

Класс DBConnector

Основное применения класса DBConnector - работа с БД из сайтов, созданных не в QP7, а также других типов приложений: десктопных приложений, сервисов и.т.д.

Конструкторы

```
public DBConnector(string strConnectionString)
```

В качестве параметра передается строка подключения к БД.

```
public DBConnector()
```

Если строка подключения не задана, значение берется из статического свойства [ConnectionString](#)

Основные свойства

ConnectionString

```
public static string ConnectionString { get; set; }
```

Строка подключения по умолчанию, используемая при создании экземпляров DBConnector, если в конструкторе строка подключения не задана явно. Инициализируется [строкой подключения qp_database](#) из [конфигурационного файла](#).

CustomConnectionString

```
public string CustomConnectionString { get; set; }
```

Строка подключения, специфичная для данного экземпляра DBConnector. Если строка подключения была передана в конструкторе, то она хранится здесь и может быть изменена.

IsStage

```
public bool IsStage { get; set; }
```

Параметр, определяющий в каком режиме (live или stage) работает экземпляр класса DBConnector. По умолчанию - false. Это оказывает влияние на следующие методы:

- [GetContentItemLinkQuery](#) и [GetContentItemLinkIds](#)
- [SendNotification](#)

- [GetMapFileContents](#) и [GetDefaultMapFileContents](#)

Если сайт сделан на объектах QP7, то параметр инициализируется в соответствии с режимом, в котором был собран сайт. В противном случае учитывается значение [параметра isLive конфигурационного файла](#). Наконец, параметр может быть задан непосредственно из кода. Если ни из одного источника значение не было получено, то используется значение по умолчанию – false.

CacheData

```
public bool CacheData { get; set; }
```

Определяет, должен ли экземпляр класса DBConnector кэшировать данные, получаемые из БД. Значение по умолчанию - true. Интервалы кэширования задаются в [конфигурационном файле](#). Данные могут кэшироваться как локально (в экземпляре), так и в общем кэше [HttpRuntime.Cache](#) в зависимости от доступности общего кэша и параметра [ForceLocalCache](#).

ForceLocalCache

```
public bool ForceLocalCache { get; set; }
```

При установке в true, заставляет экземпляр класса DBConnector принудительно использовать локальный кэш экземпляра вместо общего кэша [HttpRuntime.Cache](#). Если общий кэш недоступен, то вне зависимости от значения данного параметра будет использоваться локальный кэш. Если `CacheData == false`, то значение данного свойства игнорируется.

UpdateManyToMany

```
public bool UpdateManyToMany { get; set; }
```

Показывает, обновлять ли M2M-поля и M2O-поля при создании/обновлении статьи методом [AddFormToContent](#). Значение по умолчанию - true. Используется LINQ-to-SQL классами.

Методы

- [Кэширование](#)
- [Получение данных статей](#)
- [Обновление данных статей](#)
- [Массовое создание/изменение статей](#)
- [Выполнение произвольных SQL-запросов](#)
- [Работа с файлом отображения](#)
- [Физические и виртуальные пути](#)
- [Отправка уведомлений](#)
- [Вспомогательные методы](#)

From:

<http://wiki.qpublishing.ru/> - **QP7.Framework Docs**

Permanent link:

<http://wiki.qpublishing.ru/doku.php?id=api:dbconnector>



Last update: **2016/03/08 13:29**