

LINQ-Abfrage in DataTable umwandeln

Problem

Seit C# 3.0 kann man ja via LINQ eine Datenbank abfragen. Eine schöne Sache. Ab und zu, für bestimmte Zwecke, benötigt man aber immer noch die Ergebnisse als klassische DataTable. Wie wandelt man nun das Ergebnis einer LINQ-Abfrage in eine DataTable um?

Hier die LINQ-Abfrage die mir mein Adressbuch ausliest:

```
1. DataContext context = new DataContext();
2. var query = from s in context.Personen
3.             orderby s.Nachname
4.             select s;
```

Lösung

Die Datensätze hat man nun in query. Man könnte jetzt nun mit foreach jeden einzelnen Datensatz abfragen und in eine DataTable ablegen. Es geht aber noch etwas einfacher:

[Linq2Database.cs](#)

```
1. using System;
2. using System.Data;
3. using System.Data.SqlClient;
4. using System.Linq;
5.
6. /// <summary>
7. /// Ergebnis einer LINQ-Abfrage in eine DataTable umwandeln.
8. /// </summary>
9. /// <param name="query">Abfrageergebnis</param>
10. /// <param name="context">Context</param>
11. /// <returns>Tabelle</returns>
12. public DataTable ConvertLinqToDatatable( IQueryable query,
13.     DataContext context )
14. {
15.     DataTable table = null;
16.     IDbCommand cmd = null;
17.     SqlDataAdapter adapter = null;
18.     try
19.     {
20.         // LINQ-Abfrage in Tabelle speichern
21.         cmd = context.GetCommand( query );
```

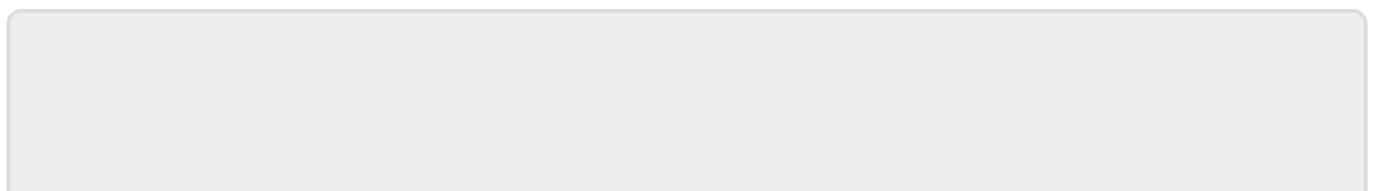
```
22.     adapter = new SqlDataAdapter();
23.     adapter.SelectCommand = (SqlCommand)cmd;
24.     table = new DataTable();
25.
26.     cmd.Connection.Open();
27.     adapter.FillSchema( table, SchemaType.Source );
28.     adapter.Fill( table );
29.
30.     return table;
31. }
32. catch
33. {
34.     throw;
35. }
36. finally
37. {
38.     if ( table != null ) table.Dispose();
39.     if ( adapter != null ) adapter.Dispose();
40.
41.     if ( cmd != null )
42.     {
43.         cmd.Connection.Close();
44.         cmd.Dispose();
45.     }
46. }
47. }
```

Somit einfach wie gewohnt eine LINQ-Abfrage durchführen und dann das Ergebnis mit der Funktion `ConvertLinqToDatatable()` umwandeln lassen:

[Linq2DatabaseExample.cs](#)

```
1. DataContext context = new DataContext();
2. var query = from s in context.Personen
3.             orderby s.Nachname
4.             select s;
5.
6. DataTable = ConvertLinqToDatatable( query, context );
```

C#



From:
<https://stefanjahn.de/> - **Stefans Wiki**

Permanent link:
https://stefanjahn.de/c-sharp/linq-abfrage_in_datatable_umwandeln

Last update: **Sat 05. Jun 2021, 01:14 Uhr**

