**20** **Параллельное программирование**

Задание 1. Создайте и реализуйте метод решения задачи и выполните его в объектах класса Task используя три варианта создания объектов класса Task: Дано трехзначное число. Найти произведение его цифр.

Листинг программы:

namespace Space

{

public class Task

{

private int \_number;

public int Number

{

get { return \_number; }

set { \_number = value; }

}

public Task()

{

\_number = 0;

}

public Task(int number)

{

\_number = number;

}

public int MultiplyDigits()

{

int product = 1;

while (\_number > 0)

{

product \*= \_number % 10;

\_number /= 10;

}

return product;

}

public static Task Create(int number)

{

return new Task(number);

}

}

public class Program

{

public static void Main()

{

Task task1 = new Task(123);

int product1 = task1.MultiplyDigits();

Console.WriteLine(product1); // Output: 6

Task task2 = new Task();

task2.Number = 456;

int product2 = task2.MultiplyDigits();

Console.WriteLine(product2); // Output: 120

Task task3 = Task.Create(789);

int product3 = task3.MultiplyDigits();

Console.WriteLine(product3); // Output: 504

}

}

}

Таблица 20.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 123  456  789 | 6  120  504 |

Анализ результатов:

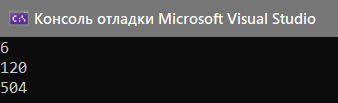


Рисунок 20.1 – Результат работы программы

Задание 2. Создайте два объекта класса Task. Первый объект возвращает результат вычисления, второй объект является задачей продолжения первого объекта и выводит результат первой задачи на консоль. Задания для реализации метода выполняемого в первом объекте класса Task: Дано трехзначное число. В нем зачеркнули первую слева цифру и приписали ее в конце. Найти полученное число.

Листинг программы:

namespace Space

{

class TaskExample

{

static void Main()

{

// Создаем первый объект задачи, который вычисляет результат

Task<int> task1 = Task<int>.Factory.StartNew(() =>

{

int number = 123;

int firstDigit = number / 100;

int restOfNumber = number % 100;

return restOfNumber \* 10 + firstDigit;

});

// Создаем второй объект задачи, который выводит результат на консоль

Task task2 = task1.ContinueWith(previousTask =>

{

Console.WriteLine("Результат: " + previousTask.Result);

});

// Ожидаем завершения второй задачи

task2.Wait();

}

}

}

Таблица 20.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 123 | 231 |

Анализ результатов:

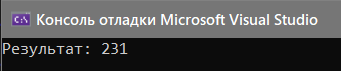


Рисунок 20.2 – Результат работы программы