

# Improve Entity Framework Performance



 Bulk Insert

 Bulk Delete

 Bulk Update

 Bulk Merge

[LEARN MORE](#)

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

## Types of Entities in Entity Framework

There are two types of Entities in Entity Framework: POCO Entities and Dynamic Proxy Entities.

### POCO Entities (Plain Old CLR Object)

A POCO entity is a class that doesn't depend on any framework-specific base class. It is like any other normal .NET CLR class, which is why it is called "Plain Old CLR Objects".

POCO entities are supported in both EF 6 and EF Core.

These POCO entities (also known as persistence-ignorant objects) support most of the same query, insert, update, and delete behaviors as entity types that are generated by the Entity Data Model. The following is an example of Student POCO entity.

```
public class Student
{
    public int StudentID { get; set; }
    public string StudentName { get; set; }
    public DateTime? DateOfBirth { get; set; }
    public byte[] Photo { get; set; }
    public decimal Height { get; set; }
    public float Weight { get; set; }

    public StudentAddress StudentAddress { get; set; }
    public Grade Grade { get; set; }
}
```

## Dynamic Proxy Entities (POCO Proxy)

Dynamic Proxy is a runtime proxy class which wraps POCO entity. Dynamic proxy entities allow **lazy loading**.

**Note:** Dynamic proxy entities are only supported in EF 6. EF Core 2.0 does not support them yet.

A POCO entity should meet the following requirements to become a POCO proxy:

1. A POCO class must be declared with public access.
2. A POCO class must not be sealed (NotInheritable in Visual Basic).
3. A POCO class must not be abstract (MustInherit in Visual Basic).
4. Each navigation property must be declared as public, virtual.
5. Each collection property must be ICollection<T>.
6. The `ProxyCreationEnabled` option must **NOT** be false (default is true) in context class.

The following POCO entity meets all of the above requirements to become a dynamic proxy entity at runtime.

```
public class Student
{
    public int StudentID { get; set; }
    public string StudentName { get; set; }
    public DateTime? DateOfBirth { get; set; }
    public byte[] Photo { get; set; }
    public decimal Height { get; set; }
    public float Weight { get; set; }

    public virtual StudentAddress StudentAddress { get; set; }
    public virtual Grade Grade { get; set; }
}
```

**Note:** By default, dynamic proxy is enabled for every entity. However, you can disable dynamic proxy by setting `context.Configuration.ProxyCreationEnabled = false;` in the context class.

At runtime, EF API will create an instance of dynamic proxy for the above `Student` entity. The type of dynamic proxy for `Student` will be `System.Data.Entity.DynamicProxies.Student`, as shown below:

```
using (SchoolDBEntities context = new SchoolDBEntities())
{
    Student studentEntity = context.Students.FirstOrDefault<Student>();
    studentEntity (System.Data.Entity.DynamicProxies.Student_8C9C606A94AFB4EB28256E8214C30B620217205B13ED3916324B4A)
}
```

Use `ObjectContext.GetObjectType()` to find the underlying wrapped type by the dynamic proxy as shown below:

```
using (SchoolDBEntities context = new SchoolDBEntities())
{
    Student studentEntity = context.Students.FirstOrDefault<Student>();
    var entityType = ObjectContext.GetObjectType(studentEntity.GetType());
    entityType {Name = "Student" FullName = "EFBasicTutorials.Student"}
}
```

[< 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.