**Embed Unicode Text in PDF TextBox Fields Using .NET Core**

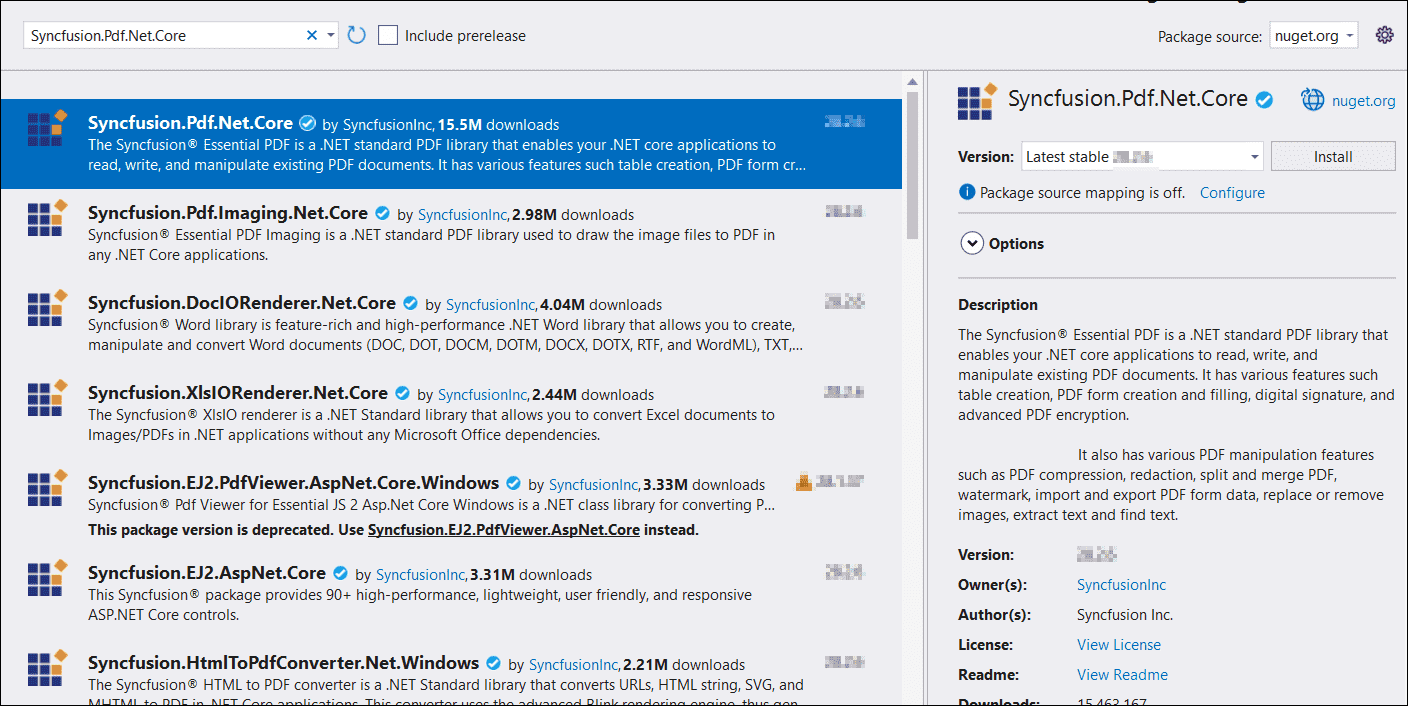
The Syncfusion Essential® PDF is a [**.NET PDF library**](https://www.syncfusion.com/document-processing/pdf-framework/net) used to create, read, and edit PDF documents. This guide will walk you through adding Unicode text to text box fields within a PDF document using C#.

**Steps to Add Unicode Text in PDF TextBox Fields**

1. **Set Up Your Project**: Create a new console application project in your development environment.A screenshot of a computer

AI-generated content may be incorrect.

2. **Install Syncfusion Package**: Add the [Syncfusion.Pdf.Net.Core](https://www.nuget.org/packages/Syncfusion.Pdf.Net.Core/) package from NuGet to your project.



3. **Include Required Namespaces**: Add these namespaces to your **Program.cs**.

**C#**

|  |
| --- |
| using **Syncfusion**.Pdf.Graphics;  using **Syncfusion**.Pdf.Parsing; |

4. **Implement Unicode Text in TextBox Fields**: Use the following code to integrate Unicode text within your PDF text box fields

**C#**

|  |
| --- |
| // Create a new PDF document.  FileStream docStream = new FileStream("Input.pdf", FileMode.Open, FileAccess.Read);  PdfLoadedDocument loadedDocument = new PdfLoadedDocument(docStream);  // Get the loaded form.  PdfLoadedForm loadedForm = loadedDocument.Form;  // Load the Unicode font.  FileStream fontStream = new FileStream("arial.ttf", FileMode.Open, FileAccess.Read);  PdfTrueTypeFont unicodeFont = new PdfTrueTypeFont(fontStream, 10);  // Iterate through each field in the form and set text for TextBox fields.  for (int i = 0; i < loadedForm.Fields.Count; i++)  {  PdfLoadedTextBoxField field = loadedForm.Fields[i] as PdfLoadedTextBoxField;  if (field is PdfLoadedTextBoxField)  {  field.Text = "ä, ü, ï, ö, ë, ÿ";  field.Font = unicodeFont;  }  }  // Disable the default appearance for the form fields.  loadedDocument.Form.SetDefaultAppearance(false);  // Save the modified document to a memory stream.  MemoryStream ms = new MemoryStream();  loadedDocument.Save(ms);  // Write the contents of the memory stream to a PDF file.  File.WriteAllBytes("output.pdf", ms.ToArray());  // Close the document.  loadedDocument.Close(true); |

A complete working sample can be downloaded from [**Draw\_unicode\_text.zip**](https://support.syncfusion.com/agent/)

By executing the program, the output PDF document will be generated as shown below.A screenshot of a computer

AI-generated content may be incorrect.

Take a moment to explore the documentation on [working with text](https://help.syncfusion.com/file-formats/pdf/working-with-text), where you can find additional options such as drawing right-to-left text, multi-column text, and using TrueType fonts, standard fonts, and CJK fonts.

**Conclusion**

I hope you enjoyed learning about how to add Unicode text in PDF text box fields using C#.

You can refer to our [**ASP.NET Core PDF**](https://www.syncfusion.com/document-processing/pdf-framework/net-core) feature tour page to know about its other groundbreaking feature representations and [**documentation**](https://help.syncfusion.com/aspnet-core/pdf/getting-started), and how to quickly get started for configuration specifications. You can also explore our [**ASP.NET Core PDF example**](https://www.syncfusion.com/demos/fileformats/pdf-library) to understand how to create and manipulate data in the .NET PDF.

For current customers, you can check out our Document processing libraries from the [**License and Downloads**](https://www.syncfusion.com/account/downloads) page. 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 ASP.NET Core PDF and other .NET Core controls.