**Методы**

Проект 1. Создайте проект с работой метода – процедурой

using System;

// Методы (Процедуры).

namespace Methods

{

class Program

{

// На 12-й строке, создаем метод с именем Procedure, который ничего не принимает и ничего не возвращает.

// В теле метода, на 14-й строке выводим на экран строку - Hello!

static void Procedure()

{

Console.WriteLine("Hello!");

}

static void Main()

{

// В теле метода Main на 21-й строке, вызываем метод Procedure.

Procedure();

// Delay.

Console.ReadKey();

}

}

}

Проект 2.

Метод – функция

using System;

// Методы (Функции).

namespace Methods

{

class Program

{

// На 12-й строке, создаем метод с именем Function, который ничего не принимает и возвращает строковое значение.

// В теле метода, используя ключевое слово return, возвращаем строку - Hello!

static string Function()

{

return "Hello!";

}

static void Main()

{

// В теле метода Main на 22-й строке, создаем строковую локальную переменную с именем @string

// и присваиваем ей возвращаемое значение метода Function.

string @string = Function();

Console.WriteLine(@string);

// Delay.

Console.ReadKey();

}

}

}

Проект 3.

using System;

// Методы (Функции).

namespace Methods

{

class Program

{

static string Function()

{

string word = "Hello!";

return word;

}

static void Main()

{

string word = Function();

Console.WriteLine(word);

// Delay.

Console.ReadKey();

}

}

}

Проект 4.

Использование параметров

using System;

namespace \_005\_Methods

{

// Методы (Функции).

class Program

{

/// <summary>

/// Сложение двух целых чисел.

/// </summary>

/// <param name="summand1">Первое слагаемое</param>

/// <param name="summand2">Второе слагаемое</param>

/// <returns>Сумма</returns>

static int Add(int summand1, int summand2)

{

return summand1 + summand2;

}

static void Main()

{

int summand1 = 2, summand2 = 3;

int sum = Add(summand1, summand2);

Console.WriteLine("{0} + {1} = {2}", summand1, summand2, sum);

// Delay.

Console.ReadKey();

}

}

}

Проект 5. Работа с логическими типами данных

using System;

// Методы (Функции).

namespace Methods

{

class Program

{

// Методы, которые возвращают логическое значение, называют методами-предикатами.

static bool And(bool a, bool b)

{

return a && b;

}

static void Main()

{

bool operand1 = true, operand2 = true;

bool result = And(operand1, operand2);

Console.WriteLine("{0} && {1} = {2}", operand1, operand2, result);

// Delay.

Console.ReadKey();

}

}

}

**Задачи:**

1. Разработать методы для формирования и вывода на экран двумерного массива
2. Создайте три метода для выполнения арифметических операций, с именами: Add– сложение, Sub– вычитание, Mul– умножение. Каждый метод должен принимать два целочисленных аргумента и выводить на экран результат выполнения арифметической операции соответствующей имени метода. Требуется предоставить пользователю возможность вводить с клавиатуры значения операндов для выполнения вычислений.

Решение задачи 1:

using System;

//Заполнение и обработка двумерного массива с помощью функций

namespace Primer

{

class Program

{

//Объявим константы и переменные в классе Program:

const byte n = 10; //количетсво строк и слобцов

static short[,] myMas = new short[n, n]; //двумерный массив

//Добавим функцию заполнения массива случайными значениями

//диапазон (минимальное и максимальное значение)

//передается в качестве параметров:

static void setArray(short \_min, short \_max)

{

Random rand = new Random();// класс генератора случ чисел

for (byte i = 0; i < n; i++)

{

for (byte j = 0; j < n; j++)

{

myMas[i, j] = (short)rand.Next(\_min, \_max);

}

}

return;

}

//Для вывода на экран элементов массива, добавим функцию,

//возвращающую в качестве результата строку:

static string printArray()

{

string outputString = "";

for (byte i = 0; i < n; i++)

{

for (byte j = 0; j < n; j++)

{

//добавляем в строку значение текущего элемента массива

//Format(String, Object) - Заменяет один или более элементов формата

//в строке строковым представлением указанного объекта.

outputString = outputString + String.Format("{0,4}", myMas[i, j]);

}

//добавляем перенос строки

outputString = outputString + '\n';

}

return outputString; //возвращаем результат функции

}

static void Main(string[] args)

{

short minValue, maxValue; //диапазон значений

// Введем с клавиатуры размеры массива

Console.WriteLine("Введите диапазон значений");

minValue = Convert.ToInt16(Console.ReadLine());

maxValue = Convert.ToInt16(Console.ReadLine());

//обращаемся к методу

setArray(minValue, maxValue);

Console.WriteLine(printArray()); //выводим на экран результат выполнения функции

}

}

}