

Working with Doxygen

Generate a default configuration file using `doxygen -g`. There is a tool named `doxywizard` that is a GUI front end for working with `doxyfile`, but we don't have it installed.

For each documentable entity, doxygen looks for a *brief comment*, and a *detailed description*. The brief comment is added to various overviews (like the class member listing).

Comment Style

There are several ways to mark a comment block as a detailed description:

1. You can use the JavaDoc style, which consist of a C-style comment block starting with two `*/`s, like this:

```
2.      /**
3.      * ... text ...
4.      */
```

5. or you can use the Qt style and add an exclamation mark (!) after the opening of a C-style comment block, as shown in this example:

```
6.      /*!
7.      * ... text ...
8.      */
```

In both cases the intermediate `*/`s are optional, so

```
/*!
... text ...
*/
```

is also valid.

See more in [Commenting the code](#) in the [doxygen documentation](#).

Other doxygen Options:

When `doxyfile` is created, many attributes have default values. You may want to change some, and you will also need to fill in somethat are specific to your project. Options you may want to change.:

- `EXTRACT_ALL` enabled: so some generation is created even for "undocumented" code
I'd strongly recommend for existing projects to first add some documentation to the most important class declarations and methods, and then turn off the `EXTRACT_ALL` option. This makes the documentation less cluttered, and gives you even more inducement to document what isn't

- JAVADOC_AUTOBRIEF: this allows to have both the brief comment and detailed description in one block (despite it's name, it works for C++ sources, too). The first line of a comment block (up to the first period) is used as brief description.
- INPUT : Consider a single "." as INPUT directory, and if you check the RECURSIVE option. this will scan the working directory
- SOURCE_BROWSER: Enable the SOURCE_BROWSER option, so the sources are included and crosslinked (doxygen removes all doxygen-style comments, but leaves all other comments in).
- HTML_OUTPUT: Set *html* or some other appropriate name for the directory in which doxygen will generate the html. (it will be created under the working directory)

Tip: When specifying directory names in doxygen config files -

- if it contains spaces, put it into quotes (this is a good idea for other options as well)
- do **not** end the full path with a backslash

Documentation Markup:

Doxygen allows for simple markup tags in the comments. They start with either a backslash, or an @. The most important are listed here:

<code>\param name description</code>	Intended for documenting function parameters. see the full sample source and documentation for how it looks like.
<code>\return description</code>	describe what a function returns
<code>\brief</code>	Short Description that will appear on list pages
<code>\b \c \e</code>	set the next word to bold, italic, or courier, respectively. e.g. /// You can make things <code>\b</code> bold, <code>\e</code> italic, or set them in <code>\c</code> courier results in You can make things bold , <i>italic</i> , or set them in courier.
<code>\code</code> <code>\endcode</code>	starts and ends a section of code, respectively. (it will be formatted nicely)
<code>\n</code>	force a newline
<code>\internal</code>	starts a paragraph with "internal information" (such as implementation details). The paragraph will be included only if the INTERNAL_DOCS option is enabled.
<code>\mainpage</code>	Indicates that the following section should appear on the main page. it's a good place to introduce your most important classes, etc. (entities will be crosslinked)
<code>\par</code> <code>\par Title</code>	Starts a new paragraph (optionally with a paragraph title), works also inside other paragraphs (such as <code>\param</code>)
Lists	Doxygen automatically creates a list if multiple lines start with a dash at the same position. Numbered lists can be created by starting the line with a dash and a hash (-#). See the doxygen documentation for more.
HTML (doxygen documentation)	Doxygen also supports HTML tags, which get converted to other outputs more or less correctly.

HTML Generation:

To generate the html, simply call:

```
doxygen doxyfile
```

You will find the generated site in the directory you specified in doxyfile. Point your browser at index.html in that directory (or the directory itself, since index.html is the default html file in a directory for most web servers).