

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

آموزش درج، بروز رسانی و حذف ذخیره شده در Code First

مدرس: مهندس افشین رفوآ

آموزش درج، بروز رسانی و حذف ذخیره شده در Code First

Entity Framework 6 Code-First توانایی ایجاد و استفاده از **Stored Procedure** را برای عملیات **add**، **delete**، **update** را فراهم می کند. این در نسخه های قبلی **Entity Framework** امکان پذیر نبود.

Student Entity:

```
class Student
{
    public Student()
    {
    }
    public int Student_ID { get; set; }
    public string StudentName { get; set; }
}
```

مثال زیر به طور خودکار یک **Stored Procedure** برای موجودیت **Student** با استفاده از **Fluent API** ایجاد می کند.

```
protected override void OnModelCreating(DbModelBuilder
modelBuilder)
{
    modelBuilder.Entity()
        .MapToStoredProcedures();
}
```

کد نشان داده در بالا سه **procedure** به ترتیب زیر می سازد.

Student_Insert

Student_Update

Student_Delete

که **procedure** های درج و بروز رسانی یک پارمتر که متناظر با اسم های مشخصه مورد نظر است، را دارد. و **procedure** حذف هم یک مشخصه کلید اولیه به نام پارامتر **StudentID** دارد.

شما همینطور می توانید **stored procedure** و اسم های پارامترهای را به شکل زیر تغییر دهید.

```
protected override void OnModelCreating(DbModelBuilder
modelBuilder)
```

```

        {
            modelBuilder.Entity()
                .MapToStoredProcedures(p => p.Insert(sp =>
                    sp.HasName("sp_InsertStudent").Parameter(pm => pm.StudentName,
                        "name").Result(rs => rs.Student_ID, "Student_ID"))
                    .Update(sp =>
                    sp.HasName("sp_UpdateStudent").Parameter(pm => pm.StudentName,
                        "name")))
                .Delete(sp =>
                    sp.HasName("sp_DeleteStudent").Parameter(pm => pm.Student_ID,
                        "Id"));
        }
    }

```

اگر می خواهید که همه موجودیت هایتان از **procedures** استفاده کند به ترتیب زیر عمل کنید.

```

protected override void OnModelCreating(DbModelBuilder
modelBuilder)
{
    modelBuilder.Types().Configure(t =>
t.MapToStoredProcedures());
}

```

محدودیت ها

تنها **Fluent API** ها می توانند با **stored procedure** برای **Map** کردن استفاده شوند. شما نمی توانید از

خصوصیات **Data Annotation** در **EF 6** برای **map** کردن **stored procedure** ها استفاده کنید.

نمی توانید ترکیبی از **stored procedure** و **query** را برای عملیات **add**، **update**، **delete** روی یک

موجودیت یکسان اعمال کنید.

شما می توانید از **stored procedure** یا **SQL query** برای عملیات **add**، **update**، **delete** روی موجودیت

ها استفاده کنید.