Quick Start

Build your technical documentation site with docfx. Converts .NET assembly, XML code comment, REST API Swagger files and markdown into rendered HTML pages, JSON model or PDF files.

Create a New Website

In this section we will build a simple documentation site on your local machine.

Prerequisites

- Familiarity with the command line
- Install .NET SDK 6.0 or higher

Make sure you have .NET SDK installed, then open a terminal and enter the following command to install the latest docfx:

```
dotnet tool update -g docfx
```

To create a new docset, run:

```
docfx init --quiet
```

This command creates a new docset under the `docfx_project` directory. To build the docset, run:

```
docfx docfx_project/docfx.json --serve
```

Now you can preview the website on http://localhost:8080.

To preview your local changes, save changes then run this command in a new terminal to rebuild the website:

```
docfx docfx_project/docfx.json
```

Publish to GitHub Pages

Docfx produces static HTML files under the `_site` folder ready for publishing to any static site hosting servers.

To publish to GitHub Pages:

1. Enable GitHub Pages.
2. Upload `_site` folder to GitHub Pages using GitHub actions.

This example uses `peaceiris/actions-gh-pages` to publish to the `gh-pages` branch:

```yaml
# Your GitHub workflow file under .github/workflows/

jobs:
  publish-docs:
    runs-on: ubuntu-latest
    steps:
    - name: Chekout
      uses: actions/checkout@v3
    - name: Dotnet Setup
      uses: actions/setup-dotnet@v3
      with:
        dotnet-version: 7.x

    - run: dotnet tool update -g docfx
    - run: docfx docfx_project/docfx.json

    - name: Deploy
      uses: peaceiris/actions-gh-pages@v3
      with:
        github_token: ${{ secrets.GITHUB_TOKEN }}
        publish_dir: docs/_site
```

Use the NuGet Library

You can also use docfx as a NuGet library:

```
<PackageReference Include="Microsoft.DocAsCode.App" Version="2.60.0" />
```

Then build a docset using:

```
await Microsoft.DocAsCode.Docset.Build("docfx.json");
```

See API References for additional APIs.

Next Steps

- Write Articles
- Organize Contents

- Configure Website
- Add .NET API Docs