Common Utils

1.0

Generated by Doxygen 1.8.11

Contents

Chapter 1

utility

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

Namespace Documentation

4.1 cbtek Namespace Reference

4.1.1 Detailed Description

Chapter 5

Class Documentation

5.1 cbtek::common::utility::StringUtils::CaseInsensitiveEqual< T > Struct Template Reference

The CaseInsensitiveEqual struct.

`#include <StringUtils.hpp>`

Public Member Functions

- **CaseInsensitiveEqual** (const T &str)
- bool **operator()** (const T &strIn)

5.1.1 Detailed Description

template<typename T>
struct cbtek::common::utility::StringUtils::CaseInsensitiveEqual< T >

The CaseInsensitiveEqual struct.

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

- common/utility/inc/StringUtils.hpp

5.2 cbtek::common::utility::Color Class Reference

Public Member Functions

- Color ()

  Color Constructor for this class.
- Color (std::uint32_t color)

  Color Constructor for this class.
- Color (const std::string &rgbaHex)

  Color Constructor for this class.
- Color (const std::string &name, const uint8_t &red, const uint8_t &green, const uint8_t &blue, const uint8_t &alpha=255)

  Color Constructor for this class.
- Color (const uint8_t &red, const uint8_t &green, const uint8_t &blue, const uint8_t &alpha=255)

  Color Constructor for this class.
- void set (const uint8_t &red, const uint8_t &green, const uint8_t &blue, const uint8_t &alpha=255)

  set Sets the color using rgba components
- void set (const std::string &htmlColor)

  set Sets the color from a html string
- void setRed (const uint8_t &red)

  setRed Sets the red component
- void setGreen (const uint8_t &green)

  setGreen Sets the green component
- void setBlue (const uint8_t &blue)

  setBlue Sets the blue component
- void setAlpha (const uint8_t &alpha)

  setAlpha Sets the alpha component
- std::uint32_t toInteger () const

  toInteger Converts color to 32-bit numeric value
- bool operator== (const Color &color) const

  operator == Determine if two colors are equal
- bool operator!= (const Color &color) const

  operator != Determine if two colors are not equal
- uint8_t getRed () const

  getRed Gets the red component
- uint8_t getGreen () const

  getGreen Gets the green component
- uint8_t getBlue () const

  getBlue Gets the blue component
- uint8_t getAlpha () const

  getAlpha Gets the alpha component
- bool isTransparent () const

  isTransparent Return true if color can be considered transparent
- std::string toString (const ColorStringStyle &style=ColorStringStyle::RGBA_255) const

  toString Convert color to string
- std::string getName () const

  getName Get the name of this color
- void setName (const std::string &name)

  setName Sets name for current color

5.2.1 Constructor & Destructor Documentation

5.2.1.1 cbtek::common::utility::Color::Color (std::uint32_t *color*)

Color Constructor for this class.

Parameters

color	32-bit color defined as 8 bit components of RGBA

5.2.1.2 cbtek::common::utility::Color::Color (const std::string & *rgbaHex*)

Color Constructor for this class.

Parameters

rgbaHex	Hexadecimal string representing the color to be set upon construction

5.2.1.3 cbtek::common::utility::Color::Color (const std::string & *name*, const uint8_t & *red*, const uint8_t & *green*, const uint8_t & *blue*, const uint8_t & *alpha* = `255`)

Color Constructor for this class.

Parameters

name	String name/id for the color
red	8-bit red component for this color [0-255]
green	8-bit green component for this color [0-255]
blue	8-bit blue component for this color [0-255]
alpha	8-bit opacity component for this color [0-255] Higher value results in less transparency

5.2.1.4 cbtek::common::utility::Color::Color (const uint8_t & *red*, const uint8_t & *green*, const uint8_t & *blue*, const uint8_t & *alpha* = `255`)

Color Constructor for this class.

Parameters

red	8-bit red component for this color [0-255]
green	8-bit green component for this color [0-255]
blue	8-bit blue component for this color [0-255]
alpha	8-bit opacity component for this color [0-255] Higher value results in less transparency

5.2.2 Member Function Documentation

5.2.2.1 uint8_t cbtek::common::utility::Color::getAlpha ( ) const

getAlpha Gets the alpha component

Returns

8-bit alpha component value

5.2.2.2 uint8_t cbtek::common::utility::Color::getBlue ( ) const

getBlue Gets the blue component

Returns

8-bit blue component value

5.2.2.3 uint8_t cbtek::common::utility::Color::getGreen ( ) const

getGreen Gets the green component

Returns

8-bit green component value

5.2.2.4 std::string cbtek::common::utility::Color::getName ( ) const

getName Get the name of this color

Returns

Return name of color

5.2.2.5 uint8_t cbtek::common::utility::Color::getRed ( ) const

getRed Gets the red component

Returns

8-bit red component value

5.2.2.6 bool cbtek::common::utility::Color::isTransparent ( ) const

isTransparent Return true if color can be considered transparent

Returns

Return true if transparent, false otherwise

5.2.2.7 bool cbtek::common::utility::Color::operator!= (const Color & *color*) const

operator != Determine if two colors are not equal

Parameters

color	The color to test against

Returns

Return true if colors don't match, false otherwise

5.2.2.8 bool cbtek::common::utility::Color::operator== (const Color & *color*) const

operator == Determine if two colors are equal

Parameters

color	The color to test against

Returns

Return true if colors match, false otherwise

5.2.2.9 void cbtek::common::utility::Color::set (const uint8_t & *red,* const uint8_t & *green,* const uint8_t & *blue,* const uint8_t & *alpha* = `255`)

set Sets the color using rgba components

Parameters

red	8-bit red component for this color [0-255]
green	8-bit green component for this color [0-255]
blue	8-bit blue component for this color [0-255]
alpha	8-bit opacity component for this color [0-255] Higher value results in less transparency

5.2.2.10 void cbtek::common::utility::Color::set (const std::string & *htmlColor*)

set Sets the color from a html string

Parameters

htmlColor	The string containing color information (e.g #AAFFCC, #abc, #ef41af)

5.2.2.11 void cbtek::common::utility::Color::setAlpha (const uint8_t & *alpha*)

setAlpha Sets the alpha component

Parameters

Alpha	intensity value for the alpha component [0-255]

5.2.2.12 void cbtek::common::utility::Color::setBlue (const uint8_t & *blue*)

setBlue Sets the blue component

Parameters

Blue	intensity value for the blue component [0-255]

5.2.2.13 void cbtek::common::utility::Color::setGreen (const uint8_t & *green*)

setGreen Sets the green component

Parameters

Green	intensity value for the green component [0-255]

5.2.2.14 void cbtek::common::utility::Color::setName (const std::string & *name*)

setName Sets name for current color

Parameters

name	String representing name

5.2.2.15 void cbtek::common::utility::Color::setRed (const uint8_t & *red*)

setRed Sets the red component

Parameters

Red	intensity value for the red component [0-255]

5.2.2.16 std::uint32_t cbtek::common::utility::Color::toInteger ( ) const

toInteger Converts color to 32-bit numeric value

Returns

Integer of converted color

5.2.2.17 std::string cbtek::common::utility::Color::toString (const ColorStringStyle & *style* = `ColorStringStyle::RGBA_255`) const

toString Convert color to string

Parameters

style	Use string style in ColorStringStyle

Returns

Converted string

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

- common/utility/inc/ColorUtils.h
- common/utility/src/ColorUtils.cpp

5.3 cbtek::common::utility::ColorFactory Class Reference

Static Public Member Functions

- static Color create (const std::string &rgba)
  create
- static Color create (const colors::ColorType &color)
  create
- static Color create (const float &r, const float &g, const float &b, const float &a=1.0f)
  create
- static Color createNextColor ()
  createNextColor
- static Color createRandomNamedColor ()
  createRandomNamedColor
- static Color createRandomColor ()
  createRandomColor
- static Color createRandomDarkColor ()
  createRandomDarkColor
- static Color createRandomLightColor ()
  createRandomLightColor
- static void resetNextColor ()
  resetNextColor

5.3.1 Member Function Documentation

5.3.1.1 Color cbtek::common::utility::ColorFactory::create (const std::string & *rgba*) [static]

create

Parameters

rgba	

Returns

5.3.1.2 Color cbtek::common::utility::ColorFactory::create (const colors::ColorType & *color*) [static]

create

Parameters

color	

Returns

5.3.1.3 Color cbtek::common::utility::ColorFactory::create (const float & *r*, const float & *g*, const float & *b*, const float & *a* = 1.0f) [static]

create

Parameters

r	
g	
b	
a	

Returns

5.3.1.4 Color cbtek::common::utility::ColorFactory::createNextColor ( ) [static]

createNextColor

Returns

5.3.1.5 Color cbtek::common::utility::ColorFactory::createRandomColor ( ) [static]

createRandomColor

Returns

5.3.1.6 Color cbtek::common::utility::ColorFactory::createRandomDarkColor ( ) [static]

createRandomDarkColor

Returns

5.3.1.7 Color cbtek::common::utility::ColorFactory::createRandomLightColor ( ) [static]

createRandomLightColor

Returns

5.3.1.8 Color cbtek::common::utility::ColorFactory::createRandomNamedColor ( ) [static]

createRandomNamedColor

Returns

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

- common/utility/inc/ColorUtils.h
- common/utility/src/ColorUtils.cpp

5.4 cbtek::common::utility::ColorLoop Class Reference

Public Member Functions

- ColorLoop & operator<< (const Color &color)
  operator <<
- ColorLoop & operator<< (const colors::ColorType &colorType)
  operator <<
- ColorLoop & operator<< (const colorFunctions::ColorFunction &function)
  operator <<
- void generateUniqueColors (const size_t &count)
  generateUniqueColors
- Color getNextColor ()
  getNextColor
- void reset ()
  reset
- void clear ()
  clear
- Color getColorAt (size_t ndx) const
  getColorAt
- size_t getCurrentColorNdx () const
  getCurrentColorNdx
- void setCurrentColorNdx (size_t ndx)
  setCurrentColorNdx

5.4.1 Member Function Documentation

5.4.1.1 void cbtek::common::utility::ColorLoop::generateUniqueColors (const size_t & *count*)

generateUniqueColors

Parameters

count	

5.4.1.2 Color cbtek::common::utility::ColorLoop::getColorAt (size_t *ndx*) const

getColorAt

Parameters

ndx	

Returns

5.4.1.3 size_t cbtek::common::utility::ColorLoop::getCurrentColorNdx ( ) const

getCurrentColorNdx

Returns

5.4.1.4 Color cbtek::common::utility::ColorLoop::getNextColor ( )

getNextColor

Returns

5.4.1.5 ColorLoop & cbtek::common::utility::ColorLoop::operator<< (const Color & *color*)

operator <<

Parameters

color	

Returns

5.4.1.6 ColorLoop & cbtek::common::utility::ColorLoop::operator<< (const colors::ColorType & *colorType*)

operator <<

Parameters

colorType	

Returns

5.4.1.7 ColorLoop & cbtek::common::utility::ColorLoop::operator<< (const colorFunctions::ColorFunction & *function*)

operator <<

Parameters

function	

Returns

5.4.1.8 void cbtek::common::utility::ColorLoop::setCurrentColorNdx (size_t *ndx*)

setCurrentColorNdx

Parameters

ndx	

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

- common/utility/inc/ColorUtils.h
- common/utility/src/ColorUtils.cpp

5.5 cbtek::common::utility::DateEntity Class Reference

The DateEntity class represents a single date object for use in DateTimeUtils and DateUtils.

`#include <DateTimeUtils.hpp>`

Public Member Functions

- DateEntity ()
  DateEntity (Constructor) Default.
- DateEntity (const size_t &month, const size_t &day, const size_t &year)
  DateEntity (Constructor) Send in integer components for month, day and year.
- DateEntity (const size_t &dateInteger)
  DateEntity (Constructor) Send in integer containing date components in form of YYYYMMDD.
- size_t getMonth () const
  getMonth Gets the month value
- size_t getDay () const
  getDay Gets the day value
- size_t getYear () const
  getYear Gets the year value
- void setMonth (const size_t &month)
  setMonth Sets the month value
- void setDay (const size_t &day)
  setDay Sets the day value
- void setYear (const size_t &year)

setYear Sets the year value

- size_t toDateInteger () const

  toDateInteger Converts date object into unique integer
- bool operator< (const DateEntity &date) const

  operator< Checks if date is less than or equal to (∗this)
- bool operator<= (const DateEntity &date) const

  operator<= Checks if date is less than or equal to (∗this)
- bool operator> (const DateEntity &date) const

  operator> Checks if date is greater than to (∗this)
- bool operator>= (const DateEntity &date) const

  operator>= Checks if date is greater than or equal to (∗this)
- bool operator== (const DateEntity &date) const

  operator!= Checks if date is equal to (∗this)
- bool operator!= (const DateEntity &date) const

  operator!= Checks if date is not equal to (∗this)
- size_t operator- (const DateEntity &date) const

  operator- Get number of days between two dates
- bool isLeapYear () const

  isLeapYear Determine if current year is leap year
- size_t getDays () const

  getDays() Gets number of days since beginning of year

5.5.1 Detailed Description

The DateEntity class represents a single date object for use in DateTimeUtils and DateUtils.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 cbtek::common::utility::DateEntity::DateEntity (const size_t & *month*, const size_t & *day*, const size_t & *year*) [inline]

DateEntity (Constructor) Send in integer components for month, day and year.

Parameters

month	The month to set
day	The day to set
year	The year to set

5.5.2.2 cbtek::common::utility::DateEntity::DateEntity (const size_t & *dateInteger*) [inline]

DateEntity (Constructor) Send in integer containing date components in form of YYYYMMDD.

Parameters

dateInteger	The date components to set

5.5.3 Member Function Documentation

5.5.3.1 size_t cbtek::common::utility::DateEntity::getDay ( ) const [inline]

getDay Gets the day value

Returns

Returns the day value

5.5.3.2 size_t cbtek::common::utility::DateEntity::getDays ( ) const [inline]

getDays() Gets number of days since beginning of year

Returns

Total number of days since beginning of year

5.5.3.3 size_t cbtek::common::utility::DateEntity::getMonth ( ) const [inline]

getMonth Gets the month value

Returns

Returns the month value

5.5.3.4 size_t cbtek::common::utility::DateEntity::getYear ( ) const [inline]

getYear Gets the year value

Returns

Returns the year value

5.5.3.5 bool cbtek::common::utility::DateEntity::isLeapYear ( ) const [inline]

isLeapYear Determine if current year is leap year

Returns

Returns true if leap year, false otherwise

5.5.3.6 bool cbtek::common::utility::DateEntity::operator!= (const DateEntity & *date*) const [inline]

operator!= Checks if date is not equal to (*this)

Parameters

date	The date to compare against

Returns

True if dates are not equal, false otherwise

5.5.3.7 size_t cbtek::common::utility::DateEntity::operator- (const DateEntity & *date*) const [inline]

operator- Get number of days between two dates

Parameters

date	The second date to subtract from

Returns

Total number of days between this->m_day and date.getDay()

5.5.3.8 bool cbtek::common::utility::DateEntity::operator< (const DateEntity & *date*) const [inline]

operator< Checks if date is less than or equal to (*this)

Parameters

date	The date to compare against

Returns

True if (*this) is less than date, false otherwise

5.5.3.9 bool cbtek::common::utility::DateEntity::operator<= (const DateEntity & *date*) const [inline]

operator<= Checks if date is less than or equal to (*this)

Parameters

date	The date to compare against

Returns

True if (*this) is less than or equal to date, false otherwise

5.5.3.10 bool cbtek::common::utility::DateEntity::operator== (const DateEntity & *date*) const [inline]

operator!= Checks if date is equal to (*this)

Parameters

date	The date to compare against

Returns

True if dates are equal, false otherwise

5.5.3.11 bool cbtek::common::utility::DateEntity::operator> (const DateEntity & *date*) const [inline]

operator> Checks if date is greater than to (*this)

Parameters

date	The date to compare against

Returns

True if (*this) is greater than to date, false otherwise

5.5.3.12 bool cbtek::common::utility::DateEntity::operator>= (const DateEntity & *date*) const [inline]

operator>= Checks if date is greater than or equal to (*this)

Parameters

date	The date to compare against

Returns

True if (*this) is greater than or equal to date, false otherwise

5.5.3.13 void cbtek::common::utility::DateEntity::setDay (const size_t & *day*) [inline]

setDay Sets the day value

Parameters

day	The day to set

5.5.3.14 void cbtek::common::utility::DateEntity::setMonth (const size_t & *month*) [inline]

setMonth Sets the month value

Parameters

month	The month to set

5.5.3.15 void cbtek::common::utility::DateEntity::setYear (const size_t & *year*) [inline]

setYear Sets the year value

Parameters

year	The year to set

5.5.3.16 size_t cbtek::common::utility::DateEntity::toDateInteger ( ) const [inline]

toDateInteger Converts date object into unique integer

Returns

Return date as integer YYYYMMDD

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

- common/utility/inc/DateTimeUtils.hpp

5.6 cbtek::common::utility::DateTimeUtils Class Reference

Static Public Member Functions

- static std::string getDisplayTimeStamp (const DateEntity &dateEntity, const TimeEntity &timeEntity)
  getDisplayTimeStamp
- static std::string getDisplayTimeStamp ()
  getDisplayTimeStamp
- static std::uint64_t getTimeStampInteger (const DateEntity &dateEntity=DateUtils::getCurrentDate(), const TimeEntity &timeEntity=TimeUtils::getCurrentTime())
  getTimeStampInteger
- static std::string getTimeStamp ()
  getTimeStamp
- static std::string getTimeStamp (const DateEntity &dateEntity, const TimeEntity &timeEntity)
  getTimeStamp

5.6.1 Member Function Documentation

5.6.1.1 static std::string cbtek::common::utility::DateTimeUtils::getDisplayTimeStamp (const DateEntity & *dateEntity,* const TimeEntity & *timeEntity*) [inline], [static]

getDisplayTimeStamp

Parameters

dateEntity	
timeEntity	

Returns

5.6.1.2 static std::string cbtek::common::utility::DateTimeUtils::getDisplayTimeStamp ( ) [inline], [static]

getDisplayTimeStamp

Returns

5.6.1.3 static std::string cbtek::common::utility::DateTimeUtils::getTimeStamp ( ) [inline], [static]

getTimeStamp

Returns

5.6.1.4 static std::string cbtek::common::utility::DateTimeUtils::getTimeStamp (const DateEntity & *dateEntity,* const TimeEntity & *timeEntity*) [inline], [static]

getTimeStamp

Parameters

dateEntity	
timeEntity	

Returns

5.6.1.5 static std::uint64_t cbtek::common::utility::DateTimeUtils::getTimeStampInteger (const DateEntity & *dateEntity* = DateUtils::getCurrentDate (), const TimeEntity & *timeEntity* = TimeUtils::getCurrentTime ()) [inline], [static]

getTimeStampInteger

Parameters

dateEntity	
timeEntity	

Returns

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

- common/utility/inc/DateTimeUtils.hpp

5.7 cbtek::common::utility::DateTimeUtils_EmbeddedUtils Class Reference

This embedded class contains utilities in support of DateTimeUtils Although the DRY is violated, this class can stay dependency free and be used freely without needing external support.

`#include <DateTimeUtils.hpp>`

Static Public Member Functions

- static void toUpperInPlace (std::string &str)

  Converts string to upper-case.
- static std::string toUpper (const std::string &str)

  Converts string to upper-case.
- static std::string replace (const std::string &inputString, const std::string &oldString, const std::string &newString, bool isCaseSensitive=c_DEFAULT_CASE_SENSITIVE)

  Perfroms string replace (in-place)
- static bool contains (const std::string &srcStr, const std::string &subStr, bool caseSensitive=c_DEFAULT_CASE_SENSITIVE)

  contains Checks to see if subStr exists within srcStr
- template<typename T >
  static T toNumber (const std::string &value)

  toNumber Converts string value to numeric type

5.7.1 Detailed Description

This embedded class contains utilities in support of DateTimeUtils Although the DRY is violated, this class can stay dependency free and be used freely without needing external support.

5.7.2 Member Function Documentation

5.7.2.1 static bool cbtek::common::utility::DateTimeUtils_EmbeddedUtils::contains (const std::string & *srcStr,* const std::string & *subStr,* bool *caseSensitive* = `c_DEFAULT_CASE_SENSITIVE`) `[inline],[static]`

contains Checks to see if subStr exists within srcStr

Parameters

srcStr	The original source string
subStr	The substring to look for
caseSensitive	

Returns

Return true if subStr exists within srcStr, false otherwise

5.7.2.2 static std::string cbtek::common::utility::DateTimeUtils_EmbeddedUtils::replace (const std::string & *inputString,* const std::string & *oldString,* const std::string & *newString,* bool *isCaseSensitive* = c_DEFAULT_CASE_SENSITIVE) [inline],[static]

Perfroms string replace (in-place)

Parameters

inputString	The string that contains oldString to replace
oldString	The old string to replace
newString	The new string to insert
isCaseSensitive	Should oldString be caseSensitive within inputString?

Returns

inputString with the replaced oldString. If oldString is not found within inputString then inputString is returned unmodified.

5.7.2.3 template<typename T > static T cbtek::common::utility::DateTimeUtils_EmbeddedUtils::toNumber (const std::string & *value*) [inline],[static]

toNumber Converts string value to numeric type

Parameters

value	The string to convert

Returns

Returns numeric type of conversion

5.7.2.4 static std::string cbtek::common::utility::DateTimeUtils_EmbeddedUtils::toUpper (const std::string & *str*) [inline],[static]

Converts string to upper-case.

Parameters

The	string to convert

Returns

The converted string

5.7.2.5 static void cbtek::common::utility::DateTimeUtils_EmbeddedUtils::toUpperInPlace (std::string & *str*) [inline], [static]

Converts string to upper-case.

Parameters

Reference	to the string to convert

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

- common/utility/inc/DateTimeUtils.hpp

5.8 cbtek::common::utility::DateUtils Class Reference

The DateUtils class.

`#include <DateTimeUtils.hpp>`

Static Public Member Functions

- static std::string toShortDateString (const DateEntity &date, const std::string &format="mm-dd-yyyy")
  toShortDateString
- static std::string toLongDateString (const DateEntity &date)
  toLongDateString
- static DateEntity getCurrentDate ()
  getCurrentDate
- static std::string toCurrentLongDateString ()
  toCurrentLongDateString
- static std::string toCurrentShortDateString ()
  toCurrentShortDateString

5.8.1 Detailed Description

The DateUtils class.

5.8.2 Member Function Documentation

5.8.2.1 static DateEntity cbtek::common::utility::DateUtils::getCurrentDate ( ) [inline], [static]

getCurrentDate

Returns

5.8.2.2 static std::string cbtek::common::utility::DateUtils::toCurrentLongDateString ( ) [inline], [static]

toCurrentLongDateString

Returns

5.8.2.3 static std::string cbtek::common::utility::DateUtils::toCurrentShortDateString ( ) [inline], [static]

toCurrentShortDateString

Returns

5.8.2.4 static std::string cbtek::common::utility::DateUtils::toLongDateString (const DateEntity & *date*) [inline], [static]

toLongDateString

Parameters

date	

Returns

5.8.2.5 static std::string cbtek::common::utility::DateUtils::toShortDateString (const DateEntity & *date,* const std::string & *format* = "mm-dd-yyyy") [inline], [static]

toShortDateString

Parameters

date	
format	

Returns

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

- common/utility/inc/DateTimeUtils.hpp

5.9 cbtek::common::utility::Font Class Reference

Public Member Functions

- Font ()
  Font.
- Font (std::string fontFamily, size_t pointSize=10, bool isBold=false, bool isItalic=false, bool isUnderlined=false)
  Font.
- bool operator== (const Font &font)
  operator ==
- void setFontFamily (const std::string &fontFamily)
  setFontFamily
- void setPointSize (const size_t &pointSize)
  setPointSize
- void setBold (const bool &enabled)
  setBold
- void setItalic (const bool &enabled)
  setItalic
- void setUnderlined (const bool &enabled)
  setUnderlined
- bool isBold () const
  isBold
- bool isItalic () const
  isItalic
- bool isUnderlined () const
  isUnderlined
- std::string getFontFamily () const
  getFontFamily
- size_t getPointSize () const
  getPointSize
- std::string toString () const
  toString
- bool hasDefaultChanged () const
  hasDefaultChanged
- void setChanged (bool flag)
  setChanged

5.9.1 Constructor & Destructor Documentation

5.9.1.1 cbtek::common::utility::Font::Font (std::string *fontFamily*, size_t *pointSize* = `10`, bool *isBold* = `false`, bool *isItalic* = `false`, bool *isUnderlined* = `false`)

Font.

Parameters

fontFamily	
pointSize	
isBold	
isItalic	
isUnderlined	

5.9.2 Member Function Documentation

5.9.2.1 std::string cbtek::common::utility::Font::getFontFamily ( ) const

getFontFamily

Returns

5.9.2.2 size_t cbtek::common::utility::Font::getPointSize ( ) const

getPointSize

Returns

5.9.2.3 bool cbtek::common::utility::Font::hasDefaultChanged ( ) const

hasDefaultChanged

Returns

5.9.2.4 bool cbtek::common::utility::Font::isBold ( ) const

isBold

Returns

5.9.2.5 bool cbtek::common::utility::Font::isItalic ( ) const

isItalic

Returns

5.9.2.6 bool cbtek::common::utility::Font::isUnderlined ( ) const

isUnderlined

Returns

5.9.2.7 bool cbtek::common::utility::Font::operator== (const Font & *font*)

operator ==

Parameters

font	

Returns

5.9.2.8 void cbtek::common::utility::Font::setBold (const bool & *enabled*)

setBold

Parameters

enabled	

5.9.2.9 void cbtek::common::utility::Font::setChanged (bool *flag*)

setChanged

Parameters

flag	

5.9.2.10 void cbtek::common::utility::Font::setFontFamily (const std::string & *fontFamily*)

setFontFamily

Parameters

fontFamily	

5.9.2.11 void cbtek::common::utility::Font::setItalic (const bool & *enabled*)

setItalic

Parameters

enabled	

5.9.2.12 void cbtek::common::utility::Font::setPointSize (const size_t & *pointSize*)

setPointSize

Parameters

pointSize	

5.9.2.13 void cbtek::common::utility::Font::setUnderlined (const bool & *enabled*)

setUnderlined

Parameters

enabled	

5.9.2.14 std::string cbtek::common::utility::Font::toString ( ) const

toString

Returns

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

- common/utility/inc/FontUtils.h
- common/utility/src/FontUtils.cpp

5.10 cbtek::common::utility::FontFactory Class Reference

Static Public Member Functions

- static Font create (const fontStyle::FontStyle &style)

  create
- static Font create (const std::string &fontString)

  create

5.10.1 Member Function Documentation

5.10.1.1 Font cbtek::common::utility::FontFactory::create (const fontStyle::FontStyle & *style*) [static]

create

Parameters

style	

Returns

5.10.1.2 Font cbtek::common::utility::FontFactory::create (const std::string & *fontString*) [static]

create

Parameters

fontString	

Returns

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

- common/utility/inc/FontUtils.h
- common/utility/src/FontUtils.cpp

5.11 cbtek::common::utility::Random Class Reference

Public Member Functions

- Random (long seed=0xABCDEF)

Random.

- void reseed (long seed)

  reseed
- double random ()

  random
- int next (int min, int max)

  next
- int next (int max)

  next

5.11.1 Constructor & Destructor Documentation

5.11.1.1 cbtek::common::utility::Random::Random (long *seed* = `0xABCDEF`)

Random.

Parameters

seed	

5.11.2 Member Function Documentation

5.11.2.1 int cbtek::common::utility::Random::next (int *min*, int *max*)

next

Parameters

min	
max	

Returns

5.11.2.2 int cbtek::common::utility::Random::next (int *max*)

next

Parameters

max	

Returns

5.11.2.3 double cbtek::common::utility::Random::random ( )

random

Returns

5.11.2.4 void cbtek::common::utility::Random::reseed (long *seed*)

reseed

Parameters

seed	

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

- common/utility/inc/Random.h
- common/utility/src/Random.cpp

5.12 cbtek::common::utility::TimeEntity Class Reference

The TimeEntity class represents a single time object for use in DateTimeUtils and TimeUtils.

`#include <DateTimeUtils.hpp>`

Public Member Functions

- TimeEntity ()

  TimeEntity (Constructor) Default.
- TimeEntity (const size_t &time)

  TimeEntity (Constructor) Create time object with integer containing time components in the form of HHMMSS.
- TimeEntity (size_t hour, size_t minute, size_t second, size_t millisecond=0)

  TimeEntity (Constructor) Creates TimeEntity object with hour, minute, second and optional millisecond components.
- bool operator> (const TimeEntity &time) const

  operator> Checks if the time object is greater than the (∗this) object
- bool operator< (const TimeEntity &time) const

  operator< Checks if the time object is less than the (∗this) object
- bool operator<= (const TimeEntity &time) const

  operator<= Checks if the time object is less than or equal to (∗this) object
- bool operator>= (const TimeEntity &time) const

operator>= Checks if the time object is greater than or equal to (∗this) object

- bool operator== (const TimeEntity &time) const

  operator== Checks if the time object is equal to (∗this) object
- void setHour (const size_t &hour)

  setHour Sets the hour value
- void setMinute (const size_t &minute)

  setMinute Sets the minute value
- void setSecond (const size_t &second)

  setSecond Sets the second value
- void setMillisecond (const size_t &millisecond)

  setMillisecond Sets the millisecond value
- size_t getHour () const

  getHour Get the hour value
- size_t getMinute () const

  getMinute Get the minute value
- size_t getSecond () const

  getSecond Get the second value
- size_t getMillisecond () const

  getMillisecond Get the millisecond value
- size_t toTimeInteger () const

  toTimeInteger Returns integer of (∗this) time value in form of HHMMSS integer
- size_t getAsMinutes () const

  getAsMinutes Get total number of minutes
- size_t getAsSeconds () const

  getAsSeconds Get total number of seconds
- size_t getAsMilliseconds () const

  getAsMilliseconds Get total number of milliseconds
- size_t getAsMicroseconds () const

  getAsMicroseconds Get total number of microseconds

5.12.1 Detailed Description

The TimeEntity class represents a single time object for use in DateTimeUtils and TimeUtils.

5.12.2 Constructor & Destructor Documentation

5.12.2.1 cbtek::common::utility::TimeEntity::TimeEntity (const size_t & *time*) [inline]

TimeEntity (Constructor) Create time object with integer containing time components in the form of HHMMSS.

Parameters

time	Integer containing HHMMSS components

5.12.2.2 cbtek::common::utility::TimeEntity::TimeEntity (size_t *hour*, size_t *minute*, size_t *second*, size_t *millisecond* = 0) [inline]

TimeEntity (Constructor) Creates TimeEntity object with hour, minute, second and optional millisecond components.

Parameters

hour	The hour to set
minute	The minute to set
second	The second to set
millisecond	The millisecond to set

5.12.3 Member Function Documentation

5.12.3.1 size_t cbtek::common::utility::TimeEntity::getAsMicroseconds ( ) const [inline]

getAsMicroseconds Get total number of microseconds

Returns

Return integer of total number of microseconds

5.12.3.2 size_t cbtek::common::utility::TimeEntity::getAsMilliseconds ( ) const [inline]

getAsMilliseconds Get total number of milliseconds

Returns

Return integer of total number of milliseconds

5.12.3.3 size_t cbtek::common::utility::TimeEntity::getAsMinutes ( ) const [inline]

getAsMinutes Get total number of minutes

Returns

Return integer of total number of minutes

5.12.3.4 size_t cbtek::common::utility::TimeEntity::getAsSeconds ( ) const [inline]

getAsSeconds Get total number of seconds

Returns

Return integer of total number of seconds

5.12.3.5 size_t cbtek::common::utility::TimeEntity::getHour ( ) const [inline]

getHour Get the hour value

Returns

Value of m_hour

5.12.3.6 size_t cbtek::common::utility::TimeEntity::getMillisecond ( ) const [inline]

getMillisecond Get the millisecond value

Returns

Value of m_millisecond

5.12.3.7 size_t cbtek::common::utility::TimeEntity::getMinute ( ) const [inline]

getMinute Get the minute value

Returns

Value of m_minute

5.12.3.8 size_t cbtek::common::utility::TimeEntity::getSecond ( ) const [inline]

getSecond Get the second value

Returns

Value of m_second

5.12.3.9 bool cbtek::common::utility::TimeEntity::operator< (const TimeEntity & *time*) const [inline]

operator< Checks if the time object is less than the (*this) object

Parameters

time	The time object to compare against

Returns

True if (*this) is less than time, false otherwise

5.12.3.10 bool cbtek::common::utility::TimeEntity::operator<= (const TimeEntity & *time*) const [inline]

operator<= Checks if the time object is less than or equal to (*this) object

Parameters

time	The time object to compare against

Returns

True if (*this) is less than or equal to time, false otherwise

5.12.3.11 bool cbtek::common::utility::TimeEntity::operator== (const TimeEntity & *time*) const [inline]

operator== Checks if the time object is equal to (*this) object

Parameters

time	The time object to compare against

Returns

True if (*this) is equal to time, false otherwise

5.12.3.12 bool cbtek::common::utility::TimeEntity::operator> (const TimeEntity & *time*) const [inline]

operator> Checks if the time object is greater than the (*this) object

Parameters

time	The time object to compare against

Returns

True if (*this) is greater than time, false otherwise

5.12.3.13 bool cbtek::common::utility::TimeEntity::operator>= (const TimeEntity & *time*) const [inline]

operator>= Checks if the time object is greater than or equal to (*this) object

Parameters

time	The time object to compare against

Returns

True if (∗this) is greater than or equal to time, false otherwise

5.12.3.14 void cbtek::common::utility::TimeEntity::setHour (const size_t & *hour*) [inline]

setHour Sets the hour value

Parameters

hour	The hour to set

5.12.3.15 void cbtek::common::utility::TimeEntity::setMillisecond (const size_t & *millisecond*) [inline]

setMillisecond Sets the millisecond value

Parameters

millisecond	The millisecond to set

5.12.3.16 void cbtek::common::utility::TimeEntity::setMinute (const size_t & *minute*) [inline]

setMinute Sets the minute value

Parameters

minute	The minute to set

5.12.3.17 void cbtek::common::utility::TimeEntity::setSecond (const size_t & *second*) [inline]

setSecond Sets the second value

Parameters

second	The second to set

5.12.3.18 size_t cbtek::common::utility::TimeEntity::toTimeInteger ( ) const [inline]

toTimeInteger Returns integer of (∗this) time value in form of HHMMSS integer

Returns

Return integer of current time components

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

- common/utility/inc/DateTimeUtils.hpp

5.13 cbtek::common::utility::TimeUtils Class Reference

The TimeUtils class contains useful functions for dealing the TimeEntity class.

`#include <DateTimeUtils.hpp>`

Static Public Member Functions

- template<typename T >
  static std::string toString (T value)
  toString
- static uint64_t getSecondsNow ()
  getSecondsNow
- static double getNanosecondsNow ()
  getNanosecondsNow
- static double getMicrosecondsNow ()
  getMicrosecondsNow
- static std::uint64_t getCurrentMilliseconds ()
  getCurrentMilliseconds
- static double getMillisecondsNow ()
  getMillisecondsNow
- static std::string toTimeString (const TimeEntity &time)
  toTimeString
- static std::string to12HourTimeString (const TimeEntity &time)
  to12HourTimeString
- static TimeEntity getCurrentTime ()
  getCurrentTime
- static TimeEntity getTimeFromSeconds (const uint64_t &secs)
  getTimeFromSeconds
- static TimeEntity getTimeFromMilliseconds (const uint64_t &ms)
  getTimeFromMilliseconds
- static std::string toCurrentTimeString ()
  toCurrentTimeString
- static std::string toCurrent12HourTimeString ()
  toCurrent12HourTimeString

5.13.1 Detailed Description

The TimeUtils class contains useful functions for dealing the TimeEntity class.

5.13.2 Member Function Documentation

5.13.2.1 static std::uint64_t cbtek::common::utility::TimeUtils::getCurrentMilliseconds ( ) [inline], [static]

getCurrentMilliseconds

Returns

5.13.2.2 static TimeEntity cbtek::common::utility::TimeUtils::getCurrentTime ( ) [inline], [static]

getCurrentTime

Returns

5.13.2.3 static double cbtek::common::utility::TimeUtils::getMicrosecondsNow ( ) [inline], [static]

getMicrosecondsNow

Returns

5.13.2.4 static double cbtek::common::utility::TimeUtils::getMillisecondsNow ( ) [inline], [static]

getMillisecondsNow

Returns

5.13.2.5 static double cbtek::common::utility::TimeUtils::getNanosecondsNow ( ) [inline], [static]

getNanosecondsNow

Returns

5.13.2.6 static uint64_t cbtek::common::utility::TimeUtils::getSecondsNow ( ) [inline], [static]

getSecondsNow

Returns

5.13.2.7 static TimeEntity cbtek::common::utility::TimeUtils::getTimeFromMilliseconds (const uint64_t & *ms*) [inline], [static]

getTimeFromMilliseconds

Parameters

ms	

Returns

5.13.2.8 static TimeEntity cbtek::common::utility::TimeUtils::getTimeFromSeconds (const uint64_t & *secs*) [inline], [static]

getTimeFromSeconds

Parameters

secs	

Returns

5.13.2.9 static std::string cbtek::common::utility::TimeUtils::to12HourTimeString (const TimeEntity & *time*) [inline], [static]

to12HourTimeString

Parameters

time	

Returns

5.13.2.10 static std::string cbtek::common::utility::TimeUtils::toCurrent12HourTimeString ( ) [inline], [static]

toCurrent12HourTimeString

Returns

5.13.2.11 static std::string cbtek::common::utility::TimeUtils::toCurrentTimeString ( ) [inline], [static]

toCurrentTimeString

Returns

5.13.2.12 template<typename T > static std::string cbtek::common::utility::TimeUtils::toString (T *value*) [inline], [static]

toString

Parameters

value	

Returns

5.13.2.13 static std::string cbtek::common::utility::TimeUtils::toTimeString (const TimeEntity & *time*) [inline], [static]

toTimeString

Parameters

time	

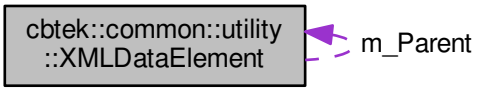
Returns

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

- common/utility/inc/DateTimeUtils.hpp

5.14 cbtek::common::utility::XMLDataElement Class Reference

Collaboration diagram for cbtek::common::utility::XMLDataElement:



Public Member Functions

- XMLDataElement ()
  XMLDataElement.
- XMLDataElement (const XMLDataElement &element)
  XMLDataElement.
- ~XMLDataElement ()
  XMLDataElement.
- const AttributeVector & getAttributes () const
  getAttributes
- const ChildElementVector & getChildren () const
  getChildren
- const std::string & getElementName () const
  getElementName
- std::string getElementData (bool trimmed=false) const
  getElementData
- std::int64_t getElementDataAsInteger () const
  getElementDataAsInteger
- double getElementDataAsFloat () const
  getElementDataAsFloat
- std::string getAttributeValue (const std::string &attributeName, bool caseSensitive=false) const
  getAttributeValue
- std::string getAttributeName (size_t index) const
  getAttributeName
- std::string getAttributeValue (size_t index) const
  getAttributeValue
- XMLDataElement * getParent () const
  getParent
- XMLDataElement * getNextSibling () const
  getNextSibling
- XMLDataElement * getChildAt (size_t index) const
  getChildAt
- XMLDataElement * getChild (const std::string &name) const
  getChild
- std::string getChildElementData (const std::string &name) const
  getChildElementData
- bool attributeExists (const std::string &attributeName, bool caseSensitive=false) const
  attributeExists
- bool hasChildren () const
  hasChildren
- bool childExists (size_t index)
  childExists
- bool childExists (const std::string &name, bool caseSensitive=false)
  childExists
- size_t getNumChildren () const
  getNumChildren
- size_t getNumAttributes () const
  getNumAttributes
- size_t getLocalIndex () const
  getLocalIndex
- size_t getChildIndex (const XMLDataElement *child) const

getChildIndex

- void setElementData (const std::string &data)

  setElementData
- void setElementName (const std::string &name)

  setElementName
- void addAttribute (const std::string &attributeName, const std::string &attributeValue)

  addAttribute
- void addChild (XMLDataElement *child)

  addChild
- void setParent (XMLDataElement *parent)

  setParent
- void setLocalIndex (size_t index)

  setLocalIndex
- XMLDataElement * find (const std::string &name, const bool &caseSensitive=false)

  find
- const XMLDataElement * find (const std::string &name, const bool &caseSensitive=false) const

  find
- void addAttribute (const std::string &attributeName, const float &attributeValue)

  addAttribute
- void addAttribute (const std::string &attributeName, const double &attributeValue)

  addAttribute
- template<typename T >
  void addAttribute (const std::string &attributeName, const T &attributeValue)

  addAttribute
- template<typename Number >
  Number getAttributeValueAsType (const std::string &attributeName) const

  getAttributeValueAsType
- template<typename Number >
  Number getChildElementDataAsType (const std::string &name) const

  getChildElementDataAsType
- bool getAttributeValueAsBool (const std::string &attributeName) const

  getAttributeValueAsBool

Protected Member Functions

- XMLDataElement * findInSubTree (const std::string &name, const XMLDataElement *element, const bool &caseSensitive)

  findInSubTree
- const XMLDataElement * findInSubTree (const std::string &name, const XMLDataElement *element, const bool &caseSensitive) const

  findInSubTree

Protected Attributes

- size_t **m_LocalIndex**
- std::string **m_Name**
- std::string **m_Data**
- std::string **m_Temp**
- AttributeVector **m_Attributes**
- ChildElementVector **m_ChildElementVector**
- XMLDataElement * **m_Parent**

5.14.1 Constructor & Destructor Documentation

5.14.1.1 cbtek::common::utility::XMLDataElement::XMLDataElement (const XMLDataElement & *element*)

XMLDataElement.

Parameters

element	

5.14.2 Member Function Documentation

5.14.2.1 void cbtek::common::utility::XMLDataElement::addAttribute (const std::string & *attributeName,* const std::string & *attributeValue*)

addAttribute

Parameters

attributeName	
attributeValue	

5.14.2.2 void cbtek::common::utility::XMLDataElement::addAttribute (const std::string & *attributeName,* const float & *attributeValue*)

addAttribute

Parameters

attributeName	
attributeValue	

5.14.2.3 void cbtek::common::utility::XMLDataElement::addAttribute (const std::string & *attributeName,* const double & *attributeValue*)

addAttribute

Parameters

attributeName	
attributeValue	

5.14.2.4 template<typename T > void cbtek::common::utility::XMLDataElement::addAttribute (const std::string & *attributeName,* const T & *attributeValue*) [inline]

addAttribute

Parameters

attributeName	
attributeValue	

5.14.2.5 void cbtek::common::utility::XMLDataElement::addChild (XMLDataElement * *child*)

addChild

Parameters

child	

5.14.2.6 bool cbtek::common::utility::XMLDataElement::attributeExists (const std::string & *attributeName,* bool *caseSensitive* = false) const

attributeExists

Parameters

attributeName	
caseSensitive	

Returns

5.14.2.7 bool cbtek::common::utility::XMLDataElement::childExists (size_t *index*)

childExists

Parameters

index	

Returns

5.14.2.8 bool cbtek::common::utility::XMLDataElement::childExists (const std::string & *name,* bool *caseSensitive* = `false`)

childExists

Parameters

name	
caseSensitive	

Returns

5.14.2.9 XMLDataElement * cbtek::common::utility::XMLDataElement::find (const std::string & *name,* const bool & *caseSensitive* = `false`)

find

Parameters

name	
caseSensitive	

Returns

5.14.2.10 const XMLDataElement * cbtek::common::utility::XMLDataElement::find (const std::string & *name,* const bool & *caseSensitive* = `false`) const

find

Parameters

name	
caseSensitive	

Returns

5.14.2.11 XMLDataElement * cbtek::common::utility::XMLDataElement::findInSubTree (const std::string & *name,* const XMLDataElement * *element,* const bool & *caseSensitive*) [protected]

findInSubTree

Parameters

name	
element	
caseSensitive	

Returns

5.14.2.12 const XMLDataElement * cbtek::common::utility::XMLDataElement::findInSubTree (const std::string & *name,* const XMLDataElement * *element,* const bool & *caseSensitive*) const [protected]

findInSubTree

Parameters

name	
element	
caseSensitive	

Returns

5.14.2.13 std::string cbtek::common::utility::XMLDataElement::getAttributeName (size_t *index*) const

getAttributeName

Parameters

index	

Returns

5.14.2.14 const AttributeVector & cbtek::common::utility::XMLDataElement::getAttributes ( ) const

getAttributes

Returns

5.14.2.15 std::string cbtek::common::utility::XMLDataElement::getAttributeValue (const std::string & *attributeName,* bool *caseSensitive* = `false`) const

getAttributeValue

Parameters

attributeName	
caseSensitive	

Returns

5.14.2.16 std::string cbtek::common::utility::XMLDataElement::getAttributeValue (size_t *index*) const

getAttributeValue

Parameters

index	

Returns

5.14.2.17 bool cbtek::common::utility::XMLDataElement::getAttributeValueAsBool (const std::string & *attributeName*) const

getAttributeValueAsBool

Parameters

attributeName	

Returns

5.14.2.18 template<typename Number > Number cbtek::common::utility::XMLDataElement::getAttributeValueAsType (const std::string & *attributeName*) const [inline]

getAttributeValueAsType

Parameters

attributeName	

Returns

5.14.2.19 XMLDataElement * cbtek::common::utility::XMLDataElement::getChild (const std::string & *name*) const

getChild

Parameters

name	

Returns

5.14.2.20 XMLDataElement * cbtek::common::utility::XMLDataElement::getChildAt (size_t *index*) const

getChildAt

Parameters

index	

Returns

5.14.2.21 std::string cbtek::common::utility::XMLDataElement::getChildElementData (const std::string & *name*) const

getChildElementData

Parameters

name	

Returns

5.14.2.22 template<typename Number > Number cbtek::common::utility::XMLDataElement::getChildElementDataAsType (const std::string & *name*) const [inline]

getChildElementDataAsType

Parameters

name	

Returns

5.14.2.23 size_t cbtek::common::utility::XMLDataElement::getChildIndex (const XMLDataElement * *child*) const

getChildIndex

Parameters

child	

Returns

5.14.2.24 const ChildElementVector & cbtek::common::utility::XMLDataElement::getChildren ( ) const

getChildren

Returns

5.14.2.25 std::string cbtek::common::utility::XMLDataElement::getElementData (bool *trimmed* = `false`) const

getElementData

Parameters

trimmed	

Returns

5.14.2.26 double cbtek::common::utility::XMLDataElement::getElementDataAsFloat ( ) const

getElementDataAsFloat

Returns

5.14.2.27 int64_t cbtek::common::utility::XMLDataElement::getElementDataAsInteger ( ) const

getElementDataAsInteger

Returns

5.14.2.28 const std::string & cbtek::common::utility::XMLDataElement::getElementName ( ) const

getElementName

Returns

5.14.2.29 size_t cbtek::common::utility::XMLDataElement::getLocalIndex ( ) const

getLocalIndex

Returns

5.14.2.30 XMLDataElement * cbtek::common::utility::XMLDataElement::getNextSibling ( ) const

getNextSibling

Returns

5.14.2.31 size_t cbtek::common::utility::XMLDataElement::getNumAttributes ( ) const

getNumAttributes

Returns

5.14.2.32 size_t cbtek::common::utility::XMLDataElement::getNumChildren ( ) const

getNumChildren

Returns

5.14.2.33 XMLDataElement * cbtek::common::utility::XMLDataElement::getParent ( ) const

getParent

Returns

5.14.2.34 bool cbtek::common::utility::XMLDataElement::hasChildren ( ) const

hasChildren

Returns

5.14.2.35 void cbtek::common::utility::XMLDataElement::setElementData (const std::string & *data*)

setElementData

Parameters

data	

5.14.2.36 void cbtek::common::utility::XMLDataElement::setElementName (const std::string & *name*)

setElementName

Parameters

name	

5.14.2.37 void cbtek::common::utility::XMLDataElement::setLocalIndex (size_t *index*)

setLocalIndex

Parameters

index	

5.14.2.38 void cbtek::common::utility::XMLDataElement::setParent (XMLDataElement * *parent*)

setParent

Parameters

parent	

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

- common/utility/inc/XMLUtils.h
- common/utility/src/XMLUtils.cpp

5.15 cbtek::common::utility::XMLReader Class Reference

Public Member Functions

- bool load (const std::string &filename)
  load
- bool loadFromString (const std::string &data)
  loadFromString
- XMLDataElement * getElement (const std::string &tagName, const bool &caseSensitive=false)
  getElement
- XMLDataElement * find (const std::string &name, const bool &caseSensitive=false)
  find
- size_t getNumElements (const std::string &tagName, const bool &caseSensitive=false)
  getNumElements
- bool exists (const std::string &tagName, const bool &caseSensitive=false)
  exists
- std::string toString () const
  toString
- XMLDataElement * getRoot ()
  getRoot
- const XMLDataElement * getRoot () const
  getRoot
- bool isValid () const
  isValid
- size_t getDepth () const
  getDepth
- size_t getNumLines () const
  getNumLines
- void reset ()
  reset
- void clear ()
  clear
- XMLDataElement * getFirstElement ()
  getFirstElement

5.15.1 Member Function Documentation

5.15.1.1 bool cbtek::common::utility::XMLReader::exists (const std::string & *tagName*, const bool & *caseSensitive* = `false`)

exists

Parameters

tagName	
caseSensitive	

Returns

5.15.1.2 XMLDataElement * cbtek::common::utility::XMLReader::find (const std::string & *name,* const bool & *caseSensitive* = `false`)

find

Parameters

name	
caseSensitive	

Returns

5.15.1.3 size_t cbtek::common::utility::XMLReader::getDepth ( ) const

getDepth

Returns

5.15.1.4 XMLDataElement * cbtek::common::utility::XMLReader::getElement (const std::string & *tagName,* const bool & *caseSensitive* = `false`)

getElement

Parameters

tagName	
caseSensitive	

Returns

5.15.1.5 XMLDataElement * cbtek::common::utility::XMLReader::getFirstElement ( )

getFirstElement

Returns

5.15.1.6 size_t cbtek::common::utility::XMLReader::getNumElements (const std::string & *tagName,* const bool & *caseSensitive* = `false`)

getNumElements

Parameters

tagName	
caseSensitive	

Returns

5.15.1.7 size_t cbtek::common::utility::XMLReader::getNumLines ( ) const

getNumLines

Returns

5.15.1.8 XMLDataElement * cbtek::common::utility::XMLReader::getRoot ( )

getRoot

Returns

5.15.1.9 const XMLDataElement * cbtek::common::utility::XMLReader::getRoot ( ) const

getRoot

Returns

5.15.1.10 bool cbtek::common::utility::XMLReader::isValid ( ) const

isValid

Returns

5.15.1.11 bool cbtek::common::utility::XMLReader::load (const std::string & *filename*)

load

Parameters

filename	

Returns

5.15.1.12 bool cbtek::common::utility::XMLReader::loadFromString (const std::string & *data*)

loadFromString

Parameters

data	

Returns

5.15.1.13 std::string cbtek::common::utility::XMLReader::toString ( ) const

toString

Returns

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

- common/utility/inc/XMLUtils.h
- common/utility/src/XMLUtils.cpp

5.16 cbtek::common::utility::XMLStreamWriter Class Reference

Public Member Functions

- XMLStreamWriter (std::ostream &out)

  Constructor for XMLStreamWriter.
- void writeStartDocument (const std::string &version="1.0", bool flag=true)

  writeStartDocument
- void writeStartElementNoAttributes (const std::string &tag)

  writeStartElementNoAttributes
- void writeStartElement (const std::string &tag)

  writeStartElement
- void writeEndElement ()

  writeEndElement
- void writeEndElement (const std::string &tag)

  writeEndElement
- void writeTextElement (const std::string &tag, const std::string &text)

  writeTextElement
- template<typename ValueType >
  void writeLastAttribute (const std::string &attributeName, const ValueType &attributeValue)

  writeLastAttribute
- void writeText (const std::string &text)

  writeText
- template<typename ValueType >
  void writeLastAttributeAndCloseTag (const std::string &attributeName, const ValueType &attributeValue)

  writeLastAttribute
- template<typename ValueType >
  void writeAttribute (const std::string &attributeName, const ValueType &attributeValue)

  writeAttribute
- ~XMLStreamWriter ()

  Destructor.

5.16.1 Member Function Documentation

5.16.1.1 template<typename ValueType > void cbtek::common::utility::XMLStreamWriter::writeAttribute (const std::string & *attributeName,* const ValueType & *attributeValue*) [inline]

writeAttribute

Parameters

attributeName	
attributeValue	

5.16.1.2 void cbtek::common::utility::XMLStreamWriter::writeEndElement (const std::string & *tag*)

writeEndElement

Parameters

tag	

5.16.1.3 template<typename ValueType > void cbtek::common::utility::XMLStreamWriter::writeLastAttribute (const std::string & *attributeName,* const ValueType & *attributeValue*) [inline]

writeLastAttribute

Parameters

attributeName	
attributeValue	

5.16.1.4 template<typename ValueType > void cbtek::common::utility::XMLStreamWriter::writeLastAttributeAndCloseTag (const std::string & *attributeName,* const ValueType & *attributeValue*) [inline]

writeLastAttribute

Parameters

attributeName	
attributeValue	

5.16.1.5 void cbtek::common::utility::XMLStreamWriter::writeStartDocument (const std::string & *version* = "1.0", bool *flag* = true)

writeStartDocument

Parameters

version	
flag	

5.16.1.6 void cbtek::common::utility::XMLStreamWriter::writeStartElement (const std::string & *tag*)

writeStartElement

Parameters

tag	

5.16.1.7 void cbtek::common::utility::XMLStreamWriter::writeStartElementNoAttributes (const std::string & *tag*)

writeStartElementNoAttributes

Parameters

tag	

5.16.1.8 void cbtek::common::utility::XMLStreamWriter::writeText (const std::string & *text*)

writeText

Parameters

text	

5.16.1.9 void cbtek::common::utility::XMLStreamWriter::writeTextElement (const std::string & *tag*, const std::string & *text*)

writeTextElement

Parameters

tag	
text	

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

- common/utility/inc/XMLUtils.h
- common/utility/src/XMLUtils.cpp

5.17 cbtek::common::utility::XMLUtils Class Reference

Static Public Member Functions

- static std::string getEncodedString (const std::string &rawString)
  getEncodedString
- static std::string getDecodedString (const std::string &xmlString)
  getDecodedString

5.17.1 Member Function Documentation

5.17.1.1 std::string cbtek::common::utility::XMLUtils::getDecodedString (const std::string & *xmlString*) [static]

getDecodedString

Parameters

xmlString	

Returns

5.17.1.2 std::string cbtek::common::utility::XMLUtils::getEncodedString (const std::string & *rawString*) [static]

getEncodedString

Parameters

rawString	

Returns

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

- common/utility/inc/XMLUtils.h
- common/utility/src/XMLUtils.cpp

Index