# BLToolkit

Business Logic Toolkit is a set of components to simplify .NET application development. BLToolkit is provided as source code that you can use "as is" or customize for your applications.

<https://github.com/igor-tkachev/bltoolkit/wiki>

# Vesmir

Ve vesmiru jsme pri dokonceni AdoNetu, meli entity bez atributu od niceho nebyly podedene.

Po instalaci balicku BLToolkit, jsme entitam pridali atributy a podedili jsme je od Mitrozovych trid, podle typu primarniho klice.

Napr Galaxie dostala takoveto atributy:

using BLToolkit.DataAccess;

using Mediaresearch.Framework.DataAccess.BLToolkit;

using Mediaresearch.Framework.DataAccess.BLToolkit.DaoFactory;

namespace WpfUniverse.Entities

{

[DaoFactory(DaoType = typeof(GalaxyDao))]

[TableName("Galaxie", Owner = "dbo")]

public class Galaxie : LightDatabaseEntityIdentityIntKey<Galaxie>

{

public string Jmeno { get; set; }

public long PolohaX { get; set; }

public long PolohaY { get; set; }

public long PolohaZ { get; set; }

}

}

GalaxyDao dostalo takoveto podedeni, a dva konstruktory, které bazovce predaji dbAlias

using Mediaresearch.Framework.DataAccess.BLToolkit.Dao;

using System.Data;

namespace WpfUniverse.Entities

{

public class GalaxyDao : EntityIdentityKeyDaoBase<Galaxie, UniverseDataModel, int, Galaxie>, IGalaxyDao

{

public GalaxyDao(string dbAlias)

:base(dbAlias)

{

}

public GalaxyDao(string dbAlias, IsolationLevel isolationLevel)

: base(dbAlias, isolationLevel)

{

}

}

public interface IGalaxyDao : ISimpleDao<Galaxie, int>

{

}

}

Abychom mohli pouzivat BLToolkit potrebujeme DataModel který je podedeny od TransactionManagera. Ma 8 konstruktoru.7

public class UniverseDataModel : TransactionDbManager

public Table<Galaxie> Galaxie => GetTable<Galaxie>();

public Table<Planeta> Planeta => GetTable<Planeta>();

public Table<Vlastnost> Vlastnost => GetTable<Vlastnost>();

public Table<VlastnostiPlanet> VlastnostiPlanet => GetTable<VlastnostiPlanet>();

GetTable<> je metoda na managerovi