Cheatography

Unity Framework Cheat Sheet

by mrtgr via cheatography.com/23460/cs/5171/

FindById

```
var entity = entityRepository.FindById(entity.Id)
//Load with properties
var entity = entityRepository.FindById(entity.Id, x=>
x.OtherEntity, t=>t.AnotherEntityCollection);
```

You can include properties with the query seperated by coma. This method calls *FindAll* with *SingleOrDefault* method.

FindAll

```
//Parameterles.
entityRepository.FindAll();
//Get entities which SomeProperty = Predicate
entity.Repository.FindAll(x=>x.SomeProperty ==
"Predicate");
//Get entity and its Child Collection
entity.Repository.FindAll(x=>x.ChildCollection)
//Get entities which SomeProperty = Predicate and
include Child Collection
entity.Repository.FindAll(x=>x.SomeProperty=
"Predicate" , t=>t.ChildCollection);
```

You can extend search result with *DynamicQueryable* methods like where, select, skip, orderby,groupby etc.

Add

```
//Entity without a parent
entity = new entity() { ... };
entityRepository.Add(entity);
//Child object that has parent
entity = new entity() { ... , parentId= 3 };
entityRepository.Add(entity);
//OR
//Child Object that has parent
entity = new entity() { ... };
parent = parentRepository.FindyById(parentId);
parent.ChildEntityCollection.Add(entity);
```

You have 3 options. If entity doesn't have a parent, simply add with repository. Else if entity has parent you must set its parentId or set parents itself.

Remove

```
//With Entity Id
someModelRepository.Remove(entity.Id);
//With Entity
someModelRepository.Remove(entity);
```

This code will raises an error if entity has child object and relation in the DB doesn't **cascade delete**.

How to delete child object

```
//One-To-One Relation
parent.Child = null;
//One-To-Many Relation
parent.ChildCollection.Remove(Child);
//OR
parent.ChildCollection.Clear()
```

One-To-Many Relations: Remember to mark *parentId* property in child entity with *[Owner]* attribute. Also in setter method of the parent navigation property, set *parentId = 0* if *value == null*. If not, you'll get orphaned entity exception

UnitOfWork

```
try
{
    using (unitOfWorkFactory.Create())
    {
        //Add, Remove, Modify
    }
}
catch (ModelValidationException mvex)
{
    foreach (var error in mvex.ValidationErrors)
    {
        ModelState.AddModelError(error.MemberNames.Fi
rstOrDefault() ?? "", error.ErrorMessage);
    }
}
```

You must use *UnitOfWork* if you want to add, edit or delete.

By **mrtgr**
cheatography.com/mrtgr/

Published 28th December, 2017.
Last updated 12th May, 2016.
Page 1 of 1.