

How to extract text from foldable brochure booklet in C# with ByteScout Document Parser SDK

ByteScout Document Parser SDK: the customizable data extraction platform for batch data extraction from documents. Relies on special templates that can be created with no special technical skills required. Supports millions of documents as input and designed to handle multiple threads. Can output data as JSON, CSV, XML or custom format.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Document Parser SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Document Parser SDK](#)

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

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

Source Code Files:

ExtractTextFromFoldableBrochure.csproj

```
<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="15.0" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props" Condition="Exists('$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props')>
  <PropertyGroup>
    <Configuration Condition="'$(Configuration)' == ''">Debug</Configuration>
    <Platform Condition="'$(Platform)' == ''">AnyCPU</Platform>
    <ProjectGuid>{A73776C6-D2B2-4E37-B852-06C6454D1B5B}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <RootNamespace>ExtractTextFromFoldableBrochure</RootNamespace>
    <AssemblyName>ExtractTextFromFoldableBrochure</AssemblyName>
    <TargetFrameworkVersion>v4.0</TargetFrameworkVersion>
    <FileAlignment>512</FileAlignment>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
```

```

<PlatformTarget>AnyCPU</PlatformTarget>
<DebugSymbols>true</DebugSymbols>
<DebugType>full</DebugType>
<Optimize>false</Optimize>
<OutputPath>bin\Debug\</OutputPath>
<DefineConstants>DEBUG;TRACE</DefineConstants>
<ErrorReport>prompt</ErrorReport>
<WarningLevel>4</WarningLevel>
</PropertyGroup>
<PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
  <PlatformTarget>AnyCPU</PlatformTarget>
  <DebugType>pdbonly</DebugType>
  <Optimize>true</Optimize>
  <OutputPath>bin\Release\</OutputPath>
  <DefineConstants>TRACE</DefineConstants>
  <ErrorReport>prompt</ErrorReport>
  <WarningLevel>4</WarningLevel>
</PropertyGroup>
<ItemGroup>
  <Reference Include="ByteScout.DocumentParser, Version=1.0.0.100, Culture=neutral, PublicKeyToken=f7dd1bd9c"
    <SpecificVersion>False</SpecificVersion>
    <HintPath>c:\Program Files\ByteScout Document Parser SDK\net40\ByteScout.DocumentParser.dll</HintPath>
  </Reference>
  <Reference Include="Newtonsoft.Json, Version=12.0.0.0, Culture=neutral, PublicKeyToken=30ad4fe6b2a6aeed, p"
    <HintPath>packages\Newtonsoft.Json.12.0.3\lib\net40\Newtonsoft.Json.dll</HintPath>
  </Reference>
  <Reference Include="System" />
  <Reference Include="System.Core" />
  <Reference Include="System.Xml.Linq" />
  <Reference Include="System.Data" />
  <Reference Include="System.Xml" />
</ItemGroup>
<ItemGroup>
  <Compile Include="Program.cs" />
</ItemGroup>
<ItemGroup>
  <None Include="..\..\SampleFoldable.pdf">
    <Link>SampleFoldable.pdf</Link>
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </None>
  <None Include="SampleFoldable.yml">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </None>
  <None Include="packages.config" />
</ItemGroup>
<Import Project="$(MSBuildToolsPath)\Microsoft.CSharp.targets" />
</Project>

```

ExtractTextFromFoldableBrochure.sln

```

Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 15
VisualStudioVersion = 15.0.27703.2018
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "ExtractTextFromFoldableBrochure", "ExtractTextFromF
EndProject
Global
  GlobalSection(SolutionConfigurationPlatforms) = preSolution
    Debug|Any CPU = Debug|Any CPU
    Release|Any CPU = Release|Any CPU
  EndGlobalSection
  GlobalSection(ProjectConfigurationPlatforms) = postSolution
    {A73776C6-D2B2-4E37-B852-06C6454D1B5B}.Debug|Any CPU.ActiveCfg = Debug|Any CPU
    {A73776C6-D2B2-4E37-B852-06C6454D1B5B}.Debug|Any CPU.Build.0 = Debug|Any CPU
    {A73776C6-D2B2-4E37-B852-06C6454D1B5B}.Release|Any CPU.ActiveCfg = Release|Any CPU
    {A73776C6-D2B2-4E37-B852-06C6454D1B5B}.Release|Any CPU.Build.0 = Release|Any CPU

```

```

EndGlobalSection
GlobalSection(SolutionProperties) = preSolution
    HideSolutionNode = FALSE
EndGlobalSection
GlobalSection(ExtensibilityGlobals) = postSolution
    SolutionGuid = {7E6DAA79-020B-421A-844A-5FE05EFC9B15}
EndGlobalSection
EndGlobal

```

Program.cs

```

using ByteScout.DocumentParser;
using Newtonsoft.Json.Linq;
using System;
using System.Diagnostics;
using System.IO;
using System.Text;

namespace ExtractTextFromFoldableBrochure
{
    class Program
    {
        static void Main(string[] args)
        {
            // Sample document containing foldable PDF
            string sampleDocument = @".\SampleFoldable.pdf";

            // Sample template
            string sampleTemplate = @".\SampleFoldable.yml";

            // Output
            var resFile = "result.txt";
            var sOutput = new StringBuilder(string.Empty);

            // Create DocumentParser instance
            using (DocumentParser documentParser = new DocumentParser("demo", "demo"))
            {
                // Add sample template
                documentParser.AddTemplate(sampleTemplate);

                // Parse document data in JSON format
                string jsonString = documentParser.ParseDocument(sampleDocument, OutputFormat.JSON);

                // Parse json to foldable structure
                var parsedRes = ParseJsonToFoldableStructure(jsonString);

                // Append all parts to return output
                sOutput.AppendLine(parsedRes.Part1);
                sOutput.AppendLine(parsedRes.Part2);
                sOutput.AppendLine(parsedRes.Part3);

                // Write output file
                File.WriteAllText(resFile, sOutput.ToString());

                Console.WriteLine($"Extracted text saved to {resFile}");

                // Open result file with default application
                Process.Start(resFile);
            }

            Console.WriteLine();
            Console.WriteLine("Press any key to continue...");
            Console.ReadLine();
        }
    }

    /// <summary>

```

```

/// Parse Json to Foldable Structure
/// </summary>
static FoldableStructure ParseJsonToFoldableStructure(string jsonData)
{
    // Get Object data from input file
    JObject jsonObj = JObject.Parse(jsonData);

    var oRet = new FoldableStructure();
    oRet.Part1 = Convert.ToString(jsonObj["fields"]["part1"]["value"]);
    oRet.Part2 = Convert.ToString(jsonObj["fields"]["part2"]["value"]);
    oRet.Part3 = Convert.ToString(jsonObj["fields"]["part3"]["value"]);

    return oRet;
}

class FoldableStructure
{
    public string Part1 { get; set; }
    public string Part2 { get; set; }
    public string Part3 { get; set; }
}

```

SampleFoldable.yml

```

templateVersion: 3
templatePriority: 0
sourceId: Untitled document kind
detectionRules:
  keywords: []
fields:
  Part1:
    type: rectangle
    rectangle:
      - 7.5
      - 33.75
      - 244.5
      - 353.25
    pageIndex: 0
  Part2:
    type: rectangle
    rectangle:
      - 273.75
      - 201.75
      - 247.5
      - 198
    pageIndex: 0
  Part3:
    type: rectangle
    rectangle:
      - 537.75
      - 27
      - 246
      - 268.5
    pageIndex: 0

```

packages.config

```
<?xml version="1.0" encoding="utf-8"?>
<packages>
  <package id="Newtonsoft.Json" version="12.0.3" targetFramework="net40" />
</packages>
```

VIDEO

<https://www.youtube.com/watch?v=MG5FfTWWSVE>

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Document Parser SDK Home Page](#)
[Explore ByteScout Document Parser SDK Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout Document Parser SDK 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