

String Class Reference

#include <String.h>

Public Member Functions

	String ()
	String (const char *s)
	String (const String &o)
	~String ()
int	size () const
char	at (int i) const
String &	operator+= (const String &o)
String &	operator+= (const char &o)
String &	operator= (String o)
String &	substr (int start, int end)

Friends

std::ostream &	operator<< (std::ostream &os, const String &o)
void	swap (String &first, String &second)

Detailed Description

My **String** Class. A wrapper for C style strings for use in C++.

Constructor & Destructor Documentation

◆ String() [1/3]

String::String ()

Default Constructor. Creates an empty string.

◆ String() [2/3]

String::String (const char * **s**)

String Literal Constructor. A constructor that makes a **String** object from a given const char * or "xxxx".

◆ String() [3 / 3]

String::String (const **String** & **o**)

Copy Constructor. Initialises the **String**'s properties by copying data from another **String** object.

◆ ~String()

String::~String ()

Strng Deconstructor. De-allocates the data pointer so no memory leakage occurs.

Member Function Documentation

◆ at()

char String::at (int **i**) const

Gets the character ata given index.

Parameters

i the index you wan't to access.

Returns

the character you want to access.

◆ operator+=() [1 / 2]

String & String::operator+= (const **String** & o)

An overload for the += operator. Allows you to concatenate another string to the current string.

Parameters

o the string you want to concatonate to this string.

Returns

the modified string.

◆ operator+=() [2 / 2]

String & String::operator+= (const char & o)

An overload for the += operator. Allows you to concatenate a single character.

Parameters

o the character you want to concatonate to this string.

Returns

the modified string.

◆ operator=()

String & String::operator= (**String** o)

An overload for the assignment operator. Allows you to set/change the string being stored to the string from another **String** class.

Parameters

o the string you want this string to become.

Returns

the modified string.

◆ size()

int String::size () const

A normal member that returns an integer.

Returns

the length of the string being held.

◆ substr()

```
String & String::substr ( int start,  
                        int end  
                        )
```

Get the substring of the held string.

Parameters

start the index you want to start copying from (0 based).

end where you want the copy to end.

Returns

a string that contains the substring.

Friends And Related Function Documentation

◆ operator<<

```
std::ostream& operator<< ( std::ostream & os,  
                           const String & o  
                           )
```

friend

An overload for the output stream operator. Needs to be a friend function defined OUTSIDE of the class as when using ostream << String you do not have access to the ostream so they can't be member operators.

Parameters

- os the ostream you're outputting the string too.
- o the string you want to output.

Returns

the ostream with the string 'inside' so you can chain it outside of this function.

◆ swap

```
void swap ( String & first,  
           String & second  
           )
```

friend

A swapping function. Swaps the data from string to another and visa-versa. Uses the std::swap function.

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

- String.h
- String.cpp