

# LINQ

## IQueryable

This is used when you want to add and/or filter results from the database, without the query being executed already. This is generally used with methods like **ValidSamples()** and **FilterSamples()**

The [IQueryable](#) interface inherits the [IEnumerable](#) interface so that if it represents a query, the results of that query can be enumerated.

## IEnumerable

Use when you deal with in process memory object collections and loop through the collection objects. IEnumerable uses Func objects that result in the query being **executed immediately and completely**, your application will see a performance boost.

Example:

```
using SignifyHR.Core;
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;

namespace SignifyHR.Data.Domain
{
    public partial class exSample : IAuditable
    {
        #region Search Parameters

        public class SearchParameters : BaseSearchParameters
        {
            public int? Someld { get; set; }
            public string Description { get; set; }
            public bool IsUsed { get; set; }
        }
    }
}
```

```
}
```

```
#endregion
```

```
#region Eager Load Parameters
```

```
public class EagerLoadParameters : BaseSearchParameters
```

```
{
```

```
    public bool IncludeSampleDocuments { get; set; }
```

```
    public bool IncludeSampleComments { get; set; }
```

```
}
```

```
#endregion
```

```
#region Protected Methods
```

```
    protected static IEnumerable<exSample> ValidSamples(SignifyHRDAL dbContext, EagerLoadParameters  
eagerLoadParms = null, SearchParameters searchParms)
```

```
{
```

```
    ///Result not executed yet
```

```
    var samples = dbContext.exSamples.AsQueryable();
```

```
    ///Result not executed yet
```

```
    if (eagerLoadParms != null)
```

```
{
```

```
    if (eagerLoadParms.IncludeSampleDocuments )
```

```
        samples = samples.Include(item => item.exSampleDocuments);
```

```
    if (eagerLoadParms.IncludeSampleComments )
```

```
        samples = samples.Include(item => item .exSampleComments);
```

```
}
```

```
    //Result not executed yet
```

```
    if (searchParms!= null)
```

```
{
```

```
    if (!String.IsNullOrEmpty(searchParms.Description))
```

```
        samples = samples.Where(item =>
```

```
item.Description.ToLower().Contains(searchParms.Description.ToLower()));
```

```
    if (searchParms.Someld.HasValue)
```

```
        samples = samples.Where(item => item.Someld == searchParms.Someld.Value);

        if (searchParms.IsUsed.HasValue)
            samples = samples.Where(item => item.IsUsed == searchParms.IsUsed.Value);
    }

    ///Result executed immediately and completely
    return samples.AsEnumerable();
}
}
```

TODO: Add example for IQueryable

As many calculations, filtering should occur in IQueryable

Naming convention for IQueryable e.g. Valid{ObjectNames}

---

Revision #7

Created 17 September 2020 01:58:16 by Theuns Pretorius

Updated 1 October 2020 05:43:39 by Theuns Pretorius