

Multiple Active Result Sets (MARS) یکی از قابلیت های SQL SERVER است. این قابلیت در واقع این امکان را برای ما فراهم می کند تا بر روی یک Connection همزمان چندین کوئری را به صورت موازی ارسال کنیم. در این حالت برای هر کوئری یک سشن مجزا در نظر گرفته می شود.

مدل:

```
namespace EnablingMARS.Models
{
    public class Product
    {
        public int Id { get; set; }
        public string Title { get; set; }
        public string Desc { get; set; }
        public float Price { get; set; }
        public Category Category { get; set; }
    }

    public enum Category
    {
        Cate1,
        Cate2,
        Cate3
    }
}
```

کلاس Context:

```
namespace EnablingMARS.Models
{
    public class ProductDbContext : DbContext
    {
        public ProductDbContext() : base("EnablingMARS") {}
        public DbSet<Product> Products { get; set; }
    }
}
```

ابتدا یک سطر جدید را توسط کد زیر به دیتابیس اضافه می کنیم:

```
MyContext.Products.Add(new Product()
{
    Title = "title1",
    Desc = "desc",
    Price = 4500f,
    Category = Category.Cate1
});
MyContext.SaveChanges();
```

اکنون می خواهیم قیمت محصولات را که در دسته بندی Cate1 قرار دارند، تغییر دهیم:

```
foreach (var product in _dvContext.Products.Where(category => category.Category == Category.Cate1))
{
    product.Price = 50000;
    MyContext.SaveChanges();
}
```

خوب؛ اکنون اگر برنامه را اجرا کنیم با خطای زیر مواجه می شویم:

There is already an open DataReader associated with this Command which must be closed first.

این استثناء زمانی اتفاق می افتد که بر روی نتایج حاصل از یک کوئری، یک کوئری دیگر را ارسال کنیم. البته استثنای صادر شده بستگی به کوئری دوم شما دارد ولی در حالت کلی و با مشاهده Stack Trace، پیام فوق نمایش داده می شود. همانطور که در کد بالا ملاحظه می کنید درون حلقه‌ی foreach ما به پراپرتی Price دسترسی پیدا کرده ایم، در حالیکه کوئری اصلی ما هنوز فعال (Active) است. MARS در اینجا به ما کمک می کند که بر روی یک Connection، بیشتر از یک کوئری فعال داشته باشیم. در حالت عادی Entity Framework Code First این ویژگی را به صورت پیش فرض برای ما فعال نمی کند. اما اگر خودمان کانکشن استرینگ را اصلاح کنیم، این ویژگی SQL SERVER فعال می گردد. برای حل این مشکل کافی است به کانکشن استرینگ، MultipleActiveResultSets=true را اضافه کنیم:

```
"Data Source=(LocalDB)\v11.0;Initial Catalog=EnablingMARS; MultipleActiveResultSets=true"
```

لازم به ذکر است که این قابلیت از نسخه 2005 SQL SERVER به بالا در دسترس می باشد. همچنین در هنگام استفاده از این قابلیت می بایستی موارد زیر را در نظر داشته باشید:

وقتی کانکشنی در حالت MARS برقرار می شود، یک سشن نیز همراه با یکسری اطلاعات اضافی برای آن ایجاد شده که باعث ایجاد Overhead خواهد شد.

دستورات مارس [thread-safe](#) نیستند.

نظرات خوانندگان

نویسنده: محسن خان
تاریخ: ۱۰:۳۲ ۱۳۹۳/۰۷/۲۴

با تشکر. یک سؤال: اگر بجای dvContext.Products.Where بنویسیم dvContext.Products.Include (x=>x.Category).Where باز هم این مشکل هست؟

نویسنده: سیروان عقیقی
تاریخ: ۱۰:۴۱ ۱۳۹۳/۰۷/۲۴

بله می شود به صورت eager loading این مشکل را نیز حل کرد. ولی مسئله performance را نیز باید در نظر داشته باشید. البته می بایست به این صورت نوشته شود:

```
foreach (var product in _dvContext.Products.Where(x => x.Category.CateName == "Cate1").Include(x =>
x.Category).ToList())
{
    product.Price = 50000;
    _dvContext.SaveChanges();
}
```

حالت فوق در صورتی است که یک Navigation property با نام Category داشته باشید.