

IniParser

Generated by Doxygen 1.8.13

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	four_bytes Struct Reference	5
3.1.1	Detailed Description	5
3.2	ini_file Struct Reference	5
3.2.1	Detailed Description	6
3.3	IniEntry Class Reference	6
3.3.1	Detailed Description	7
3.3.2	Constructor & Destructor Documentation	7
3.3.2.1	IniEntry() [1/9]	7
3.3.2.2	IniEntry() [2/9]	7
3.3.2.3	IniEntry() [3/9]	8
3.3.2.4	IniEntry() [4/9]	8
3.3.2.5	IniEntry() [5/9]	8
3.3.2.6	IniEntry() [6/9]	9
3.3.2.7	IniEntry() [7/9]	9
3.3.2.8	IniEntry() [8/9]	10
3.3.2.9	IniEntry() [9/9]	10
3.3.3	Member Function Documentation	11

3.3.3.1	getComment()	11
3.3.3.2	getDouble()	11
3.3.3.3	getFloat()	11
3.3.3.4	getInt()	12
3.3.3.5	getKey()	12
3.3.3.6	getLong()	12
3.3.3.7	getLongDouble()	13
3.3.3.8	getLongLong()	13
3.3.3.9	getString()	13
3.3.3.10	getUnsignedLong()	14
3.3.3.11	getUnsignedLongLong()	14
3.3.3.12	operator"!=()	14
3.3.3.13	operator=() [1/10]	15
3.3.3.14	operator=() [2/10]	15
3.3.3.15	operator=() [3/10]	15
3.3.3.16	operator=() [4/10]	16
3.3.3.17	operator=() [5/10]	16
3.3.3.18	operator=() [6/10]	17
3.3.3.19	operator=() [7/10]	17
3.3.3.20	operator=() [8/10]	18
3.3.3.21	operator=() [9/10]	18
3.3.3.22	operator=() [10/10]	18
3.3.3.23	operator==(())	19
3.3.3.24	setComment()	19
3.3.3.25	setValue() [1/10]	19
3.3.3.26	setValue() [2/10]	20
3.3.3.27	setValue() [3/10]	20
3.3.3.28	setValue() [4/10]	20
3.3.3.29	setValue() [5/10]	21
3.3.3.30	setValue() [6/10]	21

3.3.3.31	setValue() [7/10]	22
3.3.3.32	setValue() [8/10]	22
3.3.3.33	setValue() [9/10]	22
3.3.3.34	setValue() [10/10]	23
3.4	IniFile Class Reference	23
3.4.1	Detailed Description	23
3.4.2	Constructor & Destructor Documentation	24
3.4.2.1	IniFile()	24
3.4.3	Member Function Documentation	24
3.4.3.1	at()	24
3.4.3.2	clean()	25
3.4.3.3	clear()	25
3.4.3.4	empty()	25
3.4.3.5	erase()	25
3.4.3.6	getAllSections()	25
3.4.3.7	getComment()	26
3.4.3.8	insert()	26
3.4.3.9	operator!=()	26
3.4.3.10	operator==()	27
3.4.3.11	operator[]()	27
3.4.3.12	setComment()	27
3.4.3.13	size()	28
3.5	IniParser Class Reference	28
3.5.1	Member Function Documentation	28
3.5.1.1	getError()	29
3.5.1.2	getErrorString()	29
3.5.1.3	load()	30
3.5.1.4	store()	30
3.5.2	Member Data Documentation	31
3.5.2.1	NO_ERROR	31

3.6	IniSection Class Reference	31
3.6.1	Detailed Description	32
3.6.2	Constructor & Destructor Documentation	32
3.6.2.1	IniSection()	32
3.6.3	Member Function Documentation	32
3.6.3.1	at()	33
3.6.3.2	clean()	33
3.6.3.3	clear()	33
3.6.3.4	empty()	33
3.6.3.5	erase()	33
3.6.3.6	getAllEntries()	34
3.6.3.7	getComment()	34
3.6.3.8	getName()	34
3.6.3.9	insert()	34
3.6.3.10	operator!=()	35
3.6.3.11	operator==()	35
3.6.3.12	operator[]()	35
3.6.3.13	setComment()	37
3.6.3.14	size()	37
3.7	IniParser::ParserException Class Reference	37
3.7.1	Detailed Description	38
3.8	two_bytes Struct Reference	38
3.8.1	Detailed Description	38
	Index	39

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

exception	
IniParser::ParserException	37
four_bytes	5
ini_file	5
IniEntry	6
IniFile	23
IniParser	28
IniSection	31
two_bytes	38

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

four_bytes	5
ini_file	5
IniEntry	6
IniFile	23
IniParser	28
IniSection	31
IniParser::ParserException	37
two_bytes	38

Chapter 3

Class Documentation

3.1 four_bytes Struct Reference

```
#include <ini_helper.h>
```

Public Attributes

- unsigned char **bytes** [4]

3.1.1 Detailed Description

Four byte data structure;

The documentation for this struct was generated from the following file:

- ini_helper.h

3.2 ini_file Struct Reference

```
#include <ini_config.h>
```

Public Attributes

- enum ini_utf8_mode **utf8_mode**
- FILE * **file**
- char **equals**
- char **comment**

3.2.1 Detailed Description

Data structure for .INI files.

The documentation for this struct was generated from the following file:

- ini_config.h

3.3 IniEntry Class Reference

```
#include <IniEntry.hpp>
```

Public Member Functions

- [IniEntry](#) (std::string key, std::string value="", std::string comment="")
- [IniEntry](#) (std::string key, float value, std::string comment="")
- [IniEntry](#) (std::string key, double value, std::string comment="")
- [IniEntry](#) (std::string key, long double value, std::string comment="")
- [IniEntry](#) (std::string key, int value, std::string comment="")
- [IniEntry](#) (std::string key, long value, std::string comment="")
- [IniEntry](#) (std::string key, long long value, std::string comment="")
- [IniEntry](#) (std::string key, unsigned long value, std::string comment="")
- [IniEntry](#) (std::string key, unsigned long long value, std::string comment="")
- [IniEntry operator=](#) (const std::string &other)
- [IniEntry operator=](#) (const float &other)
- [IniEntry operator=](#) (const double &other)
- [IniEntry operator=](#) (const long double &other)
- [IniEntry operator=](#) (const int &other)
- [IniEntry operator=](#) (const long &other)
- [IniEntry operator=](#) (const long long &other)
- [IniEntry operator=](#) (const unsigned &other)
- [IniEntry operator=](#) (const unsigned long &other)
- [IniEntry operator=](#) (const unsigned long long &other)
- bool [operator==](#) (const [IniEntry](#) &other) const
- bool [operator!=](#) (const [IniEntry](#) &other) const
- void [setValue](#) (const std::string &other)
- void [setValue](#) (const float &other)
- void [setValue](#) (const double &other)
- void [setValue](#) (const long double &other)
- void [setValue](#) (const int &other)
- void [setValue](#) (const long &other)
- void [setValue](#) (const long long &other)
- void [setValue](#) (const unsigned &other)
- void [setValue](#) (const unsigned long &other)
- void [setValue](#) (const unsigned long long &other)
- std::string [getKey](#) () const
- std::string [getString](#) () const
- float [getFloat](#) () const
- double [getDouble](#) () const
- long double [getLongDouble](#) () const
- int [getInt](#) () const
- long [getLong](#) () const
- long long [getLongLong](#) () const
- unsigned long [getUnsignedLong](#) () const
- unsigned long long [getUnsignedLongLong](#) () const
- void [setComment](#) (std::string comment)
- std::string [getComment](#) ()

3.3.1 Detailed Description

Class for storing entries.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 IniEntry() [1/9]

```
IniEntry::IniEntry (
    std::string key,
    std::string value = "",
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>might</i>	throw exceptions (see std::string (constructor))
--------------	--

3.3.2.2 IniEntry() [2/9]

```
IniEntry::IniEntry (
    std::string key,
    float value,
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>std::bad_alloc</i>	or other (see std::string (constructor))
-----------------------	--

3.3.2.3 IniEntry() [3/9]

```
IniEntry::IniEntry (
    std::string key,
    double value,
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>std::bad_alloc</i>	or other (see std::string (constructor))
-----------------------	--

3.3.2.4 IniEntry() [4/9]

```
IniEntry::IniEntry (
    std::string key,
    long double value,
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>std::bad_alloc</i>	or other (see std::string (constructor))
-----------------------	--

3.3.2.5 IniEntry() [5/9]

```
IniEntry::IniEntry (
    std::string key,
```

```
int value,  
std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>std::bad_alloc</i>	or other (see std::string (constructor))
-----------------------	--

3.3.2.6 IniEntry() [6/9]

```
IniEntry::IniEntry (  
    std::string key,  
    long value,  
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>std::bad_alloc</i>	or other (see std::string (constructor))
-----------------------	--

3.3.2.7 IniEntry() [7/9]

```
IniEntry::IniEntry (  
    std::string key,  
    long long value,  
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>std::bad_alloc</i>	or other (see <code>std::string</code> (constructor))
-----------------------	---

3.3.2.8 IniEntry() [8/9]

```
IniEntry::IniEntry (  
    std::string key,  
    unsigned long value,  
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>std::bad_alloc</i>	or other (see <code>std::string</code> (constructor))
-----------------------	---

3.3.2.9 IniEntry() [9/9]

```
IniEntry::IniEntry (  
    std::string key,  
    unsigned long long value,  
    std::string comment = "" ) [inline]
```

Constructor for the entry.

Parameters

<i>key</i>	
<i>value</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<code>std::bad_alloc</code>	or other (see <code>std::string</code> (constructor))
-----------------------------	---

3.3.3 Member Function Documentation**3.3.3.1 getComment()**

```
std::string IniEntry::getComment ( ) [inline]
```

Getter for the comment.

Returns

comment

3.3.3.2 getDouble()

```
double IniEntry::getDouble ( ) const [inline]
```

Getter for the value (as double).

Returns

value

Exceptions

<code>std::invalid_argument</code>	on conversion failure and <code>std::out_of_range</code> if the value is out of range
------------------------------------	---

3.3.3.3 getFloat()

```
float IniEntry::getFloat ( ) const [inline]
```

Getter for the value (as float).

Returns

value

Exceptions

<i>std::invalid_argument</i>	on conversion failure and <i>std::out_of_range</i> if the value is out of range
------------------------------	---

3.3.3.4 getInt()

```
int IniEntry::getInt ( ) const [inline]
```

Getter for the value (as int).

Returns

value

Exceptions

<i>std::invalid_argument</i>	on conversion failure and <i>std::out_of_range</i> if the value is out of range
------------------------------	---

3.3.3.5 getKey()

```
std::string IniEntry::getKey ( ) const [inline]
```

Getter for the key.

Returns

key

3.3.3.6 getLong()

```
long IniEntry::getLong ( ) const [inline]
```

Getter for the value (as long).

Returns

value

Exceptions

<i>std::invalid_argument</i>	on conversion failure and <i>std::out_of_range</i> if the value is out of range
------------------------------	---

3.3.3.7 getLongDouble()

```
long double IniEntry::getLongDouble ( ) const [inline]
```

Getter for the value (as long double).

Returns

value

Exceptions

<i>std::invalid_argument</i>	on conversion failure and <i>std::out_of_range</i> if the value is out of range
------------------------------	---

3.3.3.8 getLongLong()

```
long long IniEntry::getLongLong ( ) const [inline]
```

Getter for the value (as long long).

Returns

value

Exceptions

<i>std::invalid_argument</i>	on conversion failure and <i>std::out_of_range</i> if the value is out of range
------------------------------	---

3.3.3.9 getString()

```
std::string IniEntry::getString ( ) const [inline]
```

Getter for the value (as string).

Returns

value

3.3.3.10 getUnsignedLong()

```
unsigned long IniEntry::getUnsignedLong ( ) const [inline]
```

Getter for the value (as unsigned long).

Returns

Exceptions

<i>std::invalid_argument</i>	on conversion failure and <i>std::out_of_range</i> if the value is out of range
------------------------------	---

3.3.3.11 getUnsignedLongLong()

```
unsigned long long IniEntry::getUnsignedLongLong ( ) const [inline]
```

Getter for the value (as unsigned long long).

Returns

value

Exceptions

<i>std::invalid_argument</i>	on conversion failure and <i>std::out_of_range</i> if the value is out of range
------------------------------	---

3.3.3.12 operator!=(())

```
bool IniEntry::operator!= (
    const IniEntry & other ) const [inline]
```

Unequal comparison operator.

Parameters

<i>value</i>	
--------------	--

Returns

false if the values are equal

3.3.3.13 operator=() [1/10]

```
IniEntry IniEntry::operator= (
    const std::string & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

3.3.3.14 operator=() [2/10]

```
IniEntry IniEntry::operator= (
    const float & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.15 operator=() [3/10]

```
IniEntry IniEntry::operator= (
    const double & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

Exceptions

<code>std::bad_alloc</code>	
-----------------------------	--

3.3.3.16 operator=() [4/10]

```
IniEntry IniEntry::operator= (
    const long double & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<code>value</code>	
--------------------	--

Returns

current object

Exceptions

<code>std::bad_alloc</code>	
-----------------------------	--

3.3.3.17 operator=() [5/10]

```
IniEntry IniEntry::operator= (
    const int & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<code>value</code>	
--------------------	--

Returns

current object

Exceptions

<code>std::bad_alloc</code>	
-----------------------------	--

3.3.3.18 operator=() [6/10]

```
IniEntry IniEntry::operator= (
    const long & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

Exceptions

<code>std::bad_alloc</code>	
-----------------------------	--

3.3.3.19 operator=() [7/10]

```
IniEntry IniEntry::operator= (
    const long long & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

Exceptions

<code>std::bad_alloc</code>	
-----------------------------	--

3.3.3.20 operator=() [8/10]

```
IniEntry IniEntry::operator= (
    const unsigned & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.21 operator=() [9/10]

```
IniEntry IniEntry::operator= (
    const unsigned long & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.22 operator=() [10/10]

```
IniEntry IniEntry::operator= (
```



```
const unsigned long long & other ) [inline]
```

Equals operator which also assigns a new value.

Parameters

<i>value</i>	
--------------	--

Returns

current object

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.23 operator==()

```
bool IniEntry::operator== (
    const IniEntry & other ) const [inline]
```

Comparison operator.

Parameters

<i>value</i>	
--------------	--

Returns

true if the values are equal

3.3.3.24 setComment()

```
void IniEntry::setComment (
    std::string comment ) [inline]
```

Setter for the comment.

Returns

comment

3.3.3.25 setValue() [1/10]

```
void IniEntry::setValue (
    const std::string & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.26 setValue() [2/10]

```
void IniEntry::setValue (
    const float & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.27 setValue() [3/10]

```
void IniEntry::setValue (
    const double & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.28 setValue() [4/10]

```
void IniEntry::setValue (
```

```
const long double & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.29 setValue() [5/10]

```
void IniEntry::setValue (
    const int & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.30 setValue() [6/10]

```
void IniEntry::setValue (
    const long & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.31 setValue() [7/10]

```
void IniEntry::setValue (
    const long long & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.32 setValue() [8/10]

```
void IniEntry::setValue (
    const unsigned & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.33 setValue() [9/10]

```
void IniEntry::setValue (
    const unsigned long & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

3.3.3.34 setValue() [10/10]

```
void IniEntry::setValue (
    const unsigned long long & other ) [inline]
```

Setter for the value.

Parameters

<i>value</i>	
--------------	--

Exceptions

<i>std::bad_alloc</i>	
-----------------------	--

The documentation for this class was generated from the following file:

- IniEntry.hpp

3.4 IniFile Class Reference

```
#include <IniFile.hpp>
```

Public Member Functions

- [IniFile](#) (std::string comment="")
- bool [operator==](#) (const [IniFile](#) &other) const
- bool [operator!=](#) (const [IniFile](#) &other) const
- size_t [size](#) () const
- bool [empty](#) () const
- std::vector< [IniSection](#) > [getAllSections](#) () const
- [IniSection](#) & [at](#) (const std::string &name="")
- [IniSection](#) & [operator\[\]](#) (const std::string &name)
- void [clean](#) ()
- void [clear](#) ()
- void [insert](#) (const [IniSection](#) §ion)
- void [erase](#) (const std::string §ionName)
- void [setComment](#) (std::string comment)
- std::string [getComment](#) ()

3.4.1 Detailed Description

Class to hold INI data.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 IniFile()

```
IniFile::IniFile (  
    std::string comment = "" ) [inline]
```

Constructor for the INI file.

Parameters

<i>comment</i>	(optional)
----------------	------------

Exceptions

<i>might</i>	throw exceptions (see std::string constructor)
--------------	--

3.4.3 Member Function Documentation

3.4.3.1 at()

```
IniSection& IniFile::at (  
    const std::string & name = "" ) [inline]
```

Returns the section with the given name.

Parameters

<i>section</i>	<i>name</i>
----------------	-------------

Returns

section

Exceptions

<i>might</i>	throw exceptions (see std::string construtor)
--------------	---

3.4.3.2 clean()

```
void IniFile::clean ( ) [inline]
```

cleans out all empty sections and empty entries.

3.4.3.3 clear()

```
void IniFile::clear ( ) [inline]
```

Deletes all sections.

3.4.3.4 empty()

```
bool IniFile::empty ( ) const [inline]
```

Tests if there are currently any sections loaded.

Returns

true, if there are sections present

3.4.3.5 erase()

```
void IniFile::erase (
    const std::string & sectionName ) [inline]
```

Erases a section with a given name.

Parameters

<i>section</i>	
----------------	--

3.4.3.6 getAllSections()

```
std::vector<IniSection> IniFile::getAllSections ( ) const [inline]
```

Returns an std::vector of all currently loaded sections

Returns

sections

3.4.3.7 getComment()

```
std::string IniFile::getComment ( ) [inline]
```

Getter for the comment.

Returns

comment

3.4.3.8 insert()

```
void IniFile::insert (
    const IniSection & section ) [inline]
```

Inserts a section.

Parameters

<i>section</i>	
----------------	--

Exceptions

<i>might</i>	throw exceptions (see std::vector::push_back)
--------------	---

3.4.3.9 operator!=(())

```
bool IniFile::operator!= (
    const IniFile & other ) const [inline]
```

Unequal comparison operator.

Parameters

<i>file</i>	
-------------	--

Returns

false if the values are equal

3.4.3.10 operator==()

```
bool IniFile::operator==(
    const IniFile & other ) const [inline]
```

Comparison operator.

Parameters

<i>file</i>	
-------------	--

Returns

true if the values are equal

3.4.3.11 operator[]()

```
IniSection& IniFile::operator[] (
    const std::string & name ) [inline]
```

Returns the section with the given name.

Parameters

<i>section</i>	name
----------------	------

Returns

section

Exceptions

<i>might</i>	throw exceptions (see std::string construtor)
--------------	---

3.4.3.12 setComment()

```
void IniFile::setComment (
    std::string comment ) [inline]
```

Setter for the comment.

Parameters

<i>comment</i>	
----------------	--

3.4.3.13 size()

```
size_t IniFile::size ( ) const [inline]
```

Returns the number of currently loaded sections.

Returns

number of sections

The documentation for this class was generated from the following file:

- [IniFile.hpp](#)

3.5 IniParser Class Reference

Classes

- class [ParserException](#)

Static Public Member Functions

- static [IniFile](#) & [load](#) (std::string path, enum ini_utf8_mode utf8_mode=INI_UTF8_MODE_FORBID, char equals='=', char comment=';')
- static void [store](#) ([IniFile](#) ini, std::string path, enum ini_utf8_mode utf8_mode=INI_UTF8_MODE_FORBID, char equals='=', char comment=';')
- static int [getError](#) ()
- static std::string [getErrorString](#) (int error)

Static Public Attributes

- static const int [NO_ERROR](#) = 0
- static const int [READ_ERROR](#) = -1
- static const int [WRITE_ERROR](#) = -2
- static const int [INVALID_STRING_ERROR](#) = -3

3.5.1 Member Function Documentation

3.5.1.1 `getError()`

```
static int IniParser::getError ( ) [static]
```

Getter for the error. This error can be converted to a string using the [getErrorString\(\)](#) method.

Returns

error

3.5.1.2 `getErrorString()`

```
static std::string IniParser::getErrorString (
    int error ) [static]
```

Returns an error string for the given error code.

Parameters

<i>error</i>	code
--------------	------

Returns

error string

Exceptions

<i>might</i>	throw exceptions (see <code>std::string</code> constructor)
--------------	---

3.5.1.3 `load()`

```
static IniFile& IniParser::load (  
    std::string path,  
    enum ini_utf8_mode utf8_mode = INI_UTF8_MODE_FORBID,  
    char equals = '=',  
    char comment = ';' ) [static]
```

Loads an INI file at a given path. If this path is empty, an empty class is returned. If this function fails, the error code can be accessed using the [getError\(\)](#) method.

Parameters

<i>path</i>	
<i>UTF8</i>	mode
<i>equals</i>	character
<i>comment</i>	character
<i>comment</i>	(stored at the beginning of the file when store() is called)

Exceptions

<i>a</i>	parser_exception and might throw others as well (see <code>std::string</code> constructor)
----------	--

Returns

the loaded ini file

3.5.1.4 `store()`

```
static void IniParser::store (  
    IniFile ini,
```

```
std::string path,
enum ini_utf8_mode utf8_mode = INI_UTF8_MODE_FORBID,
char equals = '=',
char comment = ';' ) [static]
```

Stores an INI file at a given path. If this path is empty, nothing is done. If this function fails, the error code can be accessed using the [getError\(\)](#) method.

Parameters

<i>path</i>	
<i>UTF8</i>	mode
<i>equals</i>	character
<i>comment</i>	character
<i>comment</i>	(stored at the beginning of the file when store() is called)

Exceptions

<i>a</i>	parser_exception and might throw others as well (see std::string constructor)
----------	---

3.5.2 Member Data Documentation

3.5.2.1 NO_ERROR

```
const int IniParser::NO_ERROR = 0 [static]
```

Constants representing the error state.

The documentation for this class was generated from the following file:

- IniParser.hpp

3.6 IniSection Class Reference

```
#include <IniSection.hpp>
```

Public Member Functions

- [IniSection](#) (std::string name="", std::string comment="")
- [size_t](#) [size](#) () const
- [bool](#) [empty](#) () const
- [std::vector](#)< [IniEntry](#) > [getAllEntries](#) () const
- [std::string](#) [getName](#) () const
- [IniEntry](#) & [at](#) (const std::string &key)
- [IniEntry](#) & [operator\[\]](#) (const std::string &key)
- [bool](#) [operator==](#) (const [IniSection](#) &other) const
- [bool](#) [operator!=](#) (const [IniSection](#) &other) const
- [void](#) [clean](#) ()
- [void](#) [clear](#) ()
- [void](#) [insert](#) ([IniEntry](#) entry)
- [void](#) [erase](#) (const std::string &key)
- [void](#) [setComment](#) (std::string comment)
- [std::string](#) [getComment](#) ()

3.6.1 Detailed Description

Class for storing sections.

3.6.2 Constructor & Destructor Documentation

3.6.2.1 IniSection()

```
IniSection::IniSection (
    std::string name = "",
    std::string comment = "" ) [inline]
```

Constructor for the section.

Parameters

<i>name</i>	(optional)
<i>comment</i>	(optional)

Exceptions

<i>might</i>	throw exceptions (see std::string (constructor))
--------------	--

3.6.3 Member Function Documentation

3.6.3.1 at()

```
IniEntry& IniSection::at (
    const std::string & key ) [inline]
```

Getter for the value of an entry with a given name.

Parameters

<i>entry</i>	
--------------	--

Exceptions

<i>might</i>	throw exceptions (see <code>std::vector::push_back</code>)
--------------	---

3.6.3.2 clean()

```
void IniSection::clean ( ) [inline]
```

Deletes all entries without a value.

3.6.3.3 clear()

```
void IniSection::clear ( ) [inline]
```

Deletes all entries.

3.6.3.4 empty()

```
bool IniSection::empty ( ) const [inline]
```

Returns whether the section is empty or not

Parameters

<i>true,if</i>	it is empty
----------------	-------------

3.6.3.5 erase()

```
void IniSection::erase (
    const std::string & key ) [inline]
```

Erases an entry with a given name.

Parameters

<i>entry</i>	
--------------	--

3.6.3.6 getAllEntries()

```
std::vector<IniEntry> IniSection::getAllEntries ( ) const [inline]
```

Returns a std::vector of all entries.

Parameters

<i>all</i>	entries
------------	---------

3.6.3.7 getComment()

```
std::string IniSection::getComment ( ) [inline]
```

Getter for the comment.

Returns

comment

3.6.3.8 getName()

```
std::string IniSection::getName ( ) const [inline]
```

Getter for the name.

Parameters

<i>name</i>	
-------------	--

3.6.3.9 insert()

```
void IniSection::insert (
    IniEntry entry ) [inline]
```

Inserts an entry.

Parameters

<i>entry</i>	
--------------	--

Exceptions

<i>might</i>	throw exceptions (see <code>std::vector::push_back</code>)
--------------	---

3.6.3.10 operator!=()

```
bool IniSection::operator!= (
    const IniSection & other ) const [inline]
```

Unequal comparison operator.

Parameters

<i>section</i>	
----------------	--

Returns

false if the values are equal

3.6.3.11 operator==()

```
bool IniSection::operator== (
    const IniSection & other ) const [inline]
```

Comparison operator.

Parameters

<i>section</i>	
----------------	--

Returns

true if the values are equal

3.6.3.12 operator[]()

```
IniEntry& IniSection::operator[] (
    const std::string & key ) [inline]
```

Getter for the value of an entry with a given name.

Parameters

<i>entry</i>	
--------------	--

Exceptions

<i>might</i>	throw exceptions (see <code>std::vector::push_back</code>)
--------------	---

3.6.3.13 setComment()

```
void IniSection::setComment (
    std::string comment ) [inline]
```

Setter for the comment.

Parameters

<i>comment</i>	
----------------	--

3.6.3.14 size()

```
size_t IniSection::size ( ) const [inline]
```

Returns the number of entries.

Parameters

<i>number</i>	of entries
---------------	------------

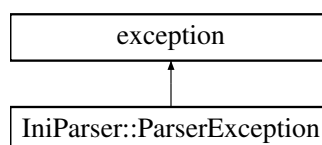
The documentation for this class was generated from the following file:

- IniSection.hpp

3.7 IniParser::ParserException Class Reference

```
#include <IniParser.hpp>
```

Inheritance diagram for IniParser::ParserException:



3.7.1 Detailed Description

Class for parser exceptions.

The documentation for this class was generated from the following file:

- IniParser.hpp

3.8 two_bytes Struct Reference

```
#include <ini_helper.h>
```

Public Attributes

- unsigned char **bytes** [2]

3.8.1 Detailed Description

Two byte data structure;

The documentation for this struct was generated from the following file:

- ini_helper.h

Index

- at
 - IniFile, [24](#)
 - IniSection, [32](#)
- clean
 - IniFile, [24](#)
 - IniSection, [33](#)
- clear
 - IniFile, [25](#)
 - IniSection, [33](#)
- empty
 - IniFile, [25](#)
 - IniSection, [33](#)
- erase
 - IniFile, [25](#)
 - IniSection, [33](#)
- four_bytes, [5](#)
- getAllEntries
 - IniSection, [34](#)
- getAllSections
 - IniFile, [25](#)
- getComment
 - IniEntry, [11](#)
 - IniFile, [25](#)
 - IniSection, [34](#)
- getDouble
 - IniEntry, [11](#)
- getError
 - IniParser, [28](#)
- getErrorString
 - IniParser, [29](#)
- getFloat
 - IniEntry, [11](#)
- getInt
 - IniEntry, [12](#)
- getKey
 - IniEntry, [12](#)
- getLong
 - IniEntry, [12](#)
- getLongDouble
 - IniEntry, [13](#)
- getLongLong
 - IniEntry, [13](#)
- getName
 - IniSection, [34](#)
- getString
 - IniEntry, [13](#)
- getUnsignedLong
 - IniEntry, [13](#)
- getUnsignedLongLong
 - IniEntry, [14](#)
- ini_file, [5](#)
- IniEntry, [6](#)
 - getComment, [11](#)
 - getDouble, [11](#)
 - getFloat, [11](#)
 - getInt, [12](#)
 - getKey, [12](#)
 - getLong, [12](#)
 - getLongDouble, [13](#)
 - getLongLong, [13](#)
 - getString, [13](#)
 - getUnsignedLong, [13](#)
 - getUnsignedLongLong, [14](#)
 - IniEntry, [7–10](#)
 - operator!=, [14](#)
 - operator=, [14–18](#)
 - operator==, [19](#)
 - setComment, [19](#)
 - setValue, [19–23](#)
- IniFile, [23](#)
 - at, [24](#)
 - clean, [24](#)
 - clear, [25](#)
 - empty, [25](#)
 - erase, [25](#)
 - getAllSections, [25](#)
 - getComment, [25](#)
 - IniFile, [24](#)
 - insert, [26](#)
 - operator!=, [26](#)
 - operator==, [26](#)
 - operator[], [27](#)
 - setComment, [27](#)
 - size, [28](#)
- IniParser, [28](#)
 - getError, [28](#)
 - getErrorString, [29](#)
 - load, [30](#)
 - NO_ERROR, [31](#)
 - store, [30](#)
- IniParser::ParserException, [37](#)
- IniSection, [31](#)
 - at, [32](#)
 - clean, [33](#)
 - clear, [33](#)

- empty, [33](#)
- erase, [33](#)
- getAllEntries, [34](#)
- getComment, [34](#)
- getName, [34](#)
- IniSection, [32](#)
- insert, [34](#)
- operator!=, [35](#)
- operator==, [35](#)
- operator[], [35](#)
- setComment, [37](#)
- size, [37](#)
- insert
 - IniFile, [26](#)
 - IniSection, [34](#)
- load
 - IniParser, [30](#)
- NO_ERROR
 - IniParser, [31](#)
- operator!=
 - IniEntry, [14](#)
 - IniFile, [26](#)
 - IniSection, [35](#)
- operator=
 - IniEntry, [14–18](#)
- operator==
 - IniEntry, [19](#)
 - IniFile, [26](#)
 - IniSection, [35](#)
- operator[]
 - IniFile, [27](#)
 - IniSection, [35](#)
- setComment
 - IniEntry, [19](#)
 - IniFile, [27](#)
 - IniSection, [37](#)
- setValue
 - IniEntry, [19–23](#)
- size
 - IniFile, [28](#)
 - IniSection, [37](#)
- store
 - IniParser, [30](#)
- two_bytes, [38](#)