|  |
| --- |
|  |
| Dao Actions Document |
|  |
|  |
| ©MetaShare Inc. |
| **8/22/2022** |

|  |
| --- |
| How To Use Dao |

|  |  |  |
| --- | --- | --- |
| **Date** | **Modifier** | **Description** |
| 2022-08-22 | MDDGroup | first draft |
|  |  |  |
|  |  |  |

目录

[1.1 Instruction 1](#_Toc112078090)

[1.2 Specific Usage 1](#_Toc112078091)

[1.2.1 Entity definition 1](#_Toc112078092)

[1.2.2 Dao definition 2](#_Toc112078093)

[1.2.3 Dao’s Methods and properties included by default 2](#_Toc112078094)

[1.2.4 DaoInterface definition 4](#_Toc112078095)

[1.2.5 DaoInterface’s Methods included by default 5](#_Toc112078096)

[1.2.6 Dao’s Use extensions for custom queries 7](#_Toc112078097)

[1.2.7 RegisterDao 8](#_Toc112078098)

[1.2.8 Use of Dao in Service 8](#_Toc112078099)

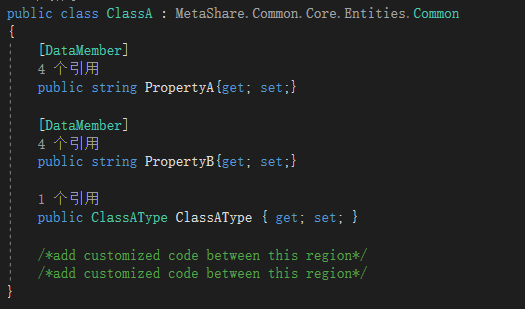
## Instruction

MMC generates project's Dao description and usage instructions;

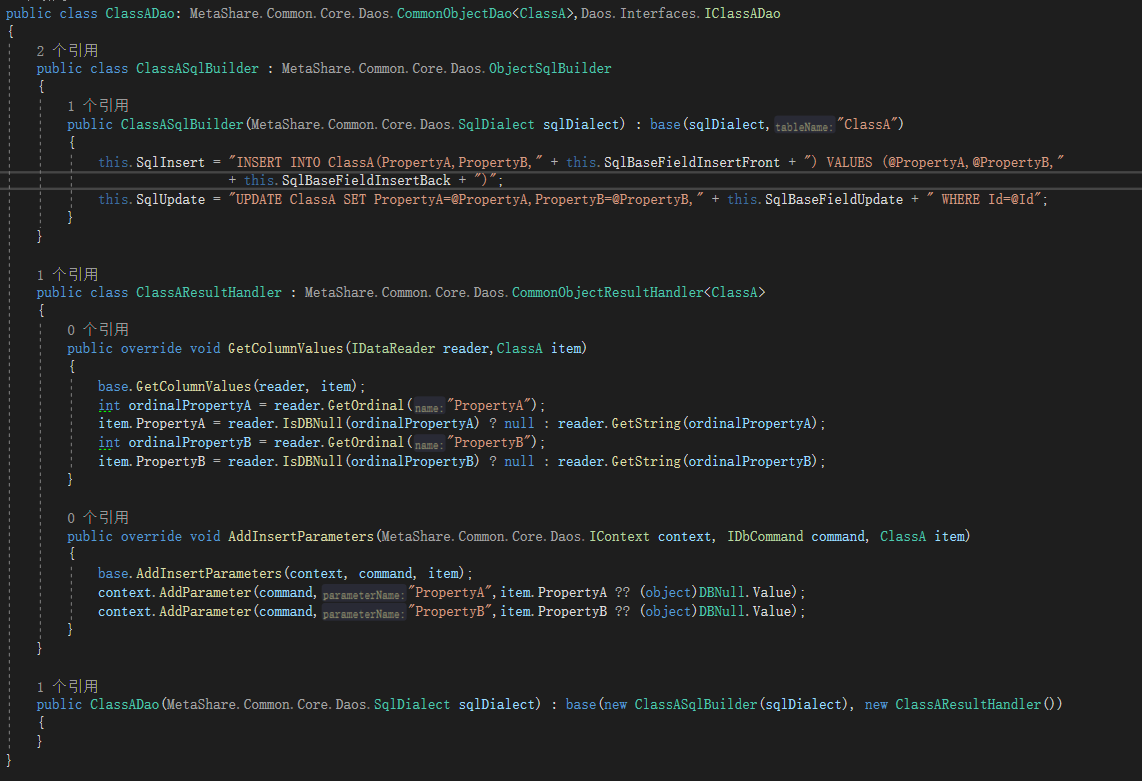
The base is on CommonCore’s Dao description and usage instructions.

## Specific Usage

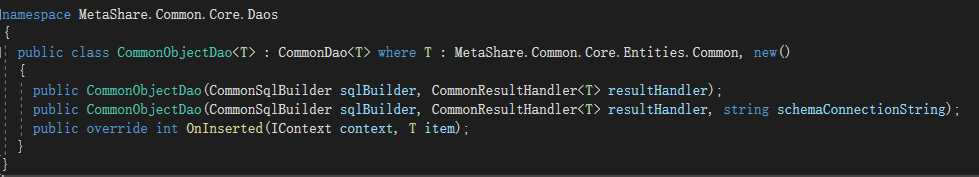
### Entity definition

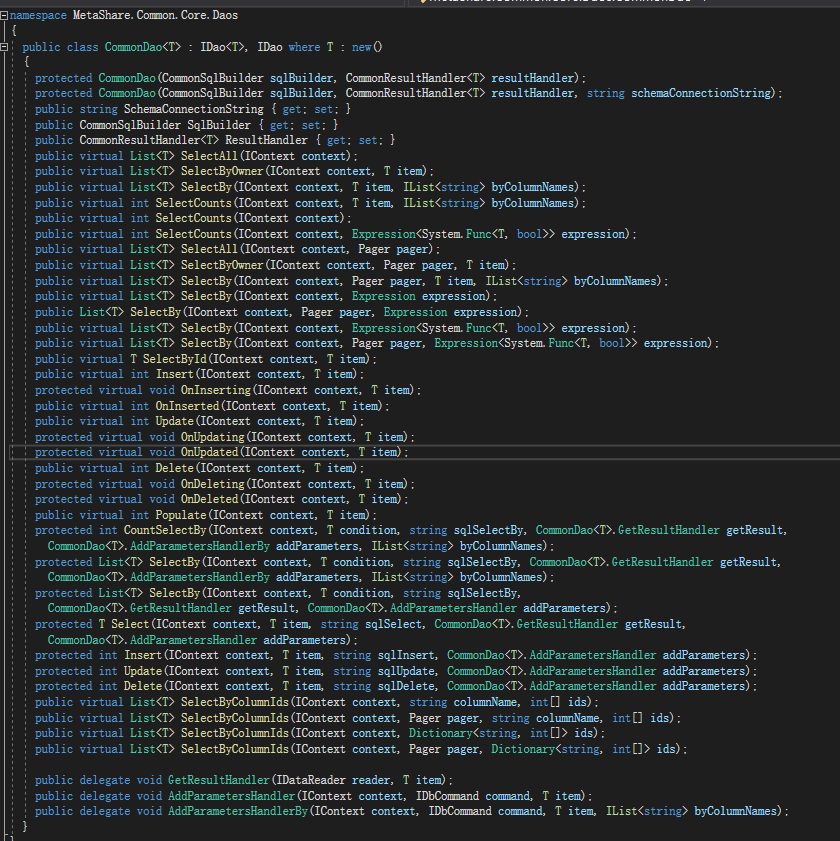


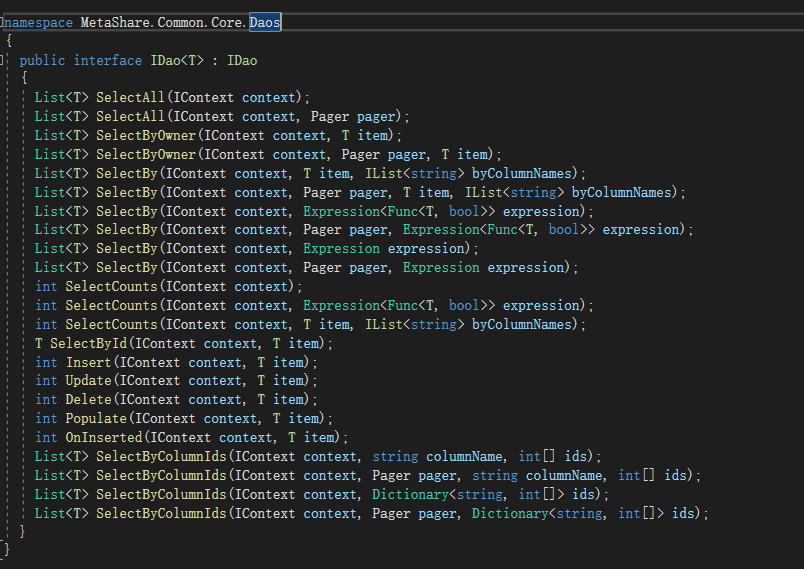
### Dao definition



### Dao’s Methods and properties included by default

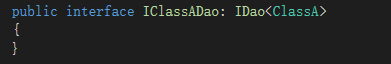




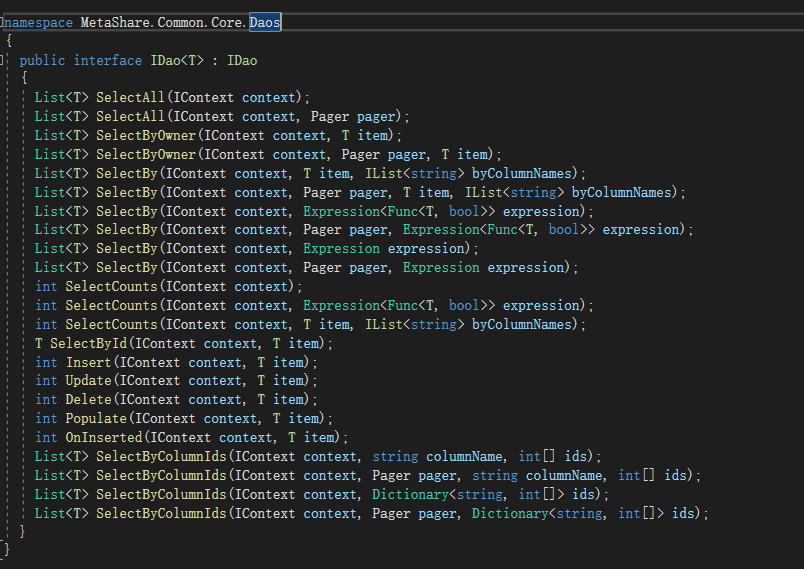


New methods can be rewritten or defined on this basis to meet usage requirements.

### DaoInterface definition



### DaoInterface’s Methods included by default



List<T> SelectAll(IContext context); --- Query all data sets

List<T> SelectAll(IContext context, Pager pager);--- Query the data collection by all data page

List<T> SelectByOwner(IContext context, T item);

--- The Owner query - method is not implemented yet

List<T> SelectByOwner(IContext context, Pager pager, T item);

--- Query through Owner and page - method not implemented yet

List<T> SelectBy(IContext context, T item, IList<string> byColumnNames);

--- Query by column criteria

List<T> SelectBy(IContext context, Pager pager, T item, IList<string> byColumnNames);

--- Query and page through column criteria

List<T> SelectBy(IContext context, Expression<Func<T, bool>> expression);

--- Query by Lambda expression condition

List<T> SelectBy(IContext context, Pager pager, Expression<Func<T, bool>> expression);

--- Query and page through Lambda expression conditions

List<T> SelectBy(IContext context, Expression expression);

--- Query by expression

List<T> SelectBy(IContext context, Pager pager, Expression expression);

---Query and page through expression

int SelectCounts(IContext context);--- Query quantity

int SelectCounts(IContext context, Expression<Func<T, bool>> expression);

--- Query the quantity by Lambda expression condition

int SelectCounts(IContext context, T item, IList<string> byColumnNames);

--- Query quantity by column

T SelectById(IContext context, T item);--- Query data by Id

int Insert(IContext context, T item);--- Insert data

int Update(IContext context, T item);--- Update data

int Delete(IContext context, T item);---Delete data

int Populate(IContext context, T item);---Insert data

int OnInserted(IContext context, T item);--- Gets the Id after the insert is successful

List<T> SelectByColumnIds(IContext context, string columnName, int[] ids);

--- Query data through the data collection of columns

List<T> SelectByColumnIds(IContext context, Pager pager, string columnName, int[] ids);

--- Query paging data through the data collection of columns

List<T> SelectByColumnIds(IContext context, Dictionary<string, int[]> ids);

--- Query data through a column data set dictionary

List<T> SelectByColumnIds(IContext context, Pager pager, Dictionary<string, int[]> ids);

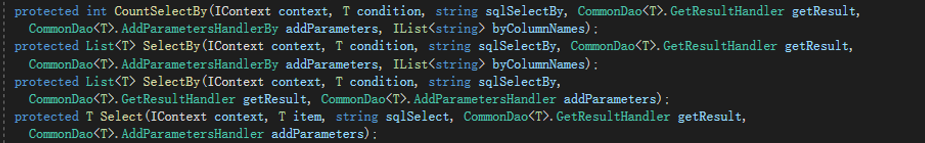
--- Query paging data through the column data set dictionary

### Dao’s Use extensions for custom queries

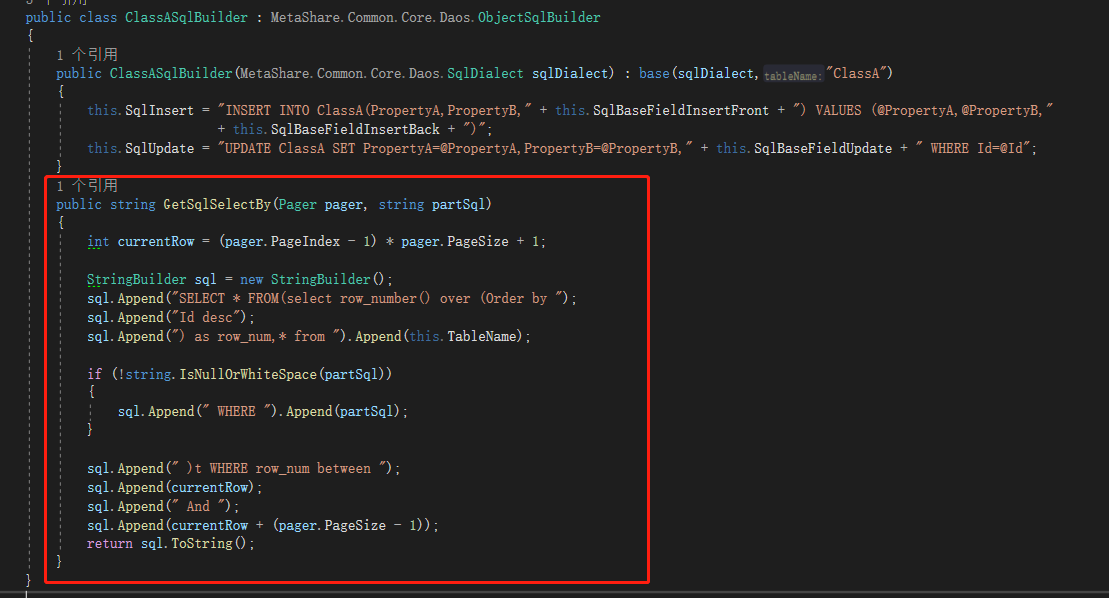
1. Define methods to get SQL statements in SqlBuilder;
2. Define methods for loading data into entities and methods for loading parameters in ResultHandler;

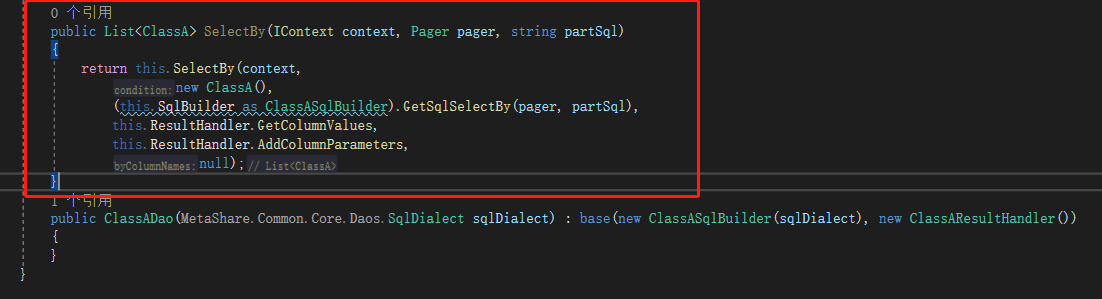
（PS：Skip this step if you can use existing methods）

1. Define the query and call the following methods to complete the query in Dao.

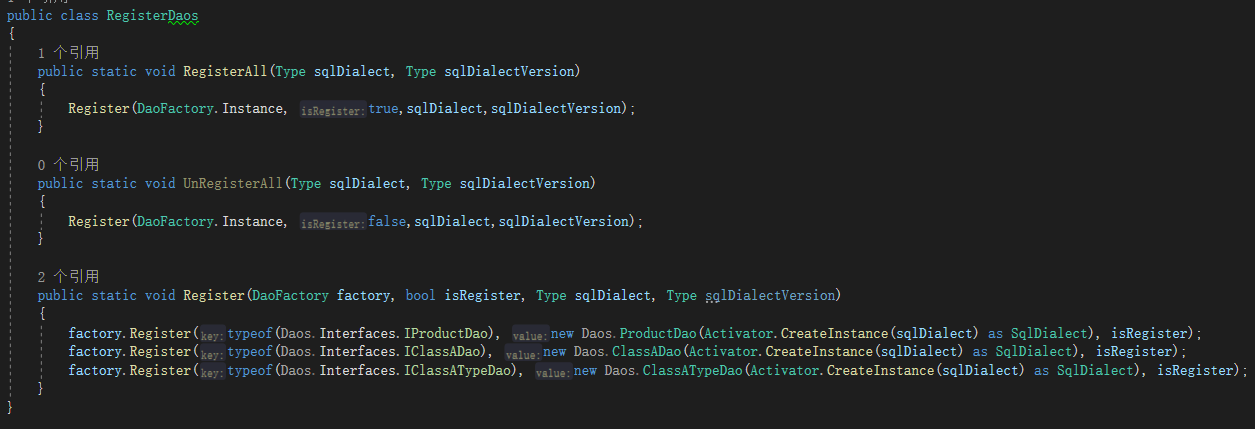


Eg.





### RegisterDao



### Use of Dao in Service

In itself Service



In other’s Service

