

How to convert csv to sql server with spreadsheet sdk in C# using ByteScout Barcode Suite

Learning is essential in computer world and the tutorial below will demonstrate how to convert csv to sql server with spreadsheet sdk in C#

These source code samples are assembled by their programming language and functions they apply. ByteScout Barcode Suite can convert csv to sql server with spreadsheet sdk. It can be applied from C#. ByteScout Barcode Suite is the set that includes three different SDK products to generate barcodes, read barcodes and read and write spreadsheets: Barcode SDK, Barcode Reader SDK and Spreadsheet SDK.

The SDK samples given below describe how to quickly make your application do convert csv to sql server with spreadsheet sdk in C# with the help of ByteScout Barcode Suite. This C# sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! Applying C# application mostly includes various stages of the software development so even if the functionality works please test it with your data and the production environment.

You can download free trial version of ByteScout Barcode Suite from our website to see and try many others source code samples for C#.

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about ByteScout Barcode Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Barcode Suite](#)

[Get Free API key for Web API](#)

[visit www.ByteScout.com](http://www.ByteScout.com)

Source Code Files:

```
using System;
using Bytescout.Spreadsheet;
using System.Data.SqlClient;

namespace ExportToSQLServer
{
    class Program
    {
        static void Main(string[] args)
        {
            try
            {
                // MODIFY THE CONNECTION STRING WITH YOUR CREDENTIALS!!!
                string connectionString = "Data Source=localhost;Initial Catalog=master";

                using (SqlConnection connection = new SqlConnection(connectionString))
                {
                    connection.Open();

                    // Drop test database if exists
                    ExecuteQueryWithoutResult(connection, "IF DB_ID ('CsvTests') IS NOT NULL DROP DATABASE CsvTests");
                    // Create empty database
                    ExecuteQueryWithoutResult(connection, "CREATE DATABASE CsvTests");
                    // Switch to created database
                    ExecuteQueryWithoutResult(connection, "USE CsvTests");
                    // Create a table for CSV data
                    ExecuteQueryWithoutResult(connection, "CREATE TABLE CsvTest (Name VARCHAR(50))");

                    // Load CSV document
                    using (Spreadsheet document = new Spreadsheet())
                    {
                        document.LoadFromFile("sample.csv", ";"); // ";" - delimiter
                        Worksheet worksheet = document.Workbook.Worksheets[0];

                        for (int row = 0; row <= worksheet.UsedRangeRowMax; row++)
                        {
                            String insertCommand = string.Format("INSERT CsvTest VALUES ('{0}')", worksheet.Cell(row, 0).Value);
                            ExecuteQueryWithoutResult(connection, insertCommand);
                        }
                    }

                    // Check the data successfully exported
                    using (SqlCommand command = new SqlCommand("SELECT * from CsvTest"))
                    {
                        SqlDataReader reader = command.ExecuteReader();

                        if (reader != null)
                        {
                            Console.WriteLine();
                            Console.WriteLine("Exported CSV data:");
                            Console.WriteLine();

                            while (reader.Read())
                            {
                                Console.WriteLine(reader.GetString(0));
                            }
                        }
                    }
                }
            }
            catch { }
        }
    }
}
```

```
        Console.WriteLine(String.Format("{0} | {1}", reader[0], reader[1]));
    }
}

Console.WriteLine();
Console.WriteLine("Press any key.");
Console.ReadKey();
}
}
catch (Exception ex)
{
    Console.WriteLine("Error: " + ex.Message);
    Console.ReadKey();
}
}

static void ExecuteQueryWithoutResult(SqlConnection connection, string query)
{
    using (SqlCommand command = new SqlCommand(query, connection))
    {
        command.ExecuteNonQuery();
    }
}
}
```

VIDEO

<https://www.youtube.com/watch?v=NEwNs2b9YN8>

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Barcode Suite Home Page](#)

[Explore ByteScout Barcode Suite Documentation](#)

[Explore Samples](#)

[Sign Up for ByteScout Barcode Suite Online Training](#)

ON-DEMAND REST WEB API

[Get Your API Key](#)

[Explore Web API Docs](#)

[Explore Web API Samples](#)

[visit www.ByteScout.com](http://www.ByteScout.com)

[visit www.PDF.co](http://www.PDF.co)

www.bytescout.com