

# Explicit Loading of Related Data

Article • 07/26/2022

## Explicit loading

You can explicitly load a navigation property via the `DbContext.Entry(...)` API.

C#

```
using (var context = new BloggingContext())
{
    var blog = context.Blogs
        .Single(b => b.BlogId == 1);

    context.Entry(blog)
        .Collection(b => b.Posts)
        .Load();

    context.Entry(blog)
        .Reference(b => b.Owner)
        .Load();
}
```

You can also explicitly load a navigation property by executing a separate query that returns the related entities. If change tracking is enabled, then when a query materializes an entity, EF Core will automatically set the navigation properties of the newly-loaded entity to refer to any entities already loaded, and set the navigation properties of the already-loaded entities to refer to the newly loaded entity.

## Querying related entities

You can also get a LINQ query that represents the contents of a navigation property.

This allows you to apply other operators over the query. For example, applying an aggregate operator over the related entities without loading them into memory.

C#

```
using (var context = new BloggingContext())
{
    var blog = context.Blogs
```

```
.Single(b => b.BlogId == 1);

var postCount = context.Entry(blog)
    .Collection(b => b.Posts)
    .Query()
    .Count();
}
```

You can also filter which related entities are loaded into memory.

C#

```
using (var context = new BloggingContext())
{
    var blog = context.Blogs
        .Single(b => b.BlogId == 1);

    var goodPosts = context.Entry(blog)
        .Collection(b => b.Posts)
        .Query()
        .Where(p => p.Rating > 3)
        .ToList();
}
```

### Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see [our contributor guide](#).

.NET

### .NET feedback

.NET is an open source project. Select a link to provide feedback:

 [Open a documentation issue](#)

 [Provide product feedback](#)