**How to load the HTML converted document into .NET Core PDF Grid Cell**

The Syncfusion Essential® PDF is a feature-rich and high performance [**.NET PDF library**](https://www.syncfusion.com/document-processing/pdf-framework/net) used to create, read, and edit PDF documents programmatically without Adobe dependencies. Using this library, you can load the HTML converted document into PDF document Grid Cell using C#.

**Steps to load the HTML converted document into PDF document Grid programmatically:**

1. Create a new console application project.A screenshot of a computer

   AI-generated content may be incorrect.
2. Install the **[Syncfusion.HtmlToPdfConverter.Net.Windows](https://www.nuget.org/packages/Syncfusion.HtmlToPdfConverter.Net.Windows" \t "_blank)** NuGet package as a reference to your console application from [**Nuget.org**](https://www.nuget.org/).

A screenshot of a computer

AI-generated content may be incorrect.

1. Include the following namespaces in the Program.cs file.

**C#**

using **Syncfusion**.Drawing;

using **Syncfusion**.HtmlConverter;

using **Syncfusion**.Pdf;

using **Syncfusion**.Pdf.Graphics;

using **Syncfusion**.Pdf.Grid;

using **Syncfusion**.Pdf.Parsing;

1. Use the following code sample in Program.cs to load the HTML converted document into PDF document Grid.

**C#**

//Initialize the HTML to PDF converter.

**HtmlToPdfConverter** htmlConverter = **new** **HtmlToPdfConverter**();

//Initialize the blink converter settings

**BlinkConverterSettings** blinkConverterSettings = **new** **BlinkConverterSettings**();

//Set blink viewport size

blinkConverterSettings.ViewPortSize = **new** **Size**(612, 792);

//Assign Blink converter settings to HTML converter

htmlConverter.ConverterSettings = blinkConverterSettings;

string htmlContent = **File**.**ReadAllText**(**Path**.**GetFullPath**("../../../sample.html"));

string baseURL = "";

//Convert URL to PDF

**PdfDocument** document = htmlConverter.**Convert**(htmlContent,baseURL);

**MemoryStream** memoryStream = **new** **MemoryStream**();

document.**Save**(memoryStream);

memoryStream.Position = 0;

//Close the document

document.**Close**(true);

//Load the converted document

**PdfLoadedDocument** doc = **new** **PdfLoadedDocument**(memoryStream);

//Get the first page from a document.

**PdfLoadedPage** page = doc.Pages[0] **as** **PdfLoadedPage**;

//Create the page as template

**PdfTemplate** pdfTemplate= page.**CreateTemplate**();

//Create a new PDF document.

**PdfDocument** pdfDocument = **new** **PdfDocument**();

//Create the page.

**PdfPage** pdfPage = pdfDocument.Pages.**Add**();

//Create a new PdfGrid.

**PdfGrid** pdfGrid = **new** **PdfGrid**();

pdfGrid.Columns.**Add**(3);

pdfGrid.Columns[0].Width = 100;

pdfGrid.Columns[1].Width = 100;

pdfGrid.Columns[2].Width = 100;

pdfGrid.Rows.**Add**();

pdfGrid.Rows.**Add**();

pdfGrid.Rows.**Add**();

pdfGrid.Rows[0].Height = 100;

pdfGrid.Rows[1].Height = 100;

pdfGrid.Rows[2].Height = 100;

pdfGrid.Rows[0].Cells[0].Value = "S.No";

pdfGrid.Rows[0].Cells[1].Value = "Name";

pdfGrid.Rows[0].Cells[2].Value = "Values";

//Call the event handler to draw the HTML converted document in a particular cell.

pdfGrid.BeginCellLayout += **PdfGrid**\_BeginCellLayout;

//Draw a grid to the page of a PDF document.

pdfGrid.**Draw**(pdfPage, **new** **PointF**(10, 250));

//Draw a grid to the page of a PDF document

**FileStream** fileStream = **new** **FileStream**("HTML-to-PDF.pdf", **FileMode**.Create, **FileAccess**.ReadWrite);

//Save and close the PDF document.

pdfDocument.**Save**(fileStream);

pdfDocument.**Close**(true);

**void** **PdfGrid**\_BeginCellLayout(object sender, **PdfGridBeginCellLayoutEventArgs** args)

{

**if** ( !args.IsHeaderRow)

{

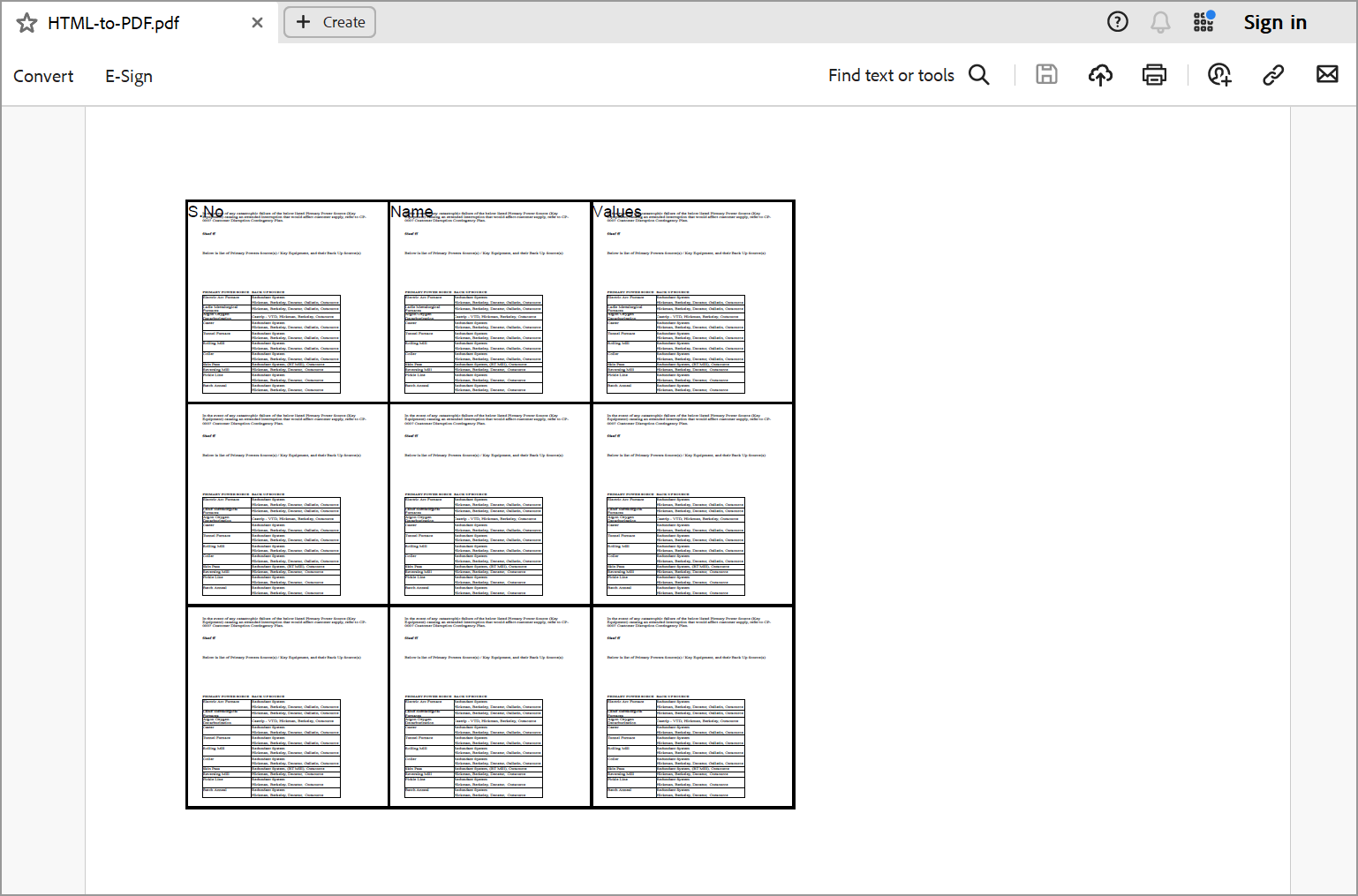
//Draw the converted document in the cell graphics

args.Graphics.**DrawPdfTemplate**(pdfTemplate, **new** **PointF**(args.Bounds.X, args.Bounds.Y) , **new** **SizeF**(args.Bounds.Width, args.Bounds.Height));

}

}

A complete working sample can be downloaded from [**HTMLtoPDF\_PdfGridCell.zip**](https://www.syncfusion.com/downloads/support/directtrac/general/ze/HTMLtoPDF_PdfGridCell-343019131.zip)

By executing the program, you will get the PDF document as follows.

Take a moment to peruse the [**documentation**](https://help.syncfusion.com/file-formats/pdf/converting-html-to-pdf), where you can find converting HTML pages to PDF document along with respective customization options and features.

Refer [**here**](https://www.syncfusion.com/document-processing/pdf-framework/net)to explore the rich set of Syncfusion Essential® PDF features.

**Conclusion**  
I hope you enjoyed learning about how to load the HTML converted document into .NET Core PDF Grid Cell.

You can refer to our [**ASP.NET Core PDF feature tour**](https://www.syncfusion.com/document-processing/pdf-framework/net-core) page to know about its other groundbreaking feature representations and [**documentation**](https://ej2.syncfusion.com/file-formats/pdf/getting-started), and how to quickly get started for configuration specifications. You can also explore our [**ASP.NET Core PDF example**](https://ej2.syncfusion.com/aspnetcore/Pdf/SalesInvoice#/bootstrap5) to understand how to create and manipulate data.

For current customers, you can check out our components from the [**License and Downloads page**](https://www.syncfusion.com/sales/teamlicense). If you are new to Syncfusion®, you can try our 30-day [**free trial**](https://www.syncfusion.com/downloads/aspnetcore-js2) to check out our other controls.

If you have any queries or require clarifications, please let us know in the comments section below. You can also contact us through our [**support forums**](https://www.syncfusion.com/forums/), [**Direct-Trac**](https://support.syncfusion.com/create), or [**feedback portal**](https://www.syncfusion.com/feedback/aspnet-core?control=pdf). We are always happy to assist you!