

C# Introduction. Урок 5

Статические методы



Статические методы



```
using System;
class StatMethDemo{
    static void show(string txt){
        Console.WriteLine(txt);
    }
    static int factorial(int n){
        int s=1;
        for (int k=1; k<=n; k++){
            s*=k;
        }
        return s;
    }
    static double power(double x, int n){
        double s=1;
        for (int k=1; k<=n; k++){
            s*=x;
        }
        return s;
    }
}
```

...

Статические методы



```
...
static void Main(){
    show("Начинаем вычисления.");
    int m=5;
    double z=3, num;
    show(m+"!="+factorial(m))
    num=power(z, m);
    show(z+" в степени " +m+": "+num);
}
}
```

Перегрузка статических методов



```
using System;
class OverloadMethDemo{
    static void show(string txt){
        Console.WriteLine("Текст: "+txt);
    }
    static void show(int num){
        Console.WriteLine("Целое число: "+num);
    }
    static void show(double num){
        Console.WriteLine("Действительное число: "+num);
    }
    static void show(char s){
        Console.WriteLine("Символ: "+s);
    }
    static void show(int num, char s){
        Console.WriteLine("Аргументы {0} и {1}: ", num, s);
    }
    ...
}
```

Перегрузка статических методов



```
...
static void Main(){
    int num=5;
    double z=12.5;
    char symb = 'W';
    show(symb);
    show ("Знакомимся с перегрузкой методов")
    show(num);
    show(z);
    show(num, 'Q');
}
```

Массив как аргумент метода



```
using System;
class ArrayToMethDemo {
    static void fillRand (int[ ] nums ) {
        Random rnd=new Random ();
        for (int k=0; k<nums.Length; k++) {
            nums [k]=rnd.Next(1,101);
        }
    }
    static void showArray (int[] nums) {
        for (int k=0; k<nums.Length; k++) {
            Console.Write ("| {0}", nums [k] );
        }
        Console.WriteLine ("|");
    }
}
```

...

Массив как аргумент метода



...

```
static void showArray (int [,] nums) {  
    for(int i=0; i<nums.GetLength(0); i++) {  
        for(int j=0; j<nums.GetLength(1); j++) {  
            Console.Write (“{0,3}”, nums[i, j]);  
        }  
        Console.WriteLine();  
    }  
}
```

```
static int findMin (int[] nums) {  
    int s=nums[0];  
    for (int K=1; k<nums.Length; k++) {  
        if (nums[k]<s) s=nums [k];  
    }  
    return s;  
}
```

...

Массив как аргумент метода



...

```
static void Main () {  
    int [] A={1,3,5,7,9,11,13,15};  
    int [] B=new int [5];  
    int[,] C={ {1,2,3,4}, {5,6,7,8}, {9,10,11,12} };  
    fillRand (B);  
    Console.WriteLine ("Одномерный массив A:");  
    showArray (A);  
    Console.WriteLine ("Одномерный массив B:");  
    showArray (B);  
    int m=findMin (B);  
    Console.WriteLine ("Наименьшее значение: {0}", m);  
    Console.WriteLine ("Двумерный массив C:");  
    showArray (C)  
}  
}
```


Массив как результат метода



```
using System;
class ArrayFromMethDemo{
    static int[] fibs (int n) {
        int[] nums=new int{n};
        nums[0]=1;
        if(nums.Length==1)
            return nums;
        nums[1]=1;
        for(int k=2; k<nums.Length; k++) {
            nums[k]=nums[k-1]+nums[k-2];
        }
        return nums;
    }
    static char [] randS (int n) {
        Random rnd=new Random ();
        char [] symbs=new char[n];
        for (int k=0; k<symbs.Length; k++) {
            symbs [k]=(char) ('A'+rnd.Next(26) );
        }
        return symbs;
    }
}
```

Массив как результат метода



...

```
statics int[,] odds(int m, int n) {  
    int[,] nums=new int[m, n];  
    int val=1;  
    for (int i=0; i<nums.GetLength(0); i++) {  
        for (int j=0; j<nums.GetLength(1); j++) {  
            nums[i, j]=val;  
            val+=2;  
        }  
    }  
    return nums;  
}
```

...

Массив как результат метода



...

```
static void Main () {  
    int[] A;  
    char [] B;  
    int[,] C;  
    A=fibs (10);  
    Console.WriteLine ("Числа Фибоначчи:");  
    foreach (int s in A) {  
        Console.WriteLine ("| {0}", s);  
    }  
    Console.WriteLine ("|");  
    C=odds (4,6);  
    Console.WriteLine ("Двумерный массив:");  
    for (int i=0; i<C.GetLength(0); i++) {  
        for (int j=0; j<C.GetLength(1); j++) {  
            Console.Write (" {0,4}", C[i, j] );  
        }  
        Console.WriteLine();  
    }  
}
```

Механизмы передачи аргументов методу



```
using System
class ArgsDemo {
    static void alpha (int n) {
        Console.WriteLine ("В методе alpha(). На входе: "+n);
        n++;
        Console.WriteLine ("В методе alpha(). На входе: "+n);
    }
    static void bravo (int [] n) {
        Console.WriteLine ("В методе bravo (). На входе: "+ArrayToText (n) );
        for (int k=0; k<n.Length; k++) {
            n[k]++;
        }
        Console.WriteLine ("В методе bravo (). На входе: "+ArrayToText (n) );
    }
    ...
}
```

Механизмы передачи аргументов методу



...

```
static void charlie (int[] n) {  
    Console.WriteLine ("В методе charlie (). На входе: "+ArrayToText (n) );  
    int[] m=new int[n.Length];  
    for (int k=0; k<n.Length; k++) {  
        m[k]=n[k]+1;  
    }  
    n=m;  
    Console.WriteLine("В методе charlie (). На входе: "+ArrayToText (n) );  
}  
static string ArrayTotext (int[] n) {  
    string res="["+n[0];  
    for(int k=1; k<n.Length; k++) {  
        res+=", "+n[k];  
    }  
    res+="]";  
    return res;  
}
```

...

Механизмы передачи аргументов методу



...

```
static void Main () {  
    int A=100;  
    Console.WriteLine ("До вызова метода alpha (): A="+A);  
    Alpha (A);  
    Console.WriteLine ("После вызова метода alpha (): A="+A);  
    int [] B={1,3,5};  
    Console.WriteLine ("До вызова метода bravo (): B="+ArrayToText(B) );  
    bravo (B);  
    Console.WriteLine ("После вызова метода bravo (): B="+ArrayToText(B) );  
    int[] C={2,4,6};  
    Console.WriteLine ("До вызова метода charlie (): C="+ArrayToText(C) );  
    charlie (C) ;  
    Console.WriteLine ("После вызова метода charlie (): C="+ArrayToText(C) );  
}  
}
```

Рекурсия



```
using System;
class RecursionDemo {
    static int factorial (int n) {
        if(n==1) return 1;
        else return n*factorial (n-1);
    }
    static int fibs (int n) {
        if(n==1||n==2) return 1;
        else return fibs(n-1)+fibs(n-2);
    }
    static int sum (int n) {
        if (n==0) return 0;
        else return n+sum (n-1);
    }
    . . .
}
```

Рекурсия



...

```
static void show (int[] a, int k) {  
    Console.Write(a[k]+" ");  
    if (k==a.Length-1) {  
        Console.WriteLine ();  
    }  
    else {  
        show(a, k+1)  
    }  
}  
static void show (int [] a) {  
    show (a,0);  
}
```

...

Рекурсия



...

```
static void Main () {  
    Console.WriteLine("Факториал числа:");  
    for (int k=1; k,<=10; k++) {  
        Console.WriteLine(k+"!="+factrorial (k) );  
        Console.WriteLine("Числа Фибоначчи:");  
        for (int k=1; k<=10; k++) {  
            Console.WriteLine(fibs (k)+" ");  
        }  
        Console.WriteLine() ;  
        Console.Write("Сумма чисел от 1 до 100: ");  
        Console.WriteLine(sum(100) );  
        int[] A={1,3,5,7,9,11,13,15,17,19,21};  
        Console.WriteLine ("Числовой массив");  
        show(A);  
        Console.WriteLine("Элементы, начиная с третьего:");  
        show(A,2);  
    }  
}
```