

Contents

- [Mark an entity for deletion](#)
- [Commit changes to the datasource](#)
- [Delete a new entity](#)
- [Undo an entity marked for deletion](#)
- [Cascaded deletes](#)
- [Other technical considerations](#)

Similar to modifying an entity, **deleting** an entity is a two step process. You first mark the entity for deletion in the [EntityManager](#)'s cache, and then you commit the changes to the datasource.

Mark an entity for deletion

You mark an entity for deletion by calling [EntityAspect.Delete](#).

```
product.EntityAspect.Delete();  
product.EntityAspect.Delete()
```

After calling *Delete*, the [EntityState](#) will be set to *Deleted*.

Commit changes to the datasource

The entity will be deleted from the datasource when you call [EntityManager.SaveChanges](#).

After a successful *SaveChanges*, the *EntityState* will become *Detached* and the reference to the entity is no longer valid and it should not be used.

You should remove the 'deleted' entity from any lists and UI datasources before calling *SaveChanges*. Otherwise, any logic that operates on the entity will likely throw an exception.

Delete a new entity

Calling *Delete* on a newly created entity (one that has not been saved) will immediately discard the entity since it does not exist in the datasource.

Undo an entity marked for deletion

If an entity is marked for deletion, you can call [EntityAspect.RejectChanges](#) to undo the scheduled deletion. (You can also call [EntityManager.RejectChanges](#) to revert everything in the cache.) Once *SaveChanges* is successfully called, you can no longer reject the changes, and another save will be required to modify the datasource. To learn more about undoing changes in the client-side cache see [Undo changes](#).

Cascaded deletes

If the Entity Data Model and the database have been configured to handle cascaded deletes, other dependent entities may be deleted as well. For example, deleting an Order may also result in the deletion of all of its OrderDetails. See [Handle cascaded deletes](#) for more information.

Other technical considerations

If you access the properties of the deleted entity, they will be the original values as they were queried from the datasource, as this represents the entity that is being deleted.