Namespace Mentor4U_Avalonia.BLL

Classes

[Roles](#)

Class Roles

Namespace: [Mentor4U_Avalonia.BLL](#)

Assembly: Mentor4U_Avalonia.BLL.dll

```csharp
public class Roles
```

Inheritance

[object](#) ← Roles

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

Roles(ICrud<Role>)

```csharp
public Roles(ICrud<Role> data)
```

Parameters

`data` [ICrud](#)<[Role](#)>

Methods

CreateAsync(Role)

```csharp
public Task<Role?> CreateAsync(Role role)
```

Parameters

`role` [Role](#)

Returns

Task<Role>

DeleteAsync(Role)

```
public Task<bool> DeleteAsync(Role role)
```

Parameters

role Role

Returns

Task<bool>

GetAllAsync()

```
public Task<IEnumerable<Role>?> GetAllAsync()
```

Returns

Task<IEnumerable<Role>>

GetByIdAsync(int)

```
public Task<Role?> GetByIdAsync(int id)
```

Parameters

id int

Returns

Task<Role>

GetByNameAsync(string)

```
public Task<Role?> GetByNameAsync(string name)
```

Parameters

name string

Returns

Task <Role>

Namespace Mentor4U_Avalonia.DAL

Classes

DbHelper

[DbService<TEntity>](DbService)

Roles

Interfaces

[ICrud<TEntity>](ICrud)

Class DbHelper

Namespace: [Mentor4U_Avalonia.DAL](#)

Assembly: Mentor4U_Avalonia.DAL.dll

```
public static class DbHelper
```

Inheritance

[object](#) ← DbHelper

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Fields

RoleTablesColumnNames

```
public static readonly Dictionary<string, string> RoleTablesColumnNames
```

Field Value

[Dictionary](#) <[string](#), [string](#)>

TableNames

```
public static readonly Dictionary<Type, string> TableNames
```

Field Value

[Dictionary](#) <[Type](#), [string](#)>

Class DbService<TEntity>

Namespace: [Mentor4U_Avalonia](#).[DAL](#)

Assembly: Mentor4U_Avalonia.DAL.dll

```
public static class DbService<TEntity> where TEntity : IModel
```

Type Parameters

`TEntity`

Inheritance

[object](#) ← DbService<TEntity>

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Methods

ConnectAsync(string)

```
public static Task ConnectAsync(string connectionString)
```

Parameters

`connectionString` [string](#)

Returns

[Task](#)

DisconnectAsync()

```
public static Task DisconnectAsync()
```

Returns

Task

ExecuteNonQueryAsync(string, object?)

```csharp
public static Task<bool> ExecuteNonQueryAsync(string sqlRaw, object? parameters = null)
```

Parameters

sqlRaw string

parameters object

Returns

Task < bool >

GetAllAsync()

```csharp
public static Task<IEnumerable<TEntity>> GetAllAsync()
```

Returns

Task < IEnumerable <TEntity> >

GetByIdAsync(int)

```csharp
public static Task<TEntity?> GetByIdAsync(int id)
```

Parameters

id int

Returns

[Task](#)⧉ <TEntity>

Interface ICrud\<TEntity\>

Namespace: [Mentor4U_Avalonia.DAL](#)

Assembly: Mentor4U_Avalonia.DAL.dll

```csharp
public interface ICrud<TEntity> where TEntity : IModel
```

Type Parameters

`TEntity`

Methods

DeleteAsync(int)

```csharp
Task<bool> DeleteAsync(int id)
```

Parameters

`id` [int](#)

Returns

[Task](#)<[bool](#)>

GetAllAsync()

```csharp
Task<IEnumerable<TEntity>?> GetAllAsync()
```

Returns

[Task](#)<[IEnumerable](#)<TEntity>>

GetAsync(int)

```
Task<TEntity?> GetAsync(int id)
```

Parameters

id [int](#)

Returns

[Task](#) <TEntity>

InsertAsync(TEntity)

```
Task<bool> InsertAsync(TEntity entity)
```

Parameters

entity TEntity

Returns

[Task](#) <[bool](#)>

UpdateAsync(TEntity)

```
Task<bool> UpdateAsync(TEntity entity)
```

Parameters

entity TEntity

Returns

[Task](#) <[bool](#)>

Class Roles

Namespace: [Mentor4U_Avalonia.DAL](#)

Assembly: Mentor4U_Avalonia.DAL.dll

```
public class Roles : ICrud<Role>
```

Inheritance

[object](#) ← Roles

Implements

[ICrud](#)<[Role](#)>

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

Roles(string)

```
public Roles(string connectionString)
```

Parameters

`connectionString` [string](#)

Methods

DeleteAsync(int)

```
public Task<bool> DeleteAsync(int id)
```

Parameters

`id` [int](#)

Returns

Task <bool>

GetAllAsync()

```
public Task<IEnumerable<Role>?> GetAllAsync()
```

Returns

Task <IEnumerable <Role>>

GetAsync(int)

```
public Task<Role?> GetAsync(int id)
```

Parameters

id int

Returns

Task <Role>

InsertAsync(Role)

```
public Task<bool> InsertAsync(Role entity)
```

Parameters

entity Role

Returns

Task <bool>

UpdateAsync(Role)

```
public Task<bool> UpdateAsync(Role entity)
```

Parameters

`entity` [Role](#)

Returns

[Task](#) < [bool](#) >

Namespace Mentor4U_Avalonia.Models

Classes

EmptyStringException

ExceptionExtensions

NegativeNumberException

Role

Interfaces

IModel

    Интерфейс модели данных

Class EmptyStringException

Namespace: [Mentor4U_Avalonia](#).[Models](#)

Assembly: Mentor4U_Avalonia.Models.dll

```csharp
public class EmptyStringException : Exception, ISerializable
```

Inheritance

[object](#) ← [Exception](#) ← EmptyStringException

Implements

[ISerializable](#)

Inherited Members

[Exception.GetBaseException()](#) , [Exception.GetType()](#) , [Exception.ToString()](#) , [Exception.Data](#) ,
[Exception.HelpLink](#) , [Exception.HResult](#) , [Exception.InnerException](#) , [Exception.Message](#) ,
[Exception.Source](#) , [Exception.StackTrace](#) , [Exception.TargetSite](#) , [Exception.SerializeObjectState](#) ,
[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#)

Constructors

EmptyStringException(string)

```csharp
public EmptyStringException(string message)
```

Parameters

`message` [string](#)

Class ExceptionExtensions

Namespace: [Mentor4U_Avalonia](#).[Models](#)

Assembly: Mentor4U_Avalonia.Models.dll

```
public static class ExceptionExtensions
```

Inheritance

[object](#) ← ExceptionExtensions

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Methods

LoggingIfException(ILogger, Action, string, string)

```
public static void LoggingIfException(ILogger logger, Action action, string moduleName,
string methodName)
```

Parameters

`logger` ILogger

`action` [Action](#)

`moduleName` [string](#)

`methodName` [string](#)

Interface IModel

Namespace: [Mentor4U_Avalonia](#).[Models](#)

Assembly: Mentor4U_Avalonia.Models.dll

Интерфейс модели данных

```
public interface IModel
```

Properties

Id

Уникальный идентификатор

```
int Id { get; set; }
```

Property Value

[int](#)↗

Class NegativeNumberException

Namespace: [Mentor4U_Avalonia](#).[Models](#)

Assembly: Mentor4U_Avalonia.Models.dll

```
public class NegativeNumberException : Exception, ISerializable
```

Inheritance

[object](#) ← [Exception](#) ← NegativeNumberException

Implements

[ISerializable](#)

Inherited Members

[Exception.GetBaseException()](#) , [Exception.GetType()](#) , [Exception.ToString()](#) , [Exception.Data](#) , [Exception.HelpLink](#) , [Exception.HResult](#) , [Exception.InnerException](#) , [Exception.Message](#) , [Exception.Source](#) , [Exception.StackTrace](#) , [Exception.TargetSite](#) , [Exception.SerializeObjectState](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#)

Constructors

NegativeNumberException(string)

```
public NegativeNumberException(string message)
```

Parameters

`message` [string](#)

Class Role

Namespace: [Mentor4U_Avalonia](#).[Models](#)

Assembly: Mentor4U_Avalonia.Models.dll

```
public record Role : IModel, IEquatable<Role>
```

Inheritance

[object](#) ← Role

Implements

[IModel](#), [IEquatable](#)<[Role](#)>

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Properties

Id

Уникальный идентификатор

```
public int Id { get; set; }
```

Property Value

[int](#)

Exceptions

[NegativeNumberException](#)

Исключение выбрасывается, если присвоить отрицательное или нулевое значение

RoleName

```csharp
public string RoleName { get; set; }
```

Property Value

[string](#)