

# Improve Entity Framework Performance



 Bulk Insert

 Bulk Delete

 Bulk Update

 Bulk Merge

[LEARN MORE](#)

[< Previous](#)[Next >](#)

## Lazy Loading in Entity Framework

Lazy loading is delaying the loading of related data, until you specifically request for it. It is the opposite of [eager loading](#). For example, the `Student` entity contains the `StudentAddress` entity. In the lazy loading, the context first loads the `Student` entity data from the database, then it will load the `StudentAddress` entity when we access the `StudentAddress` property as shown below.

```
using (var ctx = new SchoolDBEntities())
{
    //Loading students only
    IList<Student> studList = ctx.Students.ToList<Student>();

    Student std = studList[0];

    //Loads Student address for particular Student only (seperate SQL query)
    StudentAddress add = std.StudentAddress;
}
```

The code shown above will result in two SQL queries. First, it will fetch all students:

```
SELECT
[Extent1].[StudentID] AS [StudentID],
[Extent1].[StudentName] AS [StudentName],
[Extent1].[StandardId] AS [StandardId]
FROM [dbo].[Student] AS [Extent1]
```

Then, it will send the following query when we get the reference of StudentAddress:

```
exec sp_executesql N'SELECT
[Extent1].[StudentID] AS [StudentID],
[Extent1].[Address1] AS [Address1],
[Extent1].[Address2] AS [Address2],
[Extent1].[City] AS [City],
[Extent1].[State] AS [State]
FROM [dbo].[StudentAddress] AS [Extent1]
WHERE [Extent1].[StudentID] = @EntityKeyValue1',N'@EntityKeyValue1 int',@EntityKeyValue1=1
```

## Disable Lazy Loading

We can disable lazy loading for a particular entity or a context. To turn off lazy loading for a particular property, do not make it virtual. To turn off lazy loading for all entities in the context, set its configuration property to false.

```
using System;
using System.Data.Entity;
using System.Data.Entity.Infrastructure;
using System.Data.Entity.Core.Objects;
using System.Linq;

public partial class SchoolDBEntities : DbContext
{
    public SchoolDBEntities(): base("name=SchoolDBEntities")
    {
        this.Configuration.LazyLoadingEnabled = false;
    }

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

### Rules for lazy loading:

1. *context.Configuration.ProxyCreationEnabled* should be true.
2. *context.Configuration.LazyLoadingEnabled* should be true.
3. Navigation property should be defined as public, virtual. Context will **NOT** do lazy loading if the property is not defined as virtual.

---

[!\[\]\(a03a7eb2f4046e1d3c76772003e549ea\_img.jpg\) Download EF 6 DB-First Demo Project from Github](#)

---

[< Previous](#)[Next >](#)

## ENTITYFRAMEWORKTUTORIAL

Learn Entity Framework using simple yet practical examples on EntityFrameworkTutorial.net for free. Learn Entity Framework DB-First, Code-First and EF Core step by step. While using this site, you agree to have read and accepted our terms of use and [privacy policy](#).

✉ [feedback@entityframeworktutorial.net](mailto:feedback@entityframeworktutorial.net)

## TUTORIALS

> [EF Basics](#)

> [EF Core](#)

> [EF 6 DB-First](#)

> [EF 6 Code-First](#)

## E-MAIL LIST

Subscribe to EntityFrameworkTutorial email list and get EF 6 and EF Core Cheat Sheets, latest updates, tips & tricks about Entity Framework to your inbox.

We respect your privacy.

[HOME](#) [PRIVACY POLICY](#) [ADVERTISE WITH US](#)

© 2020 EntityFrameworkTutorial.net. All Rights Reserved.