How to convert xls to sql server with spreadsheet sdk in VB.NET using ByteScout Data Extraction Suite

Learn to code in VB.NET to convert xls to sql server with spreadsheet sdk with this step-by-step tutorial

The coding instructions are formulated to help you to try-out the features without the requirement to write your own code. ByteScout Data Extraction Suite is the bundle that includes three SDK tools for data extraction from PDF, scans, images and from spreadsheets: PDF Extractor SDK, Data Extraction SDK, Barcode Reader SDK. It can be applied to convert xls to sql server with spreadsheet sdk using VB.NET.

Want to quickly learn? This fast application programming interfaces of ByteScout Data Extraction Suite for VB.NET plus the guidelines and the code below will help you quickly learn how to convert xls to sql server with spreadsheet sdk. Follow the instructions from scratch to work and copy the VB.NET code. Complete and detailed tutorials and documentation are available along with installed ByteScout Data Extraction Suite if you'd like to learn more about the topic and the details of the API.

Our website gives trial version of ByteScout Data Extraction Suite for free. It also includes documentation and source code samples.

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Data Extraction Suite

Explore API Documentation

Get Free Training for ByteScout Data Extraction Suite

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

Program.vb

```vb
Imports System.IO
Imports Bytescout.Spreadsheet
Imports System.Data.SqlClient

Class Program
    Friend Shared Sub Main(args As String())
        Try
            ' MODIFY THE CONNECTION STRING WITH YOUR CREDENTIALS!!!
            Dim connectionString As String = "Data Source=localhost;Initial Catalog=mas

            Using connection As New SqlConnection(connectionString)
                connection.Open()

                ' Drop test database if exists
                ExecuteQueryWithoutResult(connection, "IF DB_ID ('XlsTests') IS NOT NU
                ' Create empty database
                ExecuteQueryWithoutResult(connection, "CREATE DATABASE XlsTests")
                ' Switch to created database
                ExecuteQueryWithoutResult(connection, "USE XlsTests")
                ' Create a table for XLS data
                ExecuteQueryWithoutResult(connection, "CREATE TABLE XlsTest (Name VARC

                ' Load XLS document
                Using document As New Spreadsheet()
                    document.LoadFromFile("SimpleReport.xls")
                    Dim worksheet As Worksheet = document.Workbook.Worksheets(0)

                    For row As Integer = 0 To worksheet.UsedRangeRowMax
                        Dim insertCommand As [String] = String.Format("INSERT XlsTest
                        ExecuteQueryWithoutResult(connection, insertCommand)
                    Next
                End Using

                ' Check the data successfully exported
                Using command As New SqlCommand("SELECT * from XlsTest", connection)
                    Dim reader As SqlDataReader = command.ExecuteReader()

                    If reader IsNot Nothing Then
                        Console.WriteLine()
                        Console.WriteLine("Exported XLS data:")
                        Console.WriteLine()

                        While reader.Read()
                            Console.WriteLine([String].Format("{0}  |  {1}", reader(0)
                        End While
                    End If
                End Using

                Console.WriteLine()
                Console.WriteLine("Press any key.")
                Console.ReadKey()
            End Using
        Catch ex As Exception
            Console.WriteLine("Error: " & ex.Message)
            Console.ReadKey()
```

```vb
            End Try
    End Sub

    Private Shared Sub ExecuteQueryWithoutResult(connection As SqlConnection, query As
        Using command As New SqlCommand(query, connection)
            command.ExecuteNonQuery()
        End Using
    End Sub
End Class
```

VIDEO

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

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Data Extraction Suite Home Page
Explore ByteScout Data Extraction Suite Documentation
Explore Samples
Sign Up for ByteScout Data Extraction Suite Online Training

ON-DEMAND REST WEB API

Get Your API Key
Explore Web API Docs
Explore Web API Samples

visit www.ByteScout.com

visit www.PDF.co