

LINQ-to-SQL классы и кэширование

У класса DBConnector существует обобщенный экземплярный метод [GetCachedEntity](#).

```
public T GetCachedEntity<T>(string key, Func<T> fillAction)
public T GetCachedEntity<T>(string key, Func<string, T> fillAction)
public T GetCachedEntity<T>(string key, double cacheInterval, Func<T>
fillAction)
public T GetCachedEntity<T>(string key, double cacheInterval, Func<string,
T> fillAction)
```

Данный метод позволяет кэшировать произвольные объекты ссылочных типов (можно даже кэшировать *null*-значения). Если мы хотим закэшировать коллекцию объектов, полученных с помощью LINQ-to-SQL, ее сначала нужно сделать вещественной с помощью любого из методов: *ToList*, *ToArray* или *ToDictionary*. Для того, чтобы кэшировать значимый тип (value type), необходимо выполнить принудительный *boxing* (число можно для этого, например, преобразовать в строку).

Пример использования:

```
DBConnector cnn = new DBConnector();
List<Banner> banners = cnn.GetCachedEntity<List<Banner>>(pageAddress, 10,
GetPageBanners);
```

Где *GetPageBanners* – метод, возвращающий *List<Banner>*, который будет вызван только при заполнении кэша. Метод может быть определен как без параметров, так и с единственным строковым параметром. Во втором случае в метод будет передано значение ключа (параметр *key*).

From:

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

Permanent link:

<http://wiki.qpublishing.ru/doku.php?id=developer:linq:caching&rev=1339602519>

Last update: **2012/06/13 18:48**

