

عنوان: مدیریت Instance در WCF

نویسنده: مسعود پاکدل

تاریخ: ۱۷:۴۰ ۱۳۹۲/۰۲/۲۷

آدرس: [www.dotnettips.info](http://www.dotnettips.info)

گروه‌ها: WCF, InstanceContextMode

## نحوه پیاده سازی و مدیریت Instance در پروژه‌های مبتنی بر WCF

نکته : آشنایی اولیه با مفاهیم WCF جهت درک صحیح مطالب الزامی است.

تشریح مسئله : در صورتی که نیاز باشد که نمونه ساخته شده از سرویس (سمت سرور) به صورت Singleton باشد بهترین روش برای پیاده سازی به چه صورت است.

برای شروع ابتدا مثال زیر را پیاده سازی می‌کنیم.

یک Contract به صورت زیر تعریف می‌کنیم:

```
[ServiceContract(SessionMode=SessionMode.Allowed)]
public interface IMyService
{
    [OperationContract]
    int GetData();
}
```

حالا یک سرویس برای پیاده سازی Interface بالا می‌نویسیم.

```
[ServiceBehavior( InstanceContextMode = InstanceContextMode.PerCall )]
public class PerCallService : IMyService
{
    int count;
    public int GetData()
    {
        return ++count;
    }
}
```

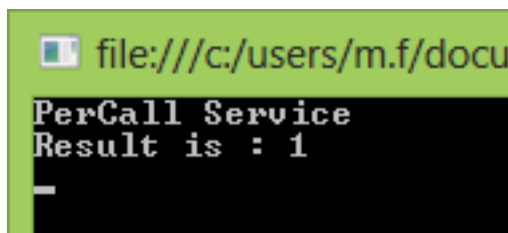
همانطور که از نام سرویس مشخص است از این سرویس به ازای هر فراخوانی یک نمونه سمت سرور ساخته می‌شود.

حالا برای مشاهده نتیجه یک پروژه ConsoleApplication ایجاد کنید و سرویس مورد نظر را از روش AddServiceReference به پروژه اضافه کرده در فایل Program کدهای زیر را کپی کنید.

```
static void Main( string[] args )
{
    Console.WriteLine( "PerCall Service" );

    MyPerCallService.MyServiceClient client = new MyPerCallService.MyServiceClient();
    int count = 0;
    for ( int i = 0 ; i < 5 ; i++ )
    {
        count = client.GetData();
    }
    Console.WriteLine( count );
    Console.ReadLine();
}
```

بعد از اجرا خروجی به صورت زیر است:



بعد از 5 بار فراخوانی متد GetData باز خروجی دارای مقدار 1 است. یعنی به ازای هر بار فراخوانی متد GetData یک نمونه از سرویس مورد نظر ساخته می‌شود. این عمل توسط خصوصیت InstanceContextMode که از نوع PerCall است به سرویس اعمال می‌شود.

حالا یک سرویس دیگر به صورت زیر ایجاد کنید.

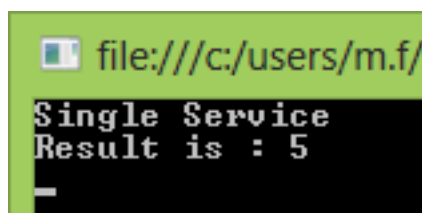
```
[ServiceBehavior( InstanceContextMode = InstanceContextMode.Single )]
public class SingleService : IMyService
{
    int count;
    public int GetData()
    {
        return ++count;
    }
}
```

تنها تفاوت این سرویس با سرویس قبلی در این است که InstanceContextMode این سرویس به صورت Single معرفی شده است. یعنی به ازای n فراخوانی فقط یک نمونه از کلاس ساخته می‌شود. این سرویس رو هم مثل روش قبلی به Client Application اضافه کنید. کد کلاس Program رو به صورت زیر تغییر دهید.

```
static void Main( string[] args )
{
    Console.WriteLine( "Single Service" );

    MySingleService.MyServiceClient client = new MySingleService.MyServiceClient();
    int count = 0;
    for ( int i = 0 ; i < 5 ; i++ )
    {
        count = client.GetData();
    }
    Console.WriteLine("Result is : {0}", count );
    Console.ReadLine();
}
```

که بعد از اجرا خروجی به صورت زیر است.



به ازای 5 بار فراخوانی سرویس متغیر Count سمت سرور مقدار قبلی خود را حفظ کرده است.

## نظرات خوانندگان

نویسنده: ahmadb7

تاریخ: ۸:۴۲ ۱۳۹۲/۰۲/۳۱

با سلام؛ امکان داره منبع خوبی برای یادگیری WCF معرفی کنید

نویسنده: مسعود م. پاکدل

تاریخ: ۹:۱۱ ۱۳۹۲/۰۲/۳۱

می تونید از کتاب 348 صفحه ای WCF 4.0 Multi-tier Services Development with LINQ to Entities نوشته Mike Liu استفاده کنید.

خیلی روان و سلیس برای سطوح مبتدی و متوسط نوشته شده.

کتاب Pro WCF 4 Practical Microsoft SOA Implementation هم گزینه‌ی خیلی مناسبه. البته MSDN رو هم فراموش نکنید.

نویسنده: احسان شیروان

تاریخ: ۱۰:۵۵ ۱۳۹۳/۰۶/۱۹

سلام

یه سوالی برام پیش اومده ممنون میشم راهنمایی فرمایید:

من یک سرویس WCF ایجاد کردم و اونو به شکل زیر تنظیم کردم :

```
[ServiceContract(SessionMode=SessionMode.Required)]
```

و همچنین برای کلاس پیاده سازی کننده اینترفیس :

```
[ServiceBehavior( InstanceContextMode = InstanceContextMode.PerSession)]
```

من داخل این کلاس یه متغیر از یک کلاس به صورت سراسری تعریف کردم که میخوام ازش توی متدهای متفاوت استفاده کنم اما ظاهرا با هر بار فراخوانی باز هم این متغیر داده‌های خودشو از دست میده البته static نیست و به دلیل ساختار اون نمیتونم استاتیکش کنم

ممنون میشم راهنمایی نمایید