语法

[SuppressMessageAttribute("Microsoft.Naming", "CA1710:IdentifiersShouldHaveCorrectSuffix", Justification = "Name is intentional")]

public class DbSet<TEntity> : DbQuery<TEntity>,

IDbSet<TEntity>, IQueryable<TEntity>, IEnumerable<TEntity>,

IQueryable, IEnumerable where TEntity : class

#### 类型参数

*TEntity*

定义集的类型。

DbSet<TEntity> 类型公开以下成员。

构造函数

|  |  |  |
| --- | --- | --- |
|  | **名称** | **描述** |
| 受保护的方法 | [DbSet<TEntity>](https://msdn.microsoft.com/zh-cn/library/dn246932) | 从派生类型的构造函数中进行调用时，创建一个 DbSet<TEntity> 实例，它将用作 DbSet 的测试替身。 将由测试替身使用的方法和属性必须由测试替身来实现，除了 AsNoTracking、AsStreaming 和 Include，它们的默认实现是空操作。 |

[顶部](https://msdn.microsoft.com/zh-cn/library/gg696460" \l "mainBody)

属性

显示: 继承 保护

|  |  |  |
| --- | --- | --- |
|  | **名称** | **描述** |
| 公共属性 | [Local](https://msdn.microsoft.com/zh-cn/library/gg696248) | 获取 [ObservableCollection<T>](https://msdn.microsoft.com/zh-cn/library/ms668604)，它表示此集中的所有“已添加”、“未更改”和“已修改”实体的本地视图。 在上下文中添加或删除实体时，该本地视图将保持同步。 同样，在本地视图中添加或删除实体也会自动在上下文中添加或删除实体。 |

[顶部](https://msdn.microsoft.com/zh-cn/library/gg696460" \l "mainBody)

方法

显示: 继承 保护

|  |  |  |
| --- | --- | --- |
|  | **名称** | **描述** |
| 公共方法 | [Add](https://msdn.microsoft.com/zh-cn/library/gg679587) | 将给定实体以“已添加”状态添加到集的基础上下文中，这样一来，当调用 SaveChanges 时，会将该实体插入到数据库中。 |
| 公共方法 | [AddRange](https://msdn.microsoft.com/zh-cn/library/dn220168) | 将给定实体集合添加到基础化集的上下文中（每个实体都置于“已添加”状态），这样当调用 SaveChanges 时，会将它插入到数据库中。 |
| 公共方法 | [AsNoTracking](https://msdn.microsoft.com/zh-cn/library/gg696268) | 返回一个新查询，其中返回的实体将不会在 [DbContext](https://msdn.microsoft.com/zh-cn/library/system.data.entity.dbcontext) 中进行缓存。 （继承自 [DbQuery<TResult>](https://msdn.microsoft.com/zh-cn/library/gg696530)。） |
| 公共方法 | [AsStreaming](https://msdn.microsoft.com/zh-cn/library/dn176837) | **已过时。** 返回将流式处理结果而非缓存它的新查询。 （继承自 [DbQuery<TResult>](https://msdn.microsoft.com/zh-cn/library/gg696530)。） |
| 公共方法 | [Attach](https://msdn.microsoft.com/zh-cn/library/gg696261) | 将给定实体附加到集的基础上下文中。 也就是说，将实体以“未更改”的状态放置到上下文中，就好像从数据库读取了该实体一样。 |
| 公共方法 | [Create()](https://msdn.microsoft.com/zh-cn/library/gg696136) | 为此集的类型创建新的实体实例。 请注意此实例不会添加或附加到此集。 如果基础上下文配置为创建代理且实体类型满足创建代理的要求，则返回的实例将是一个代理。 |
| 公共方法 | [Create<TDerivedEntity>()](https://msdn.microsoft.com/zh-cn/library/gg696685) | 为此集的类型或派生自此集类型的类型创建新的实体实例。 请注意此实例不会添加或附加到此集。 如果基础上下文配置为创建代理且实体类型满足创建代理的要求，则返回的实例将是一个代理。 |
| 公共方法 | [Equals](https://msdn.microsoft.com/zh-cn/library/gg696574) | 确定指定的 DbSet 是否等于当前 DbSet。 （重写 [DbQuery<TResult>.Equals(Object)](https://msdn.microsoft.com/zh-cn/library/gg679513)。） |
| 受保护的方法 | [Finalize](https://msdn.microsoft.com/zh-cn/library/system.object.finalize) | （继承自 [Object](https://msdn.microsoft.com/zh-cn/library/system.object)。） |
| 公共方法 | [Find](https://msdn.microsoft.com/zh-cn/library/gg696418) | 查找带给定主键值的实体。 如果上下文中存在带给定主键值的实体，则立即返回该实体，而不会向存储区发送请求。 否则，会向存储区发送查找带给定主键值的实体的请求，如果找到该实体，则将其附加到上下文并返回。 如果未在上下文或存储区中找到实体，则返回 null。 |
| 公共方法 | [FindAsync(Object[])](https://msdn.microsoft.com/zh-cn/library/dn246933) | 异步查找带给定主键值的实体。 如果上下文中存在带给定主键值的实体，则立即返回该实体，而不会向存储区发送请求。 否则，会向存储区发送查找带给定主键值的实体的请求，如果找到该实体，则将其附加到上下文并返回。 如果未在上下文或存储区中找到实体，则返回 null。 |
| 公共方法 | [FindAsync(CancellationToken,Object[])](https://msdn.microsoft.com/zh-cn/library/dn220169) | 异步查找带给定主键值的实体。 如果上下文中存在带给定主键值的实体，则立即返回该实体，而不会向存储区发送请求。 否则，会向存储区发送查找带给定主键值的实体的请求，如果找到该实体，则将其附加到上下文并返回。 如果未在上下文或存储区中找到实体，则返回 null。 |
| 公共方法 | [GetHashCode](https://msdn.microsoft.com/zh-cn/library/gg696454) | 返回此 DbSet 的哈希函数。 （重写 [DbQuery<TResult>.GetHashCode()](https://msdn.microsoft.com/zh-cn/library/gg696516)。） |
| 公共方法 | [GetType](https://msdn.microsoft.com/zh-cn/library/gg679347) | 返回当前 DbSet 的类型。 |
| 公共方法 | [Include](https://msdn.microsoft.com/zh-cn/library/gg696785) | 指定要包括在查询结果中的相关对象。 （继承自 [DbQuery<TResult>](https://msdn.microsoft.com/zh-cn/library/gg696530)。） |
| 受保护的方法 | [MemberwiseClone](https://msdn.microsoft.com/zh-cn/library/system.object.memberwiseclone) | （继承自 [Object](https://msdn.microsoft.com/zh-cn/library/system.object)。） |
| 公共方法 | [Remove](https://msdn.microsoft.com/zh-cn/library/gg679171) | 将给定实体标记为“已删除”，这样一来，当调用 SaveChanges 时，将从数据库中删除该实体。 请注意，在调用此方法之前，该实体必须以另一种状态存在于该上下文中。 |
| 公共方法 | [RemoveRange](https://msdn.microsoft.com/zh-cn/library/dn220170) | 从基础化集的上下文中删除给定实体集合（每个实体都置于“已删除”状态），这样当调用 SaveChanges 时，会从数据库中删除它。 |
| 公共方法 | [SqlQuery](https://msdn.microsoft.com/zh-cn/library/gg696332) | 创建一个原始 SQL 查询，该查询将返回此集中的实体。 默认情况下，上下文会跟踪返回的实体；可通过对返回的 [DbSqlQuery<TEntity>](https://msdn.microsoft.com/zh-cn/library/gg679457) 调用 AsNoTracking 来更改此设置。 请注意返回实体的类型始终是此集的类型，而不会是派生的类型。 如果查询的一个或多个表可能包含其他实体类型的数据，则必须编写适当的 SQL 查询以确保只返回适当类型的实体。 与接受 SQL 的任何 API 一样，对任何用户输入进行参数化以便避免 SQL 注入攻击是十分重要的。 您可以在 SQL 查询字符串中包含参数占位符，然后将参数值作为附加参数提供。 您提供的任何参数值都将自动转换为 DbParameter。 context.Blogs.SqlQuery("SELECT \* FROM dbo.Posts WHERE Author = @p0", userSuppliedAuthor); 或者，您还可以构造一个 DbParameter 并将它提供给 SqlQuery。 这允许您在 SQL 查询字符串中使用命名参数。 context.Blogs.SqlQuery("SELECT \* FROM dbo.Posts WHERE Author = @author", new SqlParameter("@author", userSuppliedAuthor)); |
| 公共方法 | [ToString](https://msdn.microsoft.com/zh-cn/library/gg696144) | 返回基础查询的 [String](https://msdn.microsoft.com/zh-cn/library/system.string) 表示形式。 （继承自 [DbQuery<TResult>](https://msdn.microsoft.com/zh-cn/library/gg696530)。） |