**业务架构使用文档**

1. **新增实体类**

注:实体名与表名一致

实例:

using System;

using System.ComponentModel.DataAnnotations;

using System.ComponentModel.DataAnnotations.Schema;

namespace Pelway.Entity

{

/// <summary>

/// 城市信息

/// </summary>

[Table("CityInfo")]

public class CityInfo

{

/// <summary>

/// 城市编码

/// </summary>

[Display(Name = "城市编码")]

[Key]

public int Id { get; set; }

/// <summary>

/// 城市名称

/// </summary>

[Column(TypeName = "VARCHAR")]

[Display(Name = "城市名称")]

[StringLength(20)]

[Required]

public string CityName { get; set; }

/// <summary>

/// 创建时间

/// </summary>

[Display(Name = "创建时间")]

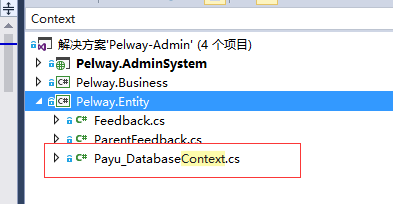
[Required]

public DateTime CreateTime { get; set; }

}

}

1. **加入到数据上下文中**
   1. 搜索: Context



b) 加入到数据上下文中

注释：数据上下文可以理解为数据库一样，DbSet中的对象是映射对应的表

/// <summary>

///城市信息集合

/// </summary>

public DbSet<CityInfo> CityInfos { get; set; }

1. **新增对应的业务**

在业务层加上对应的业务，命名规则为：实体名+Business，继承BaseBusiness

如：CityInfoBusiness

实例类：

/// <summary>

/// CityInfo 业务

/// </summary>

public class CityInfoBusiness : BaseBusiness<CityInfo>

{

}

1. **继承** BaseBusiness 后的方法以及使用

注释:如下用实例的方式调用

CityInfoBusinesscityInfoBusiness **=new** CityInfoBusiness();

* 1. 根据表达式查询返回总个数.( GetCount)

// 返回Id 小于的个数

int count=cityInfoBusiness.GetCount(t=>t.Id<100).Data;

// 生成的Sql语句为：select COUNT(1) from cityinfo where id<100

* 1. 根据 where 返回集合对象

List<City> list=cityInfoBusiness.GetByWhere(t=>t.Id<100).Data;

//生成的Sql语句为：select \* from cityinfo where id<100

* 1. 根据 where 条件返回 自定义集合对象

List< CityViewModel> list=

cityInfoBusiness.GetByWhere< CityViewModel >(t=>t.Id<100,t=>new CityViewModel {

Id=t.Id,

CityName=t.CityName

}).Data;

//生成的Sql语句为：select Id,CityName from cityinfo where id<100

//城市视图模型

public class CityViewModel{

public int Id { get; set; }

public string CityName { get; set; }

}

* 1. 根据表达式返回第一个元素

City city =cityInfoBusiness. GetFirstByWhere (t=>t.Id==1).Data;

//生成的Sql语句为：select \* from cityinfo where id=1

City city =cityInfoBusiness. GetFirstByWhere (t=>t.CityName ==”北京”).Data;

//生成的Sql语句为：select \* from cityinfo where CityName ==”北京”

* 1. 根据表达式返回第一个自定义元素

City city =cityInfoBusiness.GetFirstByWhere<> (t=>t.Id==1,t=> CityViewModel{

Id=t.Id,

CityName=t.CityName

}).Data;

//生成的Sql语句为：select Id,CityName from cityinfo where id=1

* 1. 直接调用sql 语句,返回单个对象

City city=cityInfoBusiness.ExecSelectSql(“select Id,CityName from cityinfo where id=@id”,new SqlParameter(“@id”,1));

* 1. 直接调用sql 语句,返回集合对象

List<City> cityList=cityInfoBusiness.ExecSelectSqlToList (“select Id,CityName from cityinfo where id<@id”,new SqlParameter(“@id”,100));

* 1. 调用命令型Sql，如，存储过程

cityInfoBusiness.ExecCommandSql(“ sql语句 ”,

new SqlParameter(“@参数1”,参数值)，

new SqlParameter(“@参数2”,参数值)

);

* 1. 根据表达式删除

// 删除id 等于 1 的数据

cityInfoBusiness.Remove(t=>t.Id==1);

//生成的Sql语句为：delete from cityinfo where id=1

//删除id 小于100的数据

cityInfoBusiness.Remove(t=>t.Id<100)

//生成的Sql语句为：delete from cityinfo where id<100

* 1. 删除，方法2

cityInfoBusiness.Delete(t=>t.Id==1);

//生成的Sql语句为：delete from cityinfo where id=1

* 1. 更新数据，方法1

City city= cityInfoBusiness.GetFirstByWhere (t=>t.Id==1).Data;

city.CityName=”修改字段”;

cityInfoBusiness.Update(city);

//生成的Sql语句为：update cityinfo set cityname=”修改字段”, CreateTime=”还是原来的值”

where id=1

* 1. 更新数据,方法2

cityInfoBusiness.Update(t=>t.Id==1,t=>new CityInfo{

CityName=”修改字段”

});

//生成的Sql语句为：update cityinfo set cityname=”修改字段” where id=1

* 1. 新增，添加单个数据

CityInfo cityInfo=new CityInfo();

cityInfo.CityName=””;

cityInfoBusiness. Add(cityInfo);

* 1. 新增，添加集合数据

List< CityInfo > list=new List< CityInfo >();

CityInfo cityInfo=new CityInfo();

cityInfo.CityName=””;

list.Add(cityInfo);

cityInfoBusiness.Add(list);

* 1. 数据是否存在

// true 表示已经存在了，false 表示还没有存在该数据

bool isAny=cityInfoBusiness.AnyByWhere(t=>t.Id==1).Data;

* 1. 调用分页查询

var query =cityInfoBusiness.GetQueryable();

// where 条件

// id 小于100的数据

query.where(t=>t.Id<100);

// 必须先排序

query = query.OrderBy(t => t.Id);

var paging=new Paging();

var pageResult=ExecutePage(query, paging);

// 返回值

pageResult.Data

// 分页信息

pageResult.Paging