


1

# Programska dokumentacija

2019/20.16

### ➤ XML dokumentacijski komentari

- omogućavaju pisanje dokumentacije koja se odnosi na programski kod
- označavaju se trostrukom kosom crtom : ///
- dodaju se ispred programskog bloka na koji se odnose
- sadrže posebne XML oznake koje opisuju dokumentirani kod
- Primjer:  Web \ Firma.Mvc \ Extensions \ ExceptionExtensions.cs

```
///
/// Razred sa proširenjima za iznimke
/// </summary>
public static class ExceptionExtensions {
    ///
    /// Vraća sadržaj poruka cijele hijerarhije neke iznimke.
    /// </summary>
    ///
```

# Xml datoteka s programskom dokumentacijom

3

- /doc opcija prevoditelja prilikom prevođenja uključuje traženje XML komentara u kodu te kreiranje dokumentacijske XML datoteke
  - Alternativno: kontekstni izbornik : Project properties \ Build \ XML documentation file
  - prevođenje proizvede *NazivAsemblija.XML*
  - dodavanjem reference na neki dll, uzima se i istoimena xml datoteka (ako postoji) – služi za prikaz dokumentacijskih komentara
- Služi kao pomoć prilikom programiranja
  - Nudi opise razreda i postupaka iz referenciranog projekta

## ➤ XML oznake za komentiranje programskog koda

- `<summary>` - sažetak opisa razreda ili člana
- `<remarks>` - nadopunjuje opis o razredu ili članu
- `<param>` - opis parametra metode
- `<paramref>` - oznaka da se opis odnosi na parametar
- `<permission>` - prava pristupa
- `<returns>` - povratne vrijednosti metode
- `<value>` - opis vrijednosti svojstva
- `<exception>` - iznimke koje mogu biti bačene u opisanom kodu. Sadrži referencu na iznimku.
- `<typeparam>` - opis generičkog razreda ili metode
- `<typeparamref>` - tip parametra kod generičkih razreda ili metoda

# XML oznake (nastavak)

5

- `<c>` - tekst unutar oznake je kod
- `<code>` - tekst unutar oznake je višeretkovni kod
- `<example>` - primjer
- `<include>` - uključivanje vanjske datoteke s XML dokumentacijom
- `<list>` - lista, nabrojavanje elemenata
- `<para>` - paragraf teksta
- `<see>` - omogućava stvaranje linka
- `<seealso>` - link u "See also" sekciji
  
- Za detaljniji opis oznaka pogledati:
  - <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/xml/doc/recommended-tags-for-documentation-comments>

# Generirana XML datoteka

## ► Primjer:

Web \ Firma.Mvc \ bin \ Debug \  
netcoreapp3.1 \ Firma.Mvc.xml

```
<?xml version="1.0"?>
<doc>
  <assembly>
    <name>Firma.Mvc</name>
  </assembly>
  <members>
    <member name="T:Firma.Mvc.Extensions.ExceptionExtensions">
      <summary>
        Razred sa proširenjima za iznimke
      </summary>
    </member>
    <member name="M:Firma.Mvc.Extensions.ExceptionExtensions.CompleteExceptionMessage(Sy">
      <summary>
        Vraća sadržaj poruka cijele hijerarhije neke iznimke. Za predanu iznimku provjer
        Ako da, poruka unutarne iznimke dodaje se u rezultat te se dalje provjerava post
      </summary>
      <param name="exc">Iznimka čija se kompletna hijerarhija poruka treba ispisati</p
      <returns>String formiran od poruka svih unutarnjih iznimki. Poruka svake iznimki
    </member>
    <member name="T:Firma.Mvc.TagHelpers.PageLinkTagHelper">
      <summary>
        Tag helper za kreiranje vlastitih poveznica na stranice u rezultatu nekog upravl
        Upotrebljava se kao atribut HTML oznake *div*
      <example>
        Primjer upotrebe
      <code>
        <div page-info="@Model.PagingInfo" page-action="Index" page-classes-enabled="tru
          page-class="btn" page-class-normal="btn-default"
          page-class-selected="btn-primary" class="btn-group pull-right">
        </div>
      </code>
      U datoteku *_ViewImports.cshtml* potrebno dodati `@addTagHelper *, Microsoft.Asp
      te u pogled uključiti vlastitu javascript datoteku *gotopage.js*
    </example>
  </summary>
</member>
```

# Alat za generiranje programske dokumentacije

7

- Generirana xml datoteka korisna unutar razvojnog sučelja, ali nije prikladna za čitanje
- Za izradu programske dokumentacije namijenjene za čitanje koristi se alat DocFx
  - Generira HTML sadržaj na osnovi programske dokumentacije
  - Omogućava dodavanje dodatnih sadržaja u dokumentaciju
- Instalacija DocFx-a
  - <https://github.com/dotnet/docfx/releases>
  - Preporuka dodati u putanju
- Alternativa: Unutar Visual Studio 2019 instalirati NuGet paket *docfx.Console*
- Napomena: podržava podskup oznaka iz xml dokumenacije
  - [http://dotnet.github.io/docfx/spec/triple\\_slash\\_comments\\_spec.html](http://dotnet.github.io/docfx/spec/triple_slash_comments_spec.html)

The screenshot shows the DocFx generated HTML documentation. The top navigation bar includes 'Dokumenti' and 'Programska dokumentacija'. The breadcrumb trail is 'Programska dokumentacija / Firma.Mvc.TagHelpers / PageLinkTagHelper'. A search bar is present with the placeholder 'Enter here to filter...'. On the left, a tree view shows the project structure, with 'Firma.Mvc.TagHelpers' expanded to show 'PageLinkTagHelper'. The main content area displays the title 'Class PageLinkTagHelper', a description 'Tag helper za kreiranje vlastitih poveznica na stranice u rezultatu nekog upra...', and an example of the generated HTML: `<div page-info="@Model.PagingInfo" page-action="Index" page-class="btn" page-class-normal="btn-default" page-class-selected="pull-right">`. Below the example, it notes that `@addTagHelper *, Microsoft.AspNetCore.Mvc.TagHelpers` must be added to `_ViewImports.cshtml` and that the `gotopage.js` file must be included. The 'Inheritance' section shows the hierarchy from `System.Object` to `PageLinkTagHelper`. The 'Namespace' is `Firma.Mvc.TagHelpers` and the 'Assembly' is `Firma.Mvc.dll`.

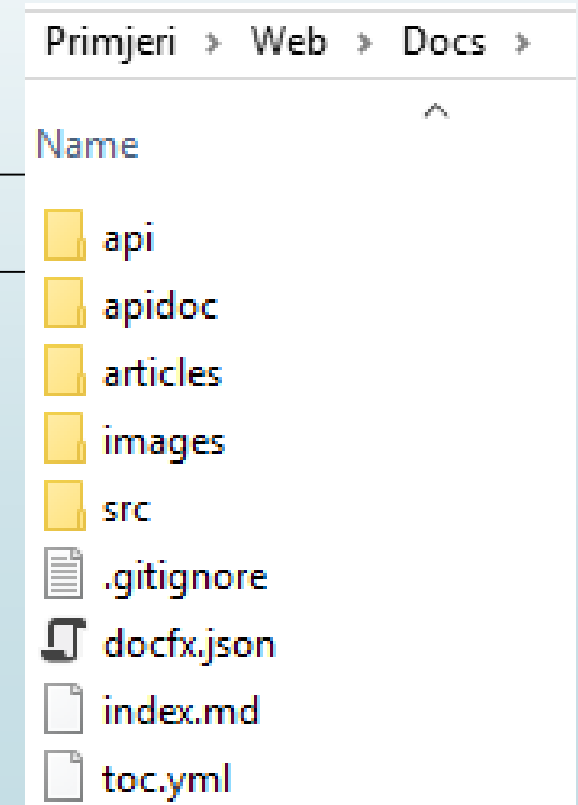
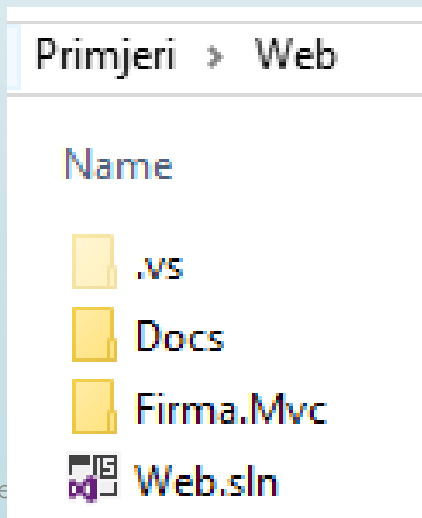
# Generiranje programske dokumentacije

8

- Upute koje slijede prilagođene su upute originalnih uputa [https://dotnet.github.io/docfx/tutorial/docfx\\_getting\\_started.html](https://dotnet.github.io/docfx/tutorial/docfx_getting_started.html)
- Preduvjet: Instaliran DocFx, dodan u putanju i uspješno kompilirana web-aplikacija s napisanom dokumentacijom
  - Alternativa: Visual Studio 2019: Projekt tipa ClassLibrary + NuGet paket docfx.Console
- U mapi u kojoj se nalazi sln datoteka (paralelno s mapom web-aplikacije) iz naredbenog retka pokrenuti

```
docfx init -q -o Docs
```



- stvara kostur projekta za programsku dokumentaciju

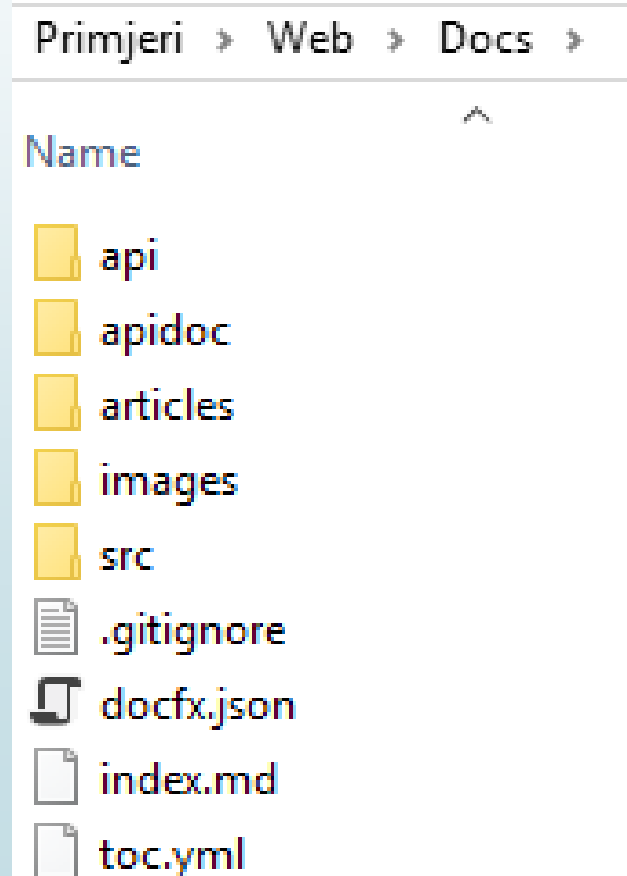




# Uređivanje početne stranice dokumentacije

9

- Docfx predstavlja strukturu projekta za izradu projektne dokumentacije
  - Mapa api će naknadno sadržavati informacije o razredima iz web-aplikacije
  - Mapa articles predstavlja proizvoljne dokumente koji se mogu uključiti u programsku dokumentaciju
    - Izmijeniti sadržaj dodatne dokumentacije po uzoru na  Web \ Docs \ articles \ index.md i toc.yml
- Toc.yml datoteke predstavljaju strukturu pojedinog dokumenta
- Sadržaj stranice piše se koristeći Markdown (ekstenzija md)
  - <https://en.wikipedia.org/wiki/Markdown>
- Izmijeniti index.md i toc.yml po uzoru na  Web \ Docs \ index.md i toc.yml







# Uključivanje web-aplikacije u izradu dokumentacije

Primjeri > Web

10

➡ Primjer  Web \ Docs \ docfx.json

Name

 .vs  
 Docs  
 Firma.Mvc  
 Web.sln

```
"metadata": [  
  {  
    "src": [  
      "files": [  
      "src/**/*.csproj"  
      ],  
      "exclude": [  
        "**/obj/**",  
        "**/bin/**",  
        "_site/**"  
      ]  
    },  
    ],  
    "dest": "api"  
  }  
],
```

```
"metadata": [  
  {  
    "src": [  
      "files": ["*.csproj"],  
      "src": "../Firma.Mvc",  
      "exclude": [  
        "**/obj/**",  
        "**/bin/**",  
        "_site/**"  
      ]  
    },  
    ],  
    "dest": "api"  
  }  
],
```

# Generiranje dokumentacije

11

- U mapi Docs pokrenuti `docfx`
  - Generira podmapu `_site` s html stranicama generiranim na osnovi markdown datoteka
- Za pregled dokumentacije pokrenuti  
`docfx serve _site`
  - Pokreće privremeni web server s dokumentacijom na adresi <http://localhost:8080>
- ...ili otvoriti `index.html` u pregledniku
  - Ako `docfx.json` za template ima `statictoc` umjesto default  
`"template": [ "statictoc"],`

Primjeri > Web > Docs > \_site >

Name

- api
- articles
- fonts
- styles
- favicon.ico
- index.html
- logo.svg
- manifest.json
- search-stopwords.json
- toc.html
- xrefmap.yml


# Izrada PDF-a s generiranom dokumentacijom (1)

12

- Potrebno instalirati *wkhtmltopdf* <https://wkhtmltopdf.org/downloads.html>

- Nakon instalacije dodati C:\Program Files\wkhtmltopdf\bin u putanju

- Kreirati podmapu pdf s definicijom poglavlja

- Primjer  Web \ Docs \ pdf \ toc.yml

```
- name: Dokumenti
  href: ../articles/toc.yml
- name: Programska dokumentacija
  href: ../api/toc.yml
```

- U docfx.json dodati blok za generiranje pdfa.

- Detaljnije u  Web \ Docs \ docfx.json

- Pokrenuti `docfx pdf`

- Nakon uspješne izgradnje pdf se nalazi u podmapi `_site_pdf`