< Previous   Next >

Deleting a Disconnected Entity in EF 6

Deleting a disconnected entity is easy. Just set its state to Delete using the `Entry()` method, as shown below.

```csharp
// disconnected entity to be deleted
var student = new Student(){ StudentId = 1 };

using (var context = new SchoolDBEntities())
{
    context.Entry(student).State = System.Data.Entity.EntityState.Deleted;

    context.SaveChanges();
}
```

In the above example, an instance of the `Student` entity contains only the `StudentId` key property. To delete an entity, it only requires a key property. `context.Entry(student).State = System.Data.Entity.EntityState.Deleted` attaches an entity to a context and sets its state to Deleted. This will execute the following DELETE command in the database.

```
delete [dbo].[Student]
where ([StudentId] = @0)',N'@0 int',@0=1
```

[⬇ Download EF 6 DB-First Demo Project from Github](#)

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