






Improve Entity Framework Performance



 **Bulk Insert**

 **Bulk Delete**

 **Bulk Update**

 **Bulk Merge**

LEARN MORE

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

One-to-One Relationship Conventions in Entity Framework Core

Entity Framework Core introduced default conventions which automatically configure a One-to-One relationship between two entities (EF 6.x or prior does not support conventions for One-to-One relationship).

In EF Core, a one-to-one relationship requires a reference navigation property at both sides. The following `Student` and `StudentAddress` entities follow the convention for the one-to-one relationship.

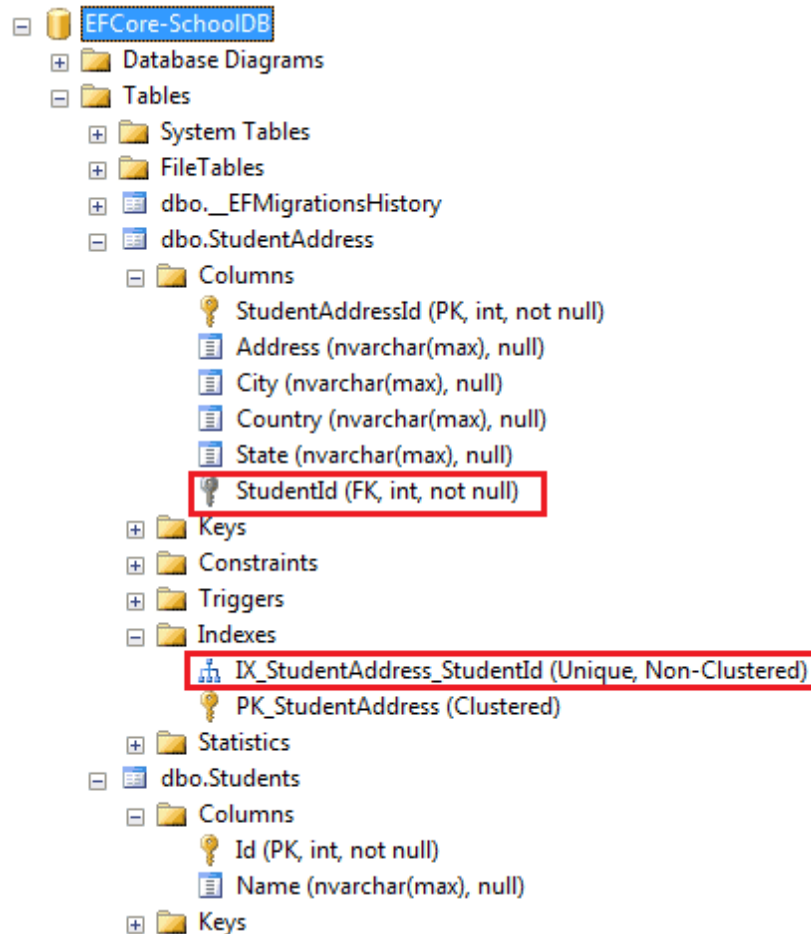
```
public class Student
{
    public int Id { get; set; }
    public string Name { get; set; }

    public StudentAddress Address { get; set; }
}

public class StudentAddress
{
    public int StudentAddressId { get; set; }
    public string Address { get; set; }
    public string City { get; set; }
    public string State { get; set; }
    public string Country { get; set; }

    public int StudentId { get; set; }
    public Student Student { get; set; }
}
```

In the example above, the `Student` entity includes a reference navigation property of type `StudentAddress` and the `StudentAddress` entity includes a foreign key property `StudentId` and its corresponding reference property `Student`. This will result in a one-to-one relationship in corresponding tables `Students` and `StudentAddresses` in the database, as shown below.



EF Core creates a unique index on the NotNull foreign key column `StudentId` in the `StudentAddresses` table, as shown above. This ensures that the value of the foreign key column `StudentId` must be unique in the `StudentAddress` table, which is necessary of a one-to-one relationship.

Note: Unique constraint is supported in Entity Framework Core but not in EF 6 and that's why EF Core includes conventions for one-to-one relationship but not EF 6.x.

Use Fluent API to configure one-to-one relationships if entities do not follow the conventions.



Further Reading

- › [Configure one-to-one relationship using Fluent API](#)
 - › [Configure one-to-many relationship using Fluent API](#)
 - › [Configure many-to-many relationship using Fluent API](#)
-

[< 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

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.

Email address

GO

We respect your privacy.

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

© 2020 EntityFrameworkTutorial.net. All Rights Reserved.