

Documenting PandaRoot with Doxygen

2015-2 Collaboration Meeting

9 June 2015, Andreas Herten

Documenting Code

- It helps others
- It helps yourself
- It's super easy and intuitive with Doxygen
- It helps structuring our 550,000 lines* of C++
- Otherwise...

Documenting Code

- It helps others
- It helps yourself
- It's super easy and intuitive with Doxygen
- It helps structuring our 550,000 lines* of C++
- Otherwise...



*: cloc --exclude-dir=genfit,genfit2,pgenerators/EvtGen,html-doc --exclude-ext=html --exclude-lang=HTML *

Documenting Code — *Why?!*

- Help people understand...
 - ... your class's **purpose**
 - ... your class's **interface**
 - ... your code's **internals**, if needed
 - ... your totally easy, extremely subtle, and super obvious 100 line method with 20 in-line `ifs`, 4 loops and running indices `i, j, k, l, m, o, i2, ii, ij`
⇒ Things might not be as obvious and easy to others as they are to you *right now*.

- Target audience

- PANDA

- Users (only using code, not writing)
 - Novice programmers (new in the collaboration / new to the topic)
 - Expert programmers (new to the topic)

- External users (our source code is *open* after all)

- **Future-proofing** for your later self and others



Documenting Code — *Why?!*

- Help people understand...
 - ... your class's **purpose**
 - ... your class's **interface**
 - ... your code's **internals**, if needed
 - ... your totally easy, extremely subtle, and super obvious 100 line method with 20 in-line `ifs`, 4 loops and running indices `i, j, k, l, m, o, i2, ii, ij`
⇒ Things might not be as obvious and easy to others as they are to you *right now*.

- Target audience

- PANDA

- Users (only using code, not writing)
 - Novice programmers (new in the collaboration / new to the topic)
 - Expert programmers (new to the topic)

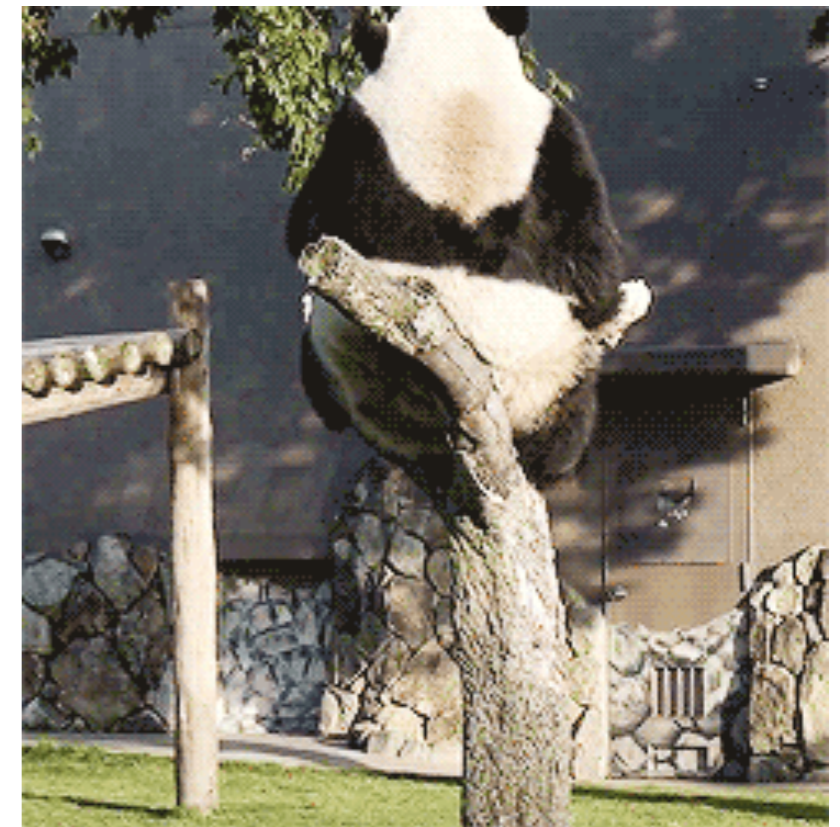
- External users (our source code is *open* after all)

- **Future-proofing** for your later self and others



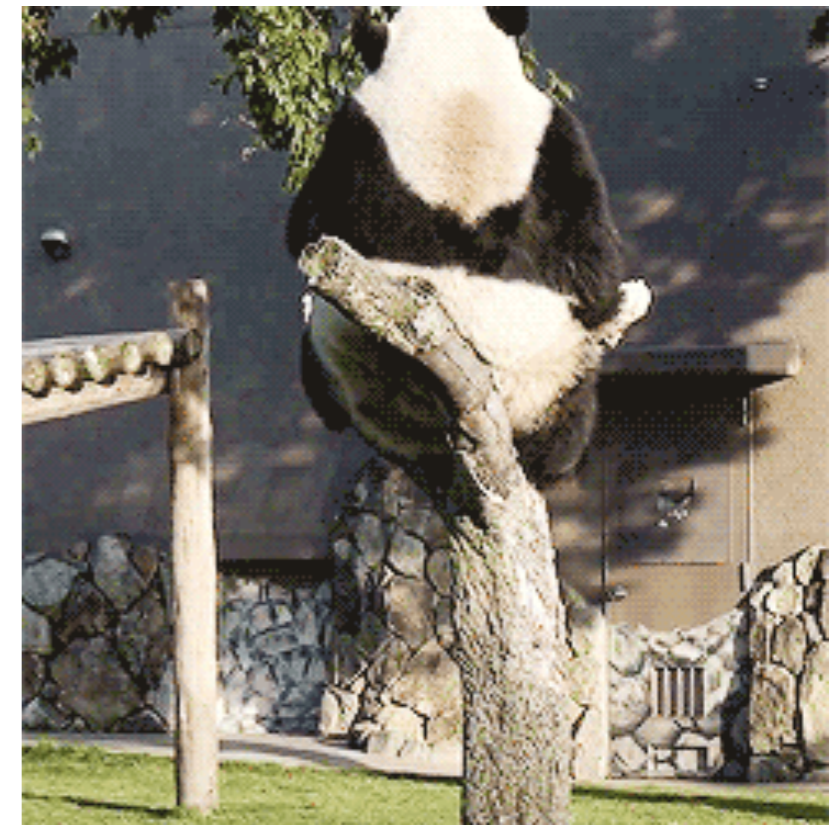
Documenting Code — What

- Class author, date, purpose
- Why (if not obvious), not how (if not complex)
- Code should be self-explanatory, but more documentation for
 - **Complex** interfaces, ambivalent interfaces
 - **Complex** methods
 - **Complex** snippets in code
 - **Caveats** of your code
- Rules of thumb
 - Self-documentation:
Use descriptive variable & method names
 - Be concise: Only add information, don't clutter it
 - Don't assume: »That's common knowledge«
 - Chances are if you spent a long time coming up with the solution, it's worth documenting



Documenting Code — What

- Class author, date, purpose
- Why (if not obvious), not how (if not complex)
- Code should be self-explanatory, but more documentation for
 - **Complex** interfaces, ambivalent interfaces
 - **Complex** methods
 - **Complex** snippets in code
 - **Caveats** of your code
- Rules of thumb
 - Self-documentation:
Use descriptive variable & method names
 - Be concise: Only add information, don't clutter it
 - Don't assume: »That's common knowledge«
 - Chances are if you spent a long time coming up with the solution, it's worth documenting



- Files considered: 2283

```
find . -type f -name "*.h" -o -name "*.cxx" -o \( -path "./genfit2" -o -path "./genfit" -o -path "./pgenerators/EvtGen" \) -prune | wc -l
```

- Files with some Doxygen-compatible comment: 1051 (46 %)

```
find . -type f -name "*.h" -o -name "*.cxx" -o \( -path "./genfit2" -o -path "./genfit" -o -path "./pgenerators/EvtGen" \) -prune | xargs pcregrep -1M "\/(\\*(\\*|\\!)[^\\/*|\\(\\|\\!))[" ]*" | wc -l
```

- Files with a sensible Doxygen-compatible comment: 412 (18 %)

```
find . -type f -name "*.h" -o -name "*.cxx" -o \( -path "./genfit2" -o -path "./genfit" -o -path "./pgenerators/EvtGen" \) -prune | xargs pcregrep -1M "\/(\\*(\\*|\\!)[^\\/*|\\(\\|\\!))[" ]*(\\|\\@)" | wc -l
```

- Files with an @author: 349 (15 %)

```
find . -type f -name "*.h" -o -name "*.cxx" -o \( -path "./genfit2" -o -path "./genfit" -o -path "./pgenerators/EvtGen" \) -prune | xargs pcregrep -1M '@author|\\author' | wc -l
```

→ That could be better!

Doxygen

- Mark-up language for documenting code in-file
- For
 - Projects; Classes; Files
 - Methods / functions; Variables
 - *Everything else*
- Generates a documentation for a code base automatically, steered and annotated by the users' Doxygen keywords
- Easy to use: descriptive keywords, little overhead
- PandaRoot Class Documentation:
http://cbmroot.gsi.de/panda_doc/daily/html/classes.html

FairRoot/PandaRoot

- Main Page
 - Related Pages
 - Modules
 - Namespaces
 - Classes**
 - Files
 - Examples
- Class List
 - Class Index**
 - Class Hierarchy
 - Class Members

Class Index

4 | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X

4

Pnd4CFitter

A

- AbsBField (genfit)
- AbsFinitePlane (genfit)
- AbsFitter (genfit)
- AbsFitterInfo (genfit)
- AbsHMatrix (genfit)
- AbsKalmanFitter (genfit)
- AbsMaterialInterface (genfit)
- AbsMeasurement (genfit)
- AbsMeasurementProducer (genfit)
- AbsTrackRep (genfit)
- AbstractLmdDataFilter (LumiFit::Comparisons)
- AdditionModel1D
- PndAnaContFact
- PndAnaCovTool
- PndAnalysis
- PndAnalysisCalcTools
- PndAnalysisTask
- PndAnalysisTaskExample

- PndEvtGenGenerator
- EvtRootRandomEngine
- Exception (genfit)
- ExponentialModel1D
- ExtrapStep (genfit)

F

- PndFastSim
- PndFieldAdaptor
- fieldCache (genfit)
- PndFieldContFact
- PndFieldCreator
- FieldManager (genfit)
- PndFieldMap
- PndFieldMapData
- PndFieldPar
- PndFileNameCreator
- FairFilteredPrimaryGenerator
- PndLmdLumiHelper::findByDimensionType
- PndLmdFitFacade::findBySelectionDimensionsIgnoringSecondaryTrackFilters
- FindTrackInXYProjection2_InputOutputData
- FindTrackInXYProjection_InputData
- fit_options_compare (LumiFit::Comparisons)

- PndHypKalmanTask
- PndHypMicroIdealWriter
- PndHypMicroWriter
- PndHypMSAnaTask
- PndHypPidCand
- PndHypPidIdealTask
- PndHypPoint
- PndHypRecoHit
- PndHypRecoSPHit
- PndHypSimpleAna
- HypStatDecay
- PndHypStrip
- PndHypStripClusterBuilder
- PndHypStripClusterTask
- PndHypStripDigiPar
- PndHypStripHitProducer

I

- IClassifierReader
- IntegralStrategy1D
- IntegralStrategy2D
- IntegralStrategyGSL1D
- IntegralStrategyGSL2D

http://cbmroot.gsi.de/panda_doc/daily/html/classes.html

- Doxygen-compatible comments start like

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

- In there: Different keywords for annotation
keywords start with @ or \ – please use @!
- Documentation block comes **before** method / class

- Keywords every class should have
 - **@author** – include email address & institute
`@author Andreas Herten <a.herten@fz-juelich.de>, FZJ`
`@author P. Anda <p.anda@gsi.de>, GSI`
 - **@date** – month in three-letter abbreviation
`@date Jun 9 2015`
 - **@brief** – one-liner about following method / ...
`@brief A talk about documentation`
 - **@details** – extensive documentation of following method / ...; examples
`@details I'm going to talk about the status and give a how-to and also unicorns`
 - **@class** – first item; name of class
`@class PndTlkDoxygen`
- **Examples follow shortly!**

Doxygen — Basics IIb: @details

- @details for extensive documentation and explanation of class (/ method)
- Place for references, examples, equation derivations, ...
- Doxygen allows to use HTML, Markdown, LaTeX for text styling
(not only in @details section, but especially)
- Markdown:
 - *italic*, **bold**, [link](<http://panda.gsi.de>)
 - # Heading 1; ## Heading 2; ...
 - * List item 1; * List item 2; ...
- LaTeX:
 - Inline: @f\$ \LaTeX @f\$
 - Block: @f[\LaTeX @f]

- For classes and methods
 - **@version** – only if you intend to keep it up to date!
 - **@note**, **@attention**, **@remark**, **@warning**, **@todo**
 - ways to steer reader's/own attention to something
 - **@namespace** – »Use namespaces sparingly«[PandaRootCodingRules](#)
 - **@name**, **@{** and **@}** – generate member block with name
- For methods / functions
 - **@param** – explanation of every parameter of method call
`@param lngth The length in cm`
 - **@return** – like **@param** but for returned parameter
`@return A std::vector of experiments`

Doxygen — Example File I

- PndTools/generalTools/DoxygenExample/PndDocuSample.{cxx,h}

```
#include "TObject.h"
```

```
/**
```

```
 * @class PndDocuSample
```

```
 * @brief Documentation Sample Class
```

```
 * @details # Sample Documentation Class
```

```
 * This file highlights documentation with `Doxygen`. Please see the individual parts  
for more information.
```

```
 * ...
```

```
 *
```

```
 * ## Markdown
```

```
 * Markdown is the a markup language which lets you write styled text.
```

```
 *
```

```
 * @author Andreas Hertel <a.herten@fz-juelich.de>, FZJ
```

```
 * @version 1.0
```

```
 * @date May 11, 2015
```

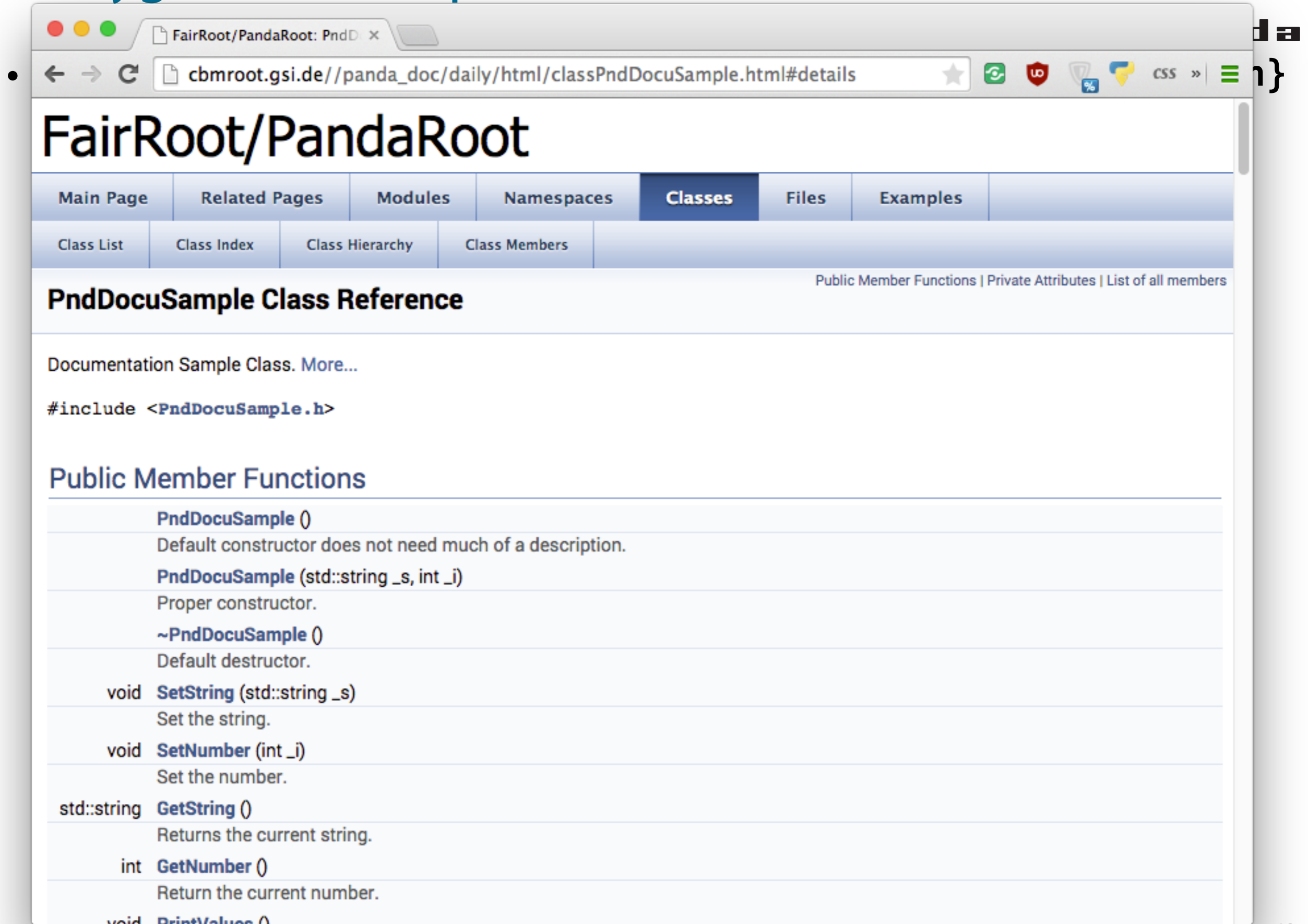
```
 */
```

```
class PndDocuSample {
```

```
public:
```

```
    // ...
```

```
};
```

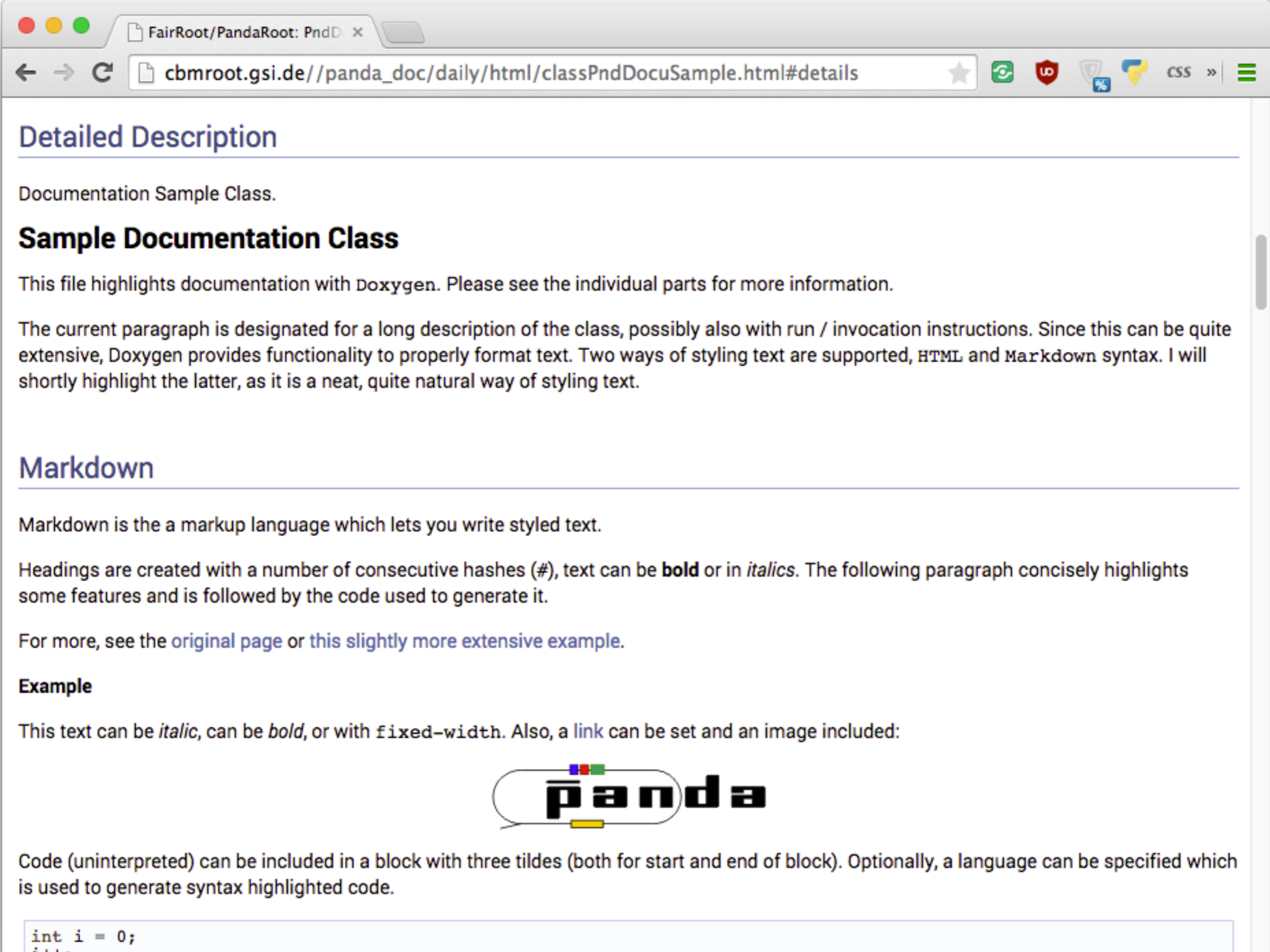


The screenshot shows a web browser window with the address bar containing `cbmroot.gsi.de//panda_doc/daily/html/classPndDocuSample.html#details`. The page title is "FairRoot/PandaRoot". The navigation menu includes "Main Page", "Related Pages", "Modules", "Namespaces", "Classes" (selected), "Files", and "Examples". Below the menu, there are sub-links for "Class List", "Class Index", "Class Hierarchy", and "Class Members". The main content area is titled "PndDocuSample Class Reference" and includes a link for "Public Member Functions | Private Attributes | List of all members". The class description starts with "Documentation Sample Class. More..." and includes the header `#include <PndDocuSample.h>`. The "Public Member Functions" section lists the following:

- `PndDocuSample ()`: Default constructor does not need much of a description.
- `PndDocuSample (std::string _s, int _i)`: Proper constructor.
- `~PndDocuSample ()`: Default destructor.
- `void SetString (std::string _s)`: Set the string.
- `void SetNumber (int _i)`: Set the number.
- `std::string GetString ()`: Returns the current string.
- `int GetNumber ()`: Return the current number.
- `void PrintValues ()`

da
n }

Doxygen — Example File I (*Pictures*)

- 

The screenshot shows a web browser window with the address bar containing `cbmroot.gsi.de//panda_doc/daily/html/classPndDocuSample.html#details`. The page content includes:

Detailed Description

Documentation Sample Class.

Sample Documentation Class

This file highlights documentation with Doxygen. Please see the individual parts for more information.

The current paragraph is designated for a long description of the class, possibly also with run / invocation instructions. Since this can be quite extensive, Doxygen provides functionality to properly format text. Two ways of styling text are supported, HTML and Markdown syntax. I will shortly highlight the latter, as it is a neat, quite natural way of styling text.

Markdown


Markdown is the a markup language which lets you write styled text.

Headings are created with a number of consecutive hashes (#), text can be **bold** or in *italics*. The following paragraph concisely highlights some features and is followed by the code used to generate it.

For more, see the [original page](#) or [this slightly more extensive example](#).

Example

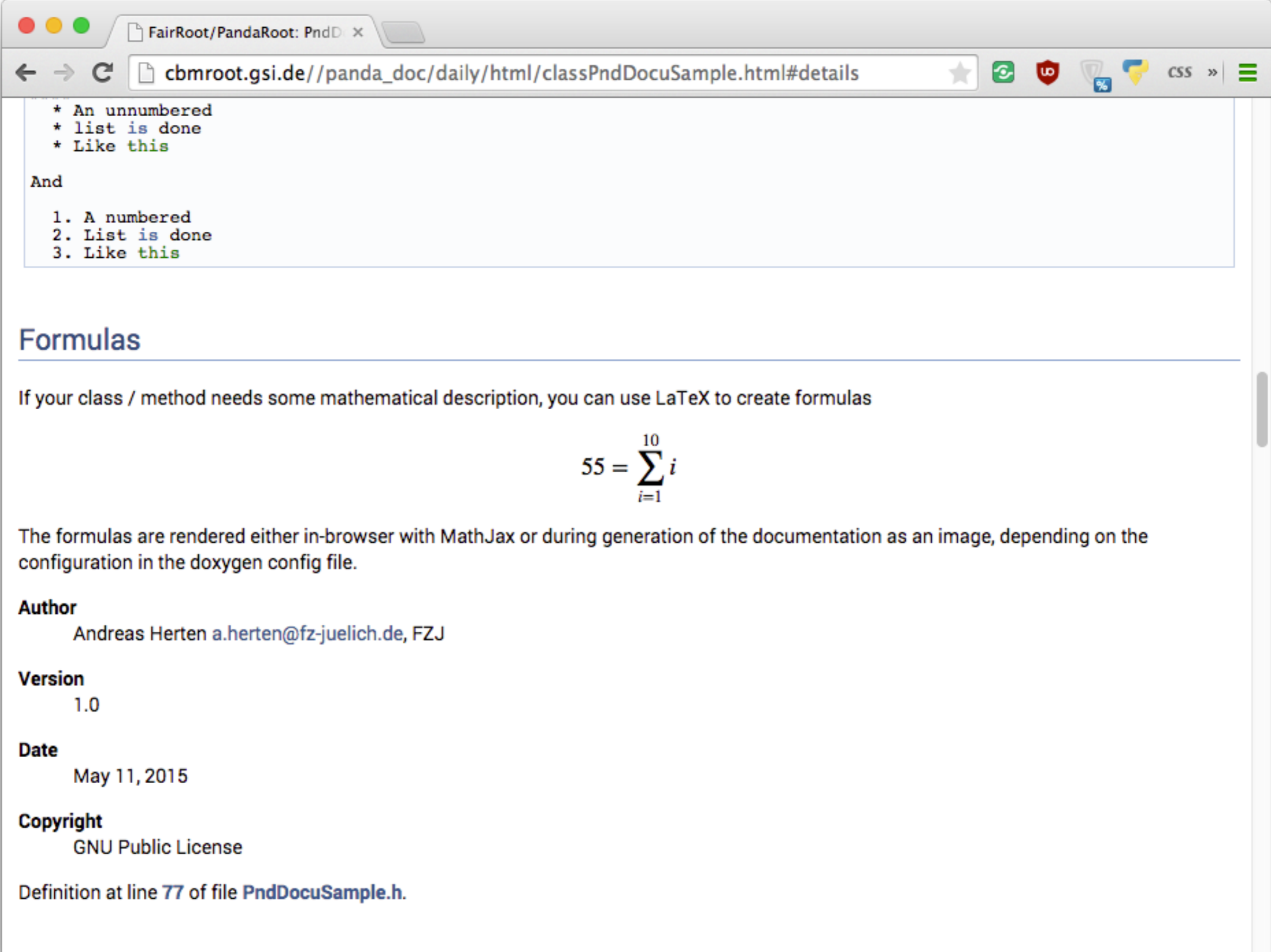
This text can be *italic*, can be **bold**, or with `fixed-width`. Also, a [link](#) can be set and an image included:



Code (uninterpreted) can be included in a block with three tildes (both for start and end of block). Optionally, a language can be specified which is used to generate syntax highlighted code.

```
int i = 0;
i++;
```

Doxygen — Example File I (*Pictures*)



• `FairRoot/PandaRoot: PndD` x

`cbmroot.gsi.de//panda_doc/daily/html/classPndDocuSample.html#details`

- * An unnumbered
- * list is done
- * Like this

And

1. A numbered
2. List is done
3. Like this

Formulas

If your class / method needs some mathematical description, you can use LaTeX to create formulas

$$55 = \sum_{i=1}^{10} i$$

The formulas are rendered either in-browser with MathJax or during generation of the documentation as an image, depending on the configuration in the doxygen config file.

Author
Andreas Herten a.herten@fz-juelich.de, FZJ

Version
1.0

Date
May 11, 2015

Copyright
GNU Public License

Definition at line **77** of file **PndDocuSample.h**.

Doxygen — Example File II

- PndTools/generalTools/DoxygenExample/PndDocuSample.{cxx,h}

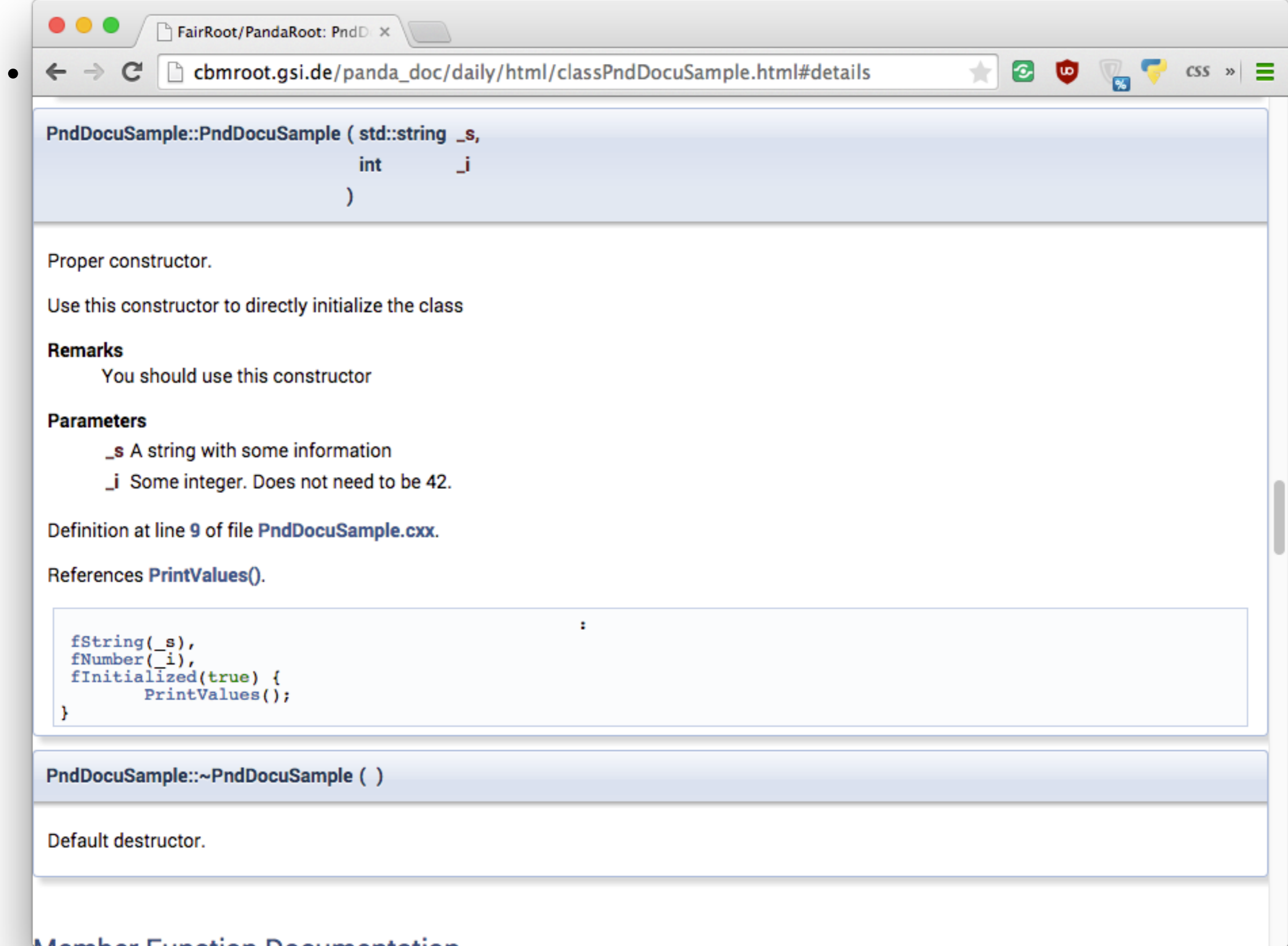
```
class PndDocuSample {
public:
    /**
     * @brief Proper constructor
     * @details Use this constructor to directly initialize the class
     * @remark You should use this constructor
     *
     * @param _s A string with some information
     * @param _i Some integer. Does not need to be 42.
     */
    PndDocuSample(std::string _s, int _i);
    ~PndDocuSample(); ///< Default destructor

    std::string GetString() { return fString; };
    /**
     * @brief Return the current number
     * @return The current content of @p fNumber
     */
    int GetNumber() { return fNumber; };

private:
    bool fInitialized; ///< Easiest way to document data member variables

    /** @cond CLASSIMP */ // As long as CLASSIMP is not set, the ClassDef() and
    ClassImp() macros are excluded from the documentation
    ClassDef(PndDocuSample, 1); ///< ClassDef/ClassImp should always be terminated with
    a semicolon
    /** @endcond */
};
```

Doxygen — Example File II (*Pictures*)



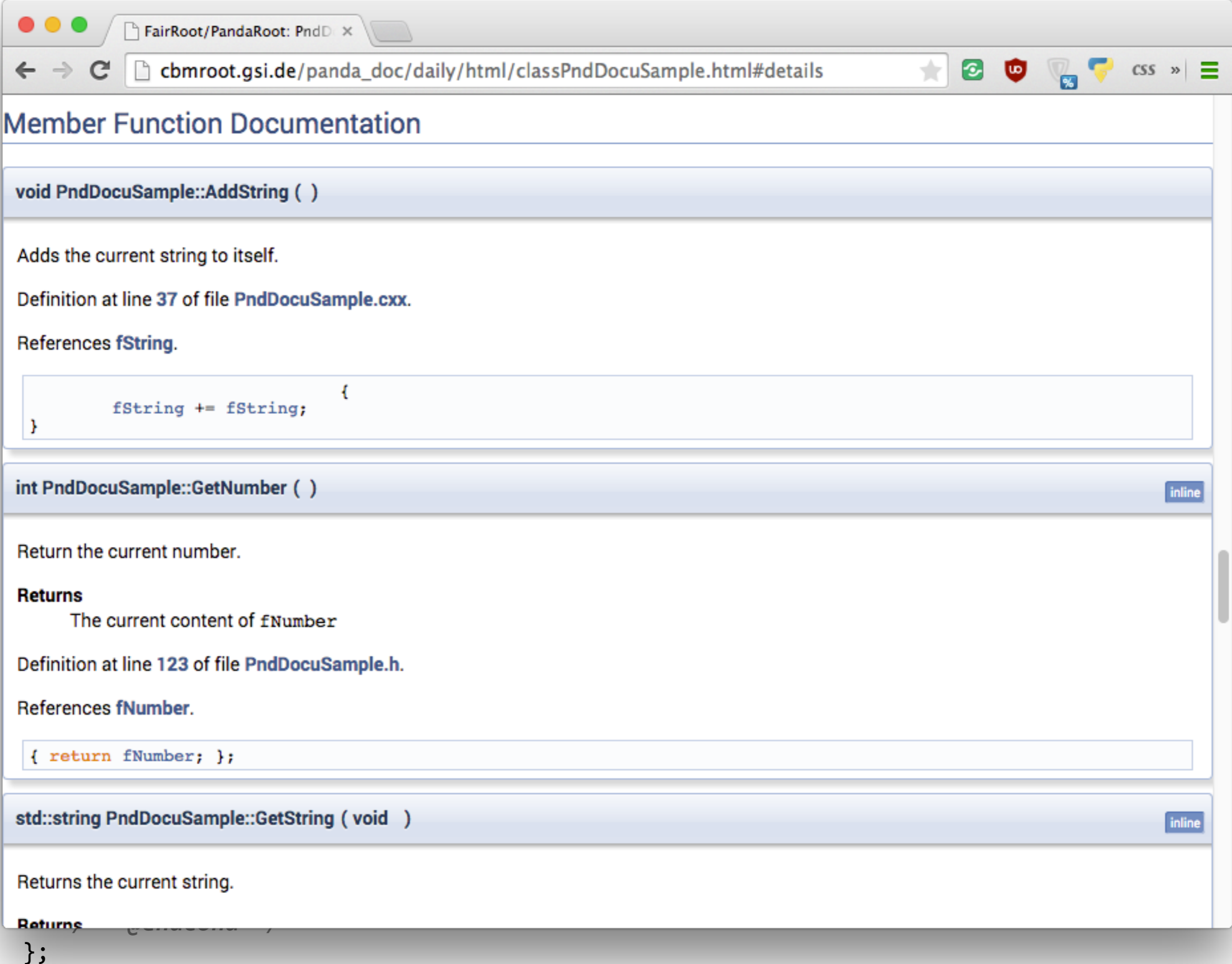
The screenshot shows a web browser window with the address bar containing `cbmroot.gsi.de/panda_doc/daily/html/classPndDocuSample.html#details`. The page content includes:

- PndDocuSample::PndDocuSample (std::string _s, int _i)**
- Proper constructor.
- Use this constructor to directly initialize the class
- Remarks**
You should use this constructor
- Parameters**
 - `_s` A string with some information
 - `_i` Some integer. Does not need to be 42.
- Definition at line 9 of file `PndDocuSample.cxx`.
- References `PrintValues()`.
- Code snippet:

```
fString(_s),  
fNumber(_i),  
fInitialized(true) {  
    PrintValues();  
}
```
- PndDocuSample::~PndDocuSample ()**
- Default destructor.

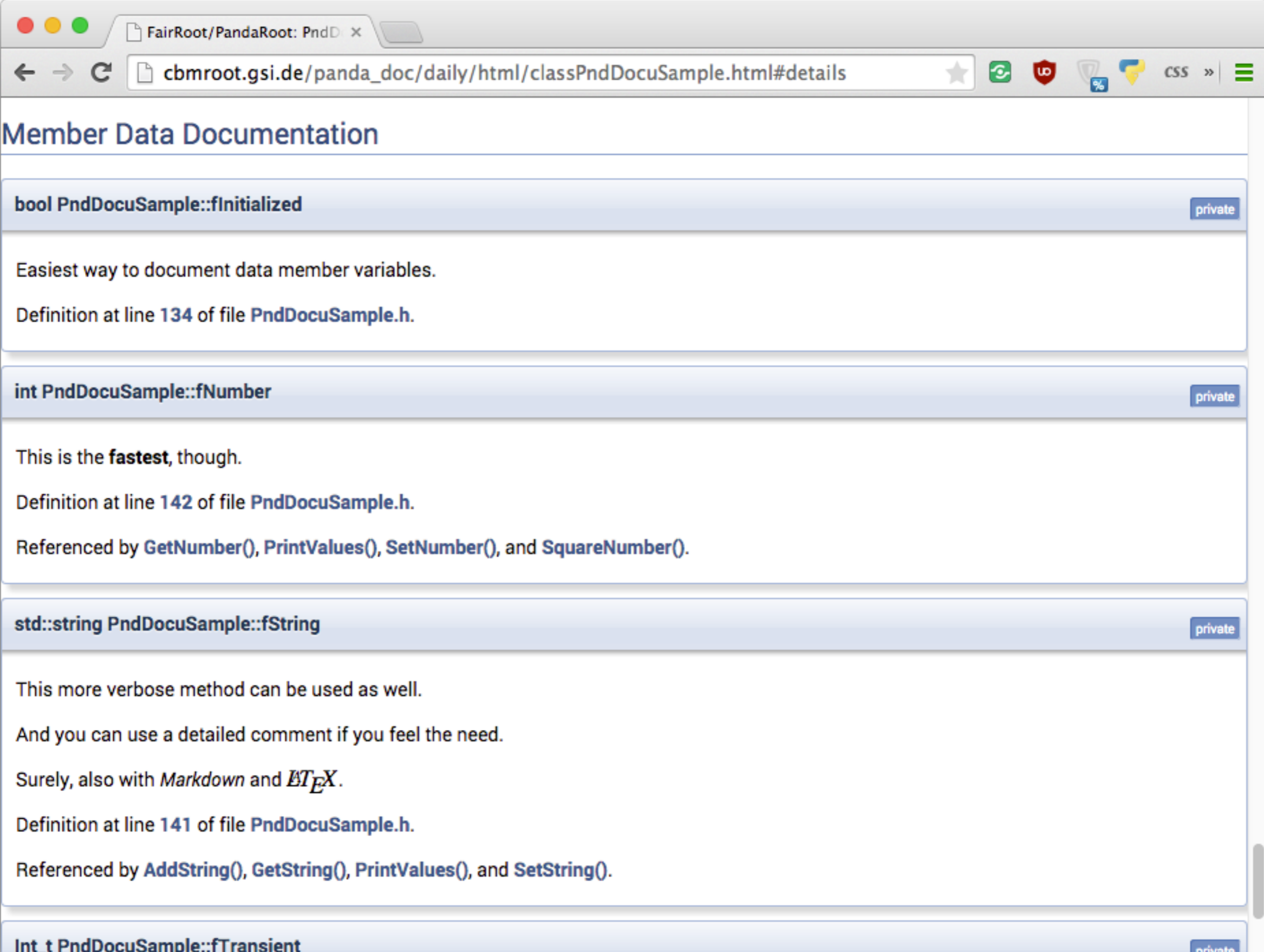
da
}

Doxygen — Example File II (*Pictures*)

- 

da
}

Doxygen — Example File II (*Pictures*)

- 

Member Data Documentation

bool PndDocuSample::fInitialized private

Easiest way to document data member variables.

Definition at line **134** of file **PndDocuSample.h**.

int PndDocuSample::fNumber private

This is the **fastest**, though.

Definition at line **142** of file **PndDocuSample.h**.

Referenced by **GetNumber()**, **PrintValues()**, **SetNumber()**, and **SquareNumber()**.

std::string PndDocuSample::fString private

This more verbose method can be used as well.

And you can use a detailed comment if you feel the need.

Surely, also with *Markdown* and *LaTeX*.

Definition at line **141** of file **PndDocuSample.h**.

Referenced by **AddString()**, **GetString()**, **PrintValues()**, and **SetString()**.

Int_t PndDocuSample::fTransient private

Conclusion & Links

- PandaRoot documentation with Doxygen
- **Please use it!**
It's easy and helps a lot



- Further reading:

- Wiki: PandaRootDoxygen

<https://panda-wiki.gsi.de/cgi-bin/view/Computing/PandaRootDoxygen>

- Example class: PndDocuSample

<https://subversion.gsi.de/trac/fairroot/browser/pandaroot/trunk/PndTools/generalTools/DoxygenExample>

- Doxygen documentation

<http://www.stack.nl/~dimitri/doxygen/manual/commands.html>

Conclusion & Links

- PandaRoot documentation with Doxygen
- **Please use it!**
It's easy and helps a lot



- Further reading:

- Wiki: PandaRootDoxygen

<https://panda-wiki.gsi.de/cgi-bin/view/Computing/PandaRootDoxygen>

- Example class: PndDocuSample

<https://subversion.gsi.de/trac/fairroot/browser/pandaroot/trunk/PndTools/generalTools/DoxygenExample>

- Doxygen documentation

<http://www.stack.nl/~dimitri/doxygen/manual/commands.html>

Conclusion & Links

- PandaRoot documentation with Doxygen
- **Please use it!**
It's easy and helps a lot



- Further reading:
 - Wiki: PandaRootDoxygen

<https://panda-wiki.gsi.de/cgi-bin/view/Computing/PandaRootDoxygen>

- Example class: PndDocuSample

<https://subversion.gsi.de/trac/fairroot/browser/pan>

- Doxygen documentation

<http://www.stack.nl/~dimitri/doxygen/manual/com>

Thank you!

Andreas Herten
a.herten@fz-juelich.de



MORE

PandaRoot Lines of Code

```
$ cloc --exclude-dir=genfit,genfit2,pgenerators/EvtGen,html-doc --exclude-ext=html --exclude-lang=HTML *  
15792 text files.  
15617 unique files.  
11004 files ignored.
```

```
http://cloc.sourceforge.net v 1.62 T=82.48 s (57.2 files/s, 11969.2 lines/s)
```

Language	files	blank	comment	code
C++	2666	123628	118833	482719
C/C++ Header	1637	33414	35831	71572
XML	18	34	4779	36376
Fortran 77	26	2216	4652	14245
Bourne Shell	107	1748	2071	11522
Perl	23	617	945	10993
Python	18	1693	997	6638
CMake	105	1255	879	5130
Bourne Again Shell	58	829	2852	3165
C	2	346	268	2042
make	43	521	250	1517
CSS	2	241	40	1228
JSON	3	47	0	205
DOS Batch	2	23	1	173
m4	2	24	0	155
Visual Basic	1	25	0	136
Javascript	2	11	68	93
CUDA	1	9	0	88
C Shell	1	11	5	26
D	1	2	0	25
SUM:	4718	166694	172471	648048