Natural Docs 4 Matlab

Users' guide

Natural Docs 4 Matlab is a documentation generator for Matlab language developed using the existing Natural Docs software (http://www.naturaldocs.org/) developed by Greg Valure.

This document is a complement of Natural Docs documentation listing changes in the generated documentation and in the use of the software.

What you need to know.

- Everything that already is on http://www.naturaldocs.org
- To document your file, you can either
  - Use classic Natural Docs tags
  - If you use line comment, you can add a line with "%%" at the beginning of your comment, and skip the "type: name" declaration of classic Natural Docs tags.
    Here's an example for a function named "getName":

```
%%
% Get the name of…
function name = getName()
      …
end
```

    - Just write the type of object you’re documenting, like the classic Natural Docs tags, but without specifying the name of the object. This is mostly for block comment use, since the “%%” option makes it already easier for line comments.
      (We couldn’t make something like we did for line comment since block comment in Matlab language not that easy to use/manipulate).
      Here’s an example for a function named “getName”:

```
%{
Function:
Get the name of …
%}
function name = getName()
      …
end
```

- It only supports object-oriented Matlab language.
    - You can’t have more than one class per file.

What has changed from the original Natural Docs

- Following modifiers for functions and properties are handled:
    - Access (private, public, protected)
    - Abstract
    - Sealed
    - Static
    - Constant

  Functions and properties have been regrouped in the summary to display under their modifiers category.

- Functions and properties are sorted alphabetically, inside whichever group they are. If there are more than 20 functions in the same group, we display them under groups for the beginning letter of each function.
- If you activate the option, the code of a function can be displayed in the documentation. A button named « Display code » will allow you to display this code that is colored using Matlab IDE color code. To allow that to happen, you just have to add « -code » in the command line you’re using to generate your documentation.

- Inheritance is now handled better.

- For a given class, you will be able to see all non-private functions inherited from its superclasses. The only documentation displayed for those methods is a link to it in its own class.
- If a function in your class is an implementation of an abstract declaration in one of its superclasses, a link will automatically be generated in the summary and body of the function, to the body of the superclass' declaration for that function. (That has been made because it often happens that you document the abstract declaration better than the implementation).
- If a function in a superclass is abstract, and implemented in the current class, or another superclass, the abstract function will not be added to the documentation.

- Command line parameters –r and –ro are no longer supported. Project will always be rebuilt completely due to the support of inheritance, which requires it.