**12 ДЕЛЕГАТЫ**

Задание 1. Объявить тип делегата, который ссылается на метод.

Листинг программы:

using System;

public delegate double CalcFigure(double r);

class Program

{

static void Main(string[] args)

{

CalcFigure CF;

CF = Get\_Length;

Console.WriteLine("Длина окружности: " + CF(5));

CF = Get\_Area;

Console.WriteLine("Площадь круга: " + CF(5));

CF = Get\_Volume;

Console.WriteLine("Объем шара: " + CF(5));

Console.ReadKey();

}

public static double Get\_Length(double r)

{

return 2 \* Math.PI \* r;

}

public static double Get\_Area(double r)

{

return Math.PI \* Math.Pow(r, 2);

}

public static double Get\_Volume(double r)

{

return 4.0 / 3.0 \* Math.PI \* Math.Pow(r, 3);

}

}

Таблица 12.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | 31,41592653589793  78,53981633974483 |

Анализ результатов:



Рисунок 12.1 – Результат работы программы

Задание 2 Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте четыре лямбда оператора для выполнения арифметических

действий: (Add – сложение, Sub – вычитание, Mul – умножение, Div – деление).

Каждый лямбда оператор должен принимать два аргумента и возвращать

результат вычисления. Лямбда оператор деления должен делать проверку

деления на ноль.

Листинг программы:

using System;

class Program

{

static void Main(string[] args)

{

Func<double, double, double> Add = (x, y) => x + y;

Func<double, double, double> Sub = (x, y) => x - y;

Func<double, double, double> Mul = (x, y) => x \* y;

Func<double, double, double> Div = (x, y) => y != 0 ? x / y : throw new DivideByZeroException();

Console.WriteLine("Введите два числа:");

double num1 = Convert.ToDouble(Console.ReadLine());

double num2 = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Введите операцию (Add, Sub, Mul, Div):");

string operation = Console.ReadLine();

double result = 0;

switch (operation)

{

case "Add":

result = Add(num1, num2);

break;

case "Sub":

result = Sub(num1, num2);

break;

case "Mul":

result = Mul(num1, num2);

break;

case "Div":

try

{

result = Div(num1, num2);

}

catch (DivideByZeroException)

{

Console.WriteLine("Ошибка: деление на ноль!");

return;

}

break;

default:

Console.WriteLine("Неизвестная операция!");

return;

}

Console.WriteLine($"Результат: {result}");

}

}

Таблица 12.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 10  4  Add | 14 |

Анализ результатов:

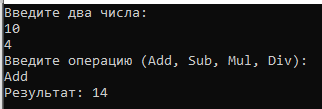


Рисунок 12.2 – Результат работы программы

Задание 3. Создать 3 метода для работы со строкой. Создать делегат. В программе вызывать все три метода при помощи делегата.

Листинг программы:

using System;

class Program

{

delegate string StringDelegate(string str);

static string Reverse(string str)

{

char[] charArray = str.ToCharArray();

Array.Reverse(charArray);

return new string(charArray);

}

static string ToUpper(string str)

{

return str.ToUpper();

}

static string ToLower(string str)

{

return str.ToLower();

}

static void Main(string[] args)

{

StringDelegate stringDelegate;

Console.WriteLine("Введите строку:");

string input = Console.ReadLine();

stringDelegate = Reverse;

string result = stringDelegate(input);

Console.WriteLine($"Перевернутая строка: {result}");

stringDelegate = ToUpper;

result = stringDelegate(input);

Console.WriteLine($"Строка в верхнем регистре: {result}");

stringDelegate = ToLower;

result = stringDelegate(input);

Console.WriteLine($"Строка в нижнем регистре: {result}");

}

}

Таблица 12.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Война и мир | Рим и анйов  ВОЙНА И МИР |

Анализ результатов:

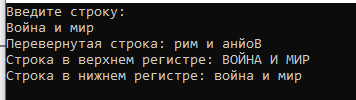


Рисунок 12.3 – Результат работы программы

Задание 4. Используя Visual Studio, создайте проект по шаблону Console Application. Создайте анонимный метод, который принимает в качестве аргумента

массив делегатов и возвращает среднее арифметическое возвращаемых значений методов сообщенных с делегатами в массиве. Методы, сообщенные с

делегатами из массива, возвращают случайное значение типа int.

Листинг программы:

using System;

class Program

{

delegate int RandomNumberMethod();

static void Main()

{

RandomNumberMethod[] methods = new RandomNumberMethod[5];

Random random = new Random();

for (int i = 0; i < methods.Length; i++)

{

methods[i] = delegate () { return random.Next(1, 100); };

}

double average = CalculateAverage(methods);

Console.WriteLine("Среднее значение: " + average);

}

static double CalculateAverage(RandomNumberMethod[] methods)

{

int sum = 0;

foreach (var method in methods)

{

sum += method();

}

return (double)sum / methods.Length;

}

}

Таблица 12.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | 61,8 |

Анализ результатов:



Рисунок 12.4 – Результат работы программы