок 17.1 – Код разработанной библиотеки

Источник: собственная разработка

Задание №2. Создать свою библиотеку на индивидуальную тему «Животные» и продемонстрировать ее функциональность.

Листинг программы:

public class Animal

{

string variety;

string food;

int weight;

string color;

public Animal(string variety, string food, int weight, string color)

{

this.variety = variety;

this.food = food;

this.weight = weight;

this.color = color;

}

public string AnimalInfo()

{

string animalInf = $"Animal variety: {variety}\nthis animal eats: {food}\nweight is {weight}\nwool color: {color}";

return animalInf; }

}

Console.WriteLine("Enter animal's data. ");

Console.Write("Enter animal variety: ");

string variety = Console.ReadLine();

Console.Write("Enter what type of food this animal eats: ");

string food = Console.ReadLine();

Console.Write("Enter animal's weight: ");

int weight = int.Parse(Console.ReadLine());

Console.Write("Enter animal's wool color : ");

string color = Console.ReadLine();

Animal inf = new Animal(variety, food, weight, color);

Console.WriteLine(inf.AnimalInfo());

Таблица 17.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Animal’s data | Herbivore, grass, 45, white |

Источник: собственная разработка

Анализ результатов представлен на рисунке 17.2.

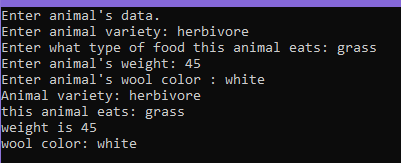


Рисунок 17.2 – Результат работы программы

Источник: собственная разработка