How to read number format in cell with spreadsheet sdk in C# using ByteScout Data Extraction Suite

Learning is essential in computer world and the tutorial below will demonstrate how to read number format in cell with spreadsheet sdk in C#

The sample shows instructions and algorithm of how to read number format in cell with spreadsheet sdk and how to make it run in your C# application. What is ByteScout Data Extraction Suite? It is the set that includes 3 SDK products for data extraction from PDF, scans, images and from spreadsheets: PDF Extractor SDK, Data Extraction SDK, Barcode Reader SDK. It can help you to read number format in cell with spreadsheet sdk in your C# application.

Want to save time? You will save a lot of time on writing and testing code as you may just take the C# code from ByteScout Data Extraction Suite for read number format in cell with spreadsheet sdk below and use it in your application. Just copy and paste the code into your C# application's code and follow the instructions. 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 Data Extraction Suite from our website with this and other source code samples for C#.

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.cs

```csharp
using System;
using System.Collections.Generic;
using System.Text;
using System.Diagnostics;
using Bytescout.Spreadsheet;
using Bytescout.Spreadsheet.Constants;
using System.IO;

namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            // Create new Spreadsheet
            Spreadsheet document = new Spreadsheet();
            document.LoadFromFile("Data.xls");

            // Get worksheet by name
            Worksheet worksheet = document.Workbook.Worksheets.ByName("Sample");

            // Check dates
            for (int i = 0; i < 8; i++)
            {
                for (int j = 0; j < 2; j++)
                {
                    // Set current cell
                    Cell currentCell = worksheet.Cell(i, j);

                    // Get format type

                    NumberFormatType formatType = currentCell.ValueDataTypeByNumberForm

                    // Write line
                    Console.Write("Cell({0}:{1}) type is {2}. Value : ", i, j, formatTy

                    switch (formatType)
                    {
                        case NumberFormatType.DateTime:
                            {
                                // Read datetime
                                DateTime date = currentCell.ValueAsDateTime;

                                // Write date to console output
                                Console.Write(date.ToString());
                            }
                            break;
                        case NumberFormatType.General:
                            {
                                // Write value to console output
                                Console.Write(currentCell.Value);
                            }
                            break;
```

```
                    }

                    Console.WriteLine();

                }
            }

            // Close document
            document.Close();
            Console.ReadKey();
        }
    }
}
```

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