

C# Part 3 – Static Keyword & Function

M.Suryo Pranoto

suryodesign@yahoo.co.id

<http://suryostudio.net>

www.suryodesign.wordpress.com

Lisensi Dokumen:

Copyright © 2003-2007 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Pendahuluan

Static Keyword dan Function mungkin sudah tidak asing lagi di mata para programmer , Static Keyword dan Function ini sudah banyak kita temui tetapi mungkin hanya ada beberapa artikel yang dapat menjelaskan tentang penggunaan Static Dan Function ini secara mudah dan jelas agar dapat langsung mudah di praktekan.

Saya akan mencoba menjelaskannya step by step agar lebih mudah di mengerti , dan juga memberikan contoh agar di ketahui penggunaan dan penerapannya.

Isi

Di Pertemuan ke 3 ini , kita akan belajar mengenai penggunaan Static Keyword dan bagaimana menggunakan function pada C#.

Pertama kita mesti mengetahui apa itu Static ?

Static disini maksudnya bisa berupa variable / function yang kita deklarasikan dengan menggunakan keyword Static di depan variable / function tersebut seperti berikut ini

```
public static double nStaticNilai;  
public static void mahasiswaClassFunction() // static  
{  
    Console.WriteLine("mahasiswa class function");  
}
```

Static hanya dapat diubah melalui dimana tempat itu di deklarasikan , contohnya apa bila variable tersebut kita deklarasikan dalam class

```
class Mahasiswa
{
    public static double nStaticNilai;
    public static void mahasiswaClassFunction() // static
    {
        Console.WriteLine("mahasiswa class function");
    }
}
```

variable static seperti berikut dapat langsung kita panggil dengan perintah berikut :

```
Mahasiswa.nStaticNilai = 3;
```

Hasilnya tentu saja akan berbeda apabila kita membuat suatu member/anggota baru dari class tersebut seperti berikut :

```
Mahasiswa mhs = new Mahasiswa();
mhs.nStaticNilai <<<<< baris perintah berikut ini tidak akan dapat di
jalankan karena variabel ini di akses via member / object yang di buat dan
bukan melalui class itu sendiri
```

saya harap contoh di atas dapat menjelaskan mengenai Static Keyword dan sekarang kita akan mencoba membahas function.

Function itu sendiri adalah Suatu kumpulan perintah yg telah kita siapkan dan dapat kita panggil sewaktu-waktu ingin kita gunakan , function ini bisa menerima nilai yang kita lempar, dalam passing parameternya biasanya menggunakan kata “ref” di depan variable yang akan dilempar , dan function itu bisa mengembalikan suatu nilai yang telah kita hitung di dalamnya., berikut contoh dari penggunaan function :

```
//membuat object mahasiswa baru
Mahasiswa mhs2 = new Mahasiswa();
//memanggil function yang telah kita gunakan
mhs2.hitungNilai(ref mhs2.nilaiUjian1, ref mhs2.nilaiUjian2 , ref no);

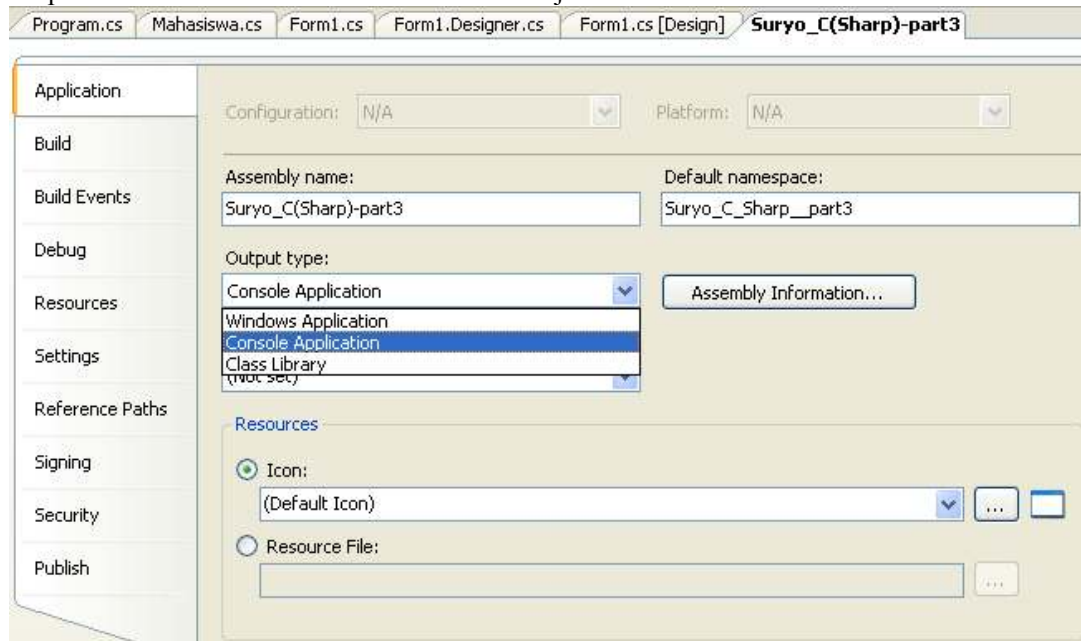
//function hitungNilai yang telah kita definisikan
public void hitungNilai(ref int a, ref int b , ref int no)
{
    nNilai++;
    Console.WriteLine("%n");
    nStaticNilai = Convert.ToDouble(a) / Convert.ToDouble(b);
    Console.WriteLine("Nilai rata2 mahasiswa ke " + no + " : "
+ nStaticNilai);
}
```

Setelah kita mengerti apa itu Static Keyword dan function dan telah mengetahui bagaimana cara

penggunaannya dalam pemrograman. maka alangkah baiknya kita langsung mencoba membuatnya dalam **aplikasi console** :

saya akan mencoba membuat nya dalam step by step agar lebih mudah untuk di ikuti :

1. Buka Visual Studio 2005 atau ketik “devenv” pada run menu.
2. Beri nama Solution dan pilih directori Save di tempat yang anda inginkan , dan Ganti Startup Properties menjadi Console



3.Add Suatu Class dan beri nama class “mahasiswa”

4.Kemudian buat variable dan function pada Class hingga menjadi seperti berikut ini :

```
using System;
using System.Collections.Generic;
using System.Text;
namespace Suryo_C_Sharp__part3
{
    class Mahasiswa
    {
        public string nama = "";
        public double nNilai ; // non-static
        public static double nStaticNilai;// static
        public static int Nomor = 0 ;
        public int nilaiUjian1 = 0 ;
        public int nilaiUjian2 = 0 ;
        public void mahasiswaMemberFunction() // non-static
        {
            Console.WriteLine("mahasiswa member function");
        }

        public static void mahasiswaClassFunction() // static
        {
            Console.WriteLine("mahasiswa class function");
        }

        public void hitungNilai(ref int a, ref int b , ref int no)
            // non-static
        {
            nNilai++;
            Console.WriteLine("¥n");
            nStaticNilai = Convert.ToDouble(a) / Convert.ToDouble(b);
            Console.WriteLine("Nilai rata2 mahasiswa ke " + no + " :
"
                + nStaticNilai);
        }
    }
}
```

5.Pada Form1.cs buat variable deklarasi dan fungsi seperti berikut ini

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace Suryo_C_Sharp__part3
{
    public partial class Form1 : Form
    {
        public Form1 ()
        {

```

```
InitializeComponent();
}

private void Form1_Load(object sender, EventArgs e)
{
    Mahasiswa mhs = new Mahasiswa(); // membuat object mhs
    Mahasiswa mhs2 = new Mahasiswa(); // membuat object mhs2
    Mahasiswa mhs3 = new Mahasiswa(); // membuat object mhs3

    mhs.mahasiswaMemberFunction();
    //memanggil function dari mhs, function ini dapat
    kita panggil
    //karena function ini tidak memiliki static
    //karena itu mhs sebagai member(object yg dibuat
    dari Mahasiswa)
    //dapat memanggil function tersebut

    //cobalah untuk memanggil mhs.mahasiswaClassFunction
    //lines ini tidak akan dapat dipanggil / dicompile
    karena memiliki
    //keyword "Static" di depan functionnya dan kita
    hanya dapat
    //memanggilnya melewati class itu sendiri dan tidak
    dapat mengakses
    //melewati object / member yang telah dibuat
    //jadi untuk memanggilnya kita langsung melewati classnya
    //seperti berikut ini :

    Mahasiswa.mahasiswaClassFunction();
    //via class dapat mengakses fungsi mahasiswa static

    //mhs.mahasiswaClassFunction
    //via object tidak dapat mengakses

    mhs.nNilai = 5 ; // via member
    Mahasiswa.nStaticNilai = 100; // via class

    Console.WriteLine("Masukkan nama mahasiswa pertama : ");
    string mhs2Nama = Console.ReadLine();

    string mhs2Nilai1 ;
    do
    {
        Console.WriteLine("Masukkan nilai pertama (max 100) : ");
        mhs2Nilai1 = Console.ReadLine();
    } while (Convert.ToInt16(mhs2Nilai1) > 100);

    string mhs2Nilai2;
    do
    {
        Console.WriteLine("Masukkan nilai kedua (max 100) : ");
        mhs2Nilai2 = Console.ReadLine();
    } while (Convert.ToInt16(mhs2Nilai2) > 100);
}
```

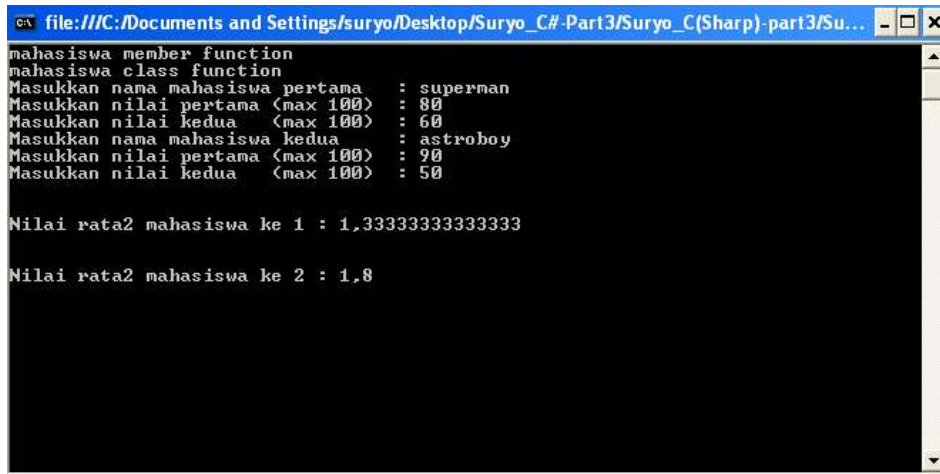
```
mhs2.nama = mhs2Nama;
mhs2.nilaiUjian1 = Convert.ToInt16(mhs2Nilai1) ;
mhs2.nilaiUjian2 = Convert.ToInt16(mhs2Nilai2) ;

Console.Write("Masukkan nama mahasiswa kedua : ");
string mhs3Nama = Console.ReadLine();
string mhs3Nilai1;
do
{
    Console.Write("Masukkan nilai pertama (max 100) : ");
    mhs3Nilai1 = Console.ReadLine();
} while (Convert.ToInt16(mhs3Nilai1) > 100);

string mhs3Nilai2;
do
{
    Console.Write("Masukkan nilai kedua (max 100) : ");
    mhs3Nilai2 = Console.ReadLine();
} while (Convert.ToInt16(mhs3Nilai2) > 100);

mhs3.nama = mhs3Nama;
mhs3.nilaiUjian1 = Convert.ToInt16(mhs3Nilai1);
mhs3.nilaiUjian2 = Convert.ToInt16(mhs3Nilai2);
int no = 1;
mhs2.hitungNilai(ref mhs2.nilaiUjian1, ref
mhs2.nilaiUjian2, ref no);
no++;
mhs3.hitungNilai(ref mhs3.nilaiUjian1, ref
mhs3.nilaiUjian2, ref no);
    }
}
```

Berikut adalah hasil Output dari aplikasi tersebut :



```
file:///C:/Documents and Settings/suryo/Desktop/Suryo_C#-Part3/Suryo_C(Sharp)-part3/Su...
mahasiswa member function
mahasiswa class function
Masukkan nama mahasiswa pertama : superman
Masukkan nilai pertama <max 100> : 80
Masukkan nilai kedua <max 100> : 60
Masukkan nama mahasiswa kedua : astroboy
Masukkan nilai pertama <max 100> : 90
Masukkan nilai kedua <max 100> : 50

Nilai rata2 mahasiswa ke 1 : 1,33333333333333
Nilai rata2 mahasiswa ke 2 : 1,8
```

Penutup

Mempelajari C# mungkin agak sulit dibandingkan bahasa pemrograman VB yang lebih bersahabat dan lebih mudah di mengerti , tetapi ini merupakan suatu tantangan bagi kita sebagai Praktisi IT untuk mendalaminya , pada pertemuan pertama ini kita mempelajari basic dari C# agar dapat kita terapkan pada pertemuan selanjutnya.

Referensi

- MSDN 2005
- <http://msdn.microsoft.com>
- www.gotdotnet.com

Biografi Penulis



M.Suryo Pranoto – Mahasiswa Perguruan Tinggi Universitas Bina Nusantara , Aktif dalam beberapa komunitas komputer , dan beberapa project terutama berbasis aplikasi seperti VB.Net atau C#, dan sedang berusaha keras untuk menabung dan berencana untuk melanjutkan Cisco CCNP setelah menyelesaikan CCNA , memiliki hobby untuk sharing mengenai komputer mulai dari software hingga jual beli hardware maupun modding komputer.
Ym : suryolovetyka , Email : suryodesign@yahoo.co.id ,
Website : www.suryostudio.net , Blog : www.suryodesign.wordpress.com

