

Ostendo

1.0

Generated by Doxygen 1.8.11

Contents

1	Namespace Index	1
1.1	Namespace List	1
2	Class Index	3
2.1	Class List	3
3	Namespace Documentation	5
3.1	ostendo Namespace Reference	5
3.1.1	Detailed Description	6
3.1.2	Enumeration Type Documentation	7
3.1.2.1	ATTRS	7
3.1.2.2	VarType	7
3.1.2.3	WINDOW_PARTS	7
3.1.3	Function Documentation	7
3.1.3.1	Add(Var in_var, int ch, Window win)	7
3.1.3.2	Backspace(Var in_var, Window win)	8
3.1.3.3	DisplayLines(ostendo::Window win, std::vector< std::string > lines, int current← _line, int &start_line)	8
3.1.3.4	DownArrow(Var in_var, Window win)	9
3.1.3.5	Enter(Var in_var, Window win)	9
3.1.3.6	Form(ostendo::Window win, std::vector< Var > fields)	9
3.1.3.7	GetVarStr(Var in_var, int width)	10
3.1.3.8	InitOstendo(bool color=false, int time_out=-1)	10
3.1.3.9	InitWindow(std::string name)	11
3.1.3.10	InitWindow(std::string name, int width, int height)	11

3.1.3.11	InitWindow(std::string name, int width, int height, int x, int y)	11
3.1.3.12	LeftArrow(Var in_var, Window win)	12
3.1.3.13	LoadColors()	12
3.1.3.14	make_pos(int x, int y, int w, int h)	12
3.1.3.15	make_pos(int x, int y)	13
3.1.3.16	Menu(ostendo::Window win, std::vector< std::string > options)	13
3.1.3.17	RightArrow(Var in_var, Window win)	13
3.1.3.18	TermOstendo()	14
3.1.3.19	TermWindow(std::string name)	14
3.1.3.20	TextBox(ostendo::Window win, std::string base="")	14
3.1.3.21	TextBox(ostendo::Window win, std::vector< std::string > base)	15
3.1.3.22	UpArrow(Var in_var, Window win)	15
3.1.3.23	VTextBox(ostendo::Window win, std::string base="")	15
3.1.3.24	VTextBox(ostendo::Window win, std::vector< std::string > base)	16
3.1.4	Variable Documentation	16
3.1.4.1	color_init	16
3.1.4.2	std_scr	16
3.1.4.3	win_index	16
3.1.4.4	windows	16
4	Class Documentation	17
4.1	ostendo::Pos Struct Reference	17
4.1.1	Detailed Description	17
4.1.2	Constructor & Destructor Documentation	17
4.1.2.1	Pos()	17
4.1.2.2	Pos(int value_x, int value_y)	17
4.1.2.3	Pos(int value_x, int value_y, int value_w, int value_h)	18
4.1.3	Member Data Documentation	18
4.1.3.1	h	18
4.1.3.2	w	18
4.1.3.3	x	18

4.1.3.4	y	18
4.2	ostendo::Var Struct Reference	19
4.2.1	Detailed Description	20
4.2.2	Constructor & Destructor Documentation	20
4.2.2.1	Var()	20
4.2.2.2	Var(int value)	20
4.2.2.3	Var(double value)	21
4.2.2.4	Var(std::string value)	21
4.2.2.5	Var(bool value)	21
4.2.2.6	Var(std::string str_name, int value)	21
4.2.2.7	Var(std::string str_name, double value)	21
4.2.2.8	Var(std::string str_name, std::string value)	22
4.2.2.9	Var(std::string str_name, bool value)	22
4.2.2.10	Var(std::string str_name, std::vector< int > set, int value=0)	22
4.2.2.11	Var(std::string str_name, std::vector< double > set, int value=0)	22
4.2.2.12	Var(std::string str_name, std::vector< std::string > set, int value=0)	23
4.2.3	Member Function Documentation	23
4.2.3.1	operator=(int value)	23
4.2.3.2	operator=(double value)	23
4.2.3.3	operator=(std::string value)	23
4.2.3.4	operator=(bool value)	24
4.2.3.5	SetSet(std::vector< int > set)	24
4.2.3.6	SetSet(std::vector< double > set)	24
4.2.3.7	SetSet(std::vector< std::string > set)	24
4.2.4	Member Data Documentation	25
4.2.4.1	bool_value	25
4.2.4.2	double_set	25
4.2.4.3	double_value	25
4.2.4.4	has_set	25
4.2.4.5	int_set	25

4.2.4.6	int_value	25
4.2.4.7	name	25
4.2.4.8	str_set	25
4.2.4.9	str_value	25
4.2.4.10	var_type	26
4.3	ostendo::Window Class Reference	26
4.3.1	Detailed Description	27
4.3.2	Constructor & Destructor Documentation	28
4.3.2.1	Window()	28
4.3.2.2	Window(int width, int height)	28
4.3.2.3	Window(double width, double height)	28
4.3.2.4	Window(int width, int height, int pos_x, int pos_y)	28
4.3.2.5	Window(int width, int height, double pos_x, double pos_y)	29
4.3.2.6	Window(double width, double height, double pos_x, double pos_y)	29
4.3.2.7	Window(double width, double height, int pos_x, int pos_y)	30
4.3.2.8	Window(Pos pos)	30
4.3.2.9	Window(const Window &win)	30
4.3.3	Member Function Documentation	30
4.3.3.1	AttrOff(int attrs)	31
4.3.3.2	AttrOn(int attrs)	31
4.3.3.3	Clear(bool all=false)	31
4.3.3.4	ClearLine(int line=-1)	31
4.3.3.5	ColorOff(int attrs)	32
4.3.3.6	ColorOn(int attrs)	32
4.3.3.7	NewWindow()	32
4.3.3.8	NewWindow(int width, int height)	32
4.3.3.9	NewWindow(double width, double height)	32
4.3.3.10	NewWindow(int width, int height, int pos_x, int pos_y)	33
4.3.3.11	NewWindow(int width, int height, double pos_x, double pos_y)	33
4.3.3.12	NewWindow(double width, double height, double pos_x, double pos_y)	33
4.3.3.13	NewWindow(double width, double height, int pos_x, int pos_y)	34
4.3.3.14	operator>()	34
4.3.3.15	Print(std::string str,...)	34
4.3.3.16	SetAutoUpdate(bool setting=true)	35
4.3.3.17	SetColor(int color, int value)	35
4.3.3.18	SetCurs(int y=-1, int x=-1)	35
4.3.3.19	ToggleBorder()	35
4.3.3.20	ToggleScroll()	36
4.3.3.21	ToggleTitle(std::string setting="")	36
4.3.3.22	Update()	36
4.3.4	Member Data Documentation	36
4.3.4.1	window_space	36

Chapter 1

Namespace Index

1.1 Namespace List

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

ostendo	Core namespace for all Ostendo methods	5
-------------------------	--	-------------------

Chapter 2

Class Index

2.1 Class List

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

ostendo::Pos	
Structure to contains position/size data	17
ostendo::Var	
Structure to use as general variable	19
ostendo::Window	
Core of everything output to the console	26

Chapter 3

Namespace Documentation

3.1 ostendo Namespace Reference

Core namespace for all Ostendo methods.

Classes

- struct [Pos](#)
Structure to contains position/size data.
- struct [Var](#)
Structure to use as general variable.
- class [Window](#)
Core of everything output to the console.

Enumerations

- enum [VarType](#) { [INT](#) = 0, [DOUBLE](#) = 1, [STRING](#) = 2, [BOOL](#) = 3 }
Declares the possible types of the variable structure.
- enum [WINDOW_PARTS](#) { [WIN_BORDER](#) = 1, [WIN_TITLE](#) = 2, [WIN_TEXT](#) = 3, [WIN_BACKGROUND](#) = 4 }
Declaration for different components of the window.
- enum [ATTRS](#) {
[ALTCHAR](#) = 1, [BLINK](#) = 2, [BOLD](#) = 3, [DIM](#) = 4,
[INVIS](#) = 5, [PROTECT](#) = 5, [REVERSE](#) = 6, [STANDOUT](#) = 7,
[UNDERLINE](#) = 8 }
Declaration for different output attributes.

Functions

- `std::vector< Var > Form (ostendo::Window win, std::vector< Var > fields)`
- `std::string GetVarStr (Var in_var, int width)`
- `Var DownArrow (Var in_var, Window win)`
- `Var UpArrow (Var in_var, Window win)`
- `Var RightArrow (Var in_var, Window win)`
- `Var LeftArrow (Var in_var, Window win)`
- `Var Enter (Var in_var, Window win)`
- `Var Backspace (Var in_var, Window win)`
- `Var Add (Var in_var, int ch, Window win)`
- `std::string Menu (ostendo::Window win, std::vector< std::string > options)`
- `void InitOstendo (bool color=false, int time_out=-1)`
Initializes Ostendo/Ncurses.
- `void TermOstendo ()`
Terminates Ostendo/Ncurses.
- `void InitWindow (std::string name)`
Creates a new window.
- `void InitWindow (std::string name, int width, int height)`
Creates a new window.
- `void InitWindow (std::string name, int width, int height, int x, int y)`
Creates a new window.
- `void TermWindow (std::string name)`
Terminates a window.
- `void LoadColors ()`
Loads console color pairs.
- `std::string TextBox (ostendo::Window win, std::string base="")`
- `std::string TextBox (ostendo::Window win, std::vector< std::string > base)`
- `std::vector< std::string > VTextBox (ostendo::Window win, std::string base="")`
- `std::vector< std::string > VTextBox (ostendo::Window win, std::vector< std::string > base)`
- `void DisplayLines (ostendo::Window win, std::vector< std::string > lines, int current_line, int &start_line)`
- `Pos make_pos (int x, int y, int w, int h)`
Creates a new position object.
- `Pos make_pos (int x, int y)`
Creates a new position object.

Variables

- `Pos std_scr`
- `std::map< std::string, Window > windows`
- `int win_index = 0`
- `bool color_init = false`

3.1.1 Detailed Description

Core namespace for all Ostendo methods.

3.1.2 Enumeration Type Documentation

3.1.2.1 enum ostendo::ATTRS

Declaration for different output attributes.

Enumerator

- ALTCHAR** Uses alternate character set.
- BLINK** Sets output to blinking (If possible in terminal emulator).
- BOLD** Sets output to bold.
- DIM** Sets output to dim colors.
- INVIS** Sets output text to be invisible.
- PROTECT** Sets output mode to protected.
- REVERSE** Sets output to reverse mode.
- STANDOUT** Sets output to be heighlighted.
- UNDERLINE** Sets output to be underlined.

3.1.2.2 enum ostendo::VarType

Declares the possible types of the variable structure.

Enumerator

- INT** Sets the variable type to be integer
- DOUBLE** Sets the variable type to be doubles
- STRING** Sets the variable type to be strings
- BOOL** Sets the variable type to be boolians

3.1.2.3 enum ostendo::WINDOW_PARTS

Declaration for different components of the window.

Enumerator

- WIN_BORDER** Selects the window border.
- WIN_TITLE** Selects the window title.
- WIN_TEXT** Selects the window test.
- WIN_BACKGROUND** Selects the window background.

3.1.3 Function Documentation

3.1.3.1 ostendo::Var ostendo::Add (Var in_var, int ch, Window win)

Handles form user input of any other key. Provides functionality for different [Var](#) types for the user input of any other key.

Parameters

<i>in_var</i>	Var to apply action to.
<i>ch</i>	Character representation of key input.
<i>win</i>	Window to display to.

Returns

[Var](#) after action is applied.

See also

[Var](#), [Window](#)

3.1.3.2 `ostendo::Var ostendo::Backspace (Var in_var, Window win)`

Handles form user input backspace key. Provides functionality for different [Var](#) types for the user input of backspace key.

Parameters

<i>in_var</i>	Var to apply action to.
<i>win</i>	Window to display to.

Returns

[Var](#) after action is applied.

See also

[Var](#), [Window](#)

3.1.3.3 `void ostendo::DisplayLines (ostendo::Window win, std::vector< std::string > lines, int current_line, int & start_line)`

Displays lines from a vector of strings. Displays the lines from a vector of string to a given window, and auto centers the current line, so that it is always on screen.

Parameters

<i>win</i>	Window to display to.
<i>lines</i>	Vector of string to display.
<i>current_line</i>	Current line to center in window.
<i>start_line</i>	line of vector to begin display from.

See also

[Window](#)

3.1.3.4 ostendo::Var ostendo::DownArrow (Var *in_var*, Window *win*)

Handles form user input arrow key down. Provides functionality for different [Var](#) types for the user input of down arrow key.

Parameters

<i>in_var</i>	Var to apply action to.
<i>win</i>	Window to display to.

Returns

[Var](#) after action is applied.

See also

[Var](#), [Window](#)

3.1.3.5 ostendo::Var ostendo::Enter (Var *in_var*, Window *win*)

Handles form user input enter key. Provides functionality for different [Var](#) types for the user input of enter key.

Parameters

<i>in_var</i>	Var to apply action to.
<i>win</i>	Window to display to.

Returns

[Var](#) after action is applied.

See also

[Var](#), [Window](#)

3.1.3.6 std::vector< ostendo::Var > ostendo::Form (ostendo::Window *win*, std::vector< Var > *fields*)

Simple form function to get several inputs from the user. This function takes a window, and a set of inputs to request of the user. After the user enters the values for the requested inputs, the values are returned.

Parameters

<i>win</i>	Window to use for displaying inputs.
<i>fields</i>	Vector of Var 's for taking in input from the user.

Returns

Vector of [Var](#)'s filled with user input.

See also

[Window](#), [Var](#)

3.1.3.7 `std::string ostendo::GetVarStr (Var in_var, int width)`

Gets the string representation of the variable to display. This function reads the [Var](#) object, and creates a string to represent the data stored by the object.

Parameters

<i>in_var</i>	Var to interpret.
<i>width</i>	Maximum width for return string.

Returns

String representation of *in_var*.

See also

[Var](#).

3.1.3.8 `void ostendo::InitOstendo (bool color = false, int time_out = -1)`

Initializes Ostendo/Ncurses.

Initializes ncurses with default settings, and enables keypad input, and reads `std_scr` data. Optionaly enables color, disables cursor, and sets input timeout time.

Parameters

<i>color</i>	Determines if colors should be enabled.
<i>cursor</i>	Determines if cursor should be disabled.
<i>time_out</i>	Sets the time out time for input. -1 for no time out.

See also

[std_scr](#), [windows](#), [color_init](#), [TermOstendo\(\)](#)

3.1.3.9 void ostendo::InitWindow (std::string *name*)

Creates a new window.

Creates a new window, and adds it to windows. The window has the same dimensions of `std_scr`.

Parameters

<i>name</i>	The name of the window for the key in windows map.
-------------	--

See also

[windows](#), [std_scr](#)

3.1.3.10 void ostendo::InitWindow (std::string *name*, int *width*, int *height*)

Creates a new window.

Creates a new window, and adds it to windows. The window is positioned at 0,0 with given width and height.

Parameters

<i>name</i>	The name of the window for the key in windows map.
<i>width</i>	The width of the window in characters.
<i>height</i>	The height of the window in characters.

See also

[windows](#), [std_scr](#)

3.1.3.11 void ostendo::InitWindow (std::string *name*, int *width*, int *height*, int *x*, int *y*)

Creates a new window.

Creates a new window, and adds it to windows. The window is at a given position with a given width and height.

Parameters

<i>name</i>	The name of the window for the key in windows map.
<i>width</i>	The width of the window in characters.
<i>height</i>	The height of the window in characters.
<i>x</i>	The x position of the window in characters from the left.
<i>y</i>	the y position of the window in characters from the top.

See also

[windows](#), [std_scr](#)

3.1.3.12 `ostendo::Var ostendo::LeftArrow (Var in_var, Window win)`

Handles form user input arrow key left. Provides functionality for different [Var](#) types for the user input of left arrow key.

Parameters

<i>in_var</i>	Var to apply action to.
<i>win</i>	Window to display to.

Returns

[Var](#) after action is applied.

See also

[Var](#), [Window](#)

3.1.3.13 `void ostendo::LoadColors ()`

Loads console color pairs.

Loads every combination of the console color pairs, if colors are initialized.

See also

[color_init](#)

3.1.3.14 `Pos ostendo::make_pos (int x, int y, int w, int h)` `[inline]`

Creates a new position object.

Creates position object given the four terms.

Parameters

<i>x</i>	X term for position.
<i>y</i>	Y term for position.
<i>w</i>	Width term for position.
<i>h</i>	Height term for position.

Returns

[Pos](#) New position structure from given values.

3.1.3.15 **Pos** ostendo::make_pos (int *x*, int *y*) `[inline]`

Creates a new position object.

Creates a position object given two terms.

Parameters

<i>x</i>	X term for position.
<i>y</i>	Y term for position.

Returns

[Pos](#) New position structure from given values.

3.1.3.16 **std::string** ostendo::Menu (ostendo::Window *win*, std::vector< std::string > *options*)

Menu function for selecting an option from a set. This function takes a set of options, and a window, then provides a simple user interface for selecting one of the provided options. This selected option is then returned.

Parameters

<i>win</i>	Window to use to display all options.
<i>options</i>	Options that the user must select from.

Returns

String from selected option of Options.

See also

[Window](#)

3.1.3.17 **ostendo::Var** ostendo::RightArrow (Var *in_var*, Window *win*)

Handles form user input arrow key right. Provides functionality for different [Var](#) types for the user input of right arrow key.

Parameters

<i>in_var</i>	Var to apply action to.
<i>win</i>	Window to display to.

Returns

[Var](#) after action is applied.

See also

[Var](#), [Window](#)

3.1.3.18 void ostendo::TermOstendo ()

Terminates Ostendo/Ncurses.

Terminates ncurses, and returns the console to standard settings.

See also

[InitOstendo\(\)](#)

3.1.3.19 void ostendo::TermWindow (std::string *name*)

Terminates a window.

Termainates a specified window from the windows map.

Parameters

<i>name</i>	The name of the window to be deleted.
-------------	---------------------------------------

See also

[windows](#)

3.1.3.20 std::string ostendo::TextBox (ostendo::Window *win*, std::string *base* = " ")

Function for basic extended text entry from user. This function allows the user to write text to a text box, and when completed, the written text is formatted into a string.

Parameters

<i>win</i>	window to display to.
<i>base</i>	Starting string to display in text box.

Returns

String of what user entered.

See also

[Window](#)

3.1.3.21 `std::string ostendo::TextBox (ostendo::Window win, std::vector< std::string > base)`

Function for basic extended text entry from user. This function allows the user to write text to a text box, and when completed, the written text is formatted into a string.

Parameters

<i>win</i>	window to display to.
<i>base</i>	Starting vector of strings representing different lines of text.

Returns

String of what user entered.

See also

[Window](#)

3.1.3.22 `ostendo::Var ostendo::UpArrow (Var in_var, Window win)`

Handles from user input arrow key up. Provides functionality for different [Var](#) types for the user input of up arrow key.

Parameters

<i>in_var</i>	Var to apply action to.
<i>win</i>	Window to display to.

Returns

[Var](#) after action is applied.

See also

[Var](#), [Window](#)

3.1.3.23 `std::vector< std::string > ostendo::VTextBox (ostendo::Window win, std::string base = " ")`

Function for basic extended text entry from user. This function allows the user to write text to a text box, and when completed, the written text is formatted into a vector of strings, each representing a line of text.

Parameters

<i>win</i>	window to display to.
<i>base</i>	Starting string to display in text box.

Returns

Vector of strings representing the different lines of input from the user.

See also

[Window](#)

3.1.3.24 `std::vector< std::string > ostendo::VTextBox (ostendo::Window win, std::vector< std::string > base)`

Function for basic extended text entry from user. This function allows the user to write text to a text box, and when completed, the written text is formatted into a vector of strings, each representing a line of text.

Parameters

<i>win</i>	window to display to.
<i>base</i>	Starting vector of strings representing different lines of text.

Returns

Vector of strings representing the different lines of input from the user.

See also

[Window](#)

3.1.4 Variable Documentation

3.1.4.1 `bool ostendo::color_init = false`

Boolean to check if colors have been initialized.

3.1.4.2 `Pos ostendo::std_scr`

Width and height data for full window, measured at startup.

3.1.4.3 `int ostendo::win_index = 0`

Current window index, from windows map.

3.1.4.4 `std::map< std::string, Window > ostendo::windows`

Optional map for windows, to link windows with given name.

Chapter 4

Class Documentation

4.1 ostendo::Pos Struct Reference

Structure to contains position/size data.

```
#include <types.hpp>
```

Public Member Functions

- [Pos](#) ()
Creates new position.
- [Pos](#) (int value_x, int value_y)
Creates new position.
- [Pos](#) (int value_x, int value_y, int value_w, int value_h)
Creates new position.

Public Attributes

- int [x](#)
- int [y](#)
- int [w](#)
- int [h](#)

4.1.1 Detailed Description

Structure to contains position/size data.

Contains the x, y positional data, and the width, height size data.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 ostendo::Pos::Pos () [inline]

Creates new position.

Creates position with zero for all terms

4.1.2.2 ostendo::Pos::Pos (int value_x, int value_y) [inline]

Creates new position.

Creates position with set x and y position, and width and height are set to zero

Parameters

<i>value</i> ↔ _x	Value for the X position.
<i>value</i> ↔ _y	Value for the y position.

4.1.2.3 `ostendo::Pos::Pos (int value_x, int value_y, int value_w, int value_h)` `[inline]`

Creates new position.

Creates position data with set x, y, width, and height data

Parameters

<i>value</i> ↔ _x	Value for the x position.
<i>value</i> ↔ _y	Value for the y position.
<i>value</i> ↔ _w	Value for the width.
<i>value</i> ↔ _h	Value for the height.

4.1.3 Member Data Documentation

4.1.3.1 `int ostendo::Pos::h`

Height term for the position data.

4.1.3.2 `int ostendo::Pos::w`

Width term for the position data.

4.1.3.3 `int ostendo::Pos::x`

X term of the position data.

4.1.3.4 `int ostendo::Pos::y`

Y term of the position data.

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

- `ostendo_files/types.hpp`

4.2 ostendo::Var Struct Reference

Structure to use as general variable.

```
#include <types.hpp>
```

Public Member Functions

- [Var](#) ()
Creates new [Var](#).
- [Var](#) (int value)
Creates new [Var](#).
- [Var](#) (double value)
Creates new [Var](#).
- [Var](#) (std::string value)
Creates new [Var](#).
- [Var](#) (bool value)
Creates new [Var](#).
- [Var](#) (std::string str_name, int value)
Creates new [Var](#).
- [Var](#) (std::string str_name, double value)
Creates new [Var](#).
- [Var](#) (std::string str_name, std::string value)
Creates new [Var](#).
- [Var](#) (std::string str_name, bool value)
Creates new [Var](#).
- [Var](#) (std::string str_name, std::vector< int > set, int value=0)
Creates new [Var](#) set.
- [Var](#) (std::string str_name, std::vector< double > set, int value=0)
Creates new [Var](#) set.
- [Var](#) (std::string str_name, std::vector< std::string > set, int value=0)
Creates new [Var](#) set.
- void [operator=](#) (int value)
Sets the value of the variable.
- void [operator=](#) (double value)
Sets the value of the variable.
- void [operator=](#) (std::string value)
Sets the value of the variable.
- void [operator=](#) (bool value)
Sets the value of the variable.
- void [SetSet](#) (std::vector< int > set)
Sets the set of the variable.
- void [SetSet](#) (std::vector< double > set)
Sets the set of the variable.
- void [SetSet](#) (std::vector< std::string > set)
Sets the set of the variable.
- int [Int](#) ()
Returns integer value of the variable.
- double [Double](#) ()
Returns the double value of the variable.

- `std::string String ()`
Returns the string value of the variable.
- `bool Bool ()`
Returns the boolean value of the variable.
- `std::string Name ()`
Returns the name of the variable.

Public Attributes

- `bool has_set`
- `std::string name`
- `int var_type`
- `int int_value`
- `double double_value`
- `std::string str_value`
- `bool bool_value`
- `std::vector< int > int_set`
- `std::vector< double > double_set`
- `std::vector< std::string > str_set`

4.2.1 Detailed Description

Structure to use as general variable.

Structure that can act as a variable for int, double, string, or bool

4.2.2 Constructor & Destructor Documentation

4.2.2.1 `ostendo::Var::Var () [inline]`

Creates new `Var`.

Creates the default variable, with standard values for all terms.

4.2.2.2 `ostendo::Var::Var (int value) [inline]`

Creates new `Var`.

Creates new integer variable.

Parameters

<i>value</i>	Integer value for variable.
--------------	-----------------------------

4.2.2.3 ostendo::Var::Var (double *value*) [inline]

Creates new [Var](#).

Creates new double variable.

Parameters

<i>value</i>	Double value for variable.
--------------	----------------------------

4.2.2.4 ostendo::Var::Var (std::string *value*) [inline]

Creates new [Var](#).

Creates new string variable.

Parameters

<i>value</i>	String value for variable.
--------------	----------------------------

4.2.2.5 ostendo::Var::Var (bool *value*) [inline]

Creates new [Var](#).

Creates new boolean variable.

Parameters

<i>value</i>	Boolean value for variable.
--------------	-----------------------------

4.2.2.6 ostendo::Var::Var (std::string *str_name*, int *value*) [inline]

Creates new [Var](#).

Creates a new integer variable.

Parameters

<i>str_name</i>	Name of variable.
<i>value</i>	Integer value for variable.

4.2.2.7 ostendo::Var::Var (std::string *str_name*, double *value*) [inline]

Creates new [Var](#).

Creates a new double variable.

Parameters

<i>str_name</i>	Name of variable.
<i>value</i>	Double value for variable.

4.2.2.8 `ostendo::Var::Var (std::string str_name, std::string value)` `[inline]`

Creates new [Var](#).

Creates a new string variable.

Parameters

<i>str_name</i>	Name of variable.
<i>value</i>	String value for variable.

4.2.2.9 `ostendo::Var::Var (std::string str_name, bool value)` `[inline]`

Creates new [Var](#).

Creates a new boolean variable.

Parameters

<i>str_name</i>	Name of variable.
<i>value</i>	Boolean value for variable.

4.2.2.10 `ostendo::Var::Var (std::string str_name, std::vector< int > set, int value = 0)` `[inline]`

Creates new [Var](#) set.

Creates new variable set of integers with given default.

Parameters

<i>str_name</i>	Name of variable.
<i>set</i>	Set of integer options.
<i>value</i>	Default option from the set.

4.2.2.11 `ostendo::Var::Var (std::string str_name, std::vector< double > set, int value = 0)` `[inline]`

Creates new [Var](#) set.

Creates new variable set of doubles with given default.

Parameters

<i>str_name</i>	Name of variable.
<i>set</i>	Set of double options.
<i>value</i>	Default option from the set.

4.2.2.12 `ostendo::Var::Var (std::string str_name, std::vector< std::string > set, int value = 0)` `[inline]`

Creates new `Var` set.

Creates new variable set of strings with given default.

Parameters

<i>str_name</i>	Name of variable.
<i>set</i>	Set of strings options.
<i>value</i>	Default option from the set.

4.2.3 Member Function Documentation

4.2.3.1 `void ostendo::Var::operator= (int value)` `[inline]`

Sets the value of the variable.

Sets the variable to the given integer value.

Parameters

<i>value</i>	Value to set variable to.
--------------	---------------------------

4.2.3.2 `void ostendo::Var::operator= (double value)` `[inline]`

Sets the value of the variable.

Sets the variable to the given double value.

Parameters

<i>value</i>	Value to set variable to.
--------------	---------------------------

4.2.3.3 `void ostendo::Var::operator= (std::string value)` `[inline]`

Sets the value of the variable.

Sets the variable to the given string value.

Parameters

<i>value</i>	Value to set variable to.
--------------	---------------------------

4.2.3.4 `void ostendo::Var::operator= (bool value) [inline]`

Sets the value of the variable.

Sets the variable to the given boolian value.

Parameters

<i>value</i>	Value to set variable to.
--------------	---------------------------

4.2.3.5 `void ostendo::Var::SetSet (std::vector< int > set) [inline]`

Sets the set of the variable.

Set the variable set to a set of integers.

Parameters

<i>set</i>	Set of integers to use in the variable.
------------	---

4.2.3.6 `void ostendo::Var::SetSet (std::vector< double > set) [inline]`

Sets the set of the variable.

Set the variable set to a set of doubles.

Parameters

<i>set</i>	Set of doubles to use in the variable.
------------	--

4.2.3.7 `void ostendo::Var::SetSet (std::vector< std::string > set) [inline]`

Sets the set of the variable.

Set the variable set to a set of strings.

Parameters

<i>set</i>	Set of strings to use in the variable.
------------	--

4.2.4 Member Data Documentation

4.2.4.1 bool ostendo::Var::bool_value

Boolean value.

4.2.4.2 std::vector<double> ostendo::Var::double_set

Set of double options.

4.2.4.3 double ostendo::Var::double_value

Double value.

4.2.4.4 bool ostendo::Var::has_set

Boolean determining if variable is a set.

4.2.4.5 std::vector<int> ostendo::Var::int_set

Set of integer options.

4.2.4.6 int ostendo::Var::int_value

Integer value.

4.2.4.7 std::string ostendo::Var::name

Name of variable.

4.2.4.8 std::vector<std::string> ostendo::Var::str_set

Set of string options.

4.2.4.9 std::string ostendo::Var::str_value

String value.

4.2.4.10 int ostendo::Var::var_type

Variable type setting from VarTypes.

See also

[VarType](#)

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

- ostendo_files/types.hpp

4.3 ostendo::Window Class Reference

Core of everything output to the console.

```
#include <window.hpp>
```

Public Member Functions

- [Window](#) ()
Creates a new window.
- [Window](#) (int width, int height)
Creates a new window.
- [Window](#) (double width, double height)
Creates a new window.
- [Window](#) (int width, int height, int pos_x, int pos_y)
Creates a new window.
- [Window](#) (int width, int height, double pos_x, double pos_y)
Creates a new window.
- [Window](#) (double width, double height, double pos_x, double pos_y)
Creates a new window.
- [Window](#) (double width, double height, int pos_x, int pos_y)
Creates a new window.
- [Window](#) (Pos pos)
Creates a new window.
- [Window](#) (const [Window](#) &win)
Copies a window.
- void [DelWin](#) ()
Deletes window.
- void [NewWindow](#) ()
Deletes window and creates a new window.
- void [NewWindow](#) (int width, int height)
Deletes window and creates a new window.
- void [NewWindow](#) (double width, double height)
Deletes window and creates a new window.
- void [NewWindow](#) (int width, int height, int pos_x, int pos_y)
Deletes window and creates a new window.

- void [NewWindow](#) (int width, int height, double pos_x, double pos_y)
Deletes window and creates a new window.
- void [NewWindow](#) (double width, double height, double pos_x, double pos_y)
Deletes window and creates a new window..
- void [NewWindow](#) (double width, double height, int pos_x, int pos_y)
Deletes window and creates a new window.
- void [Update](#) ()
Updates window display.
- void [ToggleBorder](#) ()
Toggles drawing window border.
- void [ToggleScroll](#) ()
Toggles window scrolling.
- void [ToggleTitle](#) (std::string setting="")
Toggles window title.
- void [SetAutoUpdate](#) (bool setting=true)
Sets window auto updating.
- void [ColorOn](#) (int attrs)
Enables a color for output.
- void [ColorOff](#) (int attrs)
Disables a color for output.
- void [AttrOn](#) (int attrs)
Enables an attribute for output.
- void [AttrOff](#) (int attrs)
Disables an attribute for output.
- void [SetColor](#) (int color, int value)
Sets display color of window part.
- void [Clear](#) (bool all=false)
Clears window of outputs.
- void [ClearLine](#) (int line=-1)
Clears the current line of window.
- int [Print](#) (std::string str,...)
Print output to window.
- void [SetCurs](#) (int y=-1, int x=-1)
Sets cursor position in window.
- WINDOW * [operator\(\)](#) ()
Gets ncurses window pointer.

Public Attributes

- [Pos window_space](#)

4.3.1 Detailed Description

Core of everything output to the console.

The [Window](#) class is the base of everything that is output to the console. Everything output must be part of a window, and the windows controll all outputs, and output attributes.

4.3.2 Constructor & Destructor Documentation

4.3.2.1 ostendo::Window::Window ()

Creates a new window.

Creates a new window with the same base setting of `std_scr`.

See also

[std_scr](#)

4.3.2.2 ostendo::Window::Window (int *width*, int *height*)

Creates a new window.

Creates a new window with given width and height, at 0,0.

Parameters

<i>width</i>	Width of window.
<i>height</i>	Height of window.

4.3.2.3 ostendo::Window::Window (double *width*, double *height*)

Creates a new window.

Creates a new window with width and height as a percentage of `std_scr`.

Parameters

<i>width</i>	Percentage of width.
<i>height</i>	Percentage of height.

See also

[std_scr](#)

4.3.2.4 ostendo::Window::Window (int *width*, int *height*, int *pos_x*, int *pos_y*)

Creates a new window.

Creates a new window with given width and height, at given position.

Parameters

<i>width</i>	Width of window.
--------------	------------------

Parameters

<i>height</i>	Height of window.
<i>pos</i> ↔ <i>_x</i>	X position of window.
<i>pos</i> ↔ <i>_y</i>	Y position of window.

4.3.2.5 ostendo::Window::Window (int *width*, int *height*, double *pos_x*, double *pos_y*)

Creates a new window.

Creates a new window with given width and height, and at x and y of percentage of `std_scr`.

Parameters

<i>width</i>	Width of window.
<i>height</i>	Height of window.
<i>pos</i> ↔ <i>_x</i>	Percentage of width.
<i>pos</i> ↔ <i>_y</i>	Percentage of height.

See also

[std_scr](#)

4.3.2.6 ostendo::Window::Window (double *width*, double *height*, double *pos_x*, double *pos_y*)

Creates a new window.

Creates a new window with width and height as a percentage of `std_scr`, and at x and y of percentage of `std_scr`.

Parameters

<i>width</i>	Percentage of width.
<i>height</i>	Percentage of height.
<i>pos</i> ↔ <i>_x</i>	Percentage of width.
<i>pos</i> ↔ <i>_y</i>	Percentage of height.

See also

[std_scr](#)

4.3.2.7 ostendo::Window::Window (double *width*, double *height*, int *pos_x*, int *pos_y*)

Creates a new window.

Creates a new window with given width and height, and at x and y.

Parameters

<i>width</i>	Percentage of width.
<i>height</i>	Percentage of height.
<i>pos_x</i>	X position of window.
<i>pos_y</i>	Y position of window.

See also

[std_scr](#)

4.3.2.8 ostendo::Window::Window (Pos *pos*)

Creates a new window.

Creates a window with the width, height, and position given by Position object.

Parameters

<i>pos</i>	Position definition of window.
------------	--------------------------------

See also

[Pos](#)

4.3.2.9 ostendo::Window::Window (const Window & *win*)

Copies a window.

Creates a copy of a given window.

Parameters

<i>win</i>	Window to create a copy of.
------------	---

4.3.3 Member Function Documentation

4.3.3.1 void ostendo::Window::AttrOff (int *attrs*)

Disables an attribute for output.

Disables previously enabled output attribute.

Parameters

<i>attrs</i>	Attribute value from Attrs.
--------------	-----------------------------

See also

Attrs

4.3.3.2 void ostendo::Window::AttrOn (int *attrs*)

Enables an attribute for output.

Enables output attribute.

Parameters

<i>attrs</i>	Attribute value from Attrs.
--------------	-----------------------------

See also

Attrs

4.3.3.3 void ostendo::Window::Clear (bool *all* = false)

Clears window of outputs.

Clears window of all ouputs.

Parameters

<i>all</i>	If true, then clears window of border and title aswell.
------------	---

4.3.3.4 void ostendo::Window::ClearLine (int *line* = -1)

Clears the current line of window.

Clears the current line of cursor y, and sets cursor x to begining of the line.

Parameters

<i>line</i>	Specify line to clear.
-------------	------------------------

4.3.3.5 void ostendo::Window::ColorOff (int *attrs*)

Disables a color for output.

Disables previously enabled output color.

Parameters

<i>attrs</i>	Color index represented by a two digit number.
--------------	--

4.3.3.6 void ostendo::Window::ColorOn (int *attrs*)

Enables a color for output.

Enables output color.

Parameters

<i>attrs</i>	Color index represented by a two digit number.
--------------	--

4.3.3.7 void ostendo::Window::NewWindow ()

Deletes window and creates a new window.

Creates a new window with the same base setting of `std_scr`.

See also

[std_scr](#)

4.3.3.8 void ostendo::Window::NewWindow (int *width*, int *height*)

Deletes window and creates a new window.

Creates a new window with given width and height, at 0,0.

Parameters

<i>width</i>	Width of window.
<i>height</i>	Height of window.

4.3.3.9 void ostendo::Window::NewWindow (double *width*, double *height*)

Deletes window and creates a new window.

Creates a new window with width and height as a percentage of `std_scr`.

Parameters

<i>width</i>	Percentage of width.
<i>height</i>	Percentage of height.

See also

[std_scr](#)

4.3.3.10 void ostendo::Window::NewWindow (int *width*, int *height*, int *pos_x*, int *pos_y*)

Deletes window and creates a new window.

Creates a new window with given width and height, at given position.

Parameters

<i>width</i>	Width of window.
<i>height</i>	Height of window.
<i>pos_x</i>	X position of window.
<i>pos_y</i>	Y position of window.

4.3.3.11 void ostendo::Window::NewWindow (int *width*, int *height*, double *pos_x*, double *pos_y*)

Deletes window and creates a new window.

Creates a new window with given width and height, and at x and y of percentage of std_scr.

Parameters

<i>width</i>	Width of window.
<i>height</i>	Height of window.
<i>pos_x</i>	Percentage of width.
<i>pos_y</i>	Percentage of height.

See also

[std_scr](#)

4.3.3.12 void ostendo::Window::NewWindow (double *width*, double *height*, double *pos_x*, double *pos_y*)

Deletes window and creates a new window..

Creates a new window with width and height as a percentage of std_scr, and at x and y of percentage of std_scr.

Parameters

<i>width</i>	Percentage of width.
<i>height</i>	Percentage of height.
<i>pos</i> ↔ <i>_x</i>	Percentage of width.
<i>pos</i> ↔ <i>_y</i>	Percentage of height.

See also

[std_scr](#)

4.3.3.13 void ostendo::Window::NewWindow (double *width*, double *height*, int *pos_x*, int *pos_y*)

Deletes window and creates a new window.

Creates a new window with given width and height, and at x and y.

Parameters

<i>width</i>	Percentage of width.
<i>height</i>	Percentage of height.
<i>pos</i> ↔ <i>_x</i>	X position of window.
<i>pos</i> ↔ <i>_y</i>	Y position of window.

See also

[std_scr](#)

4.3.3.14 WINDOW* ostendo::Window::operator()() [inline]

Gets ncurses window pointer.

Gets the ncurses window pointer for the current window.

Returns

pointer to ncurses window.

4.3.3.15 int ostendo::Window::Print (std::string *str*, ...)

Print output to window.

Main output function for ostendo. *str* is a formatted string such as for printf, with additional arguments as required. Accepts additional escape sequenced initialized with a '#'. '#o__' will turn on specified attribute, while '#f__' will turn off the specified attribute. Attributes can be two digit numbers (0-7)(0-7), where the first number is the foreground color, and the second number is the background color. These attributes can also be any of the following: ac(alternate character set), bl(blink), bo(bold), di(dim), in(invisible), pr(protected), re(reverse), st(standout), un(underlined).

Parameters

<i>str</i>	Format string.
...	Additional arguments as required.

Returns

-1 if window has not been initialized, otherwise 0.

4.3.3.16 void ostendo::Window::SetAutoUpdate (bool *setting* = true)

Sets window auto updating.

Sets window auto updating to specified value.

Parameters

<i>setting</i>	Value to set auto updating to.
----------------	--------------------------------

4.3.3.17 void ostendo::Window::SetColor (int *color*, int *value*)

Sets display color of window part.

Sets the color of compnents of the window.

Parameters

<i>color</i>	Color represented by two digit number.
<i>value</i>	Window component to color.

4.3.3.18 void ostendo::Window::SetCurs (int *y* = -1, int *x* = -1)

Sets cursor position in window.

Sets the position of the output cursor for the window.

Parameters

<i>x</i>	X position of the cursor.
<i>y</i>	Y position of the cursor.

4.3.3.19 void ostendo::Window::ToggleBorder ()

Toggles drawing window border.

Toggles whether the window border should be drawn.

4.3.3.20 void ostendo::Window::ToggleScroll ()

Toggles window scrolling.

Toggles if the window should handle last line output by scrolling.

4.3.3.21 void ostendo::Window::ToggleTitle (std::string *setting* = " ")

Toggles window title.

Toggles whether the window title should be drawn.

Parameters

<i>setting</i>	String to set as the new window title.
----------------	--

4.3.3.22 void ostendo::Window::Update ()

Updates window display.

Updates the window display, to show recently printed outputs. Must be called for display to be visible.

4.3.4 Member Data Documentation

4.3.4.1 Pos ostendo::Window::window_space

[Window](#) space available for output (Excluding borders/title).

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

- ostendo_files/window.hpp
- ostendo_files/window.cpp

Index

ALTCHAR
 ostendo, 7
ATTRS
 ostendo, 7
Add
 ostendo, 7
AttrOff
 ostendo::Window, 30
AttrOn
 ostendo::Window, 31

BLINK
 ostendo, 7
BOLD
 ostendo, 7
BOOL
 ostendo, 7
Backspace
 ostendo, 8
bool_value
 ostendo::Var, 25

Clear
 ostendo::Window, 31
ClearLine
 ostendo::Window, 31
color_init
 ostendo, 16
ColorOff
 ostendo::Window, 32
ColorOn
 ostendo::Window, 32

DIM
 ostendo, 7
DOUBLE
 ostendo, 7
DisplayLines
 ostendo, 8
double_set
 ostendo::Var, 25
double_value
 ostendo::Var, 25
DownArrow
 ostendo, 9

Enter
 ostendo, 9

Form
 ostendo, 9

GetVarStr
 ostendo, 10

h
 ostendo::Pos, 18
has_set
 ostendo::Var, 25

INVIS
 ostendo, 7
INT
 ostendo, 7
InitOstendo
 ostendo, 10
InitWindow
 ostendo, 11
int_set
 ostendo::Var, 25
int_value
 ostendo::Var, 25

LeftArrow
 ostendo, 12
LoadColors
 ostendo, 12

make_pos
 ostendo, 12, 13
Menu
 ostendo, 13

name
 ostendo::Var, 25
NewWindow
 ostendo::Window, 32–34

operator()
 ostendo::Window, 34
operator=
 ostendo::Var, 23, 24
ostendo, 5
 ALTCHAR, 7
 ATTRS, 7
 Add, 7
 BLINK, 7
 BOLD, 7
 BOOL, 7
 Backspace, 8
 color_init, 16
 DIM, 7
 DOUBLE, 7

- DisplayLines, 8
- DownArrow, 9
- Enter, 9
- Form, 9
- GetVarStr, 10
- INVIS, 7
- INT, 7
- InitOstendo, 10
- InitWindow, 11
- LeftArrow, 12
- LoadColors, 12
- make_pos, 12, 13
- Menu, 13
- PROTECT, 7
- REVERSE, 7
- RightArrow, 13
- STANDOUT, 7
- STRING, 7
- std_scr, 16
- TermOstendo, 14
- TermWindow, 14
- TextBox, 14, 15
- UNDERLINE, 7
- UpArrow, 15
- VTextBox, 15, 16
- VarType, 7
- WIN_BACKGROUND, 7
- WIN_BORDER, 7
- WIN_TEXT, 7
- WIN_TITLE, 7
- WINDOW_PARTS, 7
- win_index, 16
- windows, 16
- ostendo::Pos, 17
 - h, 18
 - Pos, 17, 18
 - w, 18
 - x, 18
 - y, 18
- ostendo::Var, 19
 - bool_value, 25
 - double_set, 25
 - double_value, 25
 - has_set, 25
 - int_set, 25
 - int_value, 25
 - name, 25
 - operator=, 23, 24
 - SetSet, 24
 - str_set, 25
 - str_value, 25
 - Var, 20–23
 - var_type, 25
- ostendo::Window, 26
 - AttrOff, 30
 - AttrOn, 31
 - Clear, 31
 - ClearLine, 31
 - ColorOff, 32
 - ColorOn, 32
 - NewWindow, 32–34
 - operator(), 34
 - Print, 34
 - SetAutoUpdate, 35
 - SetColor, 35
 - SetCurs, 35
 - ToggleBorder, 35
 - ToggleScroll, 36
 - ToggleTitle, 36
 - Update, 36
 - Window, 28–30
 - window_space, 36
- PROTECT
 - ostendo, 7
- Pos
 - ostendo::Pos, 17, 18
- Print
 - ostendo::Window, 34
- REVERSE
 - ostendo, 7
- RightArrow
 - ostendo, 13
- STANDOUT
 - ostendo, 7
- STRING
 - ostendo, 7
- SetAutoUpdate
 - ostendo::Window, 35
- SetColor
 - ostendo::Window, 35
- SetCurs
 - ostendo::Window, 35
- SetSet
 - ostendo::Var, 24
- std_scr
 - ostendo, 16
- str_set
 - ostendo::Var, 25
- str_value
 - ostendo::Var, 25
- TermOstendo
 - ostendo, 14
- TermWindow
 - ostendo, 14
- TextBox
 - ostendo, 14, 15
- ToggleBorder
 - ostendo::Window, 35
- ToggleScroll
 - ostendo::Window, 36
- ToggleTitle
 - ostendo::Window, 36
- UNDERLINE

- ostendo, [7](#)
- UpArrow
 - ostendo, [15](#)
- Update
 - ostendo::Window, [36](#)
- VTextBox
 - ostendo, [15](#), [16](#)
- Var
 - ostendo::Var, [20–23](#)
- var_type
 - ostendo::Var, [25](#)
- VarType
 - ostendo, [7](#)
- w
 - ostendo::Pos, [18](#)
- WIN_BACKGROUND
 - ostendo, [7](#)
- WIN_BORDER
 - ostendo, [7](#)
- WIN_TEXT
 - ostendo, [7](#)
- WIN_TITLE
 - ostendo, [7](#)
- WINDOW_PARTS
 - ostendo, [7](#)
- win_index
 - ostendo, [16](#)
- Window
 - ostendo::Window, [28–30](#)
- window_space
 - ostendo::Window, [36](#)
- windows
 - ostendo, [16](#)
- x
 - ostendo::Pos, [18](#)
- y
 - ostendo::Pos, [18](#)