



Doxygen Cheat Sheet

Doxygen is a documentation generator.

Running Doxygen

| | |
|-------------------------|---|
| <code>doxygen</code> | Run Doxygen to create documentation from annotated code |
| <code>doxygen -g</code> | Create a Doxygen template |
| <code>doxygen -x</code> | Compare existing Doxyfile against the template file |

| Block | Comments | Line |
|--|-------------------------------------|--|
| <code>/*!</code> | Start comment block (Qt style) | <code>/*!</code> |
| <code>/**</code> | Start comment block (Javadoc style) | <code>/**</code> |
| <code>*/</code> | End comment block | |
| <code>\<special-command></code> or <code>@<special-command></code> | | Use special command within a comment block |
| <pre>/*! * A brief description of the foo() function * * This is a detailed description of the foo() * function, which can span several lines. * * @param Description of the one argument named 'bar' */ void foo(int bar);</pre> | | |

Groups

| | |
|--------------------|--|
| <code>{@</code> | Mark the beginning of a group (within a special comment block) |
| <code>@}</code> | Mark the end of a group (within a special comment block) |
| <code>@name</code> | Specify name of the group |



Special commands

| | |
|----------------|---|
| @brief | Mark a paragraph as the brief description. |
| @param[in/out] | Describe the arguments of a function or method. |
| @tparam | Template parameter. |
| @mainpage | Mark a comment block as the main page (index.html). |
| @section | Start a new section in a long comment block. |
| @subsection | A section within a section. |
| @ref | Create a reference to a named section, subsection, page or anchor. |
| @internal | Generate when INTERNAL_DOCS=yes in Doxyfile. Useful for private class members. |

Special commands for arbitrary positioning

| | | | |
|----------|-------------------------|------------|------------------|
| @class | Document a C++ class | @struct | C-struct |
| @union | Document a union | @enum | Enumeration type |
| @fn | Document a function | @var | Variable |
| @def | Document a #define | @typedef | Type definition |
| @file | Document a file | @namespace | Namespace |
| @package | Document a Java package | @interface | IDL interface |

```

/*! @fn size_t ByteStream::size() const
 * @brief Returns the size of the used memory buffer
 *
 * The function behaves identically for both
 * internal and external buffers.
 */

```

