File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\CustomFontResolver.cs

```
using PdfSharp.Fonts;
using System;
using System.IO;
using System.Reflection;

namespace GetCodeAsPdf
{
public class CustomFontResolver : IFontResolver
{
public static string DefaultFontName => "Arial";

public byte[] GetFont(string faceName)
{
switch (faceName)
{
case "Arial":
return LoadFontData("GetCodeAsPdf.arial.ttf"); // Replace with your actual namespace
default:
#pragma warning disable CS8603 // Possible null reference return.
return null;
#pragma warning restore CS8603 // Possible null reference return.
}
}

public FontResolverInfo ResolveTypeface(string familyName, bool isBold, bool isItalic)
{
return new FontResolverInfo("Arial");
}

private static byte[] LoadFontData(string name)
{
using Stream? stream = Assembly.GetExecutingAssembly().GetManifestResourceStream(name) ??
throw new FileNotFoundException($"Resource not found: {name}");
byte[] data = new byte[stream.Length];
stream.Read(data, 0, data.Length);
return data;
}
}
}
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\Program.cs

```
using MigraDoc.DocumentObjectModel;
using MigraDoc.Rendering;
using PdfSharp.Fonts;
using System.Text;

namespace GetCodeAsPdf
{
class Program
{
static void Main()
{
GlobalFontSettings.FontResolver = new CustomFontResolver();

string currentDirectory = Directory.GetCurrentDirectory();
string? solutionFile = Directory.GetFiles(currentDirectory, "*.sln").FirstOrDefault();

if (solutionFile == null)
{
Console.WriteLine("No .sln file found in the current directory.");
return;
}

string solutionName = Path.GetFileNameWithoutExtension(solutionFile);
string outputFile = Path.Combine(currentDirectory, $"{solutionName}.pdf");

StringBuilder sb = new();
string[] fileExtensions = { "*.cs", "*.cshtml", "*.js", "*.css" }; // Add more if needed

foreach (string ext in fileExtensions)
{
foreach (string file in Directory.EnumerateFiles(currentDirectory, ext, SearchOption.AllDirectories))
{
sb.AppendLine($"File: {file}");
sb.AppendLine(File.ReadAllText(file));
sb.AppendLine("\n\n"); // Adding some space between files
}
}

CreatePdf(sb.ToString(), outputFile);
}

static void CreatePdf(string content, string outputPath)
{
Document document = new();
Section section = document.AddSection();
Paragraph paragraph = section.AddParagraph();
paragraph.Format.Font.Name = "Arial";
paragraph.Format.Font.Size = 10;
paragraph.AddText(content);

PdfDocumentRenderer renderer = new()
{
Document = document
};
renderer.RenderDocument();
renderer.PdfDocument.Save(outputPath);

Console.WriteLine($"PDF saved to {outputPath}");
}
}
}
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Debug\net7.0\.NETCoreApp,Version=v7.0.AssemblyAttributes.cs

```
// <autogenerated />
using System;
using System.Reflection;
[assembly:
global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v7.0",
FrameworkDisplayName = ".NET 7.0")]
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Debug\net7.0\GetCodeAsPdf.AssemblyInfo.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
// This code was generated by a tool.
// Runtime Version:4.0.30319.42000
//
// Changes to this file may cause incorrect behavior and will be lost if
// the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System;
using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyConfigurationAttribute("Debug")]
[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]
[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]
[assembly: System.Reflection.AssemblyProductAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyTitleAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generated by the MSBuild WriteCodeFragment class.
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Debug\net7.0\GetCodeAsPdf.GlobalUsings.g.cs

```
// <auto-generated/>
global using global::System;
global using global::System.Collections.Generic;
global using global::System.IO;
global using global::System.Linq;
global using global::System.Net.Http;
global using global::System.Threading;
global using global::System.Threading.Tasks;
```

File:
C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Release\net7.0\.NETCoreApp,Version=v7.0.AssemblyAttributes.cs

```
// <autogenerated />
using System;
using System.Reflection;
[assembly:
global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v7.0",
FrameworkDisplayName = ".NET 7.0")]
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Release\net7.0\GetCodeAsPdf.AssemblyInfo.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
// This code was generated by a tool.
// Runtime Version:4.0.30319.42000
//
// Changes to this file may cause incorrect behavior and will be lost if
// the code is regenerated.
// </auto-generated>
```

```
//------------------------------------------------------------------------------

using System;
using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]
[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]
[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]
[assembly: System.Reflection.AssemblyProductAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyTitleAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generated by the MSBuild WriteCodeFragment class.
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Release\net7.0\GetCodeAsPdf.GlobalUsings.g.cs

```
// <auto-generated/>
global using global::System;
global using global::System.Collections.Generic;
global using global::System.IO;
global using global::System.Linq;
global using global::System.Net.Http;
global using global::System.Threading;
global using global::System.Threading.Tasks;
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Release\net7.0\win-x64\.NETCoreApp,Version=v7.0.AssemblyAttributes.cs

```
// <autogenerated />
using System;
using System.Reflection;
[assembly:
global::System.Runtime.Versioning.TargetFrameworkAttribute(".NETCoreApp,Version=v7.0",
FrameworkDisplayName = ".NET 7.0")]
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Release\net7.0\win-x64\GetCodeAsPdf.AssemblyInfo.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
// This code was generated by a tool.
//
// Changes to this file may cause incorrect behavior and will be lost if
// the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

using System;
using System.Reflection;

[assembly: System.Reflection.AssemblyCompanyAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyConfigurationAttribute("Release")]
[assembly: System.Reflection.AssemblyFileVersionAttribute("1.0.0.0")]
[assembly: System.Reflection.AssemblyInformationalVersionAttribute("1.0.0")]
```

```
[assembly: System.Reflection.AssemblyProductAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyTitleAttribute("GetCodeAsPdf")]
[assembly: System.Reflection.AssemblyVersionAttribute("1.0.0.0")]

// Generated by the MSBuild WriteCodeFragment class.
```

File: C:\Apps\GetCodeAsPdf\GetCodeAsPdf\obj\Release\net7.0\win-x64\GetCodeAsPdf.GlobalUsings.g.cs

```
// <auto-generated/>
global using global::System;
global using global::System.Collections.Generic;
global using global::System.IO;
global using global::System.Linq;
global using global::System.Net.Http;
global using global::System.Threading;
global using global::System.Threading.Tasks;
```