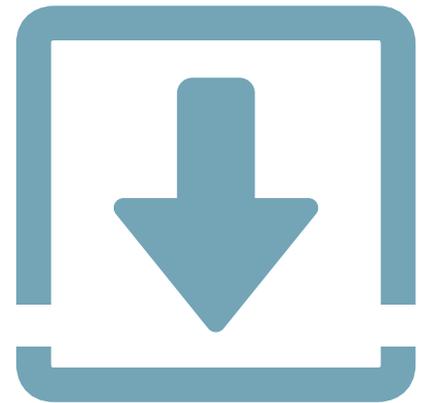


DOWNLOADS

Doppelte Portion Datenbanken

Eine Sammlung an NoSQL-Datenbanken und ein Fluent-Interface für RDBMS



Es soll sie ja noch geben: die Entwickler, die Entity Framework oder einen ähnlichen OR-Mapper nicht verwenden können oder dürfen. Das bedeutet dann aber, dass Abfragen an eine SQL-Datenbank normalerweise über SQL-Statements in String-Form abgesetzt werden. Schön ist nicht nur anders, sondern dieser Weg wird auch weder vom Compiler überwacht noch von IntelliSense unterstützt. Da kommt eine Bibliothek wie `SqlKata` gerade recht. Sie ermöglicht es, SQL-Abfragen über eine Fluent-Syntax zu formulieren. Die Suche nach dem Datensatz mit `id= 88` sieht so aus:

```
var query = new Query("Sections").Where("id", 88);
```

Die Query kann man sich dann als SQL-String ausgeben lassen. Die Bibliothek kann die Abfrage aber auch gleich ausführen und das Ergebnis anzeigen. In dem folgenden Beispiel sind beide Varianten enthalten:

```
using System;
using MySql.Data.MySqlClient;
using SqlKata;
using SqlKata.Compilers;
using SqlKata.Execution;

namespace FlowApp
{
    class Program
    {
        static void Main(string[] args)
        {
            var compiler = new MySqlCompiler();
```

● **MonoDevelop**

MonoDevelop ist eine integrierte Entwicklungsumgebung (IDE) für Mono unter Verwendung von Gtk#. Der Mono-Develop-Kern ist auch die Grundlage von Visual Studio für Mac.



● **Schwerpunkt NoSQL**

Der Aufstieg von NoSQL-Datenbanken begann vor rund zehn Jahren. Manche sangen damals schon ein Totenlied für die relationalen Datenbanksysteme (RDBMS). Doch beide Gruppen von Datenbanken haben ihre Vorteile. Kein festes Schema zu haben und damit leicht anpassbar an neue Gegebenheiten zu sein ist auf jeden Fall ein Vorzug der NoSQL-Datenbanken, von denen wir kleine und große für Sie zusammengestellt haben.



```
// Ausgabe des SQL-Strings
var query = new Query("Sections").Where("id", 88);
SqlResult sql = compiler.Compile(query);
Console.WriteLine(sql);

// direkte Abfrage an die Datenbank
var connection = new MySqlConnection(
    "Server=localhost; Database=dotnetprodata;
    Uid=dnuser; Pwd=*****;");
var db = new QueryFactory(connection, compiler);
var sections = db.Query("Sections")
    .WhereContains("name", "Front").First();
Console.WriteLine(sections);
}
}
```

Die Ausgabe auf der Konsole sieht dann so aus:

```
{DapperRow, id = '74', name = 'Frontend',
    remark = 'WPF, ASP.NET, AJAX, WinForms',
    order = '10', color = '#00A066'}
SELECT * FROM 'Sections' WHERE 'id' = 88
```

● **Schwerpunkt**

MongoDB

MongoDB ist eine universelle, dokumentenbasierte, verteilte Datenbank für Entwickler moderner Anwendungen und für die Cloud-Ära.

CouchDB

Apache CouchDB bietet Ihnen Zugang zu Ihren Daten, wo Sie ihn brauchen. Implementiert ist das CouchDB Replication Protocol in einer Vielzahl von Projekten und Produkten von Mobiltelefonen bis hin zu Webbrowsern.

NoSQL-Examples

Dies ist ein Repository mit Codebeispielen für die Verwendung der Oracle NoSQL Database.

LiteDB

LiteDB ist eine kleine und schnelle Embedded-NoSQL-Datenbank.

NoSQL-Data-Bases

Erkunden Sie NoSQL-Basics, hauptsächlich dokumentenbasierter und grafischer NoSQL-Datenbanken.

Couchbase Lite für .NET

Couchbase Lite ist eine einfache eingebettete NoSQL-Datenbank mit eingebauter Synchronisierung zu größeren Backendstrukturen, wie zum Beispiel Couchbase Server.

EasyLite

EasyLite ist eine Embedded-Open-Source-NoSQL-Datenbank basierend auf Unqlite.

KATE

JSON-basierter NoSQL-Dokumentenspeicher.

SSDB

SSDB ist ein hochperformanter Key-Value-Store und eine Alternative zu Redis. SSDB ist stabil, produktionsbereit und weit verbreitet in vielen Internetfirmen einschließlich Qihoo 360.

LedisDB

Ledisdb ist eine NoSQL-Datenbank, ähnlich wie Redis, geschrieben in Go. Sie unterstützt Datenstrukturen wie *kv*, *list*, *hash*, *zset*, *set* und verschiedene Datenbanken als Backends.

TinyDB

TinyDB ist eine dokumentenorientierte Datenbank. Sie ist in Python geschrieben und hat keine externen Abhängigkeiten. Ziel sind kleine Apps.

DensoDB

DensoDB ist eine NoSQL-Dokumenten-Datenbank, geschrieben für eine .NET-Umgebung in C#. Sie ist einfach und schnell.

● **Aktuelle Downloads**

Net-Core-DocX-HTML-To-PDF-Converter

.NET Core Library zum Erstellen von angepassten Reports basierend auf Word-DOCX- oder HTML-Dokumenten sowie zum Konvertieren in PDF.

Kinvey

Kinvey ist eine serverlose Anwendungs-Entwicklungsplattform und bietet Entwicklern Tools zum Erstellen robuster Multi-channel-Applikationen unter Zuhilfenahme eines Cloud-Backends und eines Frontend-SDK.

MonoDevelop

MonoDevelop ist eine integrierte Entwicklungsumgebung (IDE) für Mono unter Verwendung von Gtk#. Der MonoDevelop-Kern ist auch die Grundlage von Visual Studio für Mac.

FirestoreCoreSDK

FirestoreCoreSDK ist eine .NET Core Library zum Interagieren mit der Firestore-Datenbank, Cloud Messaging und Storage. Die Library sollte in serverseitigen Apps verwendet werden.

Dyalect

Dyalect ist eine dynamische Programmiersprache für die .NET-Core-Plattform. Sie ist klein, schnell und modern. Dyalect (oder abgekürzt Dy) ist in C# geschrieben und hat außer zu Standard-.NET-Core-Bibliotheken keine Abhängigkeiten. Das bedeutet, dass sie auf Windows, macOS und Linux laufen kann.

AWS Extensions für .NET CLI

Dieses Repository enthält AWS-Tool-Erweiterungen zum .NET-CLI. Die Tools helfen dabei, .NET-Core- sowie ASP.NET-Core-Anwendungen auf AWS-Diensten bereitzustellen.

.NET Performance

Dieses Repository enthält Benchmarks, die zum Performance-Testing aller .NET-Runtimes verwendet werden: .NET Core, Full .NET Framework, Mono und CoreRT.

DotNetHelper.ObjectToSql

DotNetHelper.ObjectToSql nimmt generische und dynamische Typen oder anonyme Objekte entgegen und konvertiert sie in SQL.

Simple File Logger

Ein sehr einfaches Datei-Logging-Framework, basierend auf dem NuGet-Paket Microsoft.Extensions.Logging.Abstractions. SimpleFileLogger schreibt Protokolleinträge in eine Datei. Sie können *MaxFileSizeBytes* und *MaxArchiveFileCount* spezifizieren, um zum Beispiel die Logdateigröße zu verwalten.

.NET Toolset

Diese Repo vereint die Tooling-Komponenten des .NET Core SDK (CLI Verbs, MSbuild, Compilers et cetera).