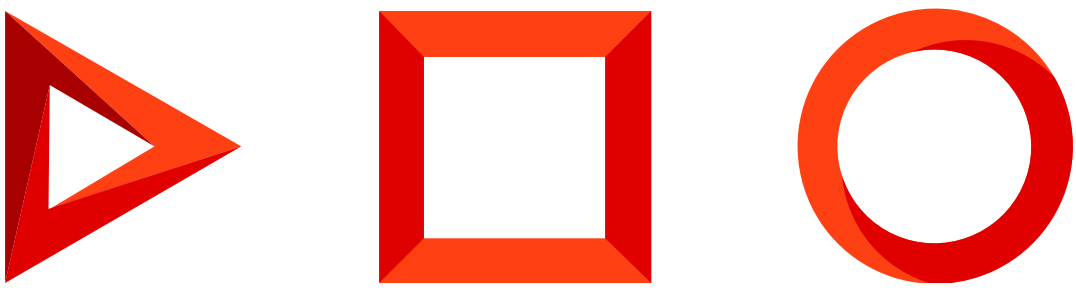
Creatio Academy

Операции с данными (back-end)

Доступ к данным через ORM

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

© 2023 Terrasoft. Все права защищены.

Содержание

© 2023 Terrasoft. Все права защищены.

© 2023 Terrasoft. Все права защищены.

Доступ к данным через ORM

Средний

Способы доступа к базе данных, которые предоставляют back-end компоненты ядра:

- Доступ через ORM-модель.
- Прямой доступ.

В этой статье будет рассмотрено выполнение запросов к базе данных через ORM-модель.

ORM (Object-relational mapping) — технология, которая позволяет работать с данными, полученными из базы данных, путем использования объектно-ориентированных языков программирования. **Назначение** ORM — связывание объектов, реализованных в коде, с записями в таблицах базы данных.

Классы, которые реализуют работу с данными через ORM-модель данных:

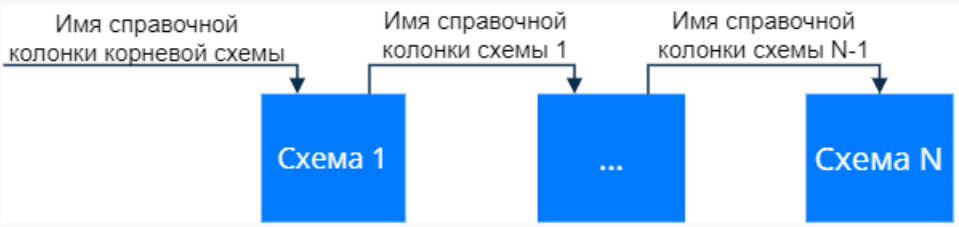
- `Terrasoft.Core.Entities.EntitySchemaQuery` — построение запросов на получение записей из таблиц базы данных с учетом прав доступа текущего пользователя.
- `Terrasoft.Core.Entities.Entity` — доступ к сущности, которая представляет собой запись в таблице базы данных.

Для доступа к данным рекомендуется использовать ORM-модель, хотя прямой доступ к базе данных также реализован в back-end компонентах ядра. Выполнение запросов к базе данных через прямой доступ подробно описано в статье Прямой доступ к данным.

Сформировать путь к колонке относительно корневой схемы

Основой механизма построения запроса на выборку с применением `EntitySchemaQuery` является корневая схема. **Корневая схема** — это таблица в базе данных, относительно которой строятся пути к колонкам в запросе, в том числе к колонкам присоединяемых таблиц. Для использования в запросе колонки из таблицы базы данных необходимо корректно задать путь к этой колонке.

При построении путей к колонкам применяется **принцип связей через справочные поля**. Имя колонки, добавляемой в запрос, строится в виде цепочки взаимосвязанных звеньев. Каждое звено представляет собой "контекст" конкретной схемы, которая связывается с предыдущей по справочной колонке.



Шаблон формирования пути к колонке из схемы N:

© 2023 Terrasoft. Все права защищены.

`КонтекстСхемы1.[...].КонтекстСхемыN.ИмяКолонкиИзСправочнойСхемы`.

Сформировать путь к колонке по прямым связям

Шаблон формирования пути к колонке по **прямым связям**:
`ИмяСправочнойКолонки.ИмяКолонкиСхемыИзСправочнойСхемы`.

Прямые связи используются, когда справочная колонка для связи присутствует в основной схеме и ссылается на присоединяемую схему. Например, есть корневая схема `[City]` со справочной колонкой `[Country]`, которая через колонку `[Id]` связана со справочной схемой `[Country]`.

Путь к колонке с наименованием страны, которой принадлежит город по прямым связям: `Country.Name`. Здесь:

- `Country` — имя справочной колонки корневой схемы `[City]` (ссылается на схему `[Country]`).
- `Name` — имя колонки из справочной схемы `[Country]`.

Сформировать путь к колонке по обратным связям

Отличие присоединения по обратным связям от присоединения по прямым связям — справочное поле для присоединения должно быть у присоединяемой сущности, а не у основной.

Шаблон формирования пути к колонке по **обратным связям**:

```
[ИмяПрисоединяемойСхемы:ИмяКолонкиДляСвязиПрисоединяемойСхемы:ИмяКолонкиДляСвязиТекущейСхемы].
ИмяКолонкиПрисоединяемойСхемы
```

.

Путь к колонке с названием контрагента, у которого в поле `[Город]` указана выбираемая в запросе запись `[City]` по обратным связям: `[Account:City:Id].Name`. Здесь:

- `Account` — имя присоединяемой схемы.
- `City` — имя колонки схемы `[Account]` для установки связи присоединяемой схемы.
- `Id` — имя справочной колонки схемы `[City]` для установки связи текущей схемы.
- `Name` — значение справочной колонки схемы `[Account]`.

Если в качестве колонки для связи у текущей схемы выступает колонка `[Id]`, то ее можно не указывать: `[ИмяПрисоединяемойСхемы:ИмяКолонкиДляСвязиПрисоединяемойСхемы].ИмяКолонкиПрисоединяемойСхемы`. Например, `[Account:City].Name`.

Получить данные из базы данных

Классы, которые реализуют получение данных из базы данных:

© 2023 Terrasoft. Все права защищены.

- `Terrasoft.Core.DB.Select` — получение данных из базы данных через прямой доступ. Подробнее читайте в статье Прямой доступ к данным.
- `Terrasoft.Core.Entities.EntitySchemaQuery` — получение данных из базы данных через ORM-модель.

Назначение класса `Terrasoft.Core.Entities.EntitySchemaQuery` — построение запросов на выборку записей из таблиц базы данных. Максимальное количество записей, которые можно получить по запросу, задается настройкой `MaxEntityRowCount` (по умолчанию — 20 000). Изменить значение настройки можно в файле `...\Terrasoft.WebApp\Web.config`.

> **Важно.** Не рекомендуется изменять настройку `MaxEntityRowCount`. Изменение настройки может привести к проблемам производительности.

После создания и конфигурирования экземпляра класса будет построен `SELECT`-запрос к базе данных приложения. В запрос можно добавить колонки, фильтры и условия ограничений. Также можно задать параметры для постраничного вывода результатов выполнения запроса. Используя класс `Terrasoft.Core.Entities.EntitySchemaQueryOptions`, можно задать параметры построения иерархического запроса. Передача одного и того же экземпляра `EntitySchemaQueryOptions` в качестве параметра метода `GetEntityCollection()` соответствующего запроса позволяет получить результат выполнения различных запросов.

Особенности класса `Terrasoft.Core.Entities.EntitySchemaQuery`:

- В результирующем запросе учитываются права доступа текущего пользователя.
- Класс позволяет управлять правами доступа текущего пользователя на таблицы, присоединенные в запрос с помощью SQL-оператора `JOIN`.
- Класс позволяет работать с данными хранилища кэша или произвольного хранилища, которое определено пользователем.

При выполнении запроса данные, полученные из базы данных, помещаются в кэш. В качестве кэша запроса может выступать произвольное хранилище, которое реализует интерфейс `Terrasoft.Core.Store.ICacheStore`. По умолчанию используется кэш Creatio уровня сессии с локальным хранением данных. Кэш запроса определяется свойством `Cache` экземпляра класса `EntitySchemaQuery`. С помощью свойства `CacheItemName` задается ключ доступа к кэшу запроса. Уровни хранилищ Creatio подробно описаны в статье Хранилища данных и кэш.

Результат выполнения запроса — экземпляр `Terrasoft.Nui.ServiceModel.DataContract.EntityCollection` или коллекция экземпляров класса `Terrasoft.Core.Entities.Entity`. Каждый экземпляр `Entity` в коллекции представляет собой строку набора данных, возвращаемого запросом.

Управлять присоединенными таблицами

Класс `EntitySchemaQuery` позволяет указать тип присоединения схемы к запросу. Для добавления в запрос колонки из присоединяемой схемы используется оператор `JOIN`.

Шаблон формирования типа соединения к колонке присоединяемой схемы:
`СпецсимволТипаСоединенияИмяКолонкиДляСвязиПрисоединяемойСхемы`.

© 2023 Terrasoft. Все права защищены.

Описание типов соединения

Тип соединения	Спецсимвол типа соединения	Пример использования
`INNER JOIN`	`=`	`=Country.Name`
`LEFT OUTER JOIN`	`>`	`>Country.Name`
`RIGHT OUTER JOIN`	`<`	`<Country.Name`
`FULL OUTER JOIN`	`<>`	`<>Country.Name`
`CROSS JOIN`	`*`	`*Country.Name`

По умолчанию используется тип присоединения `LEFT OUTER JOIN`.

Ниже представлен пример добавления колонок в запрос с использованием разных типов соединения.

Пример добавления колонок в запрос

```
/* Создание экземпляра запроса с корневой схемой City. */
var esqResult = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "City");

/* К запросу будет добавлена схема Country с типом присоединения LEFT OUTER JOIN. */
esqResult.AddColumn("Country.Name");

/* К запросу будет добавлена схема Country с типом присоединения INNER JOIN. */
esqResult.AddColumn("=Country.Name");

/* К запросу будут присоединены схема Country с типом присоединения LEFT OUTER JOIN и схема Cont
esqResult.AddColumn(">Country.<CreatedBy.Name");
```

В результате будет сформирован SQL-запрос.

SQL-запрос

```
SELECT
    [Country].[Name] [Country.Name],
    [Country1].[Name] [Country1.Name],
    [CreatedBy].[Name] [CreatedBy.Name]
FROM
    [dbo].[City] [City]
    LEFT OUTER JOIN [dbo].[Country] [Country] ON ([Country].[Id] = [City].[CountryId])
    INNER JOIN [dbo].[Country] [Country1] ON ([Country1].[Id] = [City].[CountryId])
    LEFT OUTER JOIN [dbo].[Country] [Country2] ON ([Country2].[Id] = [City].[CountryId])
    RIGHT OUTER JOIN [dbo].[Contact] [CreatedBy] ON ([CreatedBy].[Id] = [Country2].[CreatedById]
```

© 2023 Terrasoft. Все права защищены.

Если запрос содержит корневую схему и присоединяемые схемы, которые администрируются по записям, то можно применить права доступа текущего пользователя. Варианты применения прав доступа по записям к присоединенным схемам заданы перечислением `Terrasoft.Core.DB.QueryJoinRightLevel`.

Варианты применения прав доступа к присоединяемым схемам запроса:

- Всегда применяются.
- Применяются, если в присоединяемой схеме запроса используются колонки, отличные от первичной и первичной для отображения. Чаще всего это колонки `[Id]` и `[Name]`.
- Не применяются.

Порядок применения прав доступа определяется значением свойства `JoinRightState` запроса. Значение этого свойства по умолчанию определяется системной настройкой [*Способ администрирования связанных объектов*] (код `QueryJoinRightLevel`). Если значение этой системной настройки не задано, то права доступа применяются, если в присоединяемой схеме запроса используются колонки, отличные от первичной и первичной для отображения.

Управлять фильтрами в запросе

Фильтр — набор условий, применяемых при отображении данных запроса. В терминах SQL фильтр представляет собой отдельный предикат (условие) оператора `WHERE`.

Структура фильтра

```
Filter = {[AggregationType] {<LeftExpression> | <LeftExpressionColumnPath>}
    <ComparisonType>
    {{<RightExpression> | {<RightExpressionColumnPath>,...}} | {<Macros>, [MacrosValue]}}
}
```

Для создания простого фильтра в `EntitySchemaQuery` используется метод `CreateFilter()`, который возвращает экземпляр типа `EntitySchemaQueryFilter`. Для этого метода в `EntitySchemaQuery` реализован ряд перегрузок. Это позволяет создавать фильтры с разными исходными параметрами. В `EntitySchemaQuery` реализованы методы создания фильтров специального вида.

Экземпляр `EntitySchemaQuery` имеет свойство `Filters`, которое представляет собой коллекцию фильтров данного запроса (экземпляр класса `EntitySchemaQueryFilterCollection`). Экземпляр класса `EntitySchemaQueryFilterCollection` представляет собой типизированную коллекцию элементов `IEntitySchemaQueryFilterItem`.

Алгоритм добавления фильтра в запрос:

- Создайте экземпляр фильтра для данного запроса (метод `CreateFilter()`, методы создания фильтров специального вида).
- Добавьте созданный экземпляр фильтра в коллекцию фильтров запроса (метод `Add()` коллекции).

© 2023 Terrasoft. Все права защищены.

По умолчанию фильтры, добавляемые в коллекцию `Filters`, объединяются между собой логической операцией `AND`. При реализации логической операции `OR` необходимо использовать свойство `LogicalOperation` коллекции `Filters`. Это свойство принимает значения перечисления `LogicalOperationStrict` и позволяет указать логическую операцию, которой необходимо объединять фильтры.

В запросах `EntitySchemaQuery` реализована возможность управления фильтрами, участвующими в построении результирующего набора данных. Каждый элемент коллекции `Filters` имеет свойство `IsEnabled`, которое определяет, участвует ли данный элемент в построении результирующего запроса (`true` — участвует, `false` — не участвует). Аналогичное свойство `IsEnabled` также определено для коллекции `Filters`. Установив это свойство в `false`, можно отключить фильтрацию для запроса, при этом коллекция фильтров запроса останется неизменной. Таким образом, изначально сформировав коллекцию фильтров запроса, в дальнейшем можно использовать различные комбинации, не внося изменений в саму коллекцию.

Управлять сущностью базы данных

`Terrasoft.Core.Entities.Entity` — класс, который реализует работу с сущностью базы данных.

Назначение класса `Terrasoft.Core.Entities.Entity` — доступ к объекту, который является записью в таблице базы данных. Класс также можно использовать для CRUD-операций над соответствующими записями.

Управлять сущностями базы данных

Сложный

На заметку. Примеры 1-5, приведенные в этой статье, реализованы в веб-сервисе. Пакет с реализацией веб-сервиса прикреплен в блоке "Ресурсы".

Пример 1

Пример. Получить значение колонки схемы [*City*] с именем `Name`.

Метод `GetEntityColumnData`

```
public string GetEntityColumnData()
{
    var result = "";
    /* Создание запроса к схеме City, добавление в запрос колонки Name. */
    var esqResult = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "City");
    var colName = esqResult.AddColumn("Name");
    /* Выполнение запроса к базе данных и получение объекта с заданным идентификатором. UId объе
```

© 2023 Terrasoft. Все права защищены.

```
    var entity = esqResult.GetEntity(UserConnection, new Guid("100B6B13-E8BB-DF11-B00F-001D60E93
    /* Получение значения колонки объекта. */
    result += entity.GetColumnValue(colName.Name).ToString();
    return result;
}
```

Пример 2

Пример. Получить коллекцию имен колонок схемы [*City*].

Метод `GetEntityColumns`

```
public IEnumerable<string> GetEntityColumns()
{
    /* Создание объекта строки данных схемы City (по идентификатору схемы, полученному из базы д
    var entity = new Entity(UserConnection, new Guid("5CA90B6A-93E7-4448-BEFE-AB5166EC2CFE"));
    /* Получение из базы данных объекта с заданным идентификатором. UId объекта можно получить и
    entity.FetchFromDB(new Guid("100B6B13-E8BB-DF11-B00F-001D60E938C6"),true);
    /* Получение коллекции имен колонок объекта. */
    var result = entity.GetColumnValueNames();
    return result;
}
```

Пример 3

Пример. Удалить из базы данных записи схемы [*Order*].

Метод `DeleteEntity`

```
public bool DeleteEntity()
{
    /* Создание запроса к схеме Order, добавление в запрос всех колонок схемы. */
    var esqResult = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "Order");
    esqResult.AddAllSchemaColumns();
    /* Выполнение запроса к базе данных и получение объекта с заданным идентификатором. UId объе
    var entity = esqResult.GetEntity(UserConnection, new Guid("e3bfa32f-3fe9-4bae-9332-16c162c51
    /* Удаление объекта из базы данных. */
    entity.Delete();
    /* Проверка, существует ли в базе данных объект с заданным идентификатором. */
    var result = entity.ExistInDB(new Guid("e3bfa32f-3fe9-4bae-9332-16c162c51e0d"));
```

© 2023 Terrasoft. Все права защищены.

```
    return result;
}
```

Пример 4

Пример. Изменить статус заказа.

Метод `UpdateEntity`

```
public bool UpdateEntity()
{
    /* Создание запроса к схеме Order, добавление в запрос всех колонок схемы. */
    var esqResult = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "Order");
    esqResult.AddAllSchemaColumns();
    /* Выполнение запроса к базе данных и получение объекта с заданным идентификатором. UId объе
    var entity = esqResult.GetEntity(UserConnection, new Guid("58be5223-715d-4b16-a5c4-e3d4ec041
    /* Создание объекта строки данных схемы OrderStatus. */
    var statusSchema = UserConnection.EntitySchemaManager.GetInstanceByName("OrderStatus");
    var newStatus = statusSchema.CreateEntity(UserConnection);
    /* Получение из базы данных объекта с заданным названием. */
    newStatus.FetchFromDB("Name", "4. Completed");
    /* Присваивает колонке StatusId новое значение. */
    entity.SetColumnValue("StatusId", newStatus.GetTypedColumnValue<Guid>("Id"));
    /* Сохранение измененного объекта в базе данных. */
    var result = entity.Save();
    return result;
}
```

Пример 5

Пример. Добавить город с указанным названием, привязав его к указанной стране.

Метод `UpdateEntity`

```
public bool InsertEntity(string city, string country)
{
    city = city ?? "unknown city";
    country = country ?? "unknown country";
    var citySchema = UserConnection.EntitySchemaManager.GetInstanceByName("City");
    var entity = citySchema.CreateEntity(UserConnection);
```

© 2023 Terrasoft. Все права защищены.

```
    entity.FetchFromDB("Name", city);
    /* Устанавливает для колонок объекта значения по умолчанию. */
    entity.SetDefColumnValues();
    var contryEntity = new Entity(UserConnection, new Guid("09FCE1F8-515C-4296-95CD-8CD93F79A6CF
    contryEntity.FetchFromDB("Name", country);
    /* Присваивает колонке Name переданное название города. */
    entity.SetColumnValue("Name", city);
    /* Присваивает колонке CountryId UId переданной страны. */
    entity.SetColumnValue("CountryId", contryEntity.GetTypedColumnValue<Guid>("Id"));
    var result = entity.Save();
    return result;
}
```

Пример 6

Пример. Создать контакт с именем "User01".

```
EntitySchema contactSchema = UserConnection.EntitySchemaManager.GetInstanceByName("Contact");
Entity contactEntity = contactSchema.CreateEntity(UserConnection);
contactEntity.SetDefColumnValues();
contactEntity.SetColumnValue("Name", "User01");
contactEntity.Save();
```

Пример 7

Пример. Изменить имя контакта на "User02".

```
EntitySchema entitySchema = UserConnection.EntitySchemaManager.GetInstanceByName("Contact");
Entity entity = entitySchema.CreateEntity(UserConnection);
if (!entity.FetchFromDB(some_id) {
    return false;
}
entity.SetColumnValue("Name", "User02");
return entity.Save();
```

Пример 8

© 2023 Terrasoft. Все права защищены.

Пример. Удалить контакт с именем "User02".

```
EntitySchema entitySchema = UserConnection.EntitySchemaManager.GetInstanceByName("Contact");
Entity entity = entitySchema.CreateEntity(UserConnection);
var fetchConditions = new Dictionary<string, object> {
    {"Name", "User02"}
};
if (entity.FetchFromDB(fetchConditions)) {
    entity.Delete();
}
```

Получить данные из базы данных с учетом прав пользователя

Сложный

На заметку. Примеры, приведенные в этой статье, реализованы в веб-сервисе. Пакет с реализацией веб-сервиса прикреплен в блоке "Ресурсы".

Пример 1

Пример. Создать экземпляр `EntitySchemaQuery`.

Метод `CreateESQ`

```
public string CreateESQ()
{
    var result = "";
    /* Получение экземпляра менеджера схем объектов. */
    EntitySchemaManager esqManager = SystemUserConnection.EntitySchemaManager;
    /* Получение экземпляра схемы, которая будет установлена в качестве корневой для создаваемог
    var rootEntitySchema = esqManager.GetInstanceByName("City") as EntitySchema;
    /* Создание экземпляра EntitySchemaQuery, у которого в качестве корневой схемы установлена с
    var esqResult = new EntitySchemaQuery(rootEntitySchema);
    /* Добавление колонок, которые будут выбираться в результирующем запросе. */
    esqResult.AddColumn("Id");
    esqResult.AddColumn("Name");
    /* Получение экземпляра Select, ассоциированного с созданным запросом EntitySchemaQuery. */
    Select selectEsq = esqResult.GetSelectQuery(SystemUserConnection);
```

© 2023 Terrasoft. Все права защищены.

```
    /* Получение текста результирующего запроса созданного экземпляра EntitySchemaQuery. */
    result = selectEsq.GetSqlText();
    return result;
}
```

Пример 2

Пример. Создать клон экземпляра `EntitySchemaQuery`.

Метод `CreateESQClone`

```
public string CreateESQClone()
{
    var result = "";
    EntitySchemaManager esqManager = SystemUserConnection.EntitySchemaManager;
    var esqSource = new EntitySchemaQuery(esqManager, "Contact");
    esqSource.AddColumn("Id");
    esqSource.AddColumn("Name");
    /* Создание экземпляра EntitySchemaQuery, являющегося клоном экземпляра esqSource. */
    var esqClone = new EntitySchemaQuery(esqSource);
    result = esqClone.GetSelectQuery(SystemUserConnection).GetSqlText();
    return result;
}
```

Пример 3

Пример. Получить результат выполнения запроса.

Метод `GetEntitiesExample`

```
public string GetEntitiesExample()
{
    var result = "";
    /* Создание запроса к схеме City, добавление в запрос колонки Name. */
    var esqResult = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "City");
    var colName = esqResult.AddColumn("Name");

    /* Выполнение запроса к базе данных и получение всей результирующей коллекции объектов. */
    var entities = esqResult.GetEntityCollection(UserConnection);
    for (int i=0; i < entities.Count; i++) {
```

© 2023 Terrasoft. Все права защищены.

```
        result += entities[i].GetColumnValue(colName.Name).ToString();
        result += "\n";
    }

    /* Выполнение запроса к базе данных и получение объекта с заданным идентификатором. */
    var entity = esqResult.GetEntity(UserConnection, new Guid("100B6B13-E8BB-DF11-B00F-001D60E93
    result += "\n";
    result += entity.GetColumnValue(colName.Name).ToString();
    return result;
}
```

Пример 4

Пример. Использовать кэш запроса `EntitySchemaQuery`.

Метод `UsingCacheExample`

```
public Collection<string> UsingCacheExample()
{
    /* Создание запроса к схеме City, добавление в запрос колонки Name. */
    var esqResult = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "City");
    esqResult.AddColumn("Name");

    /* Определение ключа, под которым в кэше будут храниться результаты выполнения запроса. В ка
    esqResult.CacheItemName = "EsqResultItem";

    /* Коллекция, в которую будут помещены результаты выполнения запроса. */
    var esqCityNames = new Collection<string>();

    /* Коллекция, в которую будут помещаться закэшированные результаты выполнения запроса. */
    var cachedEsqCityNames = new Collection<string>();

    /* Выполнение запроса к базе данных и получение результирующей коллекции объектов.
    После выполнения этой операции результаты запроса будут помещены в кэш. */
    var entities = esqResult.GetEntityCollection(UserConnection);

    /* Обработка результатов выполнения запроса и заполнение коллекции esqCityNames. */
    foreach (var entity in entities)
    {
        esqCityNames.Add(entity.GetTypedColumnValue<string>("Name"));
    }

    /* Получение ссылки на кэш запроса esqResult по ключу CacheItemName в виде таблицы данных в
    var esqCacheStore = esqResult.Cache[esqResult.CacheItemName] as DataTable;
```

© 2023 Terrasoft. Все права защищены.

```
    /* Заполнение коллекции cachedEsqCityNames значениями из кэша запроса. */
    if (esqCacheStore != null)
    {
        foreach (DataRow row in esqCacheStore.Rows)
        {
            cachedEsqCityNames.Add(row[0].ToString());
        }
    }
    return cachedEsqCityNames;
}
```

Пример 5

Пример. Использовать дополнительные настройки запроса `EntitySchemaQuery`.

Метод `UsingCacheExample`

```
public Collection<string> ESQOptionsExample()
{
    /* Создание экземпляра запроса с корневой схемой City. */
    var esqCities = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "City");
    esqCities.AddColumn("Name");

    /* Создание запроса с корневой схемой Country. */
    var esqCountries = new EntitySchemaQuery(UserConnection.EntitySchemaManager, "Country");
    esqCountries.AddColumn("Name");

    /* Создание экземпляра настроек для возврата запросом первых 5 строк. */
    var esqOptions = new EntitySchemaQueryOptions()
    {
        PageableDirection = PageableSelectDirection.First,
        PageableRowCount = 5,
        PageableConditionValues = new Dictionary<string, object>()
    };

    /* Получение коллекции городов, которая будет содержать первые 5 городов результирующего наб
    var cities = esqCities.GetEntityCollection(UserConnection, esqOptions);

    /* Получение коллекции стран, которая будет содержать первые 5 стран результирующего набора
    var countries = esqCountries.GetEntityCollection(UserConnection, esqOptions);
        var esqStringCollection = new Collection<string>();
    foreach (var entity in cities)
    {
```

© 2023 Terrasoft. Все права защищены.

```
            esqStringCollection.Add(entity.GetTypedColumnValue<string>("Name"));
        }
        foreach (var entity in countries)
        {
            esqStringCollection.Add(entity.GetTypedColumnValue<string>("Name"));
        }
        return esqStringCollection;
    }
```

Класс EntitySchemaQuery C#

Сложный

Пространство имен `Terrasoft.Core.Entities`.

Класс `Terrasoft.Core.Entities.EntitySchemaQuery` предназначен для построения запросов выборки записей из таблиц базы данных с учетом прав доступа текущего пользователя. В результате создания и конфигурирования экземпляра этого класса будет построен запрос в базу данных приложения в виде SQL-выражения `SELECT`. В запрос можно добавить требуемые колонки, фильтры и условия ограничений.

На заметку. Полный перечень методов и свойств класса `EntitySchemaQuery`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaQuery(EntitySchema rootSchema)`

Создает экземпляр класса, в котором в качестве корневой схемы устанавливается переданный экземпляр `EntitySchema`. В качестве менеджера схем устанавливается менеджер переданного экземпляра корневой схемы.

`EntitySchemaQuery(EntitySchemaManager entitySchemaManager, string sourceSchemaName)`

Создает экземпляр класса с указанным `EntitySchemaManager` и корневой схемы с именем, переданным в качестве аргумента.

`EntitySchemaQuery(EntitySchemaQuery source)`

Создает экземпляр класса, являющийся клоном экземпляра, переданного в качестве аргумента.

Свойства

© 2023 Terrasoft. Все права защищены.

`Cache` `Terrasoft.Core.Store.ICacheStore`

Кэш запроса.

`CacheItemName` `string`

Имя элемента кэша.

`CanReadUncommitedData` `bool`

Определяет, попадут ли в результаты запроса данные, для которых не завершена транзакция.

`Caption` `Terrasoft.Common.LocalizableString`

Заголовок.

`ChunkSize` `int`

Количество строк запроса в одном чанке.

`Columns` `Terrasoft.Core.Entities.EntitySchemaQueryColumnCollection`

Коллекция колонок текущего запроса к схеме объекта.

`DataValueTypeManager` `DataValueTypeManager`

Менеджер значений типов данных.

`EntitySchemaManager` `Terrasoft.Core.Entities.EntitySchemaManager`

Менеджер схем объектов.

`Filters` `Terrasoft.Core.Entities.EntitySchemaQueryFilterCollection`

Коллекция фильтров текущего запроса к схеме объекта.

`HideSecurityValue` `bool`

Определяет, будут ли скрыты значения зашифрованных колонок.

`IgnoreDisplayValues` `bool`

Определяет, будут ли в запросе использоваться отображаемые значения колонок.

© 2023 Terrasoft. Все права защищены.

`IsDistinct` `bool`

Определяет, убирать ли дубли в результирующем наборе данных.

`IsInherited` `bool`

Определяет, является ли запрос унаследованным.

`JoinRightState` `QueryJoinRightLevel`

Определяет условие наложения прав при использовании связанных таблиц, если схема администрируется по записям.

`Manager` `Terrasoft.Core.IManager`

Менеджер схем.

`ManagerItem` `Terrasoft.Core.IManagerItem`

Элемент менеджера.

`Name` `string`

Имя.

`ParentCollection` `Terrasoft.Core.Entities.EntitySchemaQueryCollection`

Коллекция запросов, которой принадлежит текущий запрос к схеме объекта.

`ParentEntitySchema` `Terrasoft.Core.Entities.EntitySchema`

Родительская схема запроса.

`PrimaryQueryColumn` `Terrasoft.Core.Entities.EntitySchemaQueryColumn`

Колонка, созданная по первичной колонке корневой схемы. Заполняется при первом обращении.

`QueryOptimize` `bool`

Разрешает использование оптимизации запроса.

`RootSchema` `Terrasoft.Core.Entities.EntitySchema`

© 2023 Terrasoft. Все права защищены.

Корневая схема.

`RowCount` `int`

Количество строк, возвращаемых запросом.

`SchemaAliasPrefix` `string`

Префикс, используемый для создания псевдонимов схем.

`SkipRowCount` `int`

Количество строк, которые необходимо пропустить при возврате результата запроса.

`UseAdminRights` `bool`

Определяет будут ли учитываться права при построении запроса получения данных.

`UseLocalization` `bool`

Определяет, будут ли использоваться локализованные данные.

`UseOffsetFetchPaging` `bool`

Определяет возможность постраничного возврата результата запроса.

`UseRecordDeactivation` `bool`

Определяет, будут ли данные исключены из фильтрации.

`AdminUnitRoleSources` `int`

Целочисленное свойство, которое соответствует критериям фильтрации записей по источнику вхождения пользователя в роли. Значение по умолчанию: `0`.

Чтобы сформировать `AdminUnitRoleSources`, необходимо с помощью побитового `ИЛИ` "`|`" перечислить следующие константы из серверного класса `AdminUnitRoleSources`:

- `ExplicitEntry`.
- `Delegated`.
- `FuncRoleFromOrgRole`.
- `UpHierarchy`.
- `AsManager`.

© 2023 Terrasoft. Все права защищены.

- `All`.
- `None`.

В результате, отработает следующее правило: возвращать запись только, если у пользователя есть хоть одна роль, которой доступна запись, и пользователь входит в эту роль в соответствии с источниками, указанными в условиях фильтрации.

Методы

```
void AddAllSchemaColumns(bool skipSystemColumns)
```

В коллекцию колонок текущего запроса к схеме объекта добавляет все колонки корневой схемы.

```
EntitySchemaQueryColumn AddColumn(string columnPath, AggregationTypeStrict aggregationType, out
void AddColumn(EntitySchemaQueryColumn queryColumn)
EntitySchemaQueryColumn AddColumn(string columnPath)
EntitySchemaQueryColumn AddColumn(EntitySchemaQueryFunction function)
EntitySchemaQueryColumn AddColumn(object parameterValue, DataValueType parameterDataValueType)
EntitySchemaQueryColumn AddColumn(EntitySchemaQuery subQuery)
```

Создает и добавляет колонку в текущий запрос к схеме объекта.

Параметры

`columnPath`	Путь к колонке схемы относительно корневой схемы.
`aggregationType`	Тип агрегирующей функции. В качестве параметра передаются значения перечисления типов агрегирующей функции `Terrasoft.Common.AggregationTypeStrict`.
`subQuery`	Ссылка на созданный подзапрос, помещенный в колонку.
`queryColumn`	Экземпляр `EntitySchemaQueryColumn`, добавляемый в коллекцию колонок текущего запроса.
`function`	Экземпляр функции `EntitySchemaQueryFunction`.
`parameterValue`	Значение параметра, добавляемого в запрос в качестве колонки.
`parameterDataValueType`	Тип значения параметра, добавляемого в запрос в качестве колонки.

```
void ClearCache()
```

Очищает кэш текущего запроса.

© 2023 Terrasoft. Все права защищены.

```
static void ClearDefCache(string cacheItemName)
```

Удаляет из кэша запроса элемент с заданным именем `cacheItemName`.

```
object Clone()
```

Создает клон текущего экземпляра `EntitySchemaQuery`.

```
EntitySchemaQueryExpression CreateAggregationEntitySchemaExpression(string leftExprColumnPath, A
```

Возвращает выражение агрегирующей функции с заданным типом агрегации из перечисления `Terrasoft.Common.AggregationTypeStrict` для колонки, расположенной по заданному пути `leftExprColumnPath`.

```
static EntitySchemaQueryExpression CreateParameterExpression(object parameterValue)
static EntitySchemaQueryExpression CreateParameterExpression(object parameterValue, DataValueTyp
static EntitySchemaQueryExpression CreateParameterExpression(object parameterValue, string displ
```

Создает выражение для параметра запроса.

Параметры

`parameterValue`	Значение параметра.
`valueType`	Тип значения параметра.
`displayValue`	Значение для отображения параметра.

```
static IEnumerable CreateParameterExpressions(DataValueType valueType, params object[] parameter
static IEnumerable CreateParameterExpressions(DataValueType valueType, IEnumerable<object> param
```

Создает коллекцию выражений для параметров запроса с определенным типом данных `DataValueType`.

```
static EntitySchemaQueryExpression CreateSchemaColumnExpression(EntitySchemaQuery parentQuery, E
static EntitySchemaQueryExpression CreateSchemaColumnExpression(EntitySchema rootSchema, string
EntitySchemaQueryExpression CreateSchemaColumnExpression(string columnPath, bool useCoalesceFunc
```

Возвращает выражение колонки схемы объекта.

Параметры

© 2023 Terrasoft. Все права защищены.

parentQuery	Запрос к схеме объекта, для которого создается выражение колонки.
rootSchema	Корневая схема.
columnPath	Путь к колонке относительно корневой схемы.
useCoalesceFunctionForMultiLookup	Признак, использовать ли для колонки типа справочник функцию `COALESCE`. Необязательный параметр, по умолчанию равен `true`.
useDisplayValue	Признак, использовать ли для колонки значение для отображения. Необязательный параметр, по умолчанию равен `false`.

```
Enumerable CreateSchemaColumnExpressions(params string[] columnPaths)
IEnumerable CreateSchemaColumnExpressions(IEnumerable columnPaths, bool useCoalesceFunctionForMu
```

Возвращает коллекцию выражений колонок запроса к схеме объекта по заданной коллекции путей к колонкам `columnPaths`.

```
IEnumerable CreateSchemaColumnExpressionsWithoutCoalesce(params string[] columnPaths)
```

Возвращает коллекцию выражений колонок запроса к схеме объекта по заданному массиву путей к колонкам. При этом, если колонка имеет тип множественный справочник, к ее значениям не применяется функция `COALESCE`.

```
static EntitySchemaQueryExpression CreateSchemaColumnQueryExpression(string columnPath, EntitySc
static EntitySchemaQueryExpression CreateSchemaColumnQueryExpression(string columnPath, EntitySc
```

Возвращает выражение запроса к схеме объекта по заданным пути к колонке, корневой схеме и экземпляру колонки схемы. При этом для колонки можно определить, какой тип ее значения использовать в выражении — хранимое значение или значение для отображения.

```
EntitySchemaQueryExpression CreateSubEntitySchemaExpression(string leftExprColumnPath)
```

Возвращает выражение подзапроса к схеме объекта для колонки, расположенной по заданному пути `leftExprColumnPath`.

```
EntitySchemaAggregationQueryFunction CreateAggregationFunction(AggregationTypeStrict aggregatior
```

Возвращает экземпляр агрегирующей функции `EntitySchemaAggregationQueryFunction` с заданным типом агрегации из перечисления `Terrasoft.Common.AggregationTypeStrict` для колонки по указанному пути относительно корневой схемы `columnPath`.

© 2023 Terrasoft. Все права защищены.

```
EntitySchemaCaseNotNullQueryFunction CreateCaseNotNullFunction(params EntitySchemaCaseNotNullQue
```

Возвращает экземпляр `CASE`-функции `EntitySchemaCaseNotNullQueryFunction` для заданного массива выражений условий `EntitySchemaCaseNotNullQueryFunctionWhenItem`.

```
EntitySchemaCaseNotNullQueryFunctionWhenItem CreateCaseNotNullQueryFunctionWhenItem(string whenC
```

Возвращает экземпляр выражения для sql-конструкции вида `WHEN <Выражение_1> IS NOT NULL THEN <Выражение_2>`.

Параметры

whenColumnPath	Путь к колонке, содержащей выражение предложения `WHEN`.
thenParameterValue	Путь к колонке, содержащей выражение предложения `THEN`.

```
EntitySchemaCastQueryFunction CreateCastFunction(string columnPath, DBDataValueType castType)
```

Возвращает экземпляр `CAST`-функции `EntitySchemaCastQueryFunction` для выражения колонки, расположенной по заданному пути относительно корневой схемы `columnPath`, и указанным целевым типом данных `DBDataValueType`.

```
EntitySchemaCoalesceQueryFunction CreateCoalesceFunction(params string[] columnPaths)
static EntitySchemaCoalesceQueryFunction CreateCoalesceFunction(EntitySchemaQuery parentQuery, E
static EntitySchemaCoalesceQueryFunction CreateCoalesceFunction(EntitySchema rootSchema, params
```

Возвращает экземпляр функции, возвращающей первое не равное `null` выражение из списка аргументов, для заданных колонок.

Параметры

columnPaths	Массив путей к колонкам относительно корневой схемы.
parentQuery	Запрос к схеме объекта, для которого создается экземпляр функции.
rootSchema	Корневая схема.

```
EntitySchemaConcatQueryFunction CreateConcatFunction(params EntitySchemaQueryExpression[] expres
```

Возвращает экземпляр функции для формирования строки, являющейся результатом объединения строковых значений аргументов функции для заданного массива выражений `EntitySchemaQueryExpression`.

© 2023 Terrasoft. Все права защищены.

`EntitySchemaDatePartQueryFunction CreateDatePartFunction(EntitySchemaDatePartQueryFunctionInterv`

Возвращает экземпляр `DATEPART`-функции `EntitySchemaDatePartQueryFunction`, определяющей интервал даты, заданный перечислением `EntitySchemaDatePartQueryFunctionInterval` (месяц, день, час, год, день недели...), для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaDatePartQueryFunction CreateDayFunction(string columnPath)`

Возвращает экземпляр функции `EntitySchemaDatePartQueryFunction`, определяющей интервал даты [*День*] для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaDatePartQueryFunction CreateHourFunction(string columnPath)`

Возвращает экземпляр функции `EntitySchemaDatePartQueryFunction`, возвращающей часть даты [*Час*] для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaDatePartQueryFunction CreateHourMinuteFunction(string columnPath)`

Возвращает экземпляр функции `EntitySchemaDatePartQueryFunction`, возвращающей часть даты [*Минута*] для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaDatePartQueryFunction CreateMonthFunction(string columnPath)`

Возвращает экземпляр функции `EntitySchemaDatePartQueryFunction`, возвращающей часть даты [*Месяц*] для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaDatePartQueryFunction CreateWeekdayFunction(string columnPath)`

Возвращает экземпляр функции `EntitySchemaDatePartQueryFunction`, возвращающей часть даты [*День недели*] для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaDatePartQueryFunction CreateWeekFunction(string columnPath)`

Возвращает экземпляр функции `EntitySchemaDatePartQueryFunction`, возвращающей часть даты [*Неделя*] для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaDatePartQueryFunction CreateYearFunction(string columnPath)`

Возвращает экземпляр функции `EntitySchemaDatePartQueryFunction`, возвращающей часть даты [*Год*] для значения колонки, расположенной по указанному пути относительно корневой схемы.

`EntitySchemaIsNullQueryFunction CreateIsNullFunction(string checkColumnPath, string replacementC`

© 2023 Terrasoft. Все права защищены.

Возвращает экземпляр функции `EntitySchemaIsNullQueryFunction` для колонок с проверяемым и замещающим значениями, которые расположены по заданным путям относительно корневой схемы.

```
EntitySchemaLengthQueryFunction CreateLengthFunction(string columnPath)
EntitySchemaLengthQueryFunction CreateLengthFunction(params EntitySchemaQueryExpression[] expres
```

Создание экземпляра функции `LEN` (функция для возврата длины выражения) для выражения колонки по заданному пути относительно корневой схемы или для заданного массива выражений.

```
EntitySchemaTrimQueryFunction CreateTrimFunction(string columnPath)
EntitySchemaTrimQueryFunction CreateTrimFunction(params EntitySchemaQueryExpression[] expressior
```

Возвращает экземпляр функции `TRIM` (функция для удаления начальных и конечных пробелов из выражения) для выражения колонки по заданному пути относительно корневой схемы или для заданного массива выражений.

```
EntitySchemaUpperQueryFunction CreateUpperFunction(string columnPath)
```

Возвращает экземпляр функции `EntitySchemaUpperQueryFunction`, для преобразования символов выражения аргумента к верхнему регистру, для выражения колонки по заданному пути относительно корневой схемы.

```
EntitySchemaCurrenddateQueryFunction CreateCurrenddateFunction()
```

Возвращает экземпляр функции `EntitySchemaCurrenddateQueryFunction`, определяющей текущую дату.

```
EntitySchemaCurrenddateTimeQueryFunction CreateCurrenddateTimeFunction()
```

Возвращает экземпляр функции `EntitySchemaCurrenddateTimeQueryFunction`, определяющей текущие дату и время.

```
EntitySchemaCurrentTimeQueryFunction CreateCurrentTimeFunction()
```

Возвращает экземпляр функции `EntitySchemaCurrentTimeQueryFunction`, определяющей текущее время.

```
EntitySchemaCurrentUserAccountQueryFunction CreateCurrentUserAccountFunction()
```

Возвращает экземпляр функции `EntitySchemaCurrentUserAccountQueryFunction`, определяющей идентификатор контрагента текущего пользователя.

```
EntitySchemaCurrentUserContactQueryFunction CreateCurrentUserContactFunction()
```

Возвращает экземпляр функции `EntitySchemaCurrentUserContactQueryFunction`, определяющей

© 2023 Terrasoft. Все права защищены.

идентификатор контакта текущего пользователя.

```
EntitySchemaCurrentUserQueryFunction CreateCurrentUserFunction()
```

Возвращает экземпляр функции `EntitySchemaCurrentUserQueryFunction`, определяющей текущего пользователя.

```
EntitySchemaQueryFilter CreateExistsFilter(string rightExpressionColumnPath)
```

Создает фильтр сравнения типа [*Существует по заданному условию*] и устанавливает в качестве проверяемого значения выражение колонки, расположенной по пути `rightExpressionColumnPath`.

```
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExpres
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExpres
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExpres
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, EntitySchemaQuery
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, EntitySchemaQuery
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, EntitySchemaQuery
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExpres
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExpres
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExpres
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExpres
EntitySchemaQueryFilter CreateFilter(FilterComparisonType comparisonType, string leftExprColumnP
EntitySchemaQueryFilter CreateFilter(FilterComparisonType comparisonType, string leftExprColumnP
EntitySchemaQueryFilter CreateFilter(FilterComparisonType comparisonType, string leftExprColumnP
EntitySchemaQueryFilter CreateFilter(FilterComparisonType comparisonType, string leftExprColumnP
EntitySchemaQueryFilter CreateFilter(FilterComparisonType comparisonType, string leftExprColumnP
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExprCo
IEntitySchemaQueryFilterItem CreateFilter(FilterComparisonType comparisonType, string leftExprCo
```

Создает фильтр запроса для выборки записей по определенным условиям.

Параметры

© 2023 Terrasoft. Все права защищены.

comparisonType	Тип сравнения из перечисления `Terrasoft.Core.Entities.FilterComparisonType`.
leftExpressionColumnPath	Путь к колонке, содержащей выражение левой части фильтра.
leftExpression	Выражение в левой части фильтра.
leftExprAggregationType	Тип агрегирующей функции.
leftExprSubQuery	Параметр, в котором возвращается подзапрос для выражения в левой части фильтра (если он не равен `null`) либо подзапрос для первого выражения в правой части фильтра (если выражение левой части фильтра равно `null`).
rightExpressionColumnPaths	Массив путей к колонкам, содержащим выражения правой части фильтра.
rightExpression	Выражение в правой части фильтра.
rightExpressionValue	Экземпляр функции выражения в правой части фильтра (тип параметра `EntitySchemaQueryFunction`) или выражение подзапроса в правой части фильтра (тип параметра `EntitySchemaQuery`).
rightValue	Значение, которое обрабатывается макросом в правой части фильтра.
rightExprParameterValue	Значение параметра, к которому применяется агрегирующая функция в правой части фильтра.
macrosType	Тип макроса из перечисления `Terrasoft.Core.Entities.EntitySchemaQueryMacrosType`.
daysCount	Значение, к которому применяется макрос в правой части фильтра. Необязательный параметр, по умолчанию равен 0.

```
IEntitySchemaQueryFilterItem CreateFilterWithParameters(FilterComparisonType comparisonType, boo
IEntitySchemaQueryFilterItem CreateFilterWithParameters(FilterComparisonType comparisonType, str
IEntitySchemaQueryFilterItem CreateFilterWithParameters(FilterComparisonType comparisonType, str
static IEntitySchemaQueryFilterItem CreateFilterWithParameters(EntitySchemaQuery parentQuery, En
static IEntitySchemaQueryFilterItem CreateFilterWithParameters(EntitySchema rootSchema, FilterCo
```

Создает параметризированный фильтр для выборки записей по определенным условиям.

© 2023 Terrasoft. Все права защищены.

Параметры

`parentQuery`	Родительский запрос, для которого создается фильтр.
`rootSchema`	Корневая схема.
`comparisonType`	Тип сравнения из перечисления `Terrasoft.Core.Entities.FilterComparisonType`.
`useDisplayValue`	Признак типа значения колонки, которое используется в фильтре: `true` - значение для отображения; `false` - хранимое значение.
`leftExpressionColumnPath`	Путь к колонке, содержащей выражение левой части фильтра.
`rightExpressionParameterValues`	Коллекция выражений параметров в правой части фильтра.

`IEntitySchemaQueryFilterItem CreateIsNotNullFilter(string leftExpressionColumnPath)`

Создает фильтр сравнения типа [*Не является null в базе данных*], устанавливая в качестве проверяемого значения выражение колонки, расположенной по указанному в параметре `leftExpressionColumnPath` пути.

`IEntitySchemaQueryFilterItem CreateIsNullFilter(string leftExpressionColumnPath)`

Создает фильтр сравнения типа [*Является null в базе данных*], устанавливая в качестве условия проверки выражение колонки, расположенной по указанному в параметре `leftExpressionColumnPath` пути.

`EntitySchemaQueryFilter CreateNotExistsFilter(string rightExpressionColumnPath)`

Создает фильтр сравнения типа [*Не существует по заданному условию*] и устанавливает в качестве проверяемого значения выражение колонки, расположенной по пути `rightExpressionColumnPath`.

`DataTable GetDataTable(UserConnection userConnection)`

Возвращает результат выполнения текущего запроса к схеме объекта в виде таблицы данных в памяти, используя пользовательское подключение `UserConnection`.

`static int GetDayOfWeekNumber(UserConnection userConnection, DayOfWeek dayOfWeek)`

Возвращает порядковый номер дня недели для объекта `System.DayOfWeek` с учетом региональных

© 2023 Terrasoft. Все права защищены.

настроек.

```
Entity GetEntity(UserConnection userConnection, object primaryColumnValue)
```

Возвращает экземпляр Entity по первичному ключу `primaryColumnValue`, используя пользовательское подключение `UserConnection`.

```
EntityCollection GetEntityCollection(UserConnection userConnection, EntitySchemaQueryOptions opt
EntityCollection GetEntityCollection(UserConnection userConnection)
```

Возвращает коллекцию экземпляров `Entity`, представляющих результаты выполнения текущего запроса, используя пользовательское подключение `UserConnection` и заданные дополнительные настройки запроса `EntitySchemaQueryOptions`.

```
EntitySchema GetSchema()
```

Возвращает экземпляр схемы объекта `EntitySchema` текущего экземпляра `EntitySchemaQuery`.

```
Select GetSelectQuery(UserConnection userConnection)
Select GetSelectQuery(UserConnection userConnection, EntitySchemaQueryOptions options)
```

Возвращает экземпляр запроса на выборку данных, используя пользовательское подключение `UserConnection` и заданные дополнительные настройки запроса `EntitySchemaQueryOptions`.

```
EntitySchemaQueryColumnCollection GetSummaryColumns()
EntitySchemaQueryColumnCollection GetSummaryColumns(IEnumerable<string> columnNames)
```

Возвращает коллекцию выражений колонок запроса, для которых вычисляются итоговые значения.

```
Entity GetSummaryEntity(UserConnection userConnection, EntitySchemaQueryColumnCollection summary
Entity GetSummaryEntity(UserConnection userConnection)
Entity GetSummaryEntity(UserConnection userConnection, IEnumerable<string> columnNames)
Entity GetSummaryEntity(UserConnection userConnection, params string[] columnNames)
```

Возвращает экземпляр `Entity` для результата, возвращаемого запросом на выборку итоговых значений.

Параметры

© 2023 Terrasoft. Все права защищены.

`userConnection`	Пользовательское подключение.
`summaryColumns`	Коллекция колонок запроса, для которых выбираются итоговые значения.
`columnNames`	Коллекция имен колонок.

```
Select GetSummarySelectQuery(UserConnection userConnection, EntitySchemaQueryColumnCollection su
Select GetSummarySelectQuery(UserConnection userConnection)
Select GetSummarySelectQuery(UserConnection userConnection, IEnumerable<string> columnNames)
Select GetSummarySelectQuery(UserConnection userConnection, params string[] columnNames)
```

Строит запрос на выборку итоговых значений для заданной коллекции колонок текущего экземпляра `EntitySchemaQuery`.

Параметры

`userConnection`	Пользовательское подключение.
`summaryColumns`	Коллекция колонок запроса, для которых выбираются итоговые значения.
`columnNames`	Коллекция имен колонок.

```
T GetTypedColumnValue(Entity entity, string columnName)
```

Возвращает типизированное значение колонки с именем `columnName` из переданного экземпляра `Entity`.

```
void LoadDataTableData(UserConnection userConnection, DataTable dataTable)
void LoadDataTableData(UserConnection userConnection, DataTable dataTable, EntitySchemaQueryOpti
```

Загружает результат выполнения текущего запроса к схеме объекта в объект `System.Data.DataTable`, используя пользовательское подключение `UserConnection` и заданные дополнительные настройки запроса `EntitySchemaQueryOptions`.

```
void RemoveColumn(string columnName)
```

Удаляет колонку с именем `columnName` из коллекции колонок текущего запроса.

```
void ResetSchema()
```

Очищает схему текущего экземпляра `EntitySchemaQuery`.

© 2023 Terrasoft. Все права защищены.

`void ResetSelectQuery()`

Очищает запрос на выборку для текущего запроса к схеме объекта.

`void SetLocalizationCultureId(System.Guid cultureId)`

Устанавливает идентификатор локальной культуры.

Класс Entity C#

Сложный

Пространство имен `Terrasoft.Core.Entities`.

Класс `Terrasoft.Core.Entities.Entity` предназначен для доступа к объекту, который представляет собой запись в таблице базы данных.

На заметку. Полный перечень методов и свойств класса `Entity`, его родительских классов, а также реализуемых им интерфейсов, можно найти в Библиотеке .NET классов.

Конструкторы

`Entity(UserConnection userConnection)`

Создает новый экземпляр класса `Entity` для заданного пользовательского подключения `UserConnection`.

`Entity(UserConnection userConnection, Guid schemaUId)`

Создает новый экземпляр класса `Entity` для заданного пользовательского подключения `UserConnection` и схемы заданной идентификатором `schemaUId`.

`Entity(Entity source)`

Создает экземпляр класса, являющийся клоном экземпляра, переданного в качестве аргумента.

Свойства

`ChangeType` `EntityChangeType`

Тип изменения состояния объекта (добавлен, изменен, удален, без изменений).

© 2023 Terrasoft. Все права защищены.

`EntitySchemaManager` `EntitySchemaManager`

Экземпляр менеджера схемы объекта.

`EntitySchemaManagerName` `string`

Имя менеджера схемы объекта.

`HasColumnValues` `bool`

Определяет, имеет ли объект хотя бы одну колонку.

`HierarchyColumnValue` `Guid`

Значение колонки связи с родительской записью для иерархических объектов.

`InstanceUId` `Guid`

Идентификатор экземпляра объекта.

`IsDeletedFromDB` `bool`

Определяет, удален ли объект из базы данных.

`IsInColumnValueChanged` `bool`

Определяет, выполняется ли обработка события `ColumnValueChanged`.

`IsInColumnValueChanging` `bool`

Определяет, выполняется ли обработка события `ColumnValueChanging`.

`IsInDefColumnValuesSet` `bool`

Определяет, выполняется ли обработка события `DefColumnValuesSet`.

`IsInDeleted` `bool`

Определяет, выполняется ли обработка события `Deleted`.

`IsInDeleting` `bool`

Определяет, выполняется ли обработка события `Deleting`.

© 2023 Terrasoft. Все права защищены.

`IsInInserted` `bool`

Определяет, выполняется ли обработка события `Inserted`.

`IsInInserting` `bool`

Определяет, выполняется ли обработка события `Inserting`.

`IsInLoaded` `bool`

Определяет, выполняется ли обработка события `Loaded`.

`IsInLoading` `bool`

Определяет, выполняется ли обработка события `Loading`.

`IsInSaved` `bool`

Определяет, выполняется ли обработка события `Saved`.

`IsInSaveError` `bool`

Определяет, выполняется ли обработка события `SaveError`.

`IsInSaving` `bool`

Определяет, выполняется ли обработка события `Saving`.

`IsInUpdated` `bool`

Определяет, выполняется ли обработка события `Updated`.

`IsInUpdating` `bool`

Определяет, выполняется ли обработка события `Updating`.

`IsInValidating` `bool`

Определяет, выполняется ли обработка события `Validating`.

`IsSchemaInitialized` `bool`

© 2023 Terrasoft. Все права защищены.

Определяет, является ли схема объекта проинициализированной.

`LicOperationPrefix` `string`

Префикс лицензируемой операции.

`LoadState` `EntityLoadState`

Состояние загрузки объекта.

`PrimaryColumnValue` `Guid`

Идентификатор первичной колонки.

`PrimaryDisplayColumnValue` `string`

Значение для отображения первичной колонки.

`Process` `Process`

Встроенный процесс объекта.

`Schema` `EntitySchema`

Экземпляр схемы объекта.

`SchemaName` `string`

Имя схемы объекта.

`StoringState` `StoringObjectState`

Состояние объекта (изменен, добавлен, удален, без изменений).

`UseAdminRights` `bool`

Определяет, будут ли учитываться права при вставке, обновлении, удалении и получении данных.

`UseDefRights` `bool`

Определяет, использовать ли права по умолчанию на объект.

© 2023 Terrasoft. Все права защищены.

`UseLazyLoad` `bool`

Определяет, использовать ли ленивую первоначальную загрузку данных объекта.

`UserConnection` `UserConnection`

Пользовательское подключение.

`ValidationMessages` `EntityValidationMessageCollection`

Коллекция сообщений, выводимых при проверке объекта.

`ValueListSchemaManager` `ValueListSchemaManager`

Экземпляр менеджера перечислений объекта.

`ValueListSchemaManagerName` `string`

Имя менеджера перечислений объекта.

Методы

```
void AddDefRights()
void AddDefRights(Guid primaryColumnValue)
void AddDefRights(IEnumerable<Guid> primaryColumnValues)
```

Для данного объекта устанавливает права по умолчанию.

Параметры

`primaryColumnValue`	Идентификатор значения права доступа.
`primaryColumnValues`	Массив идентификаторов значений прав доступа.

```
virtual object Clone()
```

Создает клон текущего экземпляра `Entity`.

```
Insert CreateInsert(bool skipLookupColumnValues)
```

Создает запрос на добавление данных в базу.

Параметры

© 2023 Terrasoft. Все права защищены.

`skipLookupColumnValues`	Признак добавления данных с учетом справочных колонок. По умолчанию установлено значение `false`.

```
Update CreateUpdate(bool skipLookupColumnValues)
```

Создает запрос на обновление данных в базе.

Параметры

`skipLookupColumnValues`	Параметр, определяющий необходимость добавления в базу данных колонок типа справочник. Если параметр равен `true`, то колонки типа справочник не будут добавлены в базу. Значение по умолчанию — `false`.

```
virtual bool Delete()
virtual bool Delete(object keyValue)
```

Удаляет из базы данных запись объекта.

Параметры

`keyValue`	Значение ключевого поля.

```
bool DeleteWithCancelProcess()
```

Удаляет из базы данных запись объекта и отменяет запущенный процесс.

```
static Entity DeserializeFromJson(UserConnection userConnection, string jsonValue)
```

Создает объект типа `Entity`, используя пользовательское подключение `userConnection`, и заполняет значения его полей из указанной строки формата JSON `jsonValue`.

Параметры

`jsonValue`	Строка формата JSON.
`userConnection`	Пользовательское подключение.

```
bool ExistInDB(EntitySchemaColumn conditionColumn, object conditionValue)
bool ExistInDB(string conditionColumnName, object conditionValue)
bool ExistInDB(object keyValue)
```

© 2023 Terrasoft. Все права защищены.

```
bool ExistInDB(Dictionary<string,object> conditions)
```

Определяет, существует ли в базе данных запись, отвечающая заданному условию запроса `conditionValue` к заданной колонке схемы объекта `conditionColumn` либо с заданным первичным ключом `keyValue`.

Параметры

`conditionColumn`	Колонка, для которой задается условие выборки.
`conditionColumnName`	Название колонки, для которой задается условие выборки.
`conditionValue`	Значение колонки условия для выбираемых данных.
`conditions`	Набор условий фильтрации выборки записей объекта.
`keyValue`	Значение ключевого поля.

```
bool FetchFromDB(EntitySchemaColumn conditionColumn, object conditionValue, bool useDisplayValue
bool FetchFromDB(string conditionColumnName, object conditionValue, bool useDisplayValues)
bool FetchFromDB(object keyValue, bool useDisplayValues)
bool FetchFromDB(Dictionary<string,object> conditions, bool useDisplayValues)
bool FetchFromDB(EntitySchemaColumn conditionColumn, object conditionValue, IEnumerable<EntitySc
bool FetchFromDB(string conditionColumnName, object conditionValue, IEnumerable<string>columnNam
```

По заданному условию загружает объект из базы данных.

Параметры

`conditionColumn`	Колонка, для которой задается условие выборки.
`conditionColumnName`	Название колонки, для которой задается условие выборки.
`conditionValue`	Значение колонки условия для выбираемых данных.
`columnsToFetch`	Список колонок, которые будут выбраны.
`columnNamesToFetch`	Список названий колонок, которые будут выбраны.
`conditions`	Набор условий фильтрации выборки записей объекта.
`keyValue`	Значение ключевого поля.
`useDisplayValues`	Признак получения в запросе первичных отображаемых значений. Если параметр равен `true`, в запросе будут возвращены первичные отображаемые значения.

© 2023 Terrasoft. Все права защищены.

```
bool FetchPrimaryColumnFromDB(object keyValue)
```

По заданному условию `keyValue` загружает из базы данных объект с первичной колонкой.

Параметры

`keyValue`	Значение ключевого поля.

```
bool FetchPrimaryInfoFromDB(EntitySchemaColumn conditionColumn, object conditionValue)
bool FetchPrimaryInfoFromDB(string conditionColumnName, object conditionValue)
```

По заданному условию загружает из базы данных объект с первичными колонками, включая колонку, первичную для отображения.

Параметры

`conditionColumn`	Колонка, для которой задается условие выборки.
`conditionColumnName`	Название колонки, для которой задается условие выборки.
`conditionValue`	Значение колонки условия для выбираемых данных.

```
byte[] GetBytesValue(string valueName)
```

Возвращает значение заданной колонки объекта в виде массива байт.

Параметры

`valueName`	Имя колонки объекта.

```
IEnumerable<EntityColumnValue> GetChangedColumnValues()
```

Возвращает коллекцию имен колонок объекта, которые были изменены.

```
string GetColumnDisplayValue(EntitySchemaColumn column)
```

Возвращает значение для отображения свойства объекта, соответствующее заданной колонке схемы объекта.

Параметры

`column`	Определенная колонка схемы объекта.

© 2023 Terrasoft. Все права защищены.

```
object GetColumnOldValue(string valueName)
object GetColumnOldValue(EntitySchemaColumn column)
```

Возвращает предыдущее значение заданного свойства объекта.

Параметры

column	Определенная колонка схемы объекта.
valueName	Имя колонки объекта.

```
virtual object GetColumnValue(string valueName)
virtual object GetColumnValue(EntitySchemaColumn column)
```

Возвращает значение колонки объекта с заданным именем, соответствующее переданной колонке схемы объекта.

Параметры

column	Определенная колонка схемы объекта.
valueName	Имя колонки объекта.

```
IEnumerable<string> GetColumnValueNames()
```

Возвращает коллекцию имен колонок объекта.

```
virtual bool GetIsColumnValueLoaded(string valueName)
bool GetIsColumnValueLoaded(EntitySchemaColumn column)
```

Возвращает признак, определяющий, загружено ли заданное свойство объекта.

Параметры

column	Определенная колонка схемы объекта.
valueName	Имя колонки объекта.

```
virtual MemoryStream GetStreamValue(string valueName)
```

Возвращает преобразованное в экземпляр типа `System.IO.MemoryStream` значение переданной колонки схемы объекта.

© 2023 Terrasoft. Все права защищены.

Параметры

`valueName`	Имя колонки объекта.

```
TResult GetTypedColumnValue<TResult>(EntitySchemaColumn column)
```

Возвращает типизированное значение свойства объекта, соответствующее заданной колонке схемы объекта.

Параметры

`column`	Определенная колонка схемы объекта.

```
TResult GetTypedOldColumnValue<TResult>(string valueName)
TResult GetTypedOldColumnValue<TResult>(EntitySchemaColumn column)
```

Возвращает типизированное предыдущее значение свойства объекта, соответствующее заданной колонке схемы объекта.

Параметры

`column`	Определенная колонка схемы объекта.
`valueName`	Имя колонки объекта.

```
virtual bool InsertToDB(bool skipLookupColumnValues, bool validateRequired)
```

Добавляет запись текущего объекта в базу данных.

Параметры

`skipLookupColumnValues`	Параметр, определяющий необходимость добавления в базу данных колонок типа справочник. Если параметр равен `true`, то колонки типа справочник не будут добавлены в базу. Значение по умолчанию — `false`.
`validateRequired`	Параметр, определяющий необходимость проверки заполнения обязательных значений. Значение по умолчанию — `true`.

```
bool IsColumnValueLoaded(string valueName)
bool IsColumnValueLoaded(EntitySchemaColumn column)
```

Определяет, загружено ли значение свойства объекта с заданным именем.

© 2023 Terrasoft. Все права защищены.

Параметры

column	Определенная колонка схемы объекта.
valueName	Имя колонки объекта.

```
virtual bool Load(DataRow dataRow)
virtual bool Load(DataRow dataRow, Dictionary<string,string> columnMap)
virtual bool Load(IDataReader dataReader)
virtual bool Load(IDataReader dataReader, IDictionary<string,string> columnMap)
virtual bool Load(object dataSource)
virtual bool Load(object dataSource, IDictionary<string,string> columnMap)
```

Заполняет объект переданными данными.

Параметры

columnMap	Свойства объекта, заполняемые данными.
dataRow	Экземпляр `System.Data.DataRow`, из которого загружаются данные в объект.
dataReader	Экземпляр `System.Data.IDataReader`, из которого загружаются данные.
dataSource	Экземпляр `System.Object`, из которого загружаются данные.

```
void LoadColumnValue(string columnValueName, IDataReader dataReader, int fieldIndex, int binaryF
void LoadColumnValue(string columnValueName, IDataReader dataReader, int fieldIndex)
void LoadColumnValue(string columnValueName, object value)
void LoadColumnValue(EntitySchemaColumn column, object value)
```

Для свойства с заданным именем загружает его значение из переданного экземпляра.

Параметры

© 2023 Terrasoft. Все права защищены.

`binaryPackageSize`	Размер загружаемого значения.
`column`	Колонка схемы объекта.
`columnValueName`	Имя свойства объекта.
`dataReader`	Экземпляр `System.Data.IDataReader`, из которого загружается значение свойства.
`fieldIndex`	Индекс загружаемого из `System.Data.IDataReader` поля.
`value`	Загружаемое значение свойства.

```
static Entity Read(UserConnection userConnection, DataReader dataReader)
```

Возвращает значение текущего свойства типа `Entity` из потока ввода.

Параметры

`dataReader`	Экземпляр `System.Data.IDataReader`, из которого загружается значение свойства.
`userConnection`	Пользовательское подключение.

```
void ReadData(DataReader reader)
void ReadData(DataReader reader, EntitySchema schema)
```

Считывает данные из схемы объекта в заданный объект типа `System.Data.IDataReader`.

Параметры

`reader`	Экземпляр `System.Data.IDataReader`, в который загружаются данные схемы объекта.
`schema`	Схема объекта.

```
void ResetColumnValues()
```

Для всех свойств объекта отменяет изменения.

```
void ResetOldColumnValues()
```

Для всех свойств объекта отменяет изменения, устанавливая предыдущее значение.

© 2023 Terrasoft. Все права защищены.

```
bool Save(bool validateRequired)
```

Сохраняет объект в базе данных.

Параметры

`validateRequired`	Определяет необходимость проверки заполнения обязательных значений. Значение по умолчанию — `true`.

```
static string SerializeToJson(Entity entity)
```

Преобразует объект `entity` в строку формата `JSON`.

Параметры

`entity`	Экземпляр `Entity`.

```
virtual void SetBytesValue(string valueName, byte[] streamBytes)
```

Устанавливает для заданного свойства объекта переданное значение типа `System.Byte`.

Параметры

`streamBytes`	Значение типа `System.Byte`, которое устанавливается в заданную колонку объекта.
`valueName`	Имя колонки объекта.

```
bool SetColumnBothValues(EntitySchemaColumn column, object value, string displayValue)
bool SetColumnBothValues(string columnValueName, object value, string displayColumnValueName, st
```

Устанавливает свойству объекта, соответствующему заданной колонке схемы, переданные значение `value` и значение для отображения `displayValue`.

Параметры

© 2023 Terrasoft. Все права защищены.

`column`	Колонка схемы объекта.
`displayValue`	Загружаемое значение для отображения.
`displayColumnValueName`	Имя колонки, содержащей значение для отображения.
`value`	Загружаемое значение колонки.

```
bool SetColumnValue(string valueName, object value)
bool SetColumnValue(EntitySchemaColumn column, object value)
```

Устанавливает заданной колонке схемы переданное значение `value`.

Параметры

`column`	Колонка схемы объекта.
`value`	Загружаемое значение колонки.
`valueName`	Имя колонки объекта.

```
void SeddefColumnValue(string columnValueName, object defValue)
void SeddefColumnValue(string columnValueName)
```

Устанавливает значение по умолчанию свойству с заданным именем.

Параметры

`columnValueName`	Имя колонки объекта.
`defValue`	Значение по умолчанию.

```
void SeddefColumnValues()
```

Для всех свойств объекта устанавливает значения по умолчанию.

```
bool SetStreamValue(string valueName, Stream value)
```

Устанавливает для заданного свойства объекта переданное значение типа `System.IO.Stream`.

Параметры

© 2023 Terrasoft. Все права защищены.

`value`	Загружаемое значение колонки.
`valueName`	Имя колонки объекта.

```
virtual bool UpdateInDB(bool validateRequired)
```

Обновляет запись объекта в базе данных.

Параметры

`validateRequired`	Определяет необходимость проверки заполнения обязательных значений. Значение по умолчанию — `true`.

```
bool Validate()
```

Проверяет заполнение обязательных полей.

```
static void Write(DataWriter dataWriter, Entity entity, string propertyName)
static void Write(DataWriter dataWriter, Entity entity, string propertyName, bool couldConvertFo
```

Осуществляет запись значения типа `Entity` в поток вывода с заданными именем.

Параметры

`couldConvertForXml`	Разрешить преобразование для xml-сериализации.
`dataWriter`	Экземпляр класса `Terrasoft.Common.DataWriter`, предоставляющий методы последовательной записи значений в поток вывода.
`entity`	Значение для записи типа `Entity`.
`propertyName`	Имя объекта.

```
void Write(DataWriter dataWriter, string propertyName)
```

Осуществляет запись данных в поток вывода с заданным именем.

Параметры

© 2023 Terrasoft. Все права защищены.

`dataWriter`	Экземпляр класса `Terrasoft.Common.DataWriter`, предоставляющий методы последовательной записи значений в поток вывода.
`propertyName`	Имя свойства.

```
void WriteData(DataWriter writer)
void WriteData(DataWriter writer, EntitySchema schema)
```

Осуществляет запись в поток вывода для указанной либо текущей схемы объекта.

Параметры

`schema`	Схема объекта.
`writer`	Экземпляр класса `Terrasoft.Common.DataWriter`, предоставляющий методы последовательной записи значений в поток вывода.

События

```
event EventHandler<EntityColumnAfterEventArgs> ColumnValueChanged
```

Обработчик события, возникающего после изменения значения колонки объекта.

Обработчик события получает аргумент типа `EntityColumnAfterEventArgs`.

Свойства `EntityColumnAfterEventArgs` предоставляющие сведения, относящиеся к событию:

- `ColumnValueName`;
- `DisplayColumnValueName`.

```
event EventHandler<EntityColumnBeforeEventArgs> ColumnValueChanging
```

Обработчик события, возникающего перед изменением значения колонки объекта.

Обработчик события получает аргумент типа `EntityColumnBeforeEventArgs`.

Свойства `EntityColumnBeforeEventArgs` предоставляющие сведения, относящиеся к событию:

- `ColumnStreamValue`.
- `ColumnValue`.
- `ColumnValueName`.
- `DisplayColumnValue`.
- `DisplayColumnValueName`.

© 2023 Terrasoft. Все права защищены.

`event EventHandler<EventArgs> DefColumnValuesSet`

Обработчик события, возникающего после установки значений по умолчанию полей объекта.

`event EventHandler<EntityAfterEventArgs> Deleted`

Обработчик события, возникающего после удаления объекта.

Обработчик события получает аргумент типа `EntityAfterEventArgs`.

Свойства `EntityAfterEventArgs` предоставляющие сведения, относящиеся к событию:

- `ModifiedColumnValues`.
- `PrimaryColumnValue`.

`event EventHandler<EntityBeforeEventArgs> Deleting`

Обработчик события, возникающего перед удалением объекта.

Обработчик события получает аргумент типа `EntityBeforeEventArgs`.

Свойства `EntityBeforeEventArgs` предоставляющие сведения, относящиеся к событию:

- `AdditionalCondition`.
- `IsCanceled`.
- `KeyValue`.

`event EventHandler<EntityAfterEventArgs> Inserted`

Обработчик события, возникающего после вставки объекта.

Обработчик события получает аргумент типа `EntityAfterEventArgs`.

Свойства `EntityAfterEventArgs` предоставляющие сведения, относящиеся к событию:

- `ModifiedColumnValues`.
- `PrimaryColumnValue`.

`event EventHandler<EntityBeforeEventArgs> Inserting`

Обработчик события, возникающего перед вставкой объекта.

Обработчик события получает аргумент типа `EntityBeforeEventArgs`.

Свойства `EntityBeforeEventArgs` предоставляющие сведения, относящиеся к событию:

- `AdditionalCondition`.
- `IsCanceled`.
- `KeyValue`.

© 2023 Terrasoft. Все права защищены.

`event EventHandler<EntityAfterLoadEventArgs> Loaded`

Обработчик события, возникающего после загрузки объекта.

Обработчик события получает аргумент типа `EntityAfterLoadEventArgs`.

Свойства `EntityAfterLoadEventArgs` предоставляющие сведения, относящиеся к событию:

- `ColumnMap`.
- `DataSource`.

`event EventHandler<EntityBeforeLoadEventArgs> Loading`

Обработчик события, возникающего перед загрузкой объекта.

Обработчик события получает аргумент типа `EntityBeforeLoadEventArgs`.

Свойства `EntityBeforeLoadEventArgs` предоставляющие сведения, относящиеся к событию:

- `ColumnMap`.
- `DataSource`.
- `IsCanceled`.

`event EventHandler<EntityAfterEventArgs> Saved`

Обработчик события, возникающего после сохранения объекта.

Обработчик события получает аргумент типа `EntityAfterEventArgs`.

Свойства `EntityAfterEventArgs` предоставляющие сведения, относящиеся к событию:

- `ModifiedColumnValues`.
- `PrimaryColumnValue`.

`event EventHandler<EntitySaveErrorEventArgs> SaveError`

Обработчик события, возникающего при ошибке сохранения объекта.

Обработчик события получает аргумент типа `EntitySaveErrorEventArgs`.

Свойства `EntitySaveErrorEventArgs` предоставляющие сведения, относящиеся к событию:

- `Exception`.
- `IsHandled`.

`event EventHandler<EntityBeforeEventArgs> Saving`

Обработчик события, возникающего перед сохранением объекта.

Обработчик события получает аргумент типа `EntityBeforeEventArgs`.

© 2023 Terrasoft. Все права защищены.

Свойства `EntityBeforeEventArgs` предоставляющие сведения, относящиеся к событию:

- `AdditionalCondition`.
- `IsCanceled`.
- `KeyValue`.

`event EventHandler<EntityAfterEventArgs> Updated`

Обработчик события, возникающего после обновления объекта.

Обработчик события получает аргумент типа `EntityAfterEventArgs`.

Свойства `EntityAfterEventArgs` предоставляющие сведения, относящиеся к событию:

- `ModifiedColumnValues`.
- `PrimaryColumnValue`.

`event EventHandler<EntityBeforeEventArgs>Updating`

Обработчик события, возникающего перед обновлением объекта.

Обработчик события получает аргумент типа `EntityBeforeEventArgs`.

Свойства `EntityBeforeEventArgs` предоставляющие сведения, относящиеся к событию:

- `AdditionalCondition`.
- `IsCanceled`.
- `KeyValue`.

`event EventHandler<EntityValidationEventArgs> Validating`

Обработчик события, возникающего при проверке объекта.

Обработчик события получает аргумент типа `EntityValidationEventArgs`.

Свойства `EntityValidationEventArgs` предоставляющие сведения, относящиеся к событию:

- `Messages`.

Класс EntityMapper C#

Сложный

Класс `Terrasoft.Configuration.EntityMapper` — это утилитный класс конфигурации, который находится в пакете [*FinAppLending*] продукта Lending. `EntityMapper` позволяет сопоставлять данные одной сущности (`Entity`) с другой по правилам, определенным в конфигурационном файле. Использование подхода сопоставления данных разных сущностей позволяет избежать появления однообразного кода.

В продукте Lending существует два объекта, содержащих одинаковые колонки. Это объекты [*Физ. лицо*

© 2023 Terrasoft. Все права защищены.

] ([*Contact*]) и [*Анкета*] ([*AppForm*]). Также существует несколько деталей, относящихся к объекту [*Физ. лицо*] ([*Contact*]) и имеющих похожие детали, относящиеся к [*Анкета*] ([*AppForm*]). Очевидно, что при заполнении анкеты должна быть возможность по колонке [*Id*] объекта [*Физ. лицо*] ([*Contact*]) получить список всех его колонок и значений, а также список нужных деталей с их колонками и значениями, и сопоставить эти данные с данными, связанными с анкетой. После этого можно автоматически заполнить поля анкеты сопоставленными данными. Таким образом можно существенно уменьшить затраты на ручной ввод одинаковых данных.

Идея сопоставления данных разных сущностей реализована в следующих классах:

- `EntityMapper` — реализует логику сопоставления.
- `EntityResult` — определяет в каком виде вернется сопоставленная сущность.
- `MapConfig` — представляет набор правил для сопоставления.
- `DetailMapConfig` — используется для установки списка правил сопоставления деталей и связанных с ними сущностей.
- `RelationEntityMapConfig` — содержит правила для сопоставления связанных сущностей.
- `EntityFilterMap` — представляет из себя фильтр для запроса в базу данных.

Класс EntityMapper C#

Пространство имен `Terrasoft.Configuration`.

Класс реализует логику сопоставления.

Методы

```
virtual EntityResult GetMappedEntity(Guid recId, MapConfig config)
```

Возвращает сопоставленные данные для двух объектов `Entity`.

Параметры

`recId`	`GUID` записи в базе данных.
`config`	Экземпляр класса `MapConfig`, представляющий из себя набор правил сопоставления.

```
virtual Dictionary<string, object> GetColumnsValues(Guid recordId, MapConfig config, Dictionary<
```

Получает из базы данных главную сущность и сопоставляет ее колонки и значения по правилам, указанным в объекте `config`.

Параметры

© 2023 Terrasoft. Все права защищены.

`recordId`	`GUID` записи в базе данных.
`config`	Экземпляр класса `MapConfig`, представляющий из себя набор правил сопоставления.
`result`	Словарь колонок и их значений уже сопоставленной сущности.

```
virtual Dictionary<string, object> GetRelationEntityColumnsValues(List<RelationEntityMapConfig>
```

Получает из базы данных связанные сущности и сопоставляет их с основными сущностями.

Параметры

`relations`	Список правил для получения связанных записей.
`dictionaryToMerge`	Словарь с колонками и их значениями.
`columnName`	Название родительской колонки.
`entitylookup`	Объект, содержащий название и Id записи в базе.

Класс EntityResult C#

Пространство имен `Terrasoft.Configuration`.

Класс определяет в каком виде вернется сопоставленная сущность.

Свойства

`Columns` `Dictionary<string, object>`

Словарь с названиями колонок основной сущности и их значениями.

`Details` `Dictionary<string, List<Dictionary<string, object>>>`

Словарь названий деталей со списком их колонок и значений.

Класс MapConfig C#

Пространство имен `Terrasoft.Configuration`.

Класс представляет набор правил для сопоставления.

Свойства

© 2023 Terrasoft. Все права защищены.

`SourceEntityName` `string`

Название сущности в базе данных.

`Columns` `Dictionary<string, object>`

Словарь с названиями колонок одной сущности и сопоставляемыми колонками другой сущности.

`DetailsConfig` `List<DetailMapConfig>`

Список конфигурационных объектов с правилами для деталей.

`CleanDetails` `List<string>`

Список названий деталей для очистки их значений.

`RelationEntities` `List<RelationEntityMapConfig>`

Список конфигурационных объектов с правилами сопоставления связанных записей с главной сущностью.

Класс DetailMapConfig C#

Пространство имен `Terrasoft.Configuration`.

Класс используется для установки списка правил сопоставления деталей и связанных с ними сущностей.

Свойства

`DetailName` `string`

Название детали (для обеспечения уникальности экземпляра детали).

`SourceEntityName` `string`

Название сущности в базе данных.

`Columns` `Dictionary<string, object>`

Словарь с названиями колонок одной сущности и сопоставляемыми колонками другой сущности.

`Filters` `List<EntityFilterMap>`

© 2023 Terrasoft. Все права защищены.

Список конфигурационных объектов с правилами фильтрации для более точных выборок из базы данных.

`RelationEntities` `List<RelationEntityMapConfig>`

Список конфигурационных объектов с правилами сопоставления связанных записей с главной сущностью.

Класс RelationEntityMapConfig C#

Пространство имен `Terrasoft.Configuration`.

Класс содержит правила для сопоставления связанных сущностей.

Свойства

`ParentColumnName` `string`

Название родительской колонки, при нахождении которой будет срабатывать логика получения и сопоставления данных по сущности.

`SourceEntityName` `string`

Название сущности в базе данных.

`Columns` `Dictionary<string, object>`

Словарь с названиями колонок одной сущности и сопоставляемыми колонками другой сущности.

`Filters` `List<EntityFilterMap>`

Список конфигурационных объектов с правилами фильтрации для более точных выборок из базы данных.

`RelationEntities` `List<RelationEntityMapConfig>`

Список конфигурационных объектов с правилами сопоставления связанных записей с главной сущностью.

Класс EntityFilterMap C#

Пространство имен `Terrasoft.Configuration`.

Класс представляет из себя фильтр для запроса в базу данных.

© 2023 Terrasoft. Все права защищены.

Свойства

`ColumnName` `string`

Название колонки, при нахождении которой будет срабатывать логика фильтрации.

`Value` `object`

Значение, с которым необходимо сравнение.

Класс EntitySchemaQueryFunction C#

Сложный

Класс `Terrasoft.Core.Entities.EntitySchemaQueryFunction` реализует функцию выражения.

Идея функции выражения реализована в следующих классах:

- `EntitySchemaQueryFunction` — базовый класс функции выражения запроса к схеме объекта.
- `EntitySchemaAggregationQueryFunction` — реализует агрегирующую функцию выражения.
- `EntitySchemaIsNullQueryFunction` — заменяет значения `null` замещающим выражением.
- `EntitySchemaCoalesceQueryFunction` — возвращает первое выражение из списка аргументов, не равное `null`.
- `EntitySchemaCaseNotNullQueryFunctionWhenItem` — класс, описывающий выражение условия sql-оператора `CASE`.
- `EntitySchemaCaseNotNullQueryFunctionWhenItems` — коллекция выражений условий sql-оператора `CASE`.
- EntitySchemaStartOfCurrentHourQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression expression, int offset = 0) : this(parentQuery, offset)
- `EntitySchemaCaseNotNullQueryFunction` — возвращает одно из множества возможных значений в зависимости от указанных условий.
- `EntitySchemaSystemValueQueryFunction` — возвращает выражение системного значения.
- `EntitySchemaCurrentDateTimeQueryFunction` — реализует функцию выражения текущей даты и времени.
- `EntitySchemaBaseCurrentDateQueryFunction` — базовый класс функции выражения для базовой даты.
- `EntitySchemaCurrentDateQueryFunction` — реализует функцию выражения текущей даты.
- `EntitySchemaDateToCurrentYearQueryFunction` — реализует функцию выражения даты начала текущей недели.
- `EntitySchemaStartOfCurrentWeekQueryFunction` — реализует функцию, которая конвертирует выражение даты в такую же дату текущего года.
- `EntitySchemaStartOfCurrentMonthQueryFunction` — реализует функцию выражения даты начала текущего месяца.
- `EntitySchemaStartOfCurrentQuarterQueryFunction` — реализует функцию выражения даты начала

© 2023 Terrasoft. Все права защищены.

текущего квартала.

- `EntitySchemaStartOfCurrentHalfYearQueryFunction` — реализует функцию выражения даты начала текущего полугодия.
- `EntitySchemaStartOfCurrentYearQueryFunction` — реализует функцию выражения даты начала текущего года.
- `EntitySchemaBaseCurrentDateTimeQueryFunction` — базовый класс функции выражения базовых даты и времени.
- `EntitySchemaStartOfCurrentHourQueryFunction` — реализует функцию выражения начала текущего часа.
- `EntitySchemaCurrentTimeQueryFunction` — реализует функцию выражения текущего времени.
- `EntitySchemaCurrentUserQueryFunction` — реализует функцию выражения текущего пользователя.
- `EntitySchemaCurrentUserContactQueryFunction` — реализует функцию контакта текущего пользователя.
- `EntitySchemaCurrentUserAccountQueryFunction` — реализует функцию выражения контрагента текущего пользователя.
- `EntitySchemaDatePartQueryFunction` — реализует функцию запроса для части даты.
- `EntitySchemaUpperQueryFunction` — преобразовывает символы выражения аргумента к верхнему регистру.
- `EntitySchemaCastQueryFunction` — приводит выражение аргумента к заданному типу данных.
- `EntitySchemaTrimQueryFunction` — удаляет начальные и конечные пробелы из выражения.
- `EntitySchemaLengthQueryFunction` — возвращает длину выражения.
- `EntitySchemaConcatQueryFunction` — формирует строку, которая является результатом объединения строковых значений аргументов функции.
- `EntitySchemaWindowQueryFunction` — реализует функцию SQL окна.

Класс EntitySchemaQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Базовый класс функции выражения запроса к схеме объекта.

> **На заметку.** Полный перечень методов класса `EntitySchemaQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Методы

`abstract QueryColumnExpression CreateQueryColumnExpression(DBSecurityEngine dbSecurityEngine)`

Возвращает выражение колонки запроса для текущей функции, сформированное с учетом заданных прав доступа.

© 2023 Terrasoft. Все права защищены.

Параметры

dbSecurityEngine	Объект `Terrasoft.Core.DB.DBSecurityEngine`, определяющий права доступа.

`abstract DataValueType GetResultDataValueType(DataValueTypeManager dataValueTypeManager)`

Возвращает тип данных возвращаемого функцией результата, используя переданный менеджер типов данных.

Параметры

dataValueTypeManager	Менеджер типов данных.

`abstract bool GetIsSupportepataValueType(DataValueType dataValueType)`

Определяет, имеет ли возвращаемый функцией результат указанный тип данных.

Параметры

dataValueType	Тип данных.

`abstract string GetCaption()`

Возвращает заголовок функции выражения.

`virtual EntitySchemaQueryExpressionCollection GetArguments()`

Возвращает коллекцию выражений аргументов функции.

`void CheckIsSupportepataValueType(DataValueType dataValueType)`

Проверяет, имеет ли возвращаемый функцией результат указанный тип данных. В противном случае генерируется исключение.

Параметры

dataValueType	Тип данных.

Класс EntitySchemaAggregationQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

© 2023 Terrasoft. Все права защищены.

Класс реализует агрегирующую функцию выражения.

На заметку. Полный перечень методов и свойств класса `EntitySchemaAggregationQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaAggregationQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует экземпляр `EntitySchemaAggregationQueryFunction` заданного типа агрегирующей функции для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaAggregationQueryFunction(AggregationTypeStrict aggregationType, EntitySchemaQuery pa`

Инициализирует экземпляр `EntitySchemaAggregationQueryFunction` заданного типа агрегирующей функции для заданного запроса к схеме объекта.

Параметры

`aggregationType`	Тип агрегирующей функции.
`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaAggregationQueryFunction(AggregationTypeStrict aggregationType, EntitySchemaQueryExp`

Инициализирует новый экземпляр `EntitySchemaAggregationQueryFunction` для заданных типа агрегирующей функции, выражения и запроса к схеме объекта.

Параметры

`aggregationType`	Тип агрегирующей функции.
expression	Выражение запроса.
parentQuery	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaAggregationQueryFunction(EntitySchemaAggregationQueryFunction source)`

© 2023 Terrasoft. Все права защищены.

Инициализирует новый экземпляр `EntitySchemaAggregationQueryFunction`, являющийся клоном переданного экземпляра агрегирующей функции выражения.

Параметры

`source`	Экземпляр агрегирующей функции выражения, клон которой создается.

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

`AggregationType` `AggregationTypeStrict`

Тип агрегирующей функции.

`AggregationEvalType` `AggregationEvalType`

Область применения агрегирующей функции.

`Expression` `EntitySchemaQueryExpression`

Выражение аргумента агрегирующей функции.

Методы

`override void WriteMetaData(DataWriter writer)`

Выполняет сериализацию агрегирующей функции, используя заданный экземпляр `Terrasoft.Common.DataWriter`.

Параметры

`writer`	Экземпляр `Terrasoft.Common.DataWriter`, с помощью которого выполняется сериализация.

`override QueryColumnExpression CreateQueryColumnExpression(DBSecurityEngine dbSecurityEngine)`

Возвращает выражение колонки запроса для агрегирующей функции, сформированное с учетом заданных прав доступа.

© 2023 Terrasoft. Все права защищены.

Параметры

dbSecurityEngine	Объект `Terrasoft.Core.DB.DBSecurityEngine`, определяющий права доступа.

`override EntitySchemaQueryExpressionCollection GetArguments()`

Возвращает коллекцию выражений аргументов агрегирующей функции.

`override DataValueType GetResultDataValueType(DataValueTypeManager dataValueTypeManager)`

Возвращает тип данных возвращаемого агрегирующей функцией результата, используя заданный менеджер типов данных.

Параметры

dataValueTypeManager	Менеджер типов данных.

`override bool GetIsSupportepataValueType(DataValueType dataValueType)`

Определяет, имеет ли возвращаемый агрегирующей функцией результат указанный тип данных.

Параметры

dataValueType	Тип данных.

`override string GetCaption()`

Возвращает заголовок функции выражения.

`override object Clone()`

Создает клон текущего экземпляра `EntitySchemaAggregationQueryFunction`.

`EntitySchemaAggregationQueryFunction All()`

Устанавливает для текущей агрегирующей функции область применения [*Ко всем значениям*].

`EntitySchemaAggregationQueryFunction Distinct()`

Устанавливает для текущей агрегирующей функции область применения [*К уникальным значениям*].

© 2023 Terrasoft. Все права защищены.

Класс EntitySchemaIsNullQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс заменяет значения `null` замещающим выражением.

На заметку. Полный перечень методов и свойств класса `EntitySchemaIsNullQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaIsNullQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует экземпляр `EntitySchemaIsNullQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaIsNullQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression check`

Инициализирует новый экземпляр `EntitySchemaIsNullQueryFunction` для заданных запроса к схеме объекта, проверяемого выражения и замещающего выражения.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`checkExpression`	Выражение, которое проверяется на равенство `null`.
`replacementExpression`	Выражение, которое возвращается функцией, если `checkExpression` равно `null`.

`EntitySchemaIsNullQueryFunction(EntitySchemaIsNullQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaIsNullQueryFunction`, являющийся клоном переданной функции выражения.

Параметры

`source`	Экземпляр функции `EntitySchemaIsNullQueryFunction`, клон которой создается.

© 2023 Terrasoft. Все права защищены.

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

`CheckExpression` `EntitySchemaQueryExpression`

Выражение аргумента функции, которое проверяется на равенство значению `null`.

`ReplacementExpression` `EntitySchemaQueryExpression`

Выражение аргумента функции, которое возвращается, если проверяемое выражение равно `null`.

Методы

`override void WriteMetaData(DataWriter writer)`

Выполняет сериализацию функции выражения, используя переданный экземпляр `DataWriter`.

Параметры

`writer`	Экземпляр `DataWriter`, с помощью которого выполняется сериализация функции выражения.

`override QueryColumnExpression CreateQueryColumnExpression(DBSecurityEngine dbSecurityEngine)`

Возвращает выражение колонки запроса для текущей функции, сформированное с учетом заданных прав доступа.

Параметры

`dbSecurityEngine`	Объект `Terrasoft.Core.DB.DBSecurityEngine`, определяющий права доступа.

`override EntitySchemaQueryExpressionCollection GetArguments()`

Возвращает коллекцию выражений аргументов функции.

`override DataValueType GetResultDataValueType(DataValueTypeManager dataValueTypeManager)`

Возвращает тип данных возвращаемого функцией результата, используя переданный менеджер

© 2023 Terrasoft. Все права защищены.

типов данных.

Параметры

`dataValueTypeManager`	Менеджер типов данных.

Класс EntitySchemaCoalesceQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс возвращает первое выражение из списка аргументов, не равное `null`.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCoalesceQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaCoalesceQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaCoalesceQueryFunction` для заданного запроса к схеме объекта.

Параметры

`aggregationType`	Тип агрегирующей функции.
`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaCoalesceQueryFunction(EntitySchemaCoalesceQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaCoalesceQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaCoalesceQueryFunction`, клон которой создается.

Свойства

`QueryAlias string`

© 2023 Terrasoft. Все права защищены.

Псевдоним функции в sql-запросе.

`Expressions` `EntitySchemaQueryExpressionCollection`

Коллекция выражений аргументов функции.

`HasExpressions` `bool`

Признак, определяющий наличие хотя бы одного элемента в коллекции выражений аргументов функции.

Методы

`override bool GetIsSupportepataValueType(DataValueType dataValueType)`

Определяет, имеет ли возвращаемый функцией результат указанный тип данных.

Параметры

`dataValueType`	Тип данных.

Класс EntitySchemaCaseNotNullQueryFunctionWhenItem C#

Пространство имен `Terrasoft.Core.Entities`.

Класс, описывающий выражение условия sql-оператора `CASE`.

На заметку. Полный перечень методов класса `EntitySchemaCaseNotNullQueryFunctionWhenItem`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaCaseNotNullQueryFunctionWhenItem()`

Инициализирует новый экземпляр `EntitySchemaCaseNotNullQueryFunctionWhenItem`.

`EntitySchemaCaseNotNullQueryFunctionWhenItem(EntitySchemaQueryExpression whenExpression, EntityS`

Инициализирует экземпляр `EntitySchemaCaseNotNullQueryFunctionWhenItem` для заданных выражений предложений `WHEN` и `THEN`.

Параметры

© 2023 Terrasoft. Все права защищены.

whenExpression	Выражение предложения `WHEN` условия.
thenExpression	Выражение предложения `THEN` условия.

`EntitySchemaCaseNotNullQueryFunctionWhenItem(EntitySchemaCaseNotNullQueryFunctionWhenItem source`

Инициализирует экземпляр `EntitySchemaCaseNotNullQueryFunctionWhenItem`, являющийся клоном переданной функции.

Параметры

source	Функция `EntitySchemaCaseNotNullQueryFunctionWhenItem`, клон которой создается.

Свойства

`WhenExpression` `EntitySchemaQueryExpression`

Выражение предложения `WHEN`.

`ThenExpression` `EntitySchemaQueryExpression`

Выражение предложения `THEN`.

Класс EntitySchemaCaseNotNullQueryFunctionWhenItems C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует коллекцию выражений условий sql-оператора `CASE`.

На заметку. Полный перечень методов класса `EntitySchemaCaseNotNullQueryFunctionWhenItems`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaCaseNotNullQueryFunctionWhenItems()`

Инициализирует экземпляр `EntitySchemaCaseNotNullQueryFunctionWhenItems`.

`EntitySchemaCaseNotNullQueryFunctionWhenItems(EntitySchemaCaseNotNullQueryFunctionWhenItems sour`

© 2023 Terrasoft. Все права защищены.

Инициализирует новый экземпляр `EntitySchemaCaseNotNullQueryFunctionWhenItems`, являющийся клоном клоном переданной коллекции условий.

Параметры

source	Коллекция условий, клон которой создается.

Класс EntitySchemaCaseNotNullQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс возвращает одно из множества возможных значений в зависимости от указанных условий.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCaseNotNullQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`CurrentDateTimeQueryFunction()`

Инициализирует новый экземпляр `CurrentDateTimeQueryFunction`.

`EntitySchemaCaseNotNullQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaCaseNotNullQueryFunction` для заданного запроса к схеме объекта.

Параметры

parentQuery	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaCaseNotNullQueryFunction(EntitySchemaCaseNotNullQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaCaseNotNullQueryFunction`, являющийся клоном переданной функции.

Параметры

source	Функция `EntitySchemaCaseNotNullQueryFunction`, клон которой создается.

© 2023 Terrasoft. Все права защищены.

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

`WhenItems` `EntitySchemaCaseNotNullQueryFunctionWhenItems`

Коллекция условий функции выражения.

`HasWhenItems` `bool`

Признак, имеет ли функция хотя бы одно условие.

`ElseExpression` `EntitySchemaQueryExpression`

Выражение предложения `ELSE`.

Методы

`void SpecifyQueryAlias(string queryAlias)`

Определяет для текущей функции выражения заданный псевдоним в результирующем sql-запросе.

Параметры

`queryAlias`	Псевдоним, определяемый для текущей функции.

Класс EntitySchemaSystemValueQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс возвращает выражение системного значения.

На заметку. Полный перечень методов и свойств класса `EntitySchemaSystemValueQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

© 2023 Terrasoft. Все права защищены.

`SystemValueName` `string`

Имя системного значения.

Класс EntitySchemaCurrentDateTimeQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения текущей даты и времени.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCurrentDateTimeQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaCurrentDateTimeQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует экземпляр `EntitySchemaCurrentDateTimeQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaCurrentDateTimeQueryFunction(EntitySchemaCurrentDateTimeQueryFunction source)`

Инициализирует экземпляр `EntitySchemaCurrentDateTimeQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Экземпляр функции `EntitySchemaCurrentDateTimeQueryFunction`, клон которой создается.

Свойства

`SystemValueName` `string`

Имя системного значения.

Методы

© 2023 Terrasoft. Все права защищены.

```
override string GetCaption()
```

Возвращает заголовок функции выражения.

```
override object Clone()
```

Создает клон текущего экземпляра `EntitySchemaCurrentDateTimeQueryFunction`.

Класс EntitySchemaBaseCurrentDateQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Базовый класс функции выражения для базовой даты.

На заметку. Полный перечень методов и свойств класса `EntitySchemaBaseCurrentDateQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Свойства

```
SystemValueName string
```

Имя системного значения.

```
Offset int
```

Смещение.

Класс EntitySchemaCurrentDateQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения текущей даты.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCurrentDateQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaCurrentDateQueryFunction(EntitySchemaQuery parentQuery, int offset = 0) : this(parer
EntitySchemaCurrentDateQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression
```

© 2023 Terrasoft. Все права защищены.

Инициализирует экземпляр `EntitySchemaCurrentDateQueryFunction` с указанным смещением относительно базовой даты для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`offset`	Смещение в днях относительно контрольной даты. Значение по умолчанию - `0`.
`expression`	Выражение запроса.

`EntitySchemaCurrentDateQueryFunction(EntitySchemaCurrentDateQueryFunction source)`

Инициализирует экземпляр `EntitySchemaCurrentDateQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Экземпляр функции `EntitySchemaCurrentDateQueryFunction`, клон которой создается.

Методы

`override string GetCaption()`

Возвращает заголовок функции выражения.

`override object Clone()`

Создает клон текущего экземпляра `EntitySchemaCurrentDateQueryFunction`.

Класс EntitySchemaDateToCurrentYearQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию, которая конвертирует выражение даты в такую же дату текущего года.

На заметку. Полный перечень методов и свойств класса `EntitySchemaDateToCurrentYearQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

© 2023 Terrasoft. Все права защищены.

`EntitySchemaDateToCurrentYearQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaDateToCurrentYearQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaDateToCurrentYearQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpre`

Инициализирует новый экземпляр `EntitySchemaDateToCurrentYearQueryFunction` для заданного запроса к схеме объекта и переданного выражения даты.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`expression`	Выражение запроса.

`EntitySchemaDateToCurrentYearQueryFunction(EntitySchemaDateToCurrentYearQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaDateToCurrentYearQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaDateToCurrentYearQueryFunction`, клон которой создается.

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

`Expression` `EntitySchemaQueryExpression`

Выражение аргументов функции.

Класс EntitySchemaStartOfCurrentWeekQueryFunction C#

© 2023 Terrasoft. Все права защищены.

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения текущей даты.

На заметку. Полный перечень методов и свойств класса `EntitySchemaStartOfCurrentWeekQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaStartOfCurrentWeekQueryFunction(EntitySchemaQuery parentQuery, int offset = 0) : thi
EntitySchemaStartOfCurrentWeekQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpr
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentWeekQueryFunction` с указанным смещением относительно базовой даты для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`offset`	Смещение в днях относительно контрольной даты. Значение по умолчанию - `0`.
`expression`	Выражение запроса.

```
EntitySchemaStartOfCurrentWeekQueryFunction(EntitySchemaStartOfCurrentWeekQueryFunction source)
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentWeekQueryFunction`, являющийся клоном переданной функции выражения.

Параметры

`source`	Экземпляр функции `EntitySchemaStartOfCurrentWeekQueryFunction`, клон которой создается.

Методы

```
override string GetCaption()
```

Возвращает заголовок функции выражения.

```
override object Clone()
```

© 2023 Terrasoft. Все права защищены.

Создает клон текущего экземпляра `EntitySchemaStartOfCurrentWeekQueryFunction`.

Класс EntitySchemaStartOfCurrentMonthQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения даты начала текущего месяца.

На заметку. Полный перечень методов и свойств класса `EntitySchemaStartOfCurrentMonthQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaStartOfCurrentMonthQueryFunction(EntitySchemaQuery parentQuery, int offset = 0) : th
EntitySchemaStartOfCurrentMonthQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExp
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentMonthQueryFunction` с указанным смещением относительно базовой даты для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`offset`	Смещение в днях относительно контрольной даты. Значение по умолчанию - `0`.
`expression`	Выражение запроса.

```
EntitySchemaStartOfCurrentMonthQueryFunction(EntitySchemaStartOfCurrentMonthQueryFunction source
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentMonthQueryFunction`, являющийся клоном переданной функции выражения.

Параметры

`source`	Экземпляр функции `EntitySchemaStartOfCurrentMonthQueryFunction`, клон которой создается.

Методы

```
override string GetCaption()
```

© 2023 Terrasoft. Все права защищены.

Возвращает заголовок функции выражения.

```
override object Clone()
```

Создает клон текущего экземпляра `EntitySchemaStartOfCurrentMonthQueryFunction`.

Класс EntitySchemaStartOfCurrentQuarterQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения даты начала текущего месяца.

На заметку. Полный перечень методов и свойств класса `EntitySchemaStartOfCurrentQuarterQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaStartOfCurrentQuarterQueryFunction(EntitySchemaQuery parentQuery, int offset = 0) :
EntitySchemaStartOfCurrentQuarterQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryE
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentQuarterQueryFunction` с указанным смещением относительно базовой даты для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`offset`	Смещение в днях относительно контрольной даты. Значение по умолчанию - `0`.
`expression`	Выражение запроса

```
EntitySchemaStartOfCurrentQuarterQueryFunction(EntitySchemaStartOfCurrentQuarterQueryFunction sc
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentQuarterQueryFunction`, являющийся клоном переданной функции выражения.

Параметры

`source`	Экземпляр функции `EntitySchemaStartOfCurrentQuarterQueryFunction`, клон которой создается.

© 2023 Terrasoft. Все права защищены.

Методы

`override string GetCaption()`

Возвращает заголовок функции выражения.

`override object Clone()`

Создает клон текущего экземпляра `EntitySchemaStartOfCurrentQuarterQueryFunction`.

Класс EntitySchemaStartOfCurrentHalfYearQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения даты начала текущего полугодия.

> **На заметку.** Полный перечень методов и свойств класса `EntitySchemaStartOfCurrentHalfYearQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaStartOfCurrentHalfYearQueryFunction(EntitySchemaQuery parentQuery, int offset = 0) :
EntitySchemaStartOfCurrentHalfYearQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQuery
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentHalfYearQueryFunction` с указанным смещением относительно базовой даты для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`offset`	Смещение в днях относительно контрольной даты. Значение по умолчанию - `0`.
`expression`	Выражение запроса.

`EntitySchemaStartOfCurrentHalfYearQueryFunction(EntitySchemaStartOfCurrentHalfYearQueryFunction`

Инициализирует экземпляр `EntitySchemaStartOfCurrentHalfYearQueryFunction`, являющийся клоном переданной функции выражения.

Параметры

© 2023 Terrasoft. Все права защищены.

`source`	Экземпляр функции `EntitySchemaStartOfCurrentHalfYearQueryFunction`, клон которой создается.

Методы

```
override string GetCaption()
```

Возвращает заголовок функции выражения.

```
override object Clone()
```

Создает клон текущего экземпляра `EntitySchemaStartOfCurrentHalfYearQueryFunction`.

Класс EntitySchemaStartOfCurrentYearQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения даты начала текущего года.

На заметку. Полный перечень методов и свойств класса `EntitySchemaStartOfCurrentYearQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaStartOfCurrentYearQueryFunction(EntitySchemaQuery parentQuery, int offset = 0) : thi
EntitySchemaStartOfCurrentYearQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpr
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentYearQueryFunction` с указанным смещением относительно базовой даты для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`offset`	Смещение в днях относительно контрольной даты. Значение по умолчанию - `0`.
`expression`	Выражение запроса.

```
EntitySchemaStartOfCurrentYearQueryFunction(EntitySchemaStartOfCurrentYearQueryFunction source)
```

© 2023 Terrasoft. Все права защищены.

Инициализирует экземпляр `EntitySchemaStartOfCurrentYearQueryFunction`, являющийся клоном переданной функции выражения.

Параметры

`source`	Экземпляр функции `EntitySchemaStartOfCurrentYearQueryFunction`, клон которой создается.

Методы

`override string GetCaption()`

Возвращает заголовок функции выражения.

`override object Clone()`

Создает клон текущего экземпляра `EntitySchemaStartOfCurrentHalfYearQueryFunction`.

Класс EntitySchemaBaseCurrentDateTimeQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Базовый класс функции выражения базовых даты и времени.

На заметку. Полный перечень методов и свойств класса `EntitySchemaBaseCurrentDateTimeQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Свойства

`SystemValueName string`

Имя системного значения.

Класс EntitySchemaStartOfCurrentHourQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения начала текущего часа.

На заметку. Полный перечень методов и свойств класса `EntitySchemaStartOfCurrentHourQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

© 2023 Terrasoft. Все права защищены.

Конструкторы

```
EntitySchemaStartOfCurrentHourQueryFunction(EntitySchemaQuery parentQuery, int offset = 0) : bas
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentHourQueryFunction`, который является частью `parentQuery` и указан `offset` относительно базовой даты.

Параметры

`parentQuery`	Экземпляр `EntitySchemaQuery`.
`offset`	Смещение в часах относительно базовой даты.

```
EntitySchemaStartOfCurrentHourQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpr
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentHourQueryFunction`, который является частью `parentQuery`, имеет указанные аргументы `expression` и `offset` относительно базовой даты.

Параметры

`parentQuery`	Экземпляр `EntitySchemaQuery`.
`expression`	Выражение аргумента функции.
`offset`	Смещение в часах относительно базовой даты.

```
EntitySchemaStartOfCurrentHourQueryFunction(EntitySchemaStartOfCurrentHourQueryFunction source)
```

Инициализирует экземпляр `EntitySchemaStartOfCurrentHourQueryFunction`, являющийся клоном переданной функции выражения.

Параметры

`source`	Экземпляр функции `EntitySchemaStartOfCurrentHourQueryFunction`, клон которой создается.

Методы

```
override string GetCaption()
```

Возвращает заголовок функции выражения.

© 2023 Terrasoft. Все права защищены.

```
override object Clone()
```

Создает клон текущего экземпляра `EntitySchemaStartOfCurrentHourQueryFunction`.

Класс EntitySchemaCurrentTimeQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения текущего времени.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCurrentTimeQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaCurrentTimeQueryFunction(EntitySchemaQuery parentQuery) : base(parentQuery)
```

Инициализирует новый экземпляр `EntitySchemaCurrentTimeQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

```
EntitySchemaCurrentTimeQueryFunction(EntitySchemaCurrentTimeQueryFunction source) : base(source)
```

Инициализирует новый экземпляр `EntitySchemaCurrentTimeQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaCurrentTimeQueryFunction`, клон которой создается.

Свойства

```
SystemValueName string
```

Имя системного значения.

Методы

© 2023 Terrasoft. Все права защищены.

```
override string GetCaption()
```

Возвращает заголовок функции выражения.

```
override object Clone()
```

Создает клон текущего экземпляра `EntitySchemaCurrentTimeQueryFunction`.

Класс EntitySchemaCurrentUserQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения текущего пользователя.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCurrentUserQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaCurrentUserQueryFunction(EntitySchemaQuery parentQuery) : base(parentQuery)
```

Инициализирует новый экземпляр `EntitySchemaCurrentUserQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

```
EntitySchemaCurrentUserQueryFunction(EntitySchemaCurrentUserQueryFunction source) : base(source)
```

Инициализирует новый экземпляр `EntitySchemaCurrentUserQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaCurrentUserQueryFunction`, клон которой создается.

Свойства

```
SystemValueName string
```

© 2023 Terrasoft. Все права защищены.

~~SystemValueName string~~

Имя системного значения.

Методы

```
override string GetCaption()
```

Возвращает заголовок функции выражения.

```
override object Clone()
```

Создает клон текущего экземпляра `EntitySchemaCurrentUserQueryFunction`.

Класс EntitySchemaCurrentUserContactQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения контакта текущего пользователя.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCurrentUserContactQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

```
EntitySchemaCurrentUserContactQueryFunction(EntitySchemaQuery parentQuery) : base(parentQuery)
```

Инициализирует новый экземпляр `EntitySchemaCurrentUserContactQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

```
EntitySchemaCurrentUserContactQueryFunction(EntitySchemaCurrentUserContactQueryFunction source)
```

Инициализирует новый экземпляр `EntitySchemaCurrentUserContactQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaCurrentUserContactQueryFunction`, клон которой создается.

© 2023 Terrasoft. Все права защищены.

Свойства

`SystemValueName` `string`

Имя системного значения.

Методы

`override string GetCaption()`

Возвращает заголовок функции выражения.

`override object Clone()`

Создает клон текущего экземпляра `EntitySchemaCurrentUserContactQueryFunction`.

Класс EntitySchemaCurrentUserAccountQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию выражения контрагента текущего пользователя.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCurrentUserAccountQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaCurrentUserAccountQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaCurrentUserAccountQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaCurrentUserAccountQueryFunction(EntitySchemaCurrentUserAccountQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaCurrentUserAccountQueryFunction`, являющийся клоном переданной функции.

Параметры

© 2023 Terrasoft. Все права защищены.

source	Функция `EntitySchemaCurrentUserAccountQueryFunction`, клон которой создается.

Свойства

`SystemValueName` `string`

Имя системного значения.

Класс EntitySchemaDatePartQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию запроса для части даты.

На заметку. Полный перечень методов и свойств класса `EntitySchemaDatePartQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaDatePartQueryFunction(EntitySchemaQuery parentQuery) : base(parentQuery)`

Инициализирует новый экземпляр `EntitySchemaDatePartQueryFunction` для заданного запроса к схеме объекта.

Параметры

parentQuery	Экземпляр `EntitySchemaQuery`.

`EntitySchemaDatePartQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaDatePartQueryFuncti`

Инициализирует новый экземпляр `EntitySchemaDatePartQueryFunction`, который является частью `parentQuery` с указанной частью даты `interval` для запроса к схеме сущности и выражению запроса `expression`.

Параметры

© 2023 Terrasoft. Все права защищены.

`parentQuery`	Экземпляр `EntitySchemaQuery`.
`interval`	Часть даты.
`expression`	Выражение запроса.

`EntitySchemaDatePartQueryFunction(EntitySchemaDatePartQueryFunction source) : base(source)`

Инициализирует новый экземпляр `EntitySchemaDatePartQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaDatePartQueryFunction`, клон которой создается.

Свойства

`QueryAlias string`

Псевдоним функции в sql-запросе.

`EntitySchemaDatePartQueryFunctionInterval Interval`

Часть даты, возвращаемая функцией.

`EntitySchemaQueryExpression Expression`

Выражение аргумента функции.

Методы

`override void WriteMetaData(DataWriter writer)`

Выполняет сериализацию функции, используя заданный экземпляр `Terrasoft.Common.DataWriter`.

Параметры

`writer`	Экземпляр `Terrasoft.Common.DataWriter`, с помощью которого выполняется сериализация.

© 2023 Terrasoft. Все права защищены.

```
override QueryColumnExpression CreateQueryColumnExpression(DBSecurityEngine dbSecurityEngine)
```

Возвращает выражение колонки запроса для текущей функции, сформированное с учетом заданных прав доступа.

Параметры

dbSecurityEngine	Объект `Terrasoft.Core.DB.DBSecurityEngine`, определяющий права доступа.

```
override DataValueType GetResultDataValueType(DataValueTypeManager dataValueTypeManager)
```

Возвращает тип данных возвращаемого функцией результата, используя переданный менеджер типов данных.

Параметры

dataValueTypeManager	Менеджер типов данных.

```
override bool GetIsSupportedDataValueType(DataValueType dataValueType)
```

Определяет, имеет ли возвращаемый функцией результат указанный тип данных.

Параметры

dataValueType	Тип данных.

```
override string GetCaption()
```

Возвращает заголовок функции выражения.

```
override EntitySchemaQueryExpressionCollection GetArguments()
```

Возвращает коллекцию выражений аргументов функции.

```
override object Clone()
```

Создает клон текущего экземпляра `EntitySchemaUpperQueryFunction`.

Класс EntitySchemaUpperQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

© 2023 Terrasoft. Все права защищены.

Класс преобразовывает символы выражения аргумента к верхнему регистру.

На заметку. Полный перечень методов и свойств класса `EntitySchemaUpperQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaUpperQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaUpperQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaUpperQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression expres`

Инициализирует новый экземпляр `EntitySchemaUpperQueryFunction` для заданного запроса к схеме объекта и переданного выражения даты.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`expression`	Выражение запроса.

`EntitySchemaUpperQueryFunction(EntitySchemaUpperQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaUpperQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaUpperQueryFunction`, клон которой создается.

Свойства

`QueryAlias string`

© 2023 Terrasoft. Все права защищены.

Псевдоним функции в sql-запросе.

`Expression` `EntitySchemaQueryExpression`

Выражение аргументов функции.

Класс EntitySchemaCastQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс приводит выражение аргумента к заданному типу данных.

На заметку. Полный перечень методов и свойств класса `EntitySchemaCastQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaCastQueryFunction(EntitySchemaQuery parentQuery, DBDataValueType castType)`

Инициализирует новый экземпляр `EntitySchemaCastQueryFunction` для заданного запроса к схеме объекта с указанным целевым типом данных.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`castType`	Целевой тип данных.

`EntitySchemaCastQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression express`

Инициализирует новый экземпляр `EntitySchemaCastQueryFunction` с заданными выражением и целевым типом данных.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`expression`	Выражение запроса.
`castType`	Целевой тип данных.

© 2023 Terrasoft. Все права защищены.

`EntitySchemaCastQueryFunction(EntitySchemaCastQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaCastQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaCastQueryFunction`, клон которой создается.

Свойства

`QueryAlias string`

Псевдоним функции в sql-запросе.

`Expression EntitySchemaQueryExpression`

Выражение аргумента функции.

`CastType DBDataValueType`

Целевой тип данных.

Класс EntitySchemaTrimQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс удаляет начальные и конечные пробелы из выражения.

На заметку. Полный перечень методов и свойств класса `EntitySchemaTrimQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaTrimQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaTrimQueryFunction` для заданного запроса к схеме объекта.

Параметры

© 2023 Terrasoft. Все права защищены.

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaTrimQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression express`

Инициализирует новый экземпляр `EntitySchemaTrimQueryFunction` для заданного запроса к схеме объекта и переданного выражения даты.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`expression`	Выражение запроса.

`EntitySchemaTrimQueryFunction(EntitySchemaTrimQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaTrimQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaTrimQueryFunction`, клон которой создается.

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

`Expression` `EntitySchemaQueryExpression`

Выражение аргументов функции.

Класс EntitySchemaLengthQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс возвращает длину выражения.

На заметку. Полный перечень методов и свойств класса `EntitySchemaLengthQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

© 2023 Terrasoft. Все права защищены.

Конструкторы

`EntitySchemaLengthQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaLengthQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaLengthQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression expre`

Инициализирует новый экземпляр `EntitySchemaLengthQueryFunction` для заданного запроса к схеме объекта и переданного выражения даты.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`expression`	Выражение запроса.

`EntitySchemaLengthQueryFunction(EntitySchemaLengthQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaLengthQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaLengthQueryFunction`, клон которой создается.

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

`Expression` `EntitySchemaQueryExpression`

Выражение аргументов функции.

© 2023 Terrasoft. Все права защищены.

Класс EntitySchemaConcatQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс формирует строку, которая является результатом объединения строковых значений аргументов функции.

На заметку. Полный перечень методов и свойств класса `EntitySchemaConcatQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaConcatQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaConcatQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaConcatQueryFunction(EntitySchemaQuery parentQuery, EntitySchemaQueryExpression[] exp`

Инициализирует новый экземпляр `EntitySchemaConcatQueryFunction` для заданных массива выражений и запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`expressions`	Массив выражений.

`EntitySchemaConcatQueryFunction(EntitySchemaConcatQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaConcatQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaConcatQueryFunction`, клон которой создается.

© 2023 Terrasoft. Все права защищены.

Свойства

`QueryAlias` `string`

Псевдоним функции в sql-запросе.

`Expressions` `EntitySchemaQueryExpressionCollection`

Коллекция выражений аргументов функции.

`HasExpressions` `bool`

Признак, определяющий наличие хотя бы одного элемента в коллекции выражений аргументов функции.

Класс EntitySchemaWindowQueryFunction C#

Пространство имен `Terrasoft.Core.Entities`.

Класс реализует функцию SQL окна.

На заметку. Полный перечень методов и свойств класса `EntitySchemaWindowQueryFunction`, его родительских классов, а также реализуемых им интерфейсов, можно найти в документации ".NET библиотеки классов ядра платформы".

Конструкторы

`EntitySchemaWindowQueryFunction(EntitySchemaQuery parentQuery)`

Инициализирует новый экземпляр `EntitySchemaWindowQueryFunction` для заданного запроса к схеме объекта.

Параметры

`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.

`EntitySchemaWindowQueryFunction(EntitySchemaQueryExpression function, EntitySchemaQuery esq)`

Инициализирует новый экземпляр `EntitySchemaWindowQueryFunction` для заданного запроса к схеме объекта.

Параметры

© 2023 Terrasoft. Все права защищены.

`function`	Вложенная функция запроса.
`esq`	Запрос к схеме объекта.

`EntitySchemaWindowQueryFunction(EntitySchemaQueryExpression function, EntitySchemaQuery esq, Ent`

Инициализирует новый экземпляр `EntitySchemaWindowQueryFunction` для заданного запроса к схеме объекта.

Параметры

`function`	Вложенная функция запроса.
`parentQuery`	Запрос к схеме объекта, которому принадлежит функция.
`partitionBy`	Выражение для разделения запроса.
`orderBy`	Выражение для сортировки запроса.

`EntitySchemaWindowQueryFunction(EntitySchemaQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaWindowQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaQueryFunction`, клон которой создается.

`EntitySchemaWindowQueryFunction(EntitySchemaWindowQueryFunction source)`

Инициализирует новый экземпляр `EntitySchemaWindowQueryFunction`, являющийся клоном переданной функции.

Параметры

`source`	Функция `EntitySchemaWindowQueryFunction`, клон которой создается.

Свойства

`QueryAlias` `string`

© 2023 Terrasoft. Все права защищены.

Псевдоним функции в sql-запросе.

`InnerFunction` `EntitySchemaQueryExpression`

Функция для применения.

`PartitionByExpression` `EntitySchemaQueryExpression`

Разделение по пунктам.

`OrderByExpression` `EntitySchemaQueryExpression`

Сортировать по пункту.

Класс EntitySchemaQueryOptions C#

Сложный

Пространство имен `Terrasoft.Core.Entities`.

Класс `Terrasoft.Core.Entities.EntitySchemaQueryOptions` предназначен для настроек запроса к схеме объекта.

На заметку. Полный перечень методов и свойств класса `EntitySchemaQueryOptions`, его родительских классов, а также реализуемых им интерфейсов можно найти в Библиотеке .NET классов.

Конструкторы

`EntitySchemaQueryOptions`

Инициализирует экземпляр класса. В конструкторе свойству `PageableRowCount` по умолчанию устанавливается значение 14.

Свойства

`PageableRowCount` `int`

Количество записей страницы результирующего набора данных, возвращаемого запросом.

`PageableDirection` `Terrasoft.Core.DB.PageableSelectDirection`

Направление постраничного вывода.

© 2023 Terrasoft. Все права защищены.

Возможные значения (`Terrasoft.Core.DB.PageableSelectDirection`)

`Prior`	Предыдущая страница.
`First`	Первая страница.
`Current`	Текущая страница.
`Next`	Следующая страница.

`PageableConditionValues` `Dictionary<string, object>`

Значения условий постраничного вывода.

`HierarchicalMaxDepth` `int`

Максимальный уровень вложенности иерархического запроса.

`HierarchicalColumnName` `string`

Имя колонки, которая используется для построения иерархического запроса.

`HierarchicalColumnValue` `Guid`

Начальное значение иерархической колонки, от которого будет строиться иерархия.

© 2023 Terrasoft. Все права защищены.