C#中的static关键字和C++中的有很大程度的类似。但也有一些不同。

static可以用来修饰构造函数，而在C++中是不允许这样的。静态构造函数不能使用访问控制的修饰符，它是默认调用（即private）的。

例：程序static\_structure\_test1

// Copyright 2016.刘珅珅

// author：刘珅珅

// 静态构造函数

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace static\_structure\_test1

{

class StaticStructure

{

public static int alpha;

public int beta;

// 静态构造函数

static StaticStructure()

{

alpha = 99;

Console.WriteLine("Inside static constructor.");

}

public StaticStructure(int b)

{

beta = b;

Console.WriteLine("Inside instance constructor");

}

public void Show()

{

Console.WriteLine("alpha: " + alpha);

Console.WriteLine("beta: " + beta) ;

}

}

class StaticStructureTest

{

static void Main(string[] args)

{

StaticStructure obj = new StaticStructure(20);

obj.Show();

Console.WriteLine();

StaticStructure obj1 = new StaticStructure(30);

obj1.Show();

}

}

}

输出结果为：

Inside static constructor.

Inside instance constructor

alpha: 99

beta: 20

Inside instance constructor

alpha: 99

beta: 30

从结果中可以看出：

static构造函数在所有实例构造函数之前被执行了一次。

static类：在C#中Math,Abs,Array等类都是静态类，它的所有成员和方法都必须显示的声明为static，不能创建类对象。

例：程序static\_class\_test1

// Copyright 2016.刘珅珅

// author：刘珅珅

// 静态类

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace static\_class\_test1

{

static class NumericFn

{

static public double FracPart(double num)

{

return num - (int)num;

}

static public bool IsEven(double num)

{

return (num % 2) == 0 ? true : false;

}

}

class StaticClassTest

{

static void Main(string[] args)

{

if (NumericFn.IsEven(10))

{

Console.WriteLine("10 is even");

}

else

{

Console.WriteLine("10 is odd");

}

}

}

}