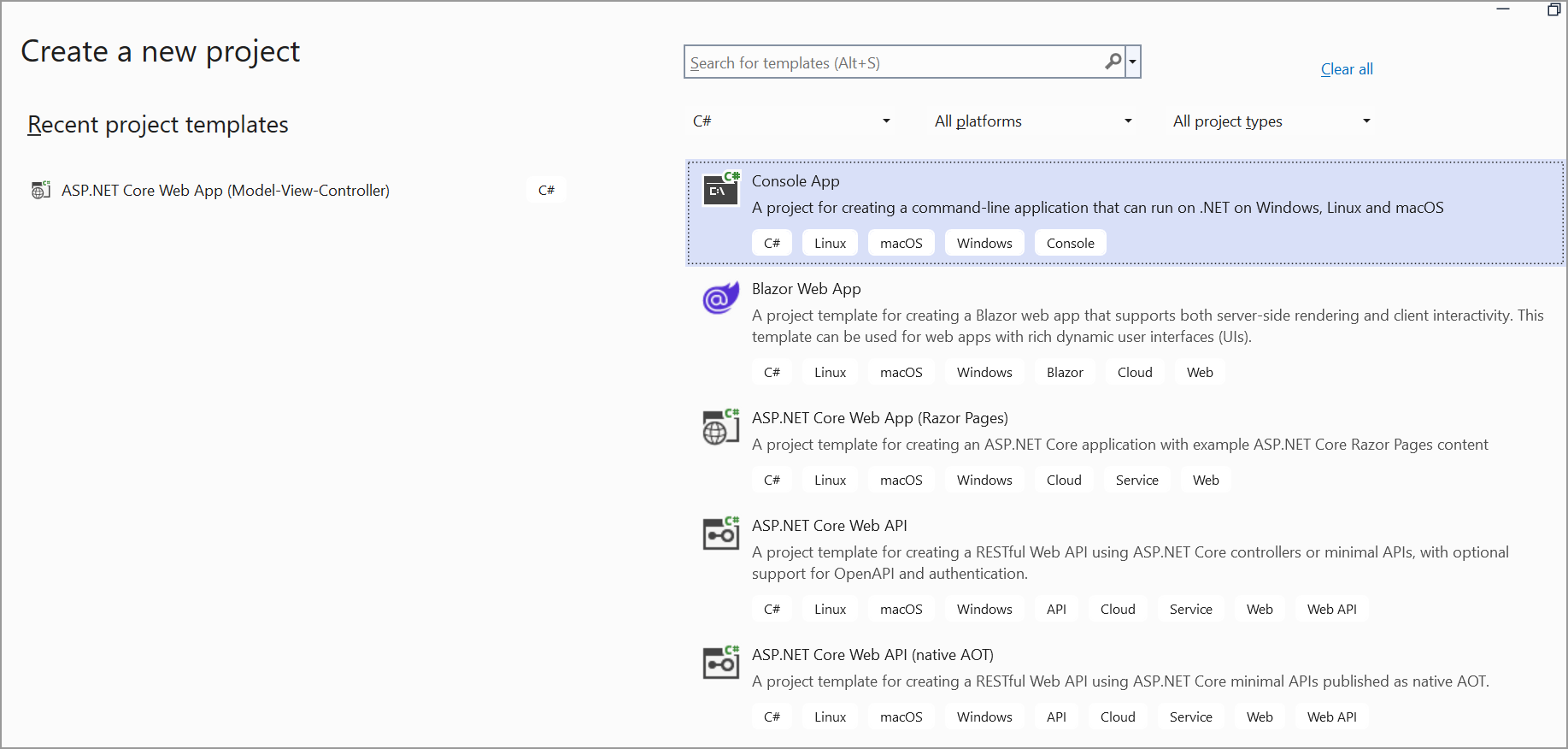
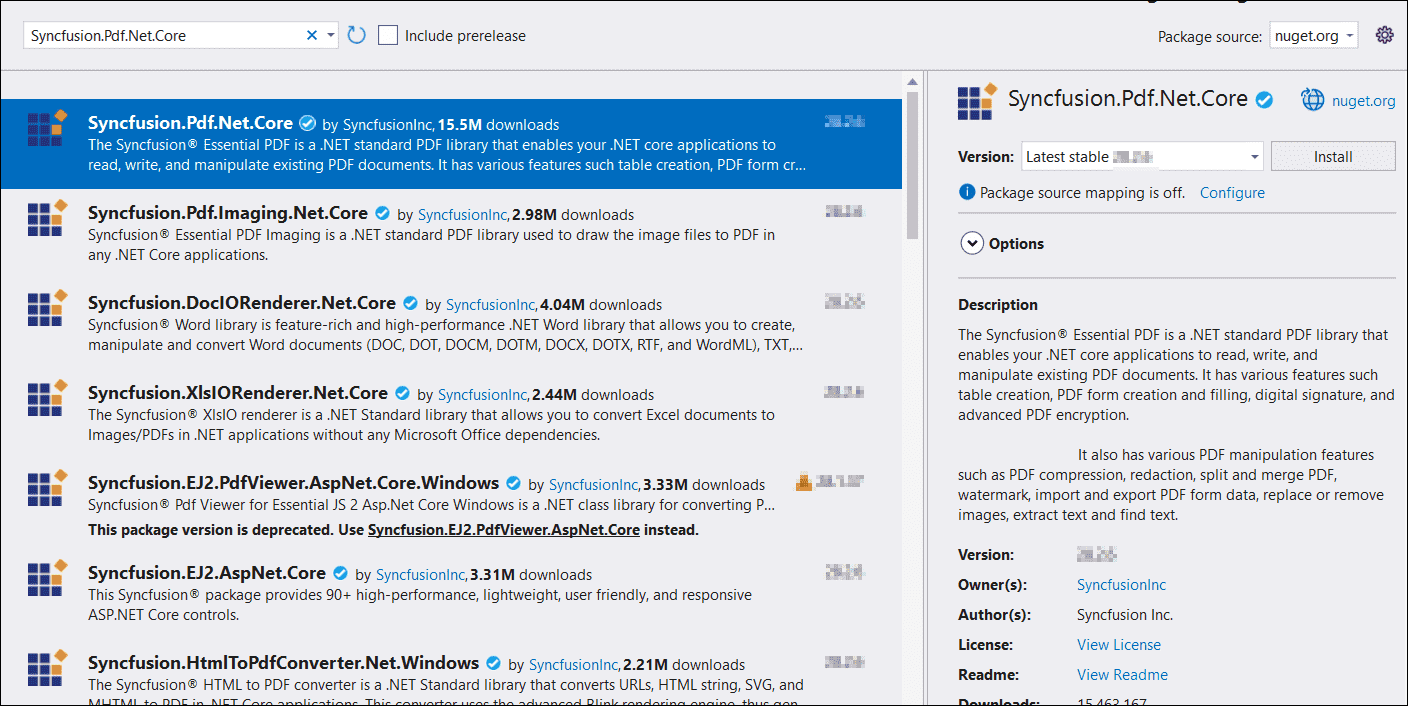
**Adding Page Labels When Importing Pages Between PDFs**

Syncfusion Essential® PDF offers a powerful and high-performance [**.NET PDF library**](https://www.syncfusion.com/document-processing/pdf-framework/net) for creating, reading, and editing PDF documents without Adobe dependencies. This tutorial demonstrates how to add page labels while importing pages from one PDF to another using C#.

**Steps to add page labels while importing pages from one PDF to another programmatically:**

**1.Create a Console Application**: Set up a new console application project.

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

3. **Include Required Namespaces** Add the following namespaces **in Program.cs.**

**C#**

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

4. **Implementation**: Use the following C# code to import pages and add page labels.  
**C#**

|  |
| --- |
| // Open the first PDF document (Barcode.pdf).  using (FileStream inputFileStream = new FileStream(@"Barcode.pdf", FileMode.Open, FileAccess.Read, FileShare.ReadWrite))  {  // Load and process the PDF document.  PdfLoadedDocument loadedDocument = new PdfLoadedDocument(inputFileStream);  PdfDocument document = new PdfDocument();  // Add pages from the first document  AddPagesWithLabels(loadedDocument, document, PdfNumberStyle.UpperRoman);  // Open the second PDF document (Invoice.pdf).  loadedDocument = new PdfLoadedDocument(new FileStream(@"Invoice.pdf", FileMode.Open));    // Add pages from the second document  AddPagesWithLabels(loadedDocument, document, PdfNumberStyle.Numeric);  // Save the final PDF document.  using (MemoryStream stream = new MemoryStream())  {  document.Save(stream);  File.WriteAllBytes(@"Output.pdf", stream.ToArray());  }  document.Close(true);  }  // Helper function to add pages with labels  static void AddPagesWithLabels(PdfLoadedDocument loadedDocument, PdfDocument document, PdfNumberStyle numberStyle)  {  PdfSection section = document.Sections.Add();  section.PageLabel = new PdfPageLabel();  section.PageLabel.StartNumber = 1;  section.PageLabel.NumberStyle = numberStyle;  section.PageSettings.Margins.All = 0;  for (int i = 0; i < loadedDocument.Pages.Count; i++)  {  PdfPage page = section.Pages.Add();  PdfTemplate template = loadedDocument.Pages[i].CreateTemplate();  page.Graphics.DrawPdfTemplate(template, PointF.Empty, template.Size);  }  loadedDocument.Close(true);  } |

A complete working sample can be downloaded: [**page\_labels\_while\_importing\_pages\_from\_one\_PDF\_to\_another.zip**](https://www.syncfusion.com/downloads/support/directtrac/general/ze/Page_labels_while_importing_pages-1777491703.zip)

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 for [working with pages](https://help.syncfusion.com/document-processing/pdf/pdf-library/net/working-with-pages), where you'll find additional options such as inserting, removing, and rearranging pages in a PDF document, adding margins, and importing pages from an existing PDF.

**Conclusion**

I hope you enjoyed learning about how to add page labels while importing pages from one PDF to another**.**

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.