LibreOffice®

LibreOffice Documentation Team

Math Guide



26.2

LibreOffice is a registered trademark of The Document Foundation
Further Information is available at libreoffice.org

Copyright

This document is Copyright © 2026 by the LibreOffice Documentation Team. Contributors are listed below. This document may be distributed and/or modified under the terms of either the GNU General Public License (https://www.gnu.org/licenses/gpl.html), version 3 or later, or the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0/), version 4.0 or later. All trademarks within this guide belong to their legitimate owners.

Contributors

To this edition

Peter Schofield

To previous editions

Agnes Belzunce
Bernard Siaud
Christian Kühl
Daniel Carrera
Dave Barton
Florian Reisinger
Frédéric Parrenin
Gisbert Friege
Hazel Russman
Ian Laurenson
Janet M. Swisher
Jean Hollis Weber
Jochen Schiffers
Laurent Balland-Poirier
Michele Zarri
Olivier Hallot
Peter Kupfer
Peter Schofield
Rafael Lima
Regina Henschel
Roman Kuznetsov
T. J. Frazier

Feedback

Please direct any comments or suggestions about this document to the Documentation Team Forum at https://community.documentfoundation.org/c/documentation/loguides/ (registration is required) or send an email to: loguides@community.documentfoundation.org.

Note

Everything sent to a forum, including email addresses and any other personal information that is written in the message, is publicly archived and cannot be deleted. Emails sent to the forum are moderated.

Publication date and software version

Published April 2026. Based on LibreOffice 26.2 Community.
Other versions of LibreOffice may differ in appearance and functionality.

Contents

Copyright....2
Contributors....2
Feedback....2
Publication date and software version....2

Preface....7
What is LibreOffice?....8
What is Math guide for?....8
Minimum requirements for LibreOffice....8
How to get LibreOffice....8
Computers and laptops....8
Tablets, iPads and Chromebooks....8
Installing LibreOffice....9
Setting up and customizing LibreOffice....9
Extensions and add-ons....9
Where to get more help....9
Help system....9
Other free online support....11
Paid support and training....11
What you see may be different....11
Illustrations....11
Icons....12
Using LibreOffice on macOS....13
Who wrote this user guide?....13
Frequently asked questions....13
What is new in LibreOffice 26.2?....15

Chapter 1, Creating and Editing Formulas....16
Introduction....17
Formulas as documents or files....17
Formula files....18
LibreOffice documents....18
Creating formulas....20
Elements panel....20
Context menu....21
Markup language....21
Greek characters....22
Formula examples....23
Editing formulas....24
Formula layout....25
Using braces....25
Brackets (parentheses) and matrices....26
Unpaired brackets....27
Recognizing functions....27
Multiple line formulas....28
Formula element spacing....28
Adding limits to summations and integrals....28
Adding derivatives....29
Control characters as normal characters....29
Text in formulas....30
Formatting text in formulas....30
Aligning formulas using equals sign....31
Changing formula appearance....31
Formula font size....31
Font size options....32

Formula fonts....33
Adjusting formula spacing....34
Adjusting formula alignment....38
Changing formula color....39
Formula library....40
Using Math....40
Using Writer, Calc, Draw, or Impress....41
Using a formula library....41

Chapter 2, Formulas in Writer....43
Introduction....44
Automatic formula numbering....44
Numbering....44
Formula cross referencing....45
Anchoring and aligning formulas....45
Anchoring position....45
Vertical alignment....46
Horizontal alignment....47
Object wrap and spacing....47
Object wrap....48
Object spacing....48
Text mode....48
Background and borders....48
Backgrounds....49
Borders....50
Quick formula insertion....50

Chapter 3, Formulas In Calc, Draw, and Impress....51
Introduction....52
Anchoring formulas....52
Calc....52
Draw and Impress....52
Formula properties....52
Formulas in charts....53
Chemical formulas....53

Chapter 4, Customization....55
Introduction....56
Formula Editor and Elements panel....56
Formula Editor....56
Elements panel....56
Adding keyboard shortcuts....57
Example keyboard shortcut....57
Customize dialog options for keyboard shortcuts....58
Catalog customization....59
Adding symbols....59
Editing symbols....61
Symbol options....62
Formula spacing....62
Scaling of code....63
Extensions....63

Chapter 5, Exporting and Importing....65
MathML format....66
Microsoft file formats....66
[L]: Load and convert the object....66
[S]: Convert and save the object....66

Appendix A, Command Reference....67

Introduction.....68
Formula commands.....68
Unary/binary operators.....68
Relations.....72
Set operations.....76
Functions.....76
Operators.....80
Attributes.....84
Brackets.....90
Formats.....96
Others.....100
Greek characters.....100
Special characters.....106
Reserved words.....106

LibreOffice

Math Guide 26.2

Preface

Equations and Formulas in LibreOffice



What is LibreOffice?

LibreOffice is a freely available, fully-featured, open-source office productivity suite compatible with other major office suites and available on various platforms. The native file format is the Open Document Format (ODF). However, LibreOffice can also open and save documents in numerous other formats, including those used by Microsoft Office versions. For further information, please refer to the *Getting Started Guide*.

What is Math guide for?

The LibreOffice Math module is the formula and equation editor included with LibreOffice. If a LibreOffice user wants to insert formulas and equations into documents, then this Math User Guide is a good reference source to use.

In LibreOffice, formulas can be inserted as objects into the LibreOffice documents created using Writer, Impress, Draw, or Calc. Regardless of the document type, formula objects are edited using Math. For more information and an introduction to all of the LibreOffice modules, see the *Getting Started Guide*.

> **Note**
>
> LibreOffice Math cannot evaluate mathematical equations or perform calculations. For more information on formulas and equations, see the *Calc Guide*.

Minimum requirements for LibreOffice

LibreOffice 26.2 requires one of the following operating systems:

Linux x64 (deb) and Linux x64 (rpm)

Mac OS X (Aarch64/Apple Silicon)

macOS x86_64 (10.14 Mojave or higher)

Windows x86_64 (Windows 7 or newer required)

For a detailed list of requirements and operating systems supported, see the LibreOffice website, https://www.libreoffice.org/get-help/system-requirements/.

How to get LibreOffice

Computers and laptops

Versions of LibreOffice for Windows, Linux, and macOS are freely available and can be downloaded from the LibreOffice website at https://www.libreoffice.org/download. For Linux users LibreOffice is included free with many of Linux distributions. For example, Ubuntu Linux versions of LibreOffice may differ in some features described in this user guide.

LibreOffice is also available for Windows in the Microsoft Store and for macOS in the Apple App Store at a low and attractive price. These versions are free software being open source, but the small charge covers the cost of placing LibreOffice in app stores. The profits from sale of LibreOffice are invested to support development of the LibreOffice project.

Tablets, iPads and Chromebooks

To use LibreOffice on tablets, iPads, or Chromebooks, a LibreOffice based app has to be downloaded and installed. This app is called **Collabora Office**, which uses the same technology

as LibreOffice and is very similar in operation to LibreOffice. For more information, go to the Collabora Office website at https://www.collaboraoffice.com.

Installing LibreOffice

Information on installing LibreOffice on supported operating systems can be found at this web page: https://www.libreoffice.org/get-help/install-howto/. If LibreOffice is acquired through official app stores, follow the installation instructions provided by the store.

Setting up and customizing LibreOffice

After installation, change the default settings (options) in LibreOffice to suit working requirements and preferences. Go to **Tools > Options** on the Menu bar (mac OS **LibreOffice > Preferences**) and change the settings as required.

Settings are described in LibreOffice Help and the *Getting Started Guide*. These two sources provide information on how to customize menus, toolbars, and keyboard shortcuts in Impress, add new menus and toolbars, and assign macros to events.

Tip

Some settings are for power users and programmers. If it is difficult to understand what an option does, it is recommended to leave an option on its default setting unless instructions in this user guide recommend changing the setting.

Extensions and add-ons

Functionality can be added to LibreOffice with extensions and add-ons. Several extensions are installed with the program and other extensions from the official extensions repository, https://extensions.libreoffice.org/ or various other sources. See the *Getting Started Guide* for more information on installing extensions and add-ons.

Where to get more help

This user guide, the Help system, and user support systems assumes users are familiar with computers and basic functions such as starting applications, opening and saving files.

Help system

LibreOffice comes with an extensive Help system and is the first line of support. Windows and Linux users can choose to download and install the offline Help for use when not connected to the Internet. Offline Help is installed with the MacOS version of LibreOffice.

To display the LibreOffice Help, press *F1* or go to **Help > LibreOffice Help** on the Menu bar. If the offline help is not installed on a computer, but connected to the Internet, a dialog opens giving the option to **Read Help Online**. Select this option and the default web browser opens at the LibreOffice online help pages in the LibreOffice website.

The Help menu includes links to other LibreOffice information and support facilities. The Help menu also includes links to other LibreOffice information and support resources.

What's This?

For quick tips when a toolbar is visible, place the cursor over a tool icon to see a small tooltip box with a brief explanation of the tool function. For a more detailed explanation, select **Help > What's This?**

Also, *Extended Tips* can be activated by going to **Tools > Options > LibreOffice > General > Help** (macOS **LibreOffice > Preferences > LibreOffice > General > Help**) on the Menu bar and selecting the option *Extended Tips*. Extended tips provide a brief description about tools and commands. To display an extended tip, use the keyboard *Shift+F1*, then move the cursor onto a tool or command.

User Guides

Opens the default browser at the Documentation page of the LibreOffice website https://documentation.libreoffice.org/en/english-documentation/. This web page gives access to LibreOffice User Guides and other useful information that can be opened in a default browser. Also, LibreOffice User Guides are available in PDF format as a free download, or to buy as printed copies.

Show Tip of the Day

Opens a small window with a random tip on how to use LibreOffice.

Search Commands

Opens a window where typing a few letters, or the name of a Menu bar command, for example, quickly finds where the command is located. Clicking on a command in the resulting list may open a relevant dialog or have other effects.

Get Help Online

Opens the default browser at the Ask LibreOffice forum of questions and answers from the LibreOffice community, https://ask.libreoffice.org/en/questions/.

Send Feedback

Opens the default browser at the Feedback page of the LibreOffice website https://www.libreoffice.org/get-help/feedback/. From this web page, bugs can be reported, new features suggested and communicated with other users in the LibreOffice community.

Restart in Safe Mode

Opens a dialog window giving options to restart LibreOffice and reset the software to its default settings. Restarting in safe mode also provides an opportunity to restore LibreOffice from a backup.

Get Involved

Opens the default browser at the Get Involved page of the LibreOffice website, https://www.libreoffice.org/community/get-involved/. Choose a topic of interest to help improve the program.

Donate to LibreOffice

Opens the default browser at the Donation page of the LibreOffice website, https://donate.libreoffice.org/ providing an opportunity to make a donation to support LibreOffice.

License Information

Outlines the licenses under which LibreOffice is made available.

Check for Updates

Opens a dialog and checks the LibreOffice website for updates to version of the software. The dialog provides an opportunity to download and install any updates to LibreOffice.

About LibreOffice

Opens a dialog displaying information about the version LibreOffice and operating system being used. This information is requested if the community asks for help or assistance with the software (in macOS, this option is found under **LibreOffice** on the Menu bar).

Other free online support

The LibreOffice community not only develops software, but provides free, volunteer-based support. See Table 1 and the web page https://www.libreoffice.org/get-help/. For comprehensive online support from the community, look at mailing lists and the Ask LibreOffice website, https://ask.libreoffice.org/en/questions/. Other user websites also offer free tips and tutorials.

Table 1: Free support for LibreOffice users

Free LibreOffice support	
FAQs	Answers to frequently asked questions https://wiki.documentfoundation.org/Faq.
Mailing lists	Free community support is provided by a network of experienced users https://www.libreoffice.org/get-help/mailing-lists/.
Questions & Answers and Knowledge Base	Free community assistance is provided in a Question & Answer formatted web service. Search similar topics or open a new one in https://ask.libreoffice.org/en/questions. The service is available in several other languages; just replace /en/ with de, es, fr, ja, ko, nl, pt, tr, and many others in the web address above.
Native language support	The LibreOffice website in various languages https://www.libreoffice.org/community/nlc/. Mailing lists for native languages https://wiki.documentfoundation.org/Local_Mailing_Lists. Information about social networking https://wiki.documentfoundation.org/Website/Web_Sites_services.
Accessibility options	Information about available accessibility options. https://www.libreoffice.org/get-help/accessibility/.
OpenOffice Forum	Another forum that provides support for LibreOffice, among other open source office suites https://forum.openoffice.org/en/forum/.

Paid support and training

Support and training is available through service contracts from a vendor or consulting firm specializing in LibreOffice. For information about certified professional support, see The Document Foundation website: https://www.documentfoundation.org/gethelp/support/.

For schools, educational and research institutions, and large organizations, see https://www.libreoffice.org/download/libreoffice-in-business/.

What you see may be different

Illustrations

LibreOffice runs on Windows, Linux, and macOS operating systems. Each operating system can be customized by users (fonts, colors, themes, window managers). The illustrations in this user guide were taken from a variety of computers and operating systems. Therefore, some illustrations may not look exactly the same as displayed on a computer monitor. Also, some of the dialogs may differ because of the settings selected in LibreOffice. Either use dialogs from the computer system (default) or dialogs provided by LibreOffice.

To change to using LibreOffice dialogs:

1) Go to **Tools > Options > LibreOffice > General** (macOS **LibreOffice > Preferences > LibreOffice > General**) on the Menu bar to open the dialog page for general options (Figure 1).

2) Select the option *Use LibreOffice dialogs* in **Open/Save dialogs** to display the LibreOffice dialogs on a computer display.

3) Click **OK** to save the settings and close the dialog.

Icons

The LibreOffice community has created icons for several icon sets, for example Breeze, Colibre, and Sifr. LibreOffice users can select a preferred set of fonts to use. Icons used to illustrate tools available in LibreOffice may differ from ones used in this user guide. User guide icons have been taken from a LibreOffice installation using the Colibre set of icons.

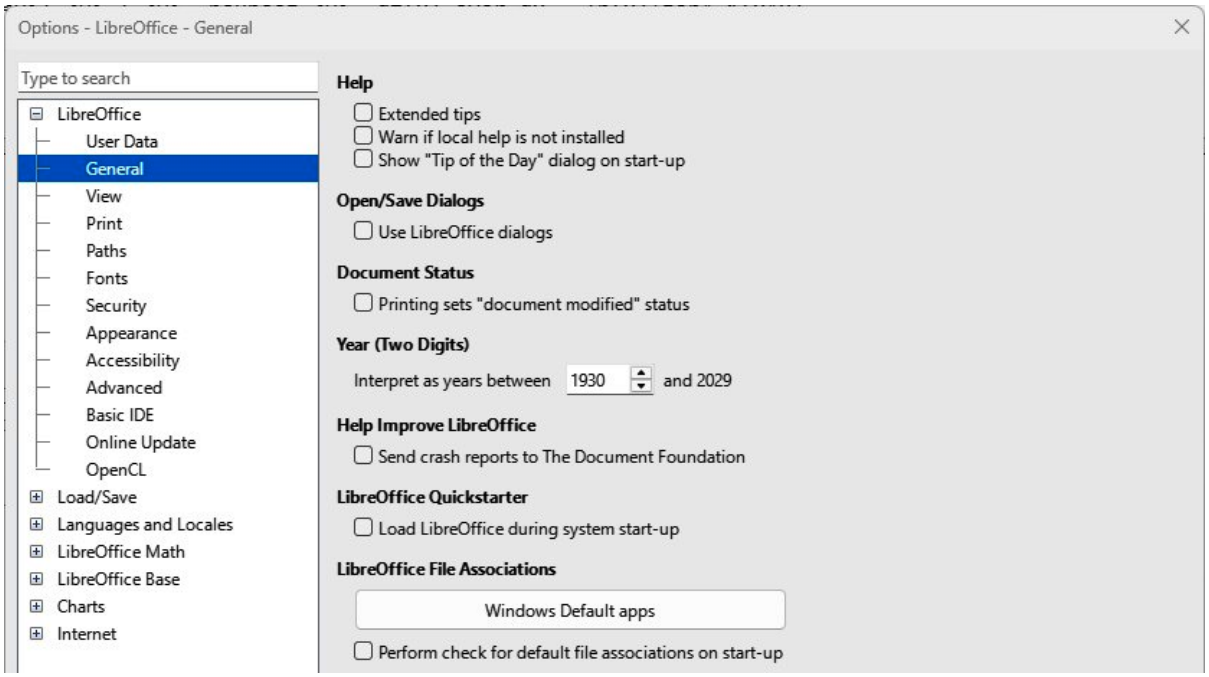


Figure 1: Options LibreOffice dialog — General page

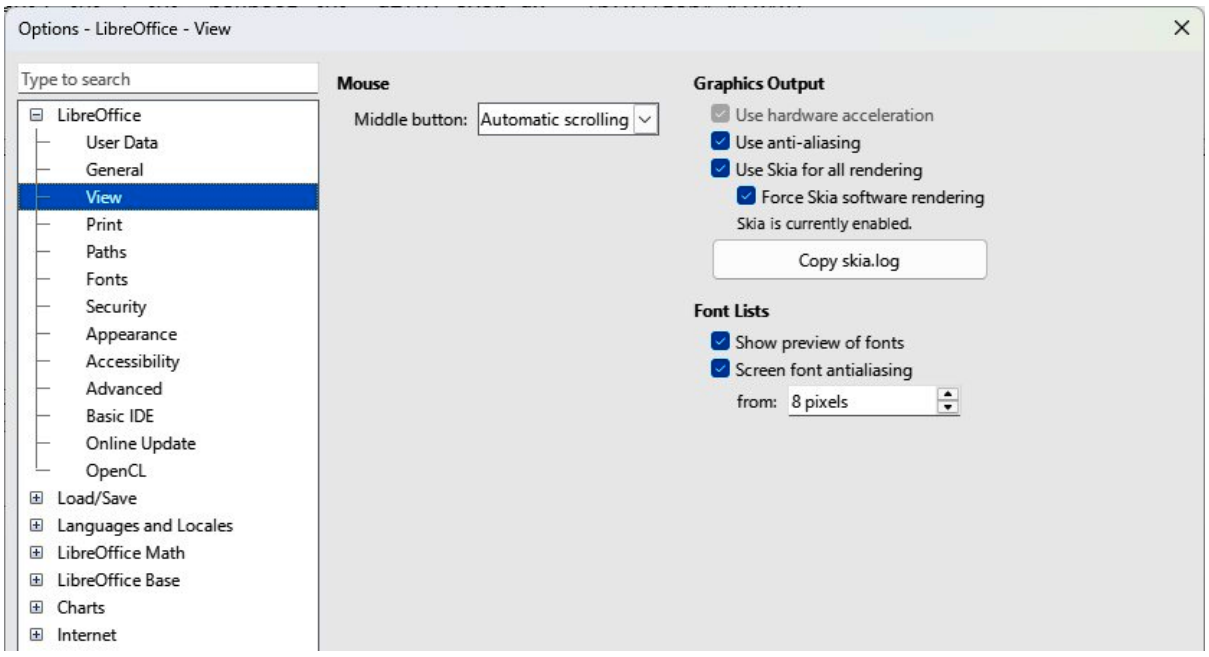


Figure 2: Options LibreOffice dialog — View page

1) On Linux and Windows operating systems, go to **Tools > Options > LibreOffice > Appearance** (macOS **LibreOffice > Preferences > LibreOffice > View**) on the Menu bar to open the dialog page for view options (Figure 2).
2) In **Icon Theme**, select a font from the options available in the drop-down list.
3) In **Icon Size**, select the required size from the drop-down lists for *Toolbar*, *Notebookbar* and *Sidebar*.
4) Click **OK** to save the settings and close the dialog.

Notes

Some Linux operating systems, for example Ubuntu, include LibreOffice as part of the installation and may not include the required icon set. Icon sets can be downloaded from the software repository for the Linux operating system being used.

Some of the previously included icon sets are now available only as extensions. Go to https://extensions.libreoffice.org/?Tags%5B%5D=49 or search for specific ones. For example, the People Gallery is available from https://extensions.libreoffice.org/en/extensions/show/people-gallery.

Using LibreOffice on macOS

Some keystrokes and menu items are different on computers operating macOS from computers using Windows and Linux operating systems. Table 2 provides some common substitutions used in this user guide. For more information on keyboard shortcuts, see **LibreOffice Help** and Appendix A, Command Reference in this user guide.

Table 2: Example of macOS keyboard shortcuts

Windows or Linux	***macOS equivalent***	***Effect***
Tools > Options	**LibreOffice > Preferences**	Access to setup options
Right-click	*Ctrl*+click and/or right-click depending on computer setup	Opens a context menu
Ctrl or *Control*	⌘ and/or *Cmd* or *Command*, depending on keyboard	Used with other keys
Alt	⌥ and/or *Alt* or *Option* depending on keyboard	Used with other keys
F11	⌘+*T*	Open the Styles deck in the Sidebar

Who wrote this user guide?

This user guide was written by volunteers from the LibreOffice community. Profits from sales of the printed edition are used to benefit the community.

Frequently asked questions

How is LibreOffice licensed?

LibreOffice is distributed under the Open Source Initiative (OSI) approved Mozilla Public License (MPL), see https://www.libreoffice.org/about-us/licenses/. It is based on code from Apache OpenOffice made available under the Apache License, 2.0, but includes software that differs from version to version under a variety of other Open Source licenses. New code is available under LGPL 3.0 and MPL 2.0.

Can LibreOffice be distributed to anyone?

Yes.

Can LibreOffice be sold?

Yes.

Can LibreOffice be used in a business?

Yes.

How many computers can LibreOffice be installed on?

As many as required.

Is LibreOffice available in different languages?

LibreOffice has been translated (localized for more than 80%, both UI and Help) into over 46 languages, so a required language is probably supported. Localization is well under way for another 30+ languages (50-80%) and for another 50+ languages help is more than welcome. In addition, over 70 spelling, hyphenation, and thesaurus dictionaries are available for languages and dialects that do not have a localized program interface. The dictionaries are available from the LibreOffice website at: https://www.libreoffice.org/.

How can LibreOffice be freely available?

LibreOffice is developed and maintained by volunteers and has the backing of several organizations. LibreOffice also relies upon donations from its users. To make a donation, go to the following web page: https://www.libreoffice.org/donate/.

Can the programming code from LibreOffice be used when developing a software application?

Yes, but follow the parameters set in the MPL and/or LGPL. Read the licenses: https://www.mozilla.org/MPL/2.0/.

Why is Java required to run LibreOffice and is it written in Java?

LibreOffice is not written in Java, but written in the C++ language. Java is one of several languages that can be used to extend the software. The Java JDK/JRE is only required for some features. The most notable one is the HSQLDB relational database engine.

Java is available at no cost. More information and download links to the appropriate edition for an operating system can be found at: https://java.com/en/download/manual.jsp.

Note

If LibreOffice features requiring Java are to be used, it is important that the correct 32-bit or 64-bit edition matches the installed version of LibreOffice. If Java is not to be used, nearly all of the LibreOffice features can still be used.

How can users contribute to LibreOffice?

Users can help with the development and user support of LibreOffice in many ways, and there is no need to be a programmer. To start, check out this webpage: https://www.libreoffice.org/community/get-involved/. An interactive web page that guides users in contributing with their best skills available at https://whatcanidoforlibreoffice.org.

Can the PDF copy of this user guide be distributed, or printed and copies sold?

Yes, as long as requirements are met for one of the licenses in the copyright statement at the beginning of this user guide. There is no need to request special permission. LibreOffice requests that users share with the LibreOffice project some of the profits made from sales of user guides, in consideration of all the work that LibreOffice volunteers have put into producing user guides.

What is new in LibreOffice 26.2?

The LibreOffice 26.2 Release Notes are available at this link https://wiki.documentfoundation.org/ReleaseNotes/26.2. Available at this link are release notes for earlier versions of LibreOffice giving more information on the features included in LibreOffice.



LibreOffice

Math Guide 26.2

Chapter 1,
Creating and Editing Formulas

Introduction

Math is used to create or edit formulas (equations) in LibreOffice and is included as a separate LibreOffice module. These formulas are then used in LibreOffice documents, or as stand alone objects. An example of the Math opening window is shown in Figure 3. The Sidebar **Elements** panel also opens.

When creating formulas in Math, a markup language is used for formulas and is designed to be easily read. For example, when *df(x) over dx = ln(x)+tan^-1(x^2)* is entered into the Formula Editor, the following formula appears in the Formula Preview.

$$df\frac{(x)}{dx}=\ln(x)+\tan^{-1}(x^2)$$

Note

Formulas used in this document are examples ONLY. How formulas are displayed when using Math depends on computer setup and computer operating system.

Formulas as documents or files

Using the Formula Editor, a formula is created as a separate file for insertion into a formula library. Alternatively, formulas can be directly inserted into documents created using Writer, Calc, Impress, or Draw. For more information refer to the user guide for the specific LibreOffice module being used.

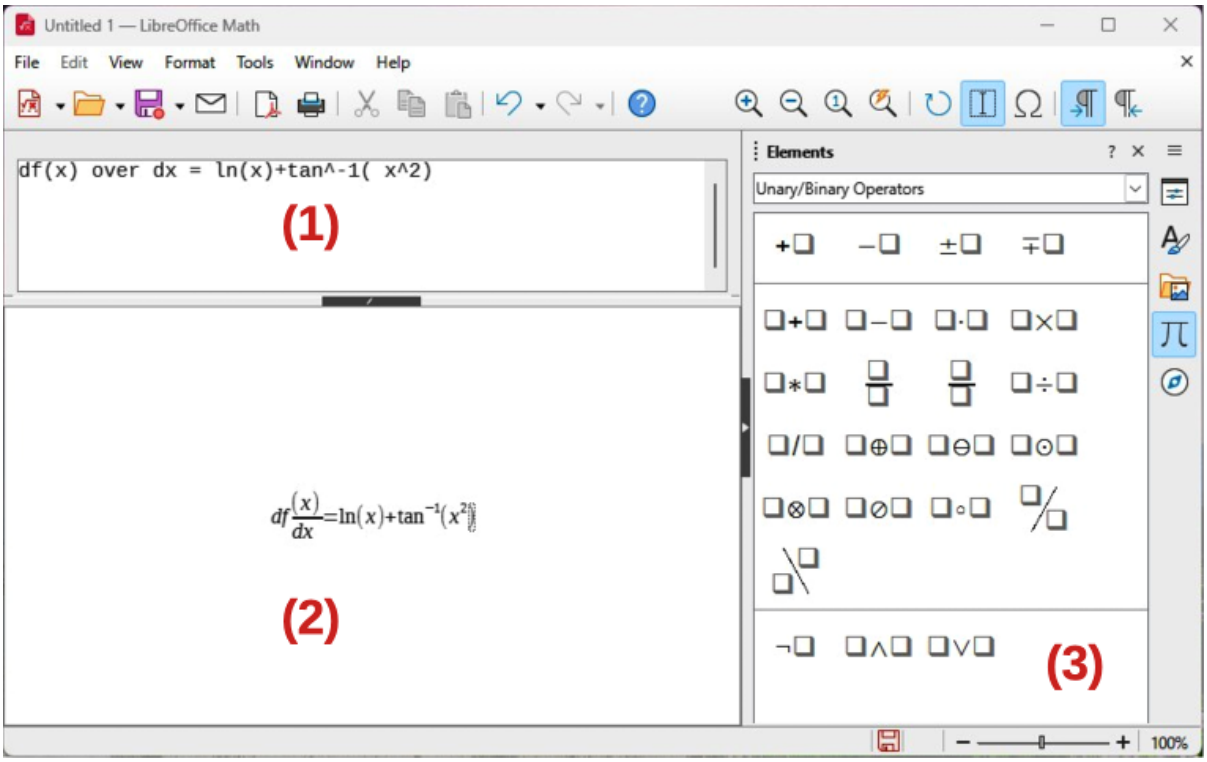


Figure 3: Math opening window

(1) Formula Editor (2) Formula Preview (3) Sidebar Elements Panel

Formula files

When characters are entered into the Formula Editor window (Item (1) in Figure 3) using markup language, the formula being created appears in the Formula Preview window (Item (2) in Figure 3). For more information on creating formulas as a separate document or file, see “Creating formulas” on page 20.

1) Use one of the following methods to create a formula as a separate document or file:
   - In the LibreOffice Start Center, click on **Math Formula**.
   - Go to **File > New > Formula** on the Menu bar.
   - On the Standard toolbar, click the triangle to the right of the **New** icon and select **Formula** from the context menu.
   - When Math is open, use the keyboard shortcut *Ctrl+N* (macOS ⌘+*N*).
2) When the formula is completed as a new document or file, save the formula in ODF format.

LibreOffice documents

A formula can also be created and inserted directly into a Writer, Calc, Draw, or Impress document. An example of creating a formula in a Writer document is shown in Figure 4. The formula box is automatically selected ready for a formula to be entered or edited. Depending on the LibreOffice module being used when inserting a formula, the cursor position changes of where to insert the formula.

- In Writer, click in the paragraph where the formula is being inserted.
- In Calc, click in the spreadsheet cell where the formula is being inserted.
- In Draw and Impress, a formula is automatically inserted at the center of a drawing or slide.

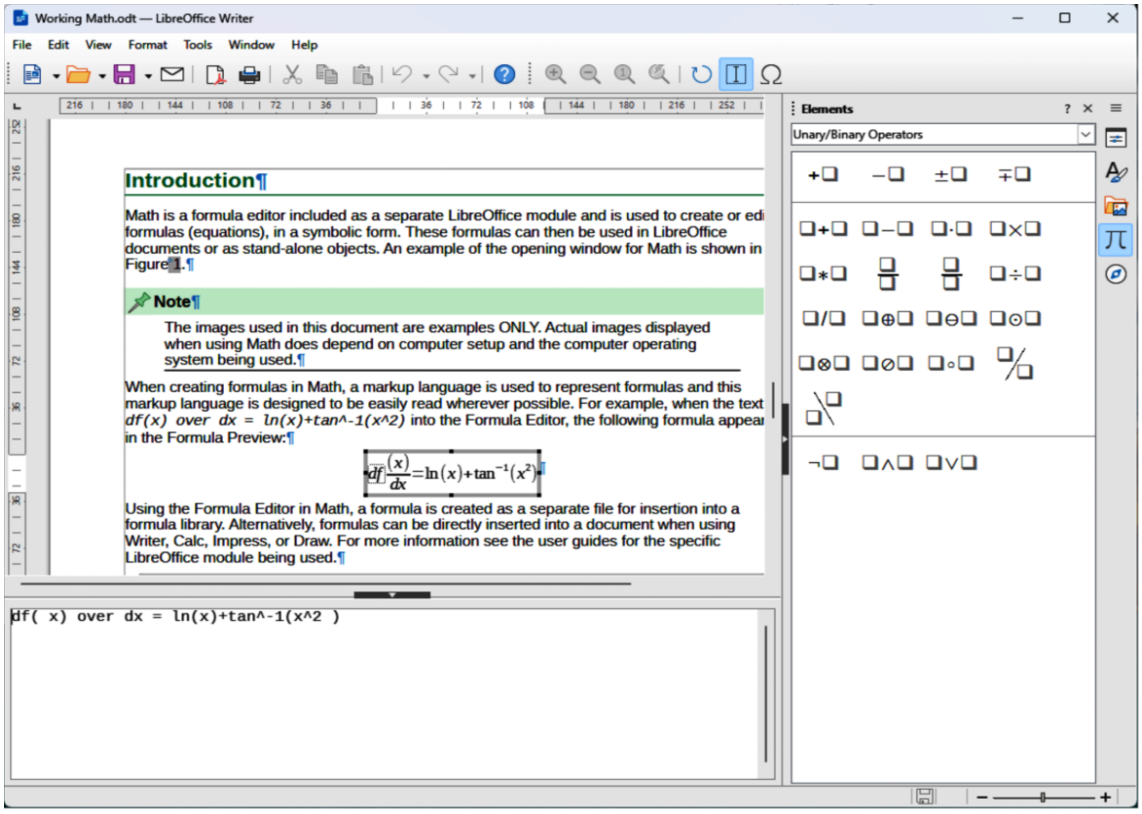


Figure 4: Example of creating a formula in Writer

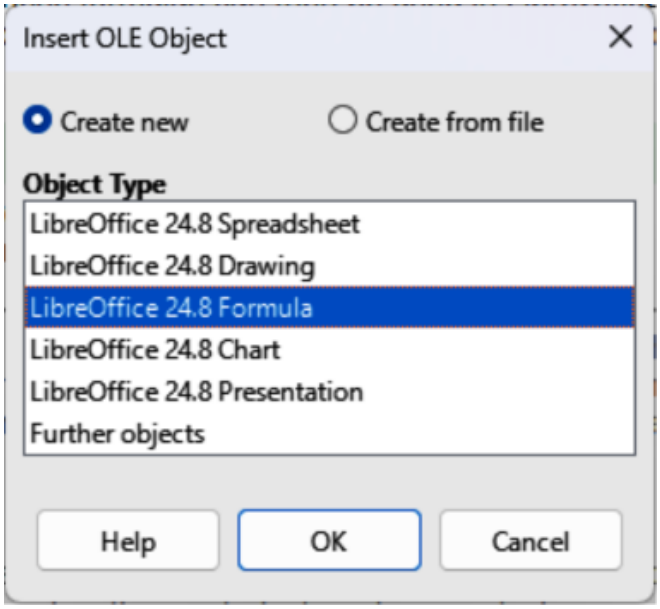


Figure 5: Insert OLE Object dialog

Formulas are inserted as OLE objects into documents, as follows:

1) Go to **Insert > OLE Object > Formula Object** on the Menu bar to open the Formula Editor.
2) Alternatively, go to **Insert > OLE Object > OLE Object** on the Menu bar to open the Insert OLE Object dialog (Figure 7).
   a) Select *Create new*, then select *LibreOffice XX Formula* in **Object Type**.
   b) Click OK to open the Formula Editor.
3) Create or edit the formula in the Formula Editor.
4) When the formula has been created or edited, close the Formula Editor using one of the following methods:
   – Press the *Esc* key.
   – Click outside the formula in the document. Double-click on the formula in the document to open the Formula Editor again to edit the formula.

For more information on inserting formulas as OLE objects, refer to the following:

Chapter 2, Formulas in Writer

Chapter 3, Formulas in Calc, Draw, and Impress

Writer, *Calc*, *Draw*, or *Impress Guides*.

Tips

If formulas are frequently inserted into documents, it is recommended to add **Formula Object** tool to the Standard toolbar, or create a keyboard shortcut. For more information, see Chapter 4, Customization or the *Getting Started Guide*.

Using the **Elements** panel on the Sidebar, or the context menus to create a formula is a convenient way to learn the markup language used used to create formulas.

When using the **Elements** panel on the Sidebar, it is recommended to have *Extended Tips* selected in LibreOffice Options. *Extended Tips* helps in identifying the categories and symbols used in formulas. Go to **Tools > Options > LibreOffice >**

General (macOS **LibreOffice > Preferences > LibreOffice > General**) on the Menu bar and select *Extended Tips* in the **Help** section.

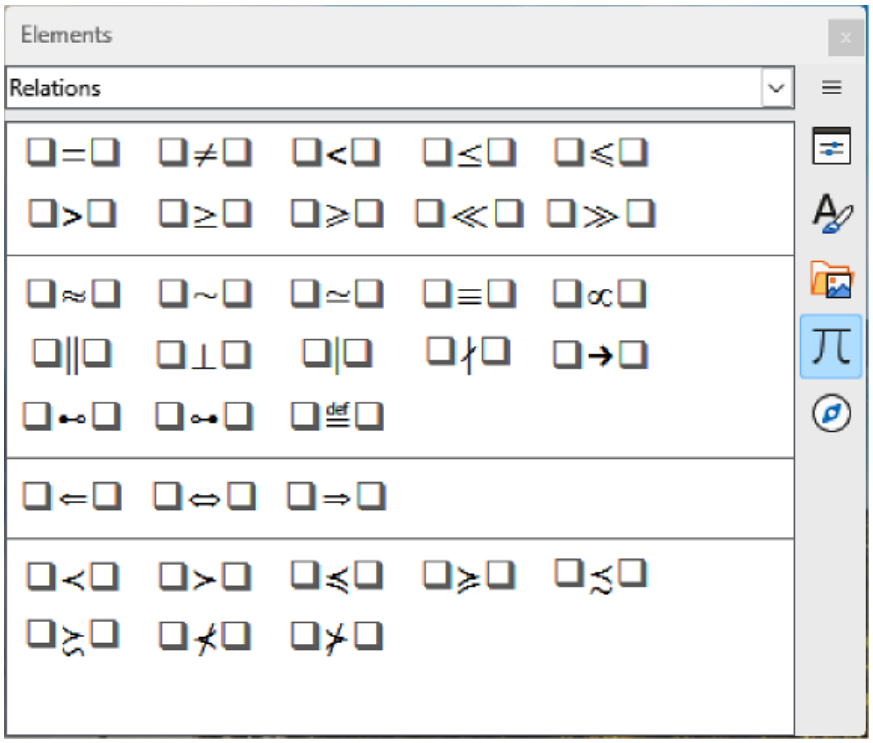


Figure 6: Sidebar Elements panel — Relations

Creating formulas

Formulas are created using one of the following methods:

- In theSidebar **Elements** panel (Figure 6), select a category from the context menu, then select a formula element from the available options.
- Right-click in the Formula Editor (Figure 3 on page 17)and select a category from the context menu, then select a formula element from the drop-down list.
- Enter markup language directly in the Formula Editor.

Elements panel

The **Elements** panel (Figure 6) on the Sidebar is a visual tool for creating and formatting formulas. The elements panel organizes mathematical symbols and operators into categories. The mathematical symbols are available in a drop-down list for each element category.

1) Select **View > Elements** on the Menu bar to open the **Elements** panel in the Sidebar.
2) Select the category required for the formula from the *Categories* drop-down list at the top of the **Elements** panel.
3) Select the symbol required from the options available in the **Elements** panel. The available symbols change according to the selected category.
4) After selecting a symbol, the symbol appears in Formula Preview and the Formula Editor is updated with the markup language used for the selected symbol.
5) Alternatively, select an example from the *Examples* list in the *Categories* drop-down list on the Sidebar (Figure 7).

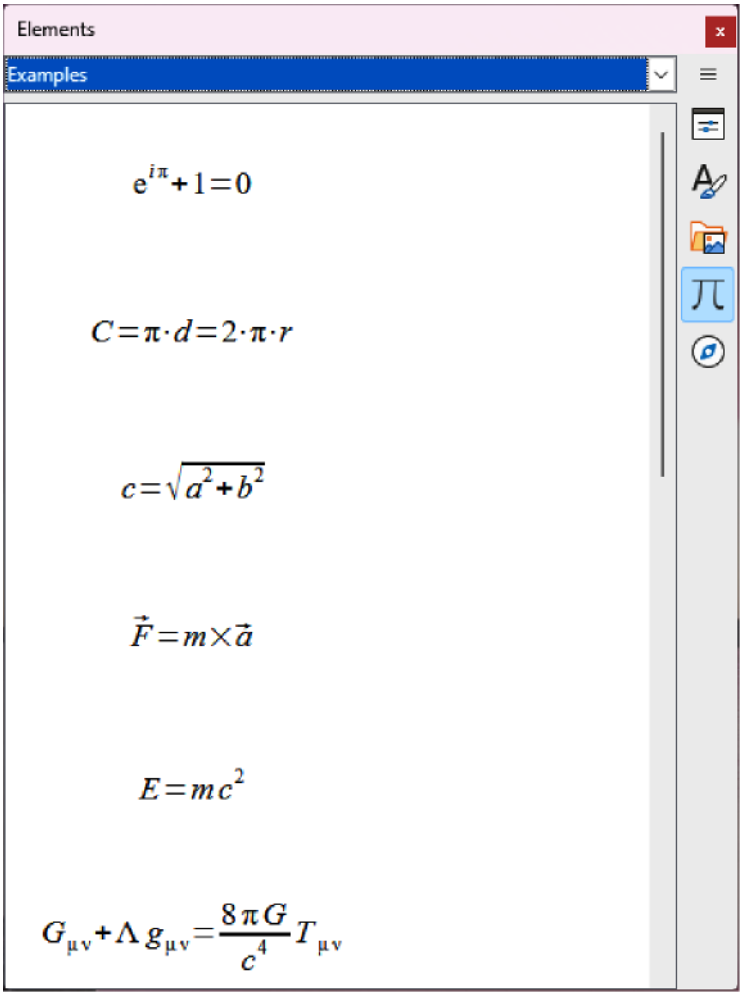


Figure 7: Sidebar Elements panel — Examples page

Context menu

The Formula Editor also provides a context menu to access categories and markup language options when creating a formula. An example is shown in Figure 9.

1) Right-click in the Formula Editor to open the context menu.
2) Select a category from the options listed in the context menu.
3) Select a markup language option from the options available in the drop-list.

Note

The **Elements** panel and the context menu in the Formula Editor provide the most common options and symbols used in formulas. Other options and symbols not listed have to be entered manually using markup language. For a complete list of commands and symbols available in Math, see Appendix A, Commands Reference.

Markup language

Markup language is entered directly into the Formula Editor. For example, typing `5 times 4` into the Formula Editor creates a simple formula. Using markup `language` is the recommended method for creating formulas. Table 3 shows examples of using markup language to enter commands. For a full list of commands that can be used in the Formula Editor, see Appendix A, Commands Reference.

Table 3: Example commands using markup language

Display	Command	Display	Command
$a=b$	`a = b`	$\sqrt{a}$	`sqrt {a}`
a^2	`a^2`	a_n	`a_n`
$\int f(x)dx$	`int f(x) dx`	$\sum a_n$	`sum a_n`
$a \leq b$	`a <= b`	∞	`infinity`
$a \times b$	`a times b`	$x \cdot y$	`x cdot y`

Greek characters

Using markup language

Greek characters are commonly used in formulas, but cannot be entered into a formula using the **Elements** panel in the sidebar, or the context menu in the Formula Editor. English names are used for Greek characters in markup language when Greek characters are required for a formula. See Appendix A, Commands Reference for a list of Greek characters that can be entered using markup language.

- For lowercase Greek characters, type percentage % sign, then the character name in lowercase using the English name. For example, typing %lambda creates the Greek character λ.
- For UPPERCASE Greek characters, type a percentage % sign, then type the character name in UPPERCASE using English. For example, typing %LAMBDA creates the Greek character Λ.
- For italic Greek characters, type a percentage % sign followed by the i character, then type the English name of the Greek character in lower or UPPER case. For example, typing %ilambda creates the italic Greek character λ and typing %iLAMBDA creates italic Greek character Λ

Symbols dialog

Greek characters can also be entered into a formula using the Symbols dialog.

1) Make sure the cursor is at the correct position in the Formula Editor.
2) Go to **Tools > Symbols** on the Menu bar, or click **Symbols** in the Tools toolbar (Figure 8) to open the Symbols dialog.
3) Select *Greek* in the **Symbol** set drop-down list (Figure 9). For italic characters, select *iGreek* in the **Symbol** drop-down list (Figure 10).
4) Double-click the Greek character required, or select the Greek character and click **Insert**. When selected, the name of the character is shown below the symbol list.
5) Click on **Close** to close the Symbols dialog.



Figure 8: Tools toolbar

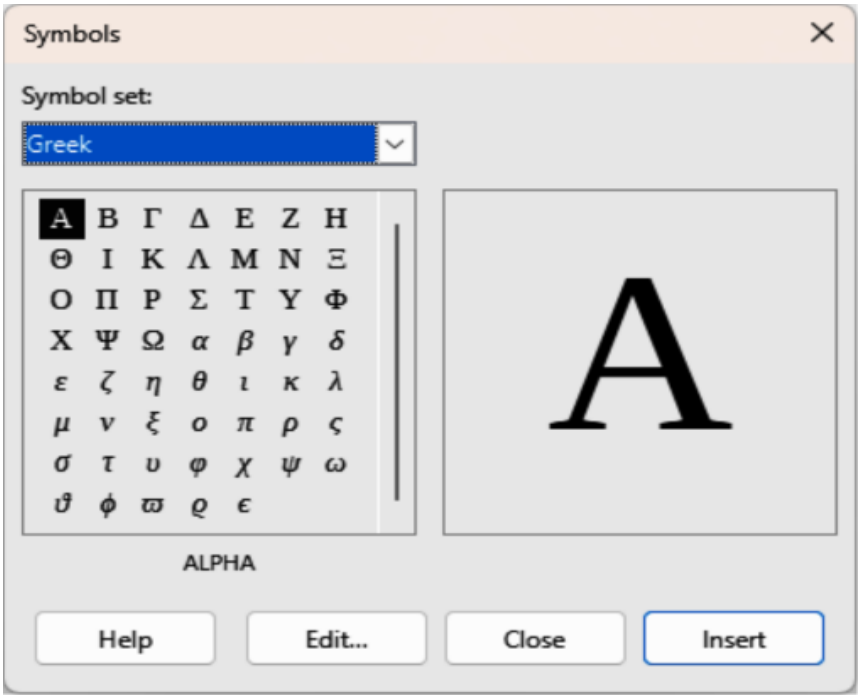


Figure 9: Symbols dialog — Greek

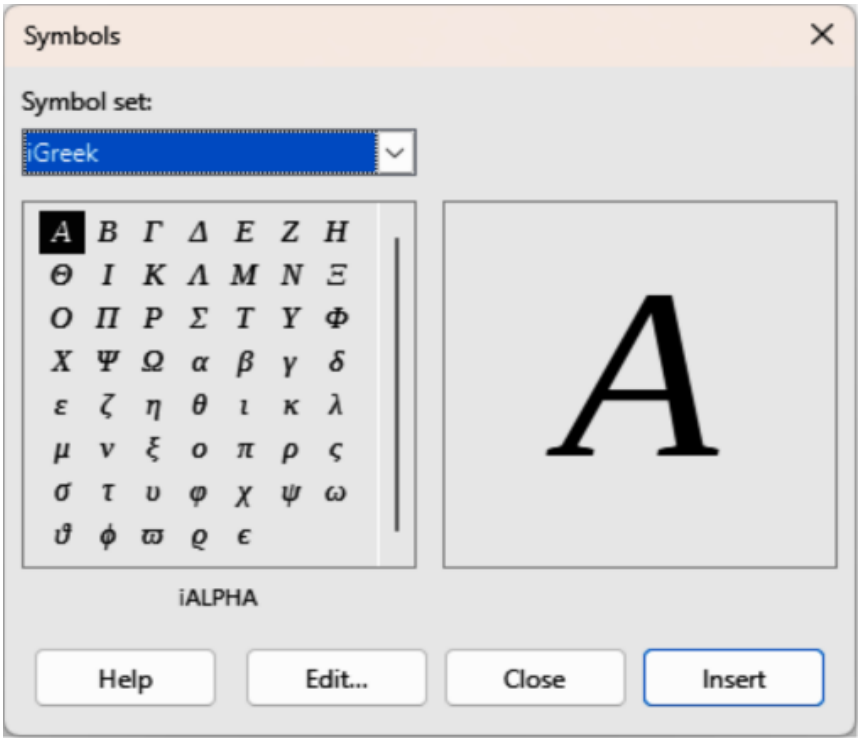


Figure 10: Symbols dialog — Greek Italic

Formula examples

Example 1

A simple formula 5×4 is created in Math as follows:

1) Make sure the cursor is located and flashing in the Formula Editor.
2) Enter Multiplication symbol using one of the following methods:

– Select the category *Unary/Binary Operators* in the **Elements** panel on the Sidebar, then select the Multiplication icon.

– Right-click in the Formula Editor and select Unary/Binary Operators, then `a times b` from the context menu.

– Using markup language, type `5 times 4` in the Formula Editor.

3) Select the first placeholder <?> before the word `times` in the Formula Editor and replace with the character `5`. The formula in the document updates automatically.

4) Select the second placeholder <?> after the word `times` in the Formula Editor and replace it with the character `4`. The formula in the document updates automatically.

Notes

The first two methods above place the formula text `<?> times <?>` in the Formula Editor and the symbol ❑×❑ appears in the document. The third method above, using markup language in the Formula Editor, places the formula directly into the document and it is not necessary to carry out the following steps.

If necessary, to prevent a formula in a document from updating automatically, go to **View** on the Menu bar and deselect **AutoUpdate Display**. The formula can be updated manually when necessary. Press the *F9* key or select **View > Update** on the Menu bar.

Example 2

The formula for Pi $\pi \simeq 3.14159$ where value is rounded to 5 decimal places is created as follows. The Greek character (pi) is known, but the required markup language `Similar Or Equal` symbol $\simeq$ is not known.

1) Make sure the cursor is located and flashing in the Formula Editor.

2) Enter `%pi` in the Formula Editor to create the Greek character for Pi (π).

3) Select the category *Relations, then s*elect the symbol `Is Similar Or Equal` using one of the following methods:

– In the **Elements** panel on the Sidebar, select *Relations* from the drop-down list and then select the `Is Similar Or Equal` icon ❑≃❑.

– Right-click in the Formula Editor and select *Relations* > `a simeq b` from the context menu.

4) Delete the first placeholder symbol <?> before the word `simeq` in the Formula Editor.

5) Select the second placeholder symbol <?> after the word `simeq` in the Formula Editor and enter 3.14159 to delete the second placeholder symbol. The formula now appears in the document.

Editing formulas

How a formula is edited and opening formula editing mode depends on whether the formula is in Math or another LibreOffice module.

1) To select the formula and open Formula Editor using one of the following methods:

– In Math, double-click on a formula element in the formula in Formula Preview to select the formula element, or directly select a formula element in the Formula Editor.

– In Writer, Calc, Impress, or Draw, double-click on the formula, or right-click on the formula and select **Edit** in the context menu, to open the Formula Editor and enter editing mode. The cursor is positioned at the start of the formula in the Formula Editor.

Note

If a formula element cannot be selected using the cursor, click on **Formula Cursor** in the Tools toolbar (Figure 8 on page 22) to activate the formula cursor.

2) Select the formula element to be changed using one of the following methods:
   – Click on the formula element in Formula Preview, then position the cursor at the beginning of the formula element in Formula Editor and select the formula element.
   – Double-click on the formula element in Formula Preview to select the formula element.
   – Position the cursor in the Formula Editor at the formula element to edited, then select the formula element.
   – Double-click directly on the formula element in the Formula Editor to select it.
3) Make all necessary changes to the selected formula element.
4) Go to **View > Update** on the Menu bar, press the *F9* key, or click on **Update** on the Tools toolbar to update the formula.
5) Save the changes to the formula using one of the following methods:
   – In Math, save the changes to the formula after editing before closing Math.
   – In Writer, Calc, Impress, or Draw, click anywhere in the document away from the formula to exit editing mode and save the document.

Formula layout

Using braces

Braces (curly brackets) in a formula have to be used to define the order of formula operations. The following examples show how braces are used in a formula.

Example 1

`2 over x + 1` creates the formula $\frac{2}{x}+1$

Math recognized the 2 before and the x after the over command belong to a fraction and has represented them accordingly. If $x+1$ is required rather than x to be a denominator, $x+1$ must be bracketed together using braces to create the correct formula. Inserting braces as shown in `2 over {x + 1}` creates the correct result $\frac{2}{x+1}$ where $x+1$ is now the denominator.

Example 2

`– 1 over 2` creates the formula $\frac{-1}{2}$

Math has recognized the minus sign as a prefix for the `1` and has placed it as the numerator of the fraction. If it is required that the whole fraction is negative with the minus sign in front of the fraction, the fraction must be placed in braces to instruct Math that all the characters are part of

the whole fraction. Adding braces into the markup language `– {1 over 2}` creates the fraction $-\frac{1}{2}$ and the whole fraction is now negative.

Example 3

Braces are used in markup language to define the formula layout and are not displayed or printed. If braces used within a formula are to be printed, use the commands `lbrace` and `rbrace` within the markup language.

x over {–x + 1} gives the formula $\frac{x}{-x+1}$

Replace the braces with `lbrace` and `rbrace` in the markup language as follows:

`x over lbrace –x + 1 rbrace` creates the formula $\frac{x}{\{-x+1\}}$

Brackets (parentheses) and matrices

To use a matrix in a formula, the matrix command has to be used.

Example 1

To create a simple example of a 2x2 matrix. In matrices, rows are separated by two hashes (##) and entries within each row are separated by one hash (#).

`( matrix { a # b ## c # d } ` ⇨ $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Example 2

Normally, when brackets are used within a matrix, the brackets do not scale as the matrix increases in size. The following example shows a formula where the parentheses do not scale to the size of the resulting matrix.

`( matrix { a # b ## c # d }  )` ⇨ $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Example 3

To solve this problem, Math provides scalable brackets that change in size to match the matrix size. The commands `left(` and `right)` have to be used to create scalable brackets around a matrix. The following example shows how to create a matrix with scalable parentheses.

`left( matrix { a # b ## c # d } right)` ⇨ $\left(\begin{matrix} a & b \\ c & d \end{matrix}\right)$

Scalable brackets can be used with any formula element, such as fraction, square root, and so on.

To create a matrix where some values are empty, the grave accent (`) is used so that Math places a small space in the position where grave accent (`) has been placed, as shown in the following example:

`left( matrix { 1 # 2 # 3 ## 4 # ` # 6 } right)` ⇨ $\left(\begin{matrix} 1 & 2 & 3 \\ 4 & & 6 \end{matrix}\right)$

Tip

Use the commands `left[` and `right]` to obtain square brackets. A list of all brackets available within Math can be found in Appendix A, Commands Reference. If all brackets are to be scalable, go to **Format > Spacing** on the Menu bar to open the

Spacing dialog. Click on **Category** and select *Brackets* from the drop-down list, then select the option *Scale all brackets* from the drop-down list.

Unpaired brackets

When using brackets in a formula, Math expects that for every opening bracket there will be a closing bracket. If a closing bracket is not used, Math places an inverted question mark next to where the closing bracket should have been placed.

For example, `lbrace a; b` creates ¿ because the right bracket `rbrace` is missing. This inverted question mark disappears when all brackets are paired. This is corrected by using `lbrace a; b rbrace`, which creates $\{a;b\}$. However, there are cases where an unpaired bracket is required.

Non scalable brackets

A backslash `\` is placed before a non scalable bracket to indicate that the subsequent character is not a bracket, but a literal character.

Example

The unpaired brackets in the formula [a; b [results in an inverted question mark being entered because Math expects that `[` will be closed by `]`. To correct the error, use a backslash and insert `\ [ a; b \ [` into the Formula Editor to create the correct result $[a;b[$.

Scalable brackets

To create unpaired scalable brackets or braces in a formula, the markup commands `left`, `right`, and `none` are used.

Example

The following formula uses the stack command to create a two-line formula and adds a bracket only to the right side of the stack. This effect is achieved by using the left `lbrace` command combined with the right `none` command. The first command indicates that the left bracket is a `lbrace` and the second command `none` tells Math that there is no right bracket added to the right side of the formula.

```
abs x = left lbrace stack {x "for" x
>= 0 # -x "for" x < 0} right none
```

⇨ $|x| = \begin{cases} x \text{ for } x \geq 0 \\ -x \text{ for } x < 0 \end{cases}$

Recognizing functions

In the normal installation of Math, Math creates functions using normal characters and variables in *italic* characters. However, if Math fails to recognize a function, Math can be forced to recognize that a function has been entered. Using the markup command `func` before a function forces Math to recognize following text as a function. For a full list of functions available in Math, see Appendix A, Commands Reference.

Some Math functions have to be followed by a number or variable. If a number or variable is missing, Math places an inverted question mark where the missing number or variable should be. To remove the inverted question mark and correct the formula, enter a number, variable, or pair of empty brackets as a placeholder.

Tip

Navigate through formula errors using the function key *F3* to move to the next error, or use the function key combination *Shift+F3* to move to the previous error.

Multiple line formulas

When creating a formula that requires more than one line, for example $\begin{matrix} x=3 \\ y=1 \end{matrix}$, the first reaction is to use the *Enter* key. However, when the *Enter* key is used, the markup language in the Formula Editor goes to a new line, but the resulting formula does not have two lines. To add a new line into a formula, the markup command `newline` has to be used

Table 4 shows how the markup command `newline` is used to add a new line to a formula. The first example simply adds an `Enter` command to break the line in the markup language. This does not create a new line in the formula. The second example uses the `newline` command. The `newline` command creates a two-line formula.

Table 4: Examples of multiple line formulas

Markup Language	***Resulting Formula***
`x = 3` `y = 1`	$x=3\ y=1$
`x = 3 newline y = 1`	$\begin{matrix} x=3 \\ y=1 \end{matrix}$

It is not possible in Math to create multiple line formulas when a line ends with an equals sign and continue the formula on a new line. A term on the right side of the equals sign must be entered to create multiple line formulas.

If a multiple line formula is required to have an equals sign at the end of a line without a term after the equals sign, use empty quotes `“”`, empty braces `{}`, grave character `, or tilde character ~.

By default, the alignment of a multiple line formula is center aligned. For more information on alignment using the equals sign, see “Aligning formulas using equals sign” on page 31.

Formula element spacing

Spacing between elements in a formula is not set by using space characters in the markup language. To add spaces into a formula, use one of the following options:

- Grave `` ` `` to add a small space.
- Tilde `~` to add a large space.
- Space characters between quotes `“ ”`. Space characters are considered as text.

Note

Any spaces at the end of a line in the markup language are ignored by default. For more information, see Chapter 4, Customization.

Adding limits to summations and integrals

The `sum` and `int` commands are used for summations and integrals with the parameters `from` and `to` used to set the lower and upper limits. The parameters from and to can be used singly or together as shown by the following examples. For more information on the `sum` and `int` commands, see Appendix A, Commands Reference.

Table 5 shows how to add upper and lower bounds to summations and integrals. Note that brackets are used to define which parts of the formula correspond to the desired bounds.

Table 5: Example of using summations and integrals

Markup Language	Resulting Formula
`sum from k = 1 to n a_k`	$\sum_{k=1}^{n} a_k$
`sum to infinity 2^{-n}`	$\sum^{\infty} 2^{-n}$
`sum from{ i=1 } to{ n } sum from{ j=1; i <> j } to{ m } x_ij`	$\sum_{i=1}^{n} \sum_{j=1;i\neq j}^{m} x_{ij}$
`int from 0 to x f(t) dt`	$\int_{0}^{x} f(t)\,dt$
`int_0^x f(t) dt`	$\int_{0}^{x} f(t)\,dt$
`int from Re f`	$\int_{\Re} f$

Adding derivatives

When adding derivatives to a formula, indicate in Math that it is a fraction by using the `over` command.

- The `over` command is combined with the character `d` for a total derivative, or the `partial` command for a partial derivative to achieve the effect of a derivative.
- Braces `{}` are used on each side of elements to enclose the elements and make the derivative as shown in Table 6.

Table 6: Examples of adding derivatives

Markup Language	Resulting Formula
`{df} over {dx}`	$\frac{df}{dx}$
`{partial f} over {partial y}`	$\frac{\partial f}{\partial y}$
`{partial^2 f} over {partial t^2}`	$\frac{\partial^2 f}{\partial t^2}$

Note

To write function names with primes, as is normal in school notation, add the symbols to the catalog first. For more information, see Chapter 4, Customization.

Control characters as normal characters

Control characters are: `%, {, }, &, |, _, ^` and `"`. For example, `2% = 0.02` cannot be entered into markup language and expect the same characters to appear in a formula. To overcome this limitation in markup language, use one of the following methods:

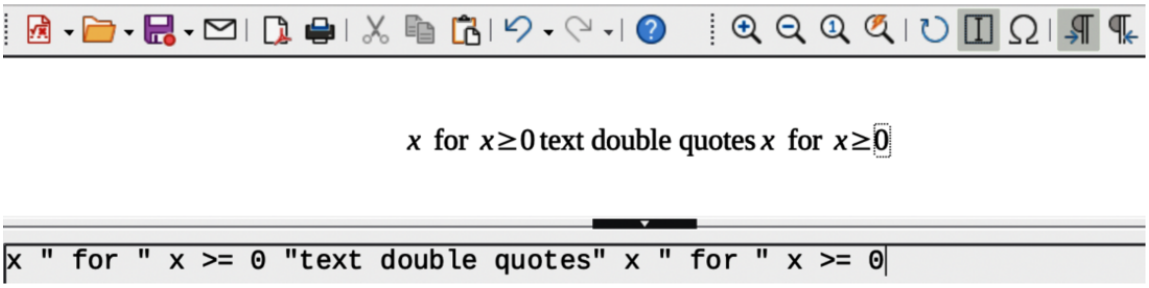


Figure 11: Example of using double quotes to create formula text

- Use double quotes to mark that character as text, for example `2"%"= 0.02` displays in a formula as $2\%=0.02$. However, double quotes cannot be used to enter a double quote character as text, see “Text in formulas” below.
- Add the character to the Math Catalog, for example the `double quote` character.
- Use commands, for example `lbrace` and `rbrace` for literal braces.

Note

Characters that are used as controls in markup language cannot be entered directly as normal characters.

Text in formulas

To include text in a formula, enclose any text in double-quotes, for example `x " for " x >= 0` in markup language creates the formula $x \text{ for } x \geq 0$. All characters, except double quotes, can be used in text.

If double quotes are required in formula text, any text created must be contained within double quotes. Text is then copied and pasted as text into the Formula Editor (Figure 11).

The font used for text in a formula is the default font that has been set in the Fonts dialog. For more information on how to change fonts used in formulas. For more information, see Appendix A, Commands Reference for more information.

By default, text alignment is left-justified in formulas. For more information, see Appendix A, Commands Reference for more information.

Formatting text in formulas

Formatting commands are not interpreted within text used in formulas. To use formatting commands within formula text, the text flow must be broken by using double quotes in the Formula Editor.

Example

Enter the following text in the Formula Editor using markup language to create text in a formula:

```
"In " color blue bold "isosceles" "triangles, the base angles are equal"
```

In **isosceles** triangles, the base angles are equal

The example above shows how markup language is used to change the text color to blue and use bold text for the word “isosceles” in the formula. Note that the commands are applied to any text placed between the double quotes. The formatting command `color blue bold` is not applied to the rest of the text in the example because it is a separate quote block.

Aligning formulas using equals sign

Math does not have a command for aligning formulas on a character. However, a matrix can be used to align formulas on the equals sign (=) character. The markup commands `alignr`, `alignl` and `alignc` are used to align each value inside a matrix to the right, left or center, respectively.

Example

The following uses a matrix to align formulas on the equals sign and uses alignment commands to align the contents of each position within the matrix. The spacing on each side of the equals sign cab be reduced if the inter-column spacing of the matrix is adjusted. See “Adjusting formula spacing” on page 34 for more information.

```
matrix{ alignr x+y # {}={} # alignl 2 ## alignr
x # {}={} # alignl 2-y }
```

$$\begin{matrix} x+y & = & 2 \\ x & = & 2-y \end{matrix}$$

Note

The empty braces each side of the equals sign are required. The equals sign is a binary operator and requires an expression on each side. Spaces, or characters ` or ~ on each side of the equals sign can be used, but it is recommended to use braces because braces are easier to see within markup language.

Changing formula appearance

Note

Only the default font or font size can be changed in a formula. Any formulas that were already inserted into a document will continue to use the previous settings for default font and font size. The font and font size of these previously inserted formulas have to be changed individually to use the new setting for default font and font size.

Tip

Extension *Formatting of All Math Formulas* can be used to change font name and font size for all or selected formulas in a document. The extension and instructions for use can be downloaded from the LibreOffice website using the following link: https://extensions.libreoffice.org/en/extensions/show/formatting-of-all-math-formulas.

Formula font size

Changing formula font size

To change the font size used in a formula already inserted into a LibreOffice document:

1) Click in the markup language in the Formula Editor.
2) Go to **Format > Font size** on the Menu bar to open the Font Sizes dialog (Figure 12).
3) Select a different font size from the options available in *Base size*, or enter a new font size in the text box.
4) Click **OK** to save the changes and close the dialog.

Example

Default font size 12pt: $\pi \simeq 3.14159$. Font size changed to 18pt: $\pi \simeq 3.14159$

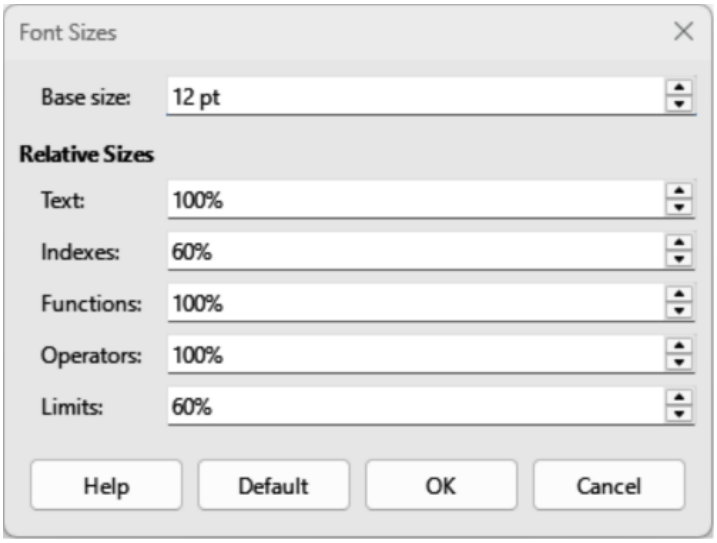


Figure 12: Font Sizes dialog

Default formula font size

To change the default font size used for all formulas in a LibreOffice document:

1) Before inserting any formulas into a document, go to **Format > Font size** on the Menu bar to open the Font Sizes dialog (Figure 12).
2) Select a different font size from the options available in *Base size*, or enter a new font size in the text box.
3) Click on **Default** and select **Yes** in the conformation dialog that opens to save the change in font size.
4) Click on **OK** to save the changes and close the Font Sizes dialog. Any new formulas created will use the new base size font for formulas.

Font size options

The Font Sizes dialog ((Figure 12) specifies the font sizes used in a formula. Change the base size and all elements in the formula created after the change will use the new base font size.

Base size

All elements of a formula are proportionally scaled to the base size. To change the base size, select or type in the desired point (pt) size. You can also use other units of measure or other metrics, which are then automatically converted to points.

Relative Sizes

Determines the relative sizes for each type of element with reference to the base size.

Text

Select the text size in a formula relative to the base size.

Indexes

Select the relative size for indexes in a formula in proportion to the base size.

Functions

Select the relative size for names and function elements in a formula in proportion to the base size.

Operators

Select the relative size of the mathematical operators used in a formula in proportion to the base size.

Limits

Select the relative size for the limits in a formula in proportion to the base size. This is used to determine the size for commands `from` and to used in summations and integrals.

Default

Select **Default** to save any changes to the font size options. The new font sizes are then used as default for all new formulas.

Formula fonts

Changing formula fonts

To change the fonts used for a formula in a LibreOffice document:

1) Click in the markup language in the Formula Editor.
2) Go to **Format > Fonts** on the Menu bar to open the Fonts dialog (Figure 13).
3) Select a new font for each of the various options in the drop-down lists in the Fonts dialog.
4) If the font required does not appear in the drop-down list, click **Modify** and select the option from the context menu to open a fonts dialog.
5) Select the font required and click **OK** to add the font to the drop-down list for the selected option.
6) Click **OK** to save the changes and close the Fonts dialog.

Default formula fonts

To change the default fonts used for all formulas in a LibreOffice document:

1) Before inserting any formulas into a document, go to **Format > Font size** on the Menu bar to open the Fonts dialog (Figure 13).
2) Select a new font for each the various options from the drop-down lists.
3) If the font required does not appear in the drop-down list, select **Modify** and then select an option from the context menu to open a fonts dialog.

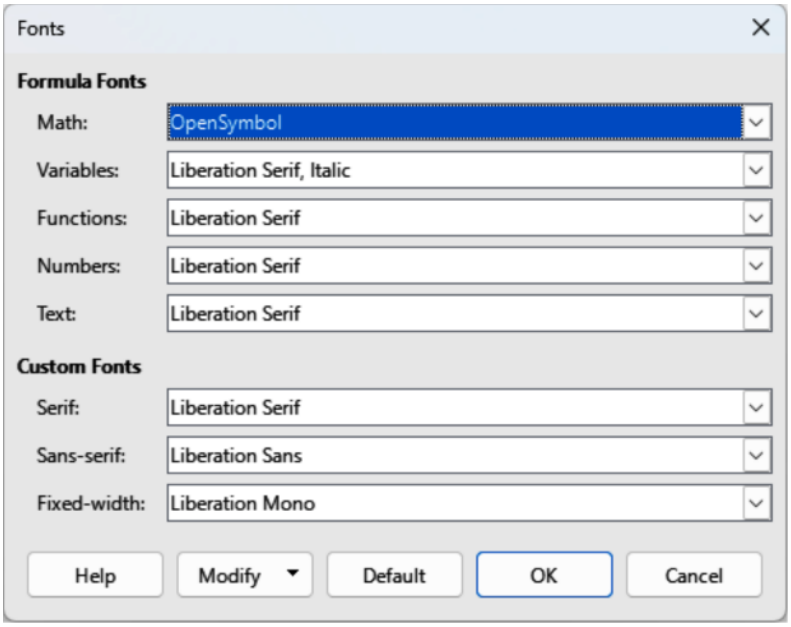


Figure 13: Fonts dialog

4) Select the font required and click **OK** to add the new font to the drop-down list for the selected option.
5) Click on **Default** and select **Yes** in the conformation dialog that opens to save the change in font type.
6) Click **OK** to save the changes and close the Fonts dialog.

Formula font options

Formula Fonts

Defines the fonts used for the variables, functions, numbers and inserted text used for formula elements.

Variables

Selects the fonts used for the variables in a formula.

Functions

Selects the fonts used for the function names and properties.

Numbers

Selects the fonts for the numbers in a formula.

Text

Defines the font used for text in a formula.

Custom Fonts

Defines fonts to format text components in a formula. The three basic fonts *Serif*, *Sans-serif* and *Fixed-width* are available. Other fonts can be added using the **Modify** option. All fonts installed on a computer system are available for use.

Serif

Specifies the font used for a serif font format.

Sans

Specifies the font used for a sans font format.

Fixed

Specifies the font used for a fixed font format.

Modify

Select an option from the context menu to access the Fonts dialog to define the font and font attributes for a formula and for any custom fonts being used.

Default

Click on **Default** to save any changes as the default for all new formulas. A confirmation message appears before any changes are saved.

Notes

When a new font is used for a formula, the old font remains available and can be selected again.

Variables are written in *italics.* For all other formula elements, the basic font form is used. The font style can be altered in the formula by using the commands `italic` or `bold`. To stop using an *italic* or **bold** font, use `nitalic` or `nbold` after the variable

Adjusting formula spacing

The alignment settings determine how formula elements located above one another are aligned horizontally relative to each other.

Current formula spacing

To change the spacing used in the current formula in a LibreOffice document:

1) Click in the markup language in the Formula Editor.
2) Go to **Format > Spacing** on the Menu bar to open the Spacing dialog (Figure 14).
3) Click **Category** and select one of the options from the drop-down list. The options in the Spacing dialog change depending on which category is selected.
4) Enter new values for the spacing category and click **OK**.
5) Check the spacing format in the formula. If the spacing format is not correct, repeat the steps above.

Default formula spacing

To change the default spacing used for all formulas in a LibreOffice document:

1) Before inserting any formulas into a document, go to **Format > Spacing** on the Menu bar to open the Spacing dialog (Figure 14).
2) Click **Category** and select one of the options from the drop-down list. The options in the Spacing dialog change according to the category selected.
3) Make the required changes to the category option, then click **Default** and confirm the changes to the formula spacing. Any new formulas created will use the new spacing for formulas.
4) Click **OK** to save the changes and close the Spacing dialog.

Note

Only new formulas inserted after the change in formula spacing will use the new setting for formula spacing. For any formulas inserted into a document before the default spacing was changed, the new default spacing will not be applied. The formula spacing on each formula already inserted into a document will have to be changed manually.

Formula spacing options

Use the **Category** options available in the Spacing dialog (Figure 14) to change the spacing for a formula element. The dialog that opens depends on the selected category. A preview window displays which spacing is being changed.

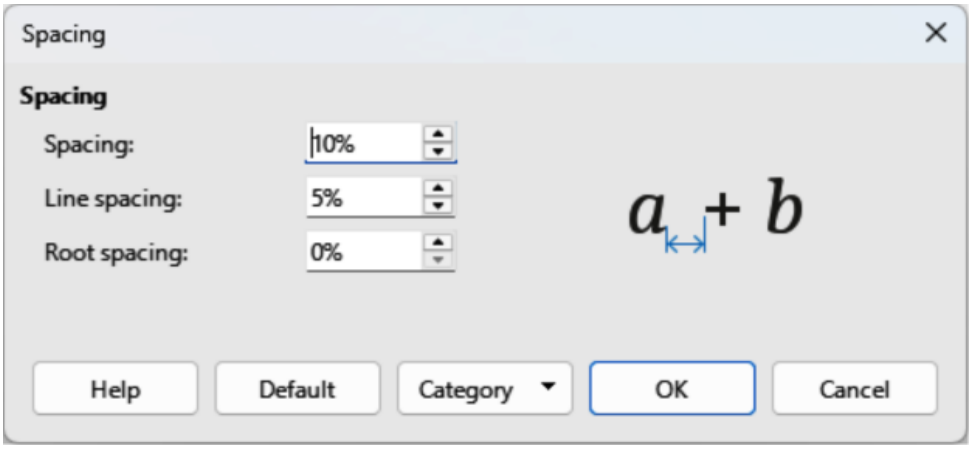


Figure 14: Spacing dialog

Category

Select a category to change the formula spacing.

Spacing

Defines the spacing between variables and operators, lines, and root signs and radicals.

Spacing

Determines the spacing between variables and operators

Line Spacing

Determines the spacing between lines.

Root Spacing

Determines the spacing between the root sign and radicals.

Indexes

Defines the spacing for superscript and subscript indexes.

Superscript

Determines the spacing for superscript indexes.

Subscript

Determines the spacing for subscript indexes.

Fractions

Defines the spacing between the fraction bar and the numerator or denominator.

Numerator

Determines the spacing between the fraction bar and the numerator.

Denominator

Determines the spacing between the fraction bar and the denominator.

Fraction Bars

Defines the excess length and line weight of the fraction bar.

Excess length

Determines the excess length of the fraction line.

Weight

Determines the weight of the fraction line.

Limits

Defines the spacing between the sum symbol and the limit conditions.

Upper limit

Determines the spacing between the sum symbol and the upper limit.

Lower limit

Determines the spacing between the sum symbol and the lower limit.

Brackets

Defines the spacing between brackets and the content.

Excess size (left/right)

Determines the vertical distance between the upper edge of the contents and the upper end of the brackets.

Spacing

Determines the horizontal distance between the contents and the upper end of the brackets.

Scale all brackets

Scales all types of brackets. If `( a over b)` is entered into the Formula Editor, the brackets surrounds the whole height of the argument.

Excess size

Adjusts the percentage excess size. At 0% the brackets are set so that they surround the argument at the same height. The higher the value that is entered, the larger the vertical gap between the contents of the brackets and the external border of the brackets. The option can only be used in combination with *Scale all brackets*.

Matrices

Defines the relative spacing for the elements in a matrix.

Line spacing

Determines the spacing between matrix elements in a row.

Column spacing

Determines the spacing between matrix elements in a column.

Symbols

Defines the spacing of symbols in relation to variables

Primary height

Defines the height of the symbols in relation to the baseline.

Minimum spacing

Determines the minimum distance between a symbol and variable.

Operators

Defines the spacing between operators and variables or numbers.

Excess size

Determines the height from the variable to the operator upper edge.

Spacing

Determines the horizontal distance between operators and variables.

Borders

Adds a border to a formula. This option is particularly useful if you want to integrate the formula into a text file in LibreOffice Writer by combining mathematical formulas and text inside the same paragraph.

Left

The left border is positioned between the formula and background.

Right

The right border is positioned between the formula and background.

Top

The top border is positioned between the formula and background.

Bottom

The bottom border is positioned between the formula and background.

Preview Field

Displays a preview of the current selection.

Default

Saves any changes as default settings for all new formulas. A security response will appear before saving these changes.

Example

Math objects are combined with text by removing all spacing both from the formula and the frame style surrounding it. Because spacing is set to zero, the spacing between text and formulas is achieved by simply adding a space character using the keyboard space bar.

“Let c_{ij} be a cost matrix and $x_{ij} \in \{0;1\}$ be a decision variable defined for $i=1,2,...,m$ origins and for $j=1,2,...,n$ destinations”.

Notes

When adding formulas into a LibreOffice document, there may be spacing between the formula and surrounding text, even if all formula borders are set to zero. This is because formulas are OLE objects embedded into a frame that, by default, adds 0.2 cm to the left and right of the formula.

Extra spacing can be removed using one of two methods. First method is to edit the object properties by right-clicking the formula and then going to **Properties > Wrap** to adjust the **Spacing** fields to the required values. The second method is to change the spacing options for the formula frame style or create a new style. For more information on changing the formula spacing, see Chapter 2, Formulas in Writer. For more information on how to edit styles in general, see the *Writer Guide*

Adjusting formula alignment

The formula alignment is used for determining how formula elements are located above one another and aligned horizontally relative to each formula element.

Notes

It is not possible to align formulas on a character and formula alignment cannot be used to align text elements. Text elements are always aligned to the left.

It is possible to align sections of a formula using the commands `alignl`, `alignc` and `alignr`. Using these commands, matrices and text elements in a formula can be aligned.

Current formula alignment

To change the alignment used for a formula in a LibreOffice document:

1) Click in the markup language in the Formula Editor.
2) Go to **Format > Alignment** on the Menu bar to open the Alignment dialog (Figure 15).
3) Select *Left*, *Centered*, or *Right* for horizontal alignment.
4) Select **OK** and check the result in the Formula Preview. If necessary, repeat steps 1 to 3 to correct the formula alignment.

Default formula alignment

To change the default alignment used for all formulas in a LibreOffice document:

1) Click in the markup language in the Formula Editor.
2) Go to **Format > Alignment** on the Menu bar to open the Alignment dialog (Figure 15).

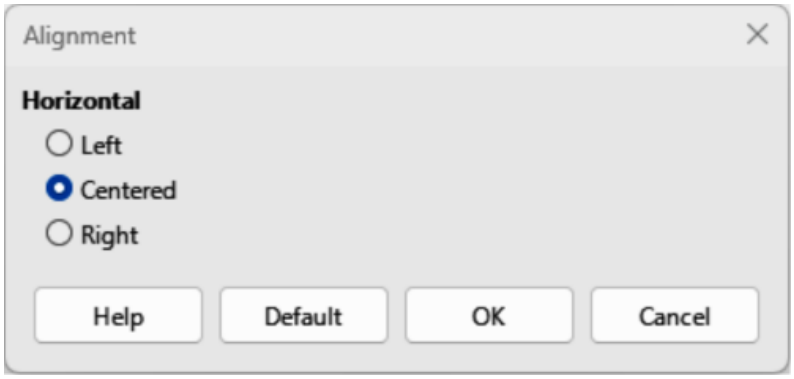


Figure 15: Alignment dialog

3) Select *Left*, *Centered*, or *Right* for horizontal alignment.
4) Select **Default** and confirm the changes to the formula alignment. Any new formulas created will use the new formula alignment.
5) Select **OK** and check the result in the Formula Preview. If necessary, repeat steps 1 to 4 to correct the formula alignment.

Note

Only formulas inserted after changing in formula alignment use the new default settings. For any formulas already inserted into a document before the default alignment was changed, the new default alignment is not applied. The formula alignment on each formula already inserted has to be changed manually.

Changing formula color

The color of formula characters can be changed using markup language. Use the command `color` followed by a color name, RGB value, or HTML hex value.

- The `color` command only works if the `color` command immediately follows the color name, RGB value, or hex value. For example, entering `color red 5 times 4` gives the result 5×4 . Only the number 5 is colored red.
- To change the color of the whole formula, the complete formula has to be placed inside brackets. For example, entering `color red {5 times 4}` gives the result 5×4 .

Named colors

For information on the named colors available in Math, see Appendix A. These colors are listed in the *Attributes* category of the **Elements** panel on the Sidebar (Figure 16).

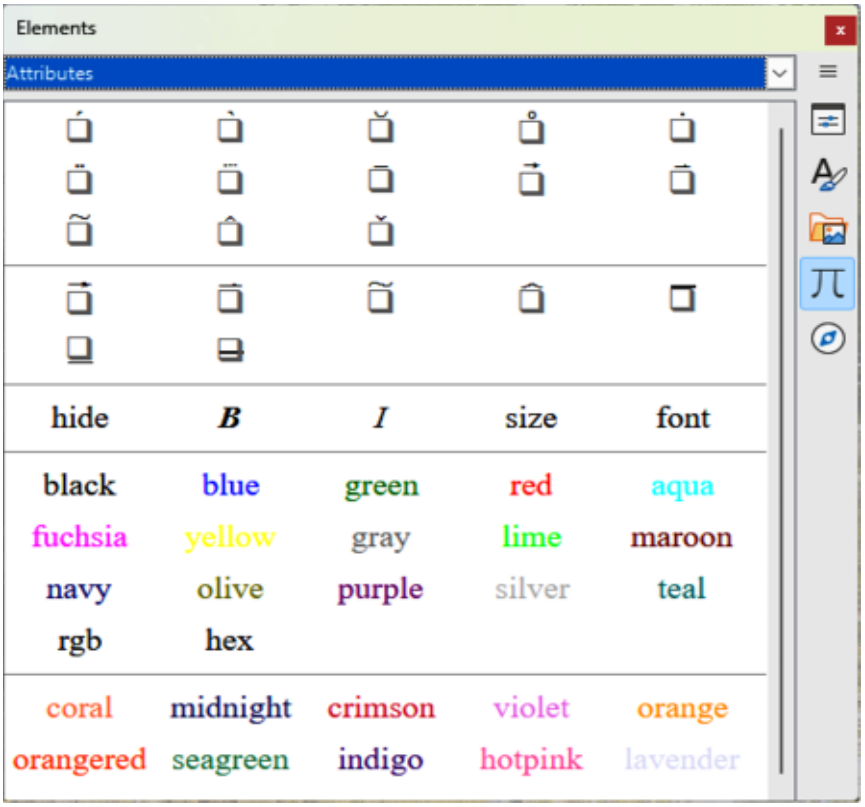


Figure 16: Elements panel in Sidebar — Attributes

RGB values

RGB (Red, Green and Blue) values for colors are from 0 to 255. To use RGB values, enter the command `color rgb R G B` followed by the RGB values of the required color.

Example

In this example, the formula element `decision variable` uses a color defined with the RGB values of 160, 82, 45.

`"Let " x_ij "` be a `" color rgb 160 82 45 "decision variable " "` in the problem under consideration"

Creates the following in a formula:

Let x_{ij} be a decision variable in the problem under consideration

Hex values

HTML colors can be used in a formula and these are defined using a hex number. Use the command `hex 000000`, where `00000` is the corresponding hex number for the color required. For example, using the command `color hex FF0000 decision` creates the output *decision*.

Background color

It is not possible to use a background color for formulas. The formula background color is, by default, the same color as the document or frame where the formula has been inserted. In Writer, object properties can be used to change the background color of a formula. For more information, see Chapter 2, Formulas in Writer.

Formula library

If formulas are going to be regularly inserted documents, a formula library can be created using the Formula Editor. Individual formulas are saved as separate files using the ODF format file type `ODF`, or the MathML format for file type `MML`. Math, Writer, Calc, Draw, or Impress can be used to create formulas and added to a formula library.

Note

MathML 2.0 stands for Mathematical Markup Language (MathML) Version 2.0. MathML is an XML format that describes mathematical notation and enables the use of mathematics in the World Wide Web. For more information on MathML format, visit the website at https://www.w3.org/TR/MathML2/overview.html.

Using Math

1) Create a computer folder to contain formula files and give the folder a memorable name, for example Formula Library.
2) Go to **File > New > Formula** on the Menu bar, or click on **Math Formula** in the Start Center to open LibreOffice Math and the Formula Editor. See “Formula files” on page 18 for more information.
3) Create the formula required.
4) Go to **File > Save As** on the main Menu bar, or use the keyboard shortcut *Ctrl*+*Shift*+*S* (macOS ⌘+*Shift*+*S*) to open the Save As dialog.
5) Navigate to the folder created for a formula library.
6) Type a memorable name for the formula in the file name or save as box.

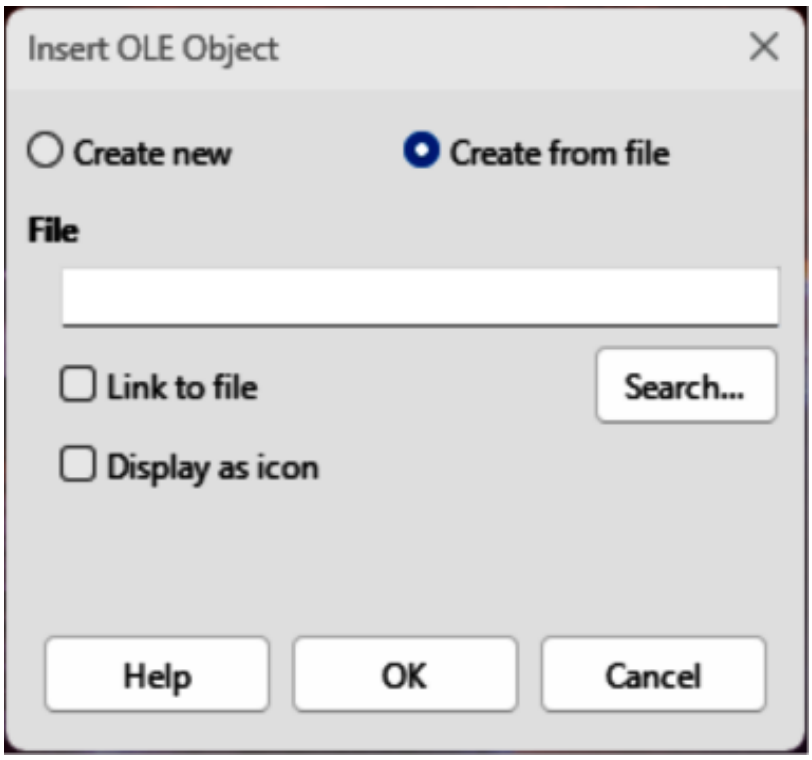


Figure 17: Insert OLE Object dialog — Create from file

7) Select in the File type drop-down list select ODF Formula (`.odf`) or MathML 2.0 (`.mml`) as the file type for the formula.

8) Click **Save** to save the formula and close the Save As dialog.

Using Writer, Calc, Draw, or Impress

1) Create a computer folder to contain formula files and give the folder a memorable name, for example Formula Library.

2) Open a document using Writer, Calc, Draw, or Impress.

3) Go to **Insert > Object > Formula Object** on the Menu bar to open the Formula Editor and create a formula. See “LibreOffice documents” on page 18 for more information.

4) Right-click on the formula object and select **Save Copy** as in the context menu to open a Save As dialog.

5) Navigate to the folder you have created for your formula library.

6) Type a memorable name for your formula in the File name box.

7) Select in the File type drop-down list either *ODF Formula (.odf)* or *MathML 2.0 (.mml)* as the file type for the formula.

8) Click **Save** to save the formula and close the Save As dialog.

Using a formula library

A formula in the formula library cannot be inserted into a document by dragging and dropping, or by using **Insert > File** on the Menu bar. A formula from the formula library must be inserted into a document as an OLE object.

1) Open a document using Writer, Calc, Draw, or Impress.

2) Go to **Insert > OLE Object > OLE Object** on the Menu bar to open the Insert OLE Object dialog (Figure 17).

3) Select the option *Create from file*.

4) Click **Search** to open the file browser and navigate to the formula library folder.

5) Select the file for formula required and click **Open** in the file browser, or double-click on the formula required.

6) If required, select the option *Link to file* or *Display as icon*.

7) Click **OK** to insert the formula as an OLE object in the document and close the Insert OLE Object dialog.

LibreOffice

Math Guide 26.2

Chapter 2,
Formulas in Writer



Introduction

When a formula is inserted into a document, Writer inserts the formula into a frame and treats the formula as an OLE object. For more information on creating and editing formulas, refer to Chapter 1, Creating and Editing Formulas.

- When a formula is selected for editing, the Writer Formula Editor opens, as shown by the example in Figure 18.
- Double-clicking on an inserted formula opens Math with the formula selected for editing in the Math Formula Editor.

Note

This chapter explains what options can changed for a formula within a Writer document. See the *Writer Guide* for information on how to change the settings for frame styles for OLE objects.

Automatic formula numbering

Automatic numbering of formulas to create formula cross references can only be carried out in Writer. The AutoText entry `fn` (formula numbered) is used to automatically add a numbered cross reference to formulas.

Numbering

1) Start a new line in a document at the required position for the formula.
2) Type `fn` and press the *F3* key to insert a two column table with no borders. The left column contains a sample formula and the right column contains the cross reference number, as shown by the following example.
3) Delete the sample formula and insert the required formula into the left column.

$$E = mc^2 \tag{1}$$

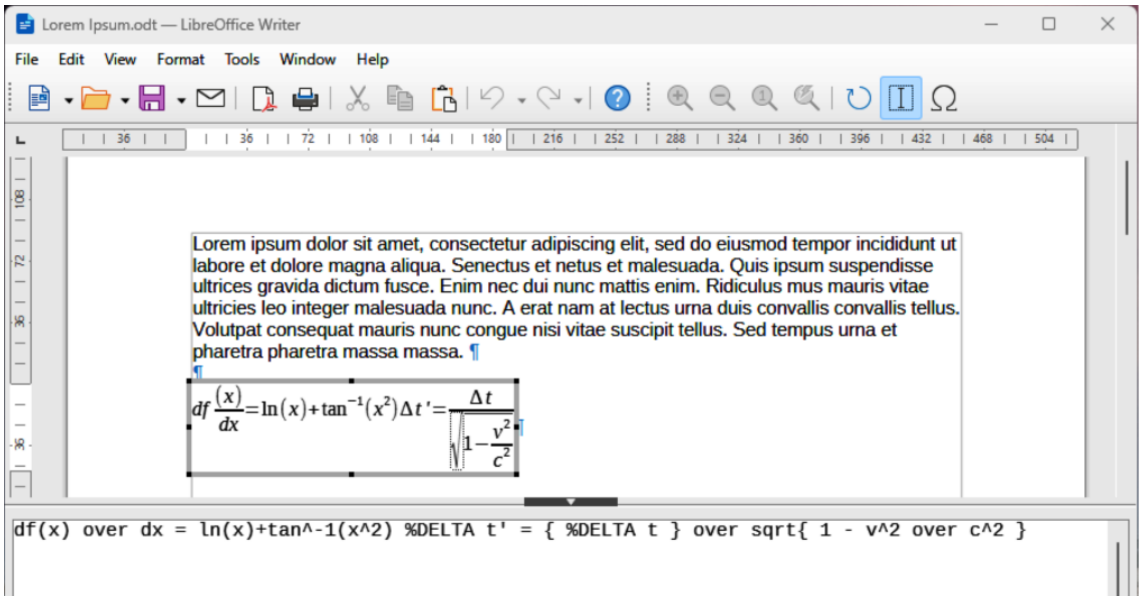


Figure 18: Example formula selected for editing in Writer

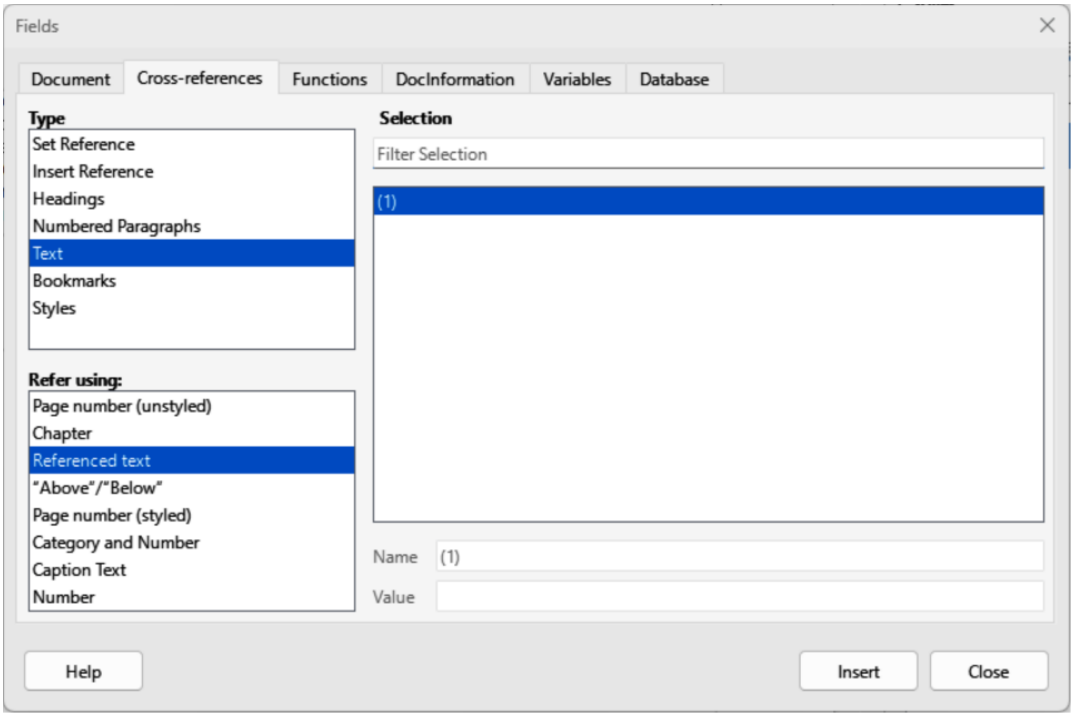


Figure 19: Fields dialog — Cross-references — Referenced text

Formula cross referencing

1) Click in the document at the position where a cross reference for a formula is required.
2) Go to **Insert > Cross-reference** on the Menu bar to open the Fields dialog (Figure 19).
3) Click on **Cross-references**, then select *Text* in **Type**.
4) In **Selection**, select the formula number required for the cross reference.
5) In **Refer using**, select *Referenced text*, then click on **Insert** to insert the cross reference for the selected formula.
6) Click on **Close** to close the Fields dialog.

Note

To use square parentheses instead of curved parentheses, or require the cross reference number to be separated from the formula by tabs instead of using a table, see the *Writer Guide* for more information.

Anchoring and aligning formulas

Anchoring position

A formula is an object when inserted into a Writer document. The default anchoring for a formula is *As character* when inserted into a document. To change the formula anchoring in a document:

1) Right-click on the selected formula and select **Anchor** from the context menu.
2) Select an anchoring option from the context menu. The anchoring options available are *To paragraph*, *To character*, or *As character*.
3) Alternatively, right-click on the selected formula and select **Properties** from the context menu, or go to **Format > Frame and Object > Properties** on the Menu bar to open the OLE Object dialog (Figure 20).

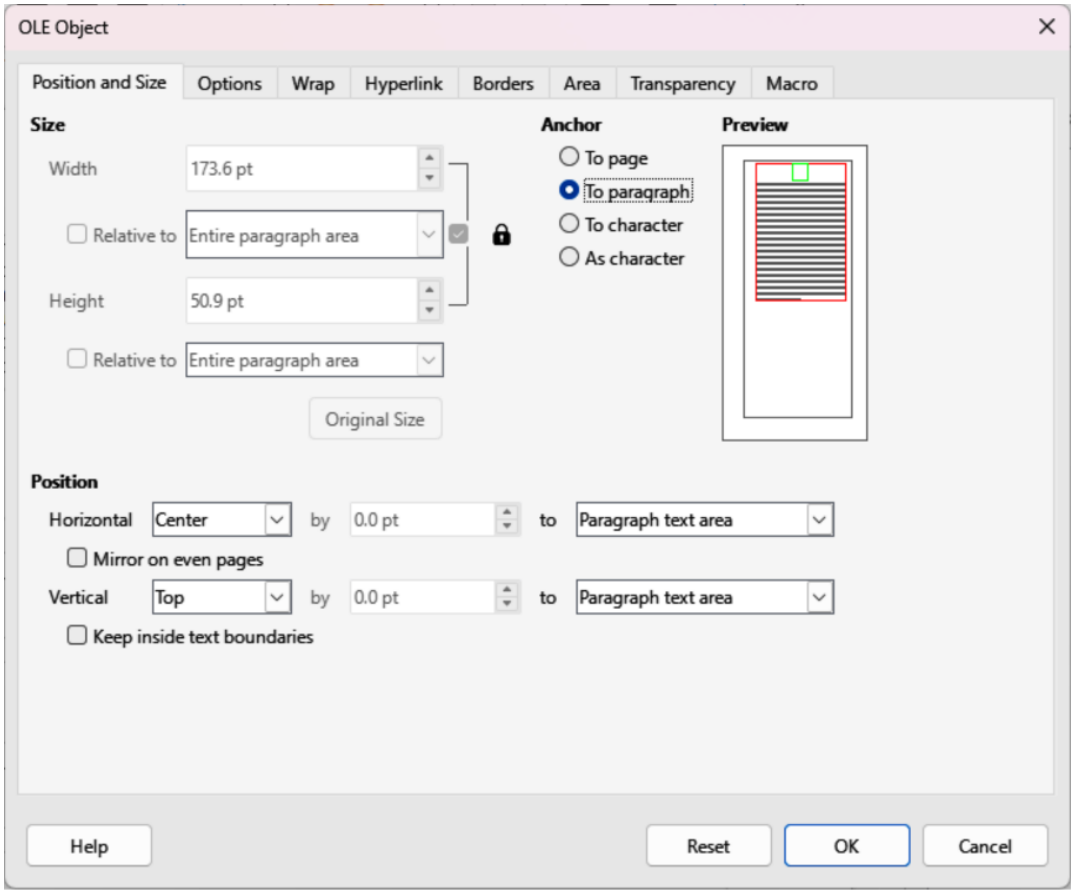


Figure 20: OLE Object dialog — Position and Size page

4) Make sure **Position and Size** is selected, then select a new anchoring position from the options available in **Anchor**. The anchoring options available are *To page*, *To paragraph*, *To character*, or *As character*.
5) Click **OK** to save the changes and close the OLE Object dialog.

Vertical alignment

The normal default setting for vertical alignment for formulas is to use the text base line as a reference. This default setting can be changed by modifying the frame style for formulas, see the *Writer Guide* for more information.

1) Make sure the To *paragraph* or *To character* option for formula anchoring is selected to change vertical formula alignment. Vertical alignment is NOT available for the *As character* anchoring option.
2) Right-click on the selected formula and select **Align Objects** from the context menu.
3) Select a vertical alignment option from the context menu. The options available are *Align Top to Anchor*, *Align Middle to Anchor*, and *Align Bottom to Anchor*.
4) Alternatively, right-click on the selected formula and select **Properties** from the context menu, or go to **Format > Frame and Object > Properties** on the Menu bar to open the Object dialog (Figure 20).
5) Make sure **Position and Size** is selected, then select a new vertical alignment from the options available in **Position** *Vertical*. The vertical alignment options available are *Center*, *Top*, *Bottom*, or *From bottom*.
6) If the *From bottom* option is selected for vertical alignment, enter a plus or minus value for vertical alignment.

7) Select the type of text alignment from the drop-down list in **Position** *Vertical*. The text alignment options available are *Base line*, *Character*, and *Row*.

8) Click **OK** to save the changes and close the OLE Object dialog.

Horizontal alignment

The normal default setting for horizontal alignment for formulas is to use the text base line as a reference. This default setting can be changed by modifying the frame style for formulas, see the *Writer Guide* for more information.

1) Right-click on the selected formula and select **Align Objects** from the context menu.

2) Select a horizontal alignment option from the context menu. The options available are *Left*, *Centered*, and *Right*.

3) Alternatively, right-click on the selected formula and select **Properties** from the context menu, or go to **Format > Frame and Object > Properties** on the Menu bar to open the Object dialog (Figure 20).

4) Make sure **Position and Size** is selected, then select a new horizontal alignment from the options available in **Position** *Horizontal*. The horizontal alignment options available are *Left*, *Right*, *Center*, or *From left*.

5) If the *From left* option is selected for horizontal alignment, enter a plus or minus value for horizontal alignment.

6) Select the type of text alignment from the drop-down list in **Position** *Horizontal*. The text alignment options available are *Entire paragraph area*, *Left of paragraph text area*, *Right of paragraph text area*, *Left of page text area*, *Right of page text area, Entire page, Page text area*, and *Character.*

7) Click **OK** to save the changes and close the OLE Object dialog.

Object wrap and spacing

A formula, when inserted into a document, has text wrap and spacing on each side of the formula. The default settings used for wrap and spacing is set within the frame style for formulas. This default settings can be changed by modifying the formula frame style, see the *Writer Guide* for more information.

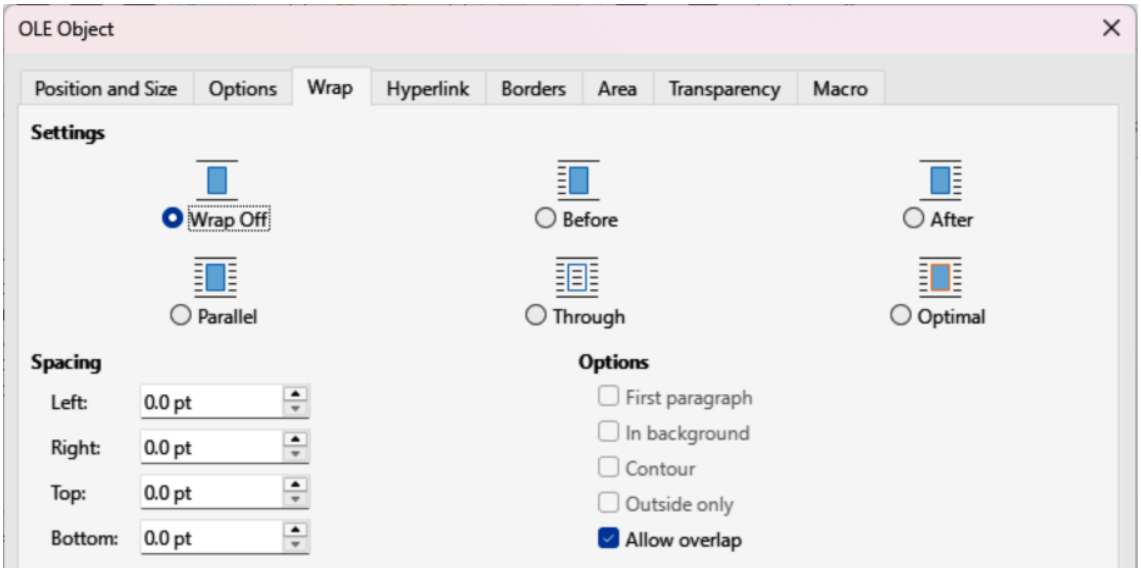


Figure 21: OLE Object dialog — Wrap page

Object wrap

To individually adjust the wrap for a formula positioned in a Writer document:

1) Make sure the To *paragraph* or *To character* option for formula anchoring is selected to change object wrap. Object wrap is NOT available for the *As character* anchoring option.
2) Right-click on the selected formula and select **Wrap** from the context menu.
3) Select a wrap option from the context menu. The options available are *None*, *Parallel*, *Optimal*, *Before*, *After*, *Through In Background* and *Edit Contour*.
4) Right-click on the selected formula and select **Properties** in the context menu, or go to **Format > Frame and Object > Properties** to open the Object dialog.
5) Click on **Wrap** to open the Wrap page in the Object dialog (Figure 21).
6) In S**ettings**, select the type of wrap required for the formula. The options available are *Wrap Off*, *Before*, *After*, *Parallel*, *Through* and *Optimal*.
7) Click **OK** to save the changes and close the Object dialog.

Object spacing

To individually adjust the spacing for a formula positioned in a Writer document:

1) Right-click on the selected formula and select **Align Objects** from the context menu.
2) Select a horizontal alignment option from the context menu. The options available are *Left*, *Centered*, and *Right*.
3) Right-click on the selected formula and select **Properties** in the context menu, or go to **Format > Frame and Object > Properties** to open the Object dialog.
4) Click on **Wrap** to open the Wrap page in the Object dialog (Figure 21).
5) In **Spacing**, enter the spacing value for *Left*, *Right*, *Top*, and *Bottom* spacing.
6) Click **OK** to save the changes and close the Object dialog.

Text mode

When large formulas are placed within a line of text, the formula elements can be higher than the text height. To make large formulas easier to read, it is recommended to always insert large formulas into a separate paragraph to separate a formula from the text.

$$\sum_{i=2}^{5} i^2$$

If it is necessary to place a large formula within a line of text, double-click on the formula to open the Formula Editor and then go to **Format > Text Mode** on the Menu bar. The Formula Editor reduces the formula size to match text height, as shown in the following example.

The same formula embedded into a line of text using text mode format $\sum_{i=2}^{5} i^2$:

Background and borders

The default setting for background (area fill) and borders for formulas is set by the Formula frame style. For more information on changing the background and borders for Formula frame style, see the *Writer Guide*. If required, individual formulas in a document can use different backgrounds and borders.

Note

The size of the frame that a formula is placed in when inserted into a document cannot be changed. The frame size for a formula depends on the setting of the formula font size. See Chapter 1, Creating and Editing Formulas for more information.

The following example is a formula using background color and borders. Note that the background and border properties are part of the Math OLE object and cannot be defined using markup language.

$$df\frac{(x)}{dx}=\ln(x)+\tan^{-1}(x^2)\Delta t'=\frac{\Delta t}{\sqrt{1-\frac{v^2}{c^2}}}$$

Backgrounds

1) Select a formula in a document to change the background.
2) Right-click on the formula and select **Properties** from the context menu, or go to **Format > Frame and Object > Properties** on the Menu bar to open the OLE Object dialog.
3) Click on **Area**, then select the type of fill required for the formula from the available options (Figure 22). The options available are *None*, *Color*, *Gradient*, *Image*, *Pattern*, and *Hatch*.

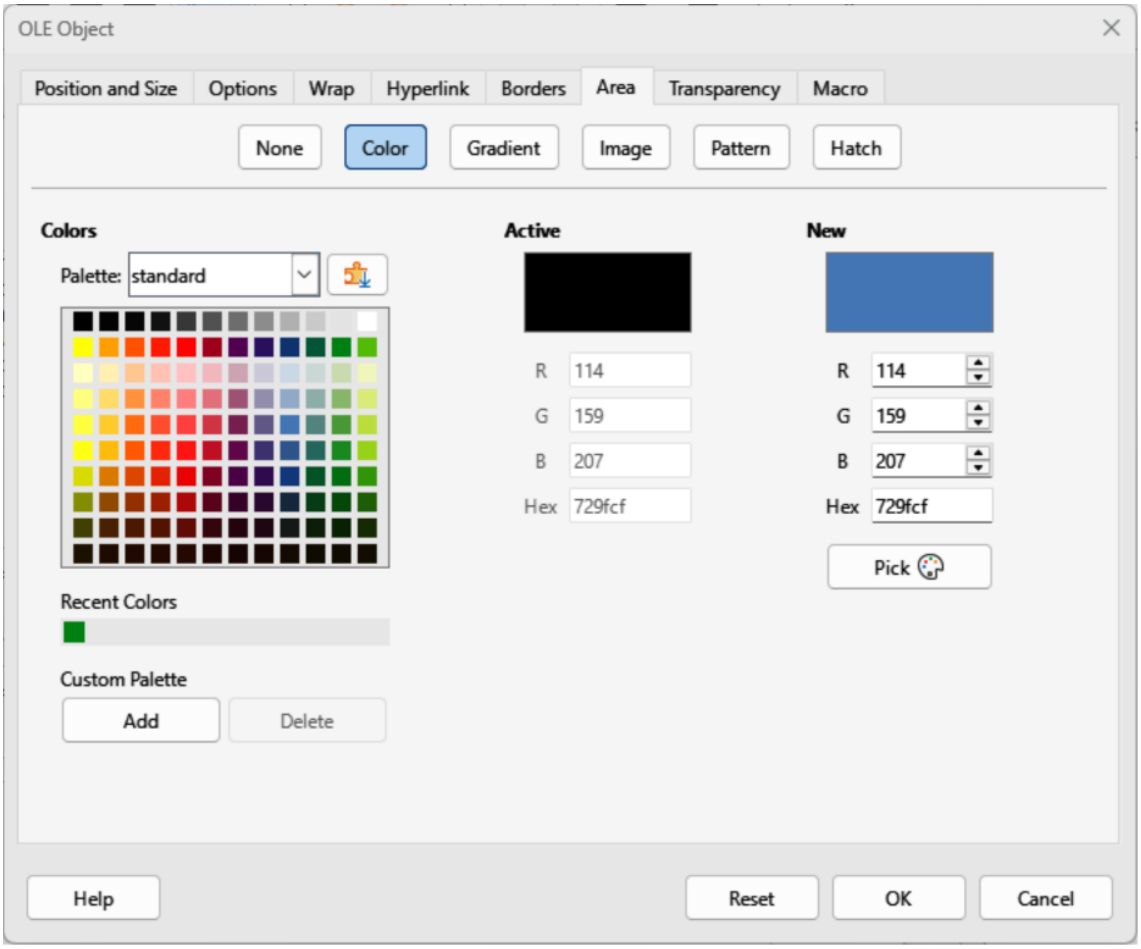


Figure 22: OLE Object dialog — Area page

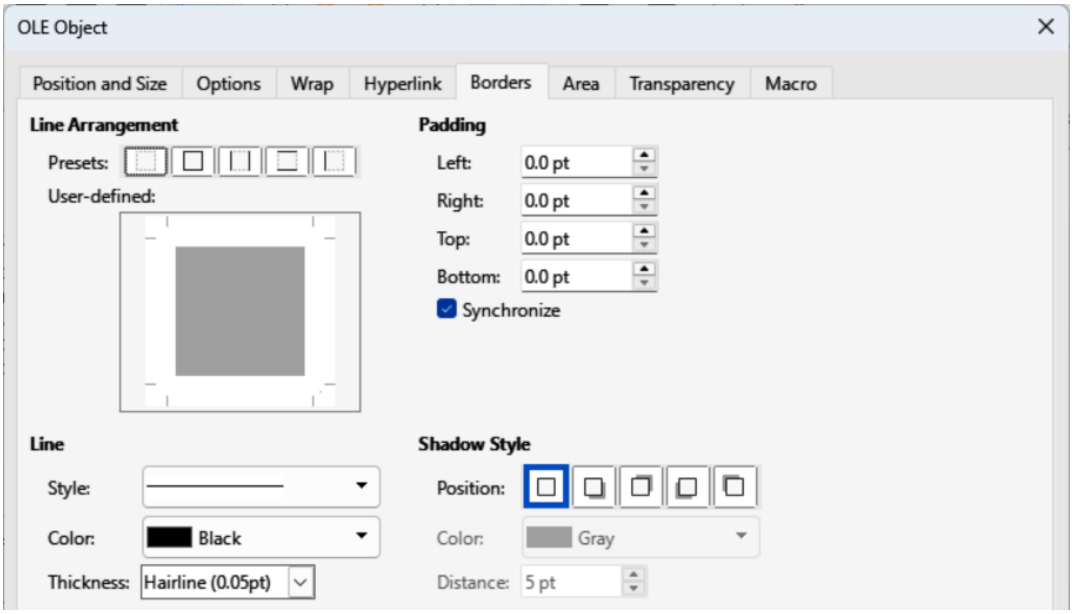


Figure 23: OLE Object dialog — Borders page

4) After selecting the background type, select the properties for the formula background. The area properties change depending on the type of area fill selected.

5) Click **OK** to save the changes and close the OLE Object dialog.

Borders

1) Select a formula in a document to change the borders.

2) Right-click on the formula and select **Properties** from the context menu, or go to **Format > Frame and Object > Properties** on the Menu bar to open the OLE Object dialog.

3) Click on **Borders**, then select the of border properties required for the formula from the available options (Figure 23). The options available are *Line Arrangement*, *Line*, *Padding*, and *Shadow Type*.

4) Click **OK** to save the changes and close the OLE Object dialog.

Quick formula insertion

If the markup language for a formula is available, the formula can be quickly inserted into a document without opening the Formula Editor:

```
df(x) over dx = ln(x)+tan^-1(x^2) %DELTA t' = { %DELTA t } over sqrt{ 1 - v^2 over c^2 }
```

$$df\frac{(x)}{dx}=\ln(x)+\tan^{-1}(x^2)\Delta t'=\frac{\Delta t}{\sqrt{1-\frac{v^2}{c^2}}}$$

1) Copy the formula markup language required, then paste into the document at the required position, as shown by the above example.

2) Select all of the formula markup language.

3) Go to **Insert > OLE Object** on the Menu bar and select **Formula Object** to create a formula using the selected markup language. The formula is inserted replacing formula markup language.

LibreOffice

Math Guide 26.2

Chapter 3,
Formulas In Calc, Draw, and Impress



Introduction

When a formula is inserted into a Calc, Draw, or Impress document, the formula is inserted as an OLE object without any background or borders. For more information on creating and editing formulas, please refer to Chapter 1, Creating and Editing Formulas. Each formula is inserted into a spreadsheet, drawing, or slide as follows:

- In Calc, formulas are inserted into a selected cell in a spreadsheet with no style assigned to the formula. For more information, see the *Calc Guide*.
- In Draw and Impress, formulas are inserted in a central position on a drawing or slide. By default, formulas are assigned the drawing object style *Object with no fill and no line*. For more information on how to modify or assign drawing object styles, see the *Draw Guide* or *Impress Guide*.

Anchoring formulas

Calc

By default, a formula is anchored into a Calc spreadsheet using the option *To Page.* Change the default formula anchoring in a Calc spreadsheet as follows:

1) Select the formula in the spreadsheet.
2) Right-click on the formula and select **Anchor** from the context menu. Anchoring options available in the context menu are *To Cell*, *To Cell (resize with cell)*, and *To Page*.
3) Alternatively, go to **Format > Anchor** on the Menu bar and select an anchoring option from *To Cell*, *To Cell (resize with cell)*, and *To Page*.

Tip

If a formula is inserted into a Calc spreadsheet and appears out of scale, right click the formula and select **Original Size** from the context menu.

Draw and Impress

When a formula is inserted into a drawing or slide, it is inserted as a floating OLE object in the center of a drawing or slide. The inserted formula is not anchored and can be moved to any particular position in a drawing or slide. See the *Draw Guide* or *Impress Guide* for information on moving objects in a drawing or slide.

Formula properties

Formulas in Calc, Draw, and Impress can be modified just like any other object that has been placed in a spreadsheet, drawing, or presentation. However, formula size and changing text format in a formula cannot be carried out. For more information on how to change properties, see the *Calc Guide*, *Draw Guide*, or *Impress Guide*. For more information on formula size and formatting formula text, see Chapter 1, Creating and Editing Formulas.

- For formula backgrounds, use the options in the Area dialog pages.
- For formula borders, use the options in the Line dialog.

Note

Formula borders are separate from cell borders in a Calc spreadsheet.

- To accurately re-position a formula, use the options in the Position and Size dialog.
- In Draw and Impress, arrange, align, group, flip, convert, break, combine, and edit points can be used for formula objects.
- Text attributes of a formula cannot be changed. The formula text is set when a formula is created using the Formula Editor.
- Formula size is set by the formula font size when a formula is created. Formula size is protected in the Position and Size dialog, but this can be deselected if required. However, it is not recommended as resizing a formula using the Position and Size dialog. This may distort a formula making it difficult to read.

Formulas in charts

A Calc chart is an OLE object created from a Calc spreadsheet. However, the Formula Editor cannot be used to create and insert a formula directly into a chart. A formula has to be created, then copied and pasted into a chart:

1) Create the chart using a Calc spreadsheet. For more information, see the *Calc Guide*.
2) Click in a spreadsheet cell to deselect a chart.
3) Insert a formula by clicking on **Insert > OLE Object > Formula Object** on the Menu bar to open the Formula Editor in Calc.
4) Create the required formula in the Formula Editor.
5) Select the formula and copy the formula.
6) Double-click the chart object to open editing mode, then paste the formula into the chart.
7) Move the formula to the required position inside the chart.

Note

To edit a formula, the above procedure has to be repeated to create a new formula, or edit an existing formula. The new or edited formula is then pasted into the chart.

Chemical formulas

The primary purpose of Math is to create mathematical formulas. However, Math can create chemical formulas. However, chemical symbols are normally written in uppercase using non-italic characters. The following table shows examples of chemical formulas.

Table 7: Example chemical formulas

Construction	***Example***	***Markup Language***
Molecules	H_2SO_4	`H_2 SO_4`
Isotopes	${}_{92}^{238}U$	`U lsub 92 lsup 238`
Ions	SO_4^{2-} or SO_4^{2-}	`SO_4^{2-{}}` *or* `SO_4^{2"-"}`

Creating chemical formulas using Math, the font used for variables is changed to a non-italic font. For more information on formula fonts, see Chapter 1, Creating and Editing Formulas.

For reversible reactions in chemical formulas, there is no symbol for a double arrow in Math. If there is a font available with symbols for use in chemical formulas, then add these symbols to the Catalog. See Chapter 4, Customization for more information on how to add symbols.

LibreOffice

Math Guide 26.2

Chapter 4,
Customization



Introduction

This chapter explains how to customize LibreOffice Math when creating formulas for use in LibreOffice documents. For more information on customizing LibreOffice refer to the *Getting Started Guide*.

Formula Editor and Elements panel

Formula Editor

Floating window

If required, the Formula Editor can become a floating window allowing the Formula Editor to be positioned at a convenient location when creating formulas.

1) Position the cursor on the frame of the Formula Editor, NOT in the Formula Editor.
2) Press and hold *Ctrl* key (macOS ⌘) and double-click on the frame. This turns the Formula Editor into a floating window labelled **Commands** (Figure 24).

To return the Formula Editor back to its position in the Math window:

1) Position the cursor on the frame of the **Commands** floating window, NOT on the title bar at the top of the window.
2) Press and hold *Ctrl* key (macOS ⌘) and double-click. The floating window closes and the Formula Editor appears at its previous position in the Math window.

Window static position

The Formula Editor can be positioned at the top or bottom of the Math window. The default position is the bottom of the Math window.

1) Position the cursor on the frame of the Formula Editor, NOT in the Formula Editor.
2) Press and hold *Alt* key (macOS ⌥) and drag the Formula Editor to a new position at the top or bottom of the Math window.

Elements panel

The Sidebar **Elements** panel can be used as a floating dialog allowing the **Elements** panel to be positioned at a convenient location when creating formulas.

1) Position the cursor on the title bar of the **Elements** panel on the Sidebar.
2) Press and hold *Alt* key (macOS ⌥), then double-click to create a floating **Elements** panel (Figure 25).

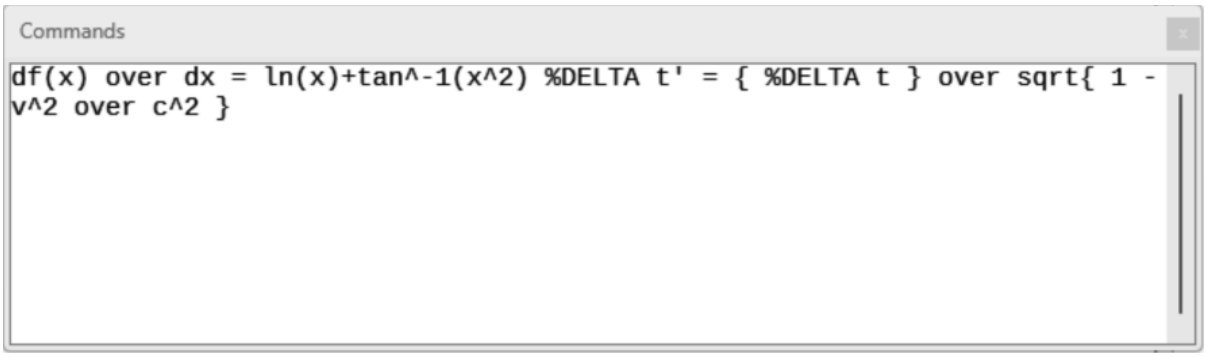


Figure 24: Example of floating Formula Editor window

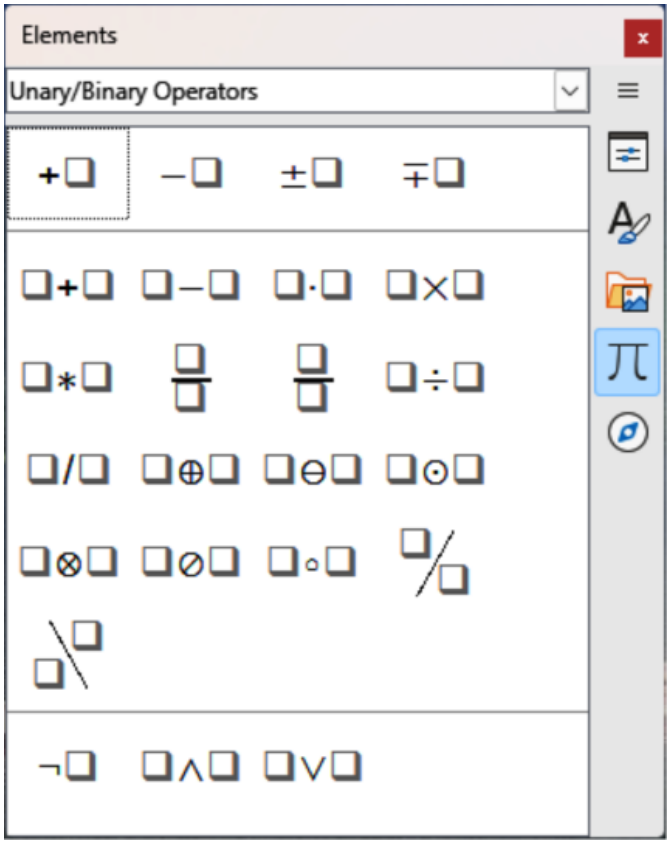


Figure 25: Example of a floating Elements panel

3) To move the **Elements** panel back to its position as a Sidebar, press and hold `Alt` key (macOS ⌥), then drag the **Elements** panel back to its Sidebar position.

Adding keyboard shortcuts

Keyboard shortcuts can be added to LibreOffice when creating documents. Following is an example of how to add a keyboard shortcut for inserting a formula into a LibreOffice document. For more information on creating keyboard shortcuts, see the *Getting Started Guide*.

Note

When creating new keyboard shortcuts, make sure that the keyboard shortcut is NOT already in use by LibreOffice or the computer system.

Example keyboard shortcut

1) Go to **Tools > Customize** on the Menu bar to open the Customize dialog, then click on **Keyboard** to access the options available for creating keyboard shortcuts (Figure 26).
2) Select the type of the new keyboard shortcut, as follows:
   - Select **LibreOffice** to create a new keyboard shortcut for all components of LibreOffice.
   - Select **Math** to create a new keyboard shortcut for Math.
3) In the **Category**, select *Insert*.
4) In the **Function**, select *Import Formula*.

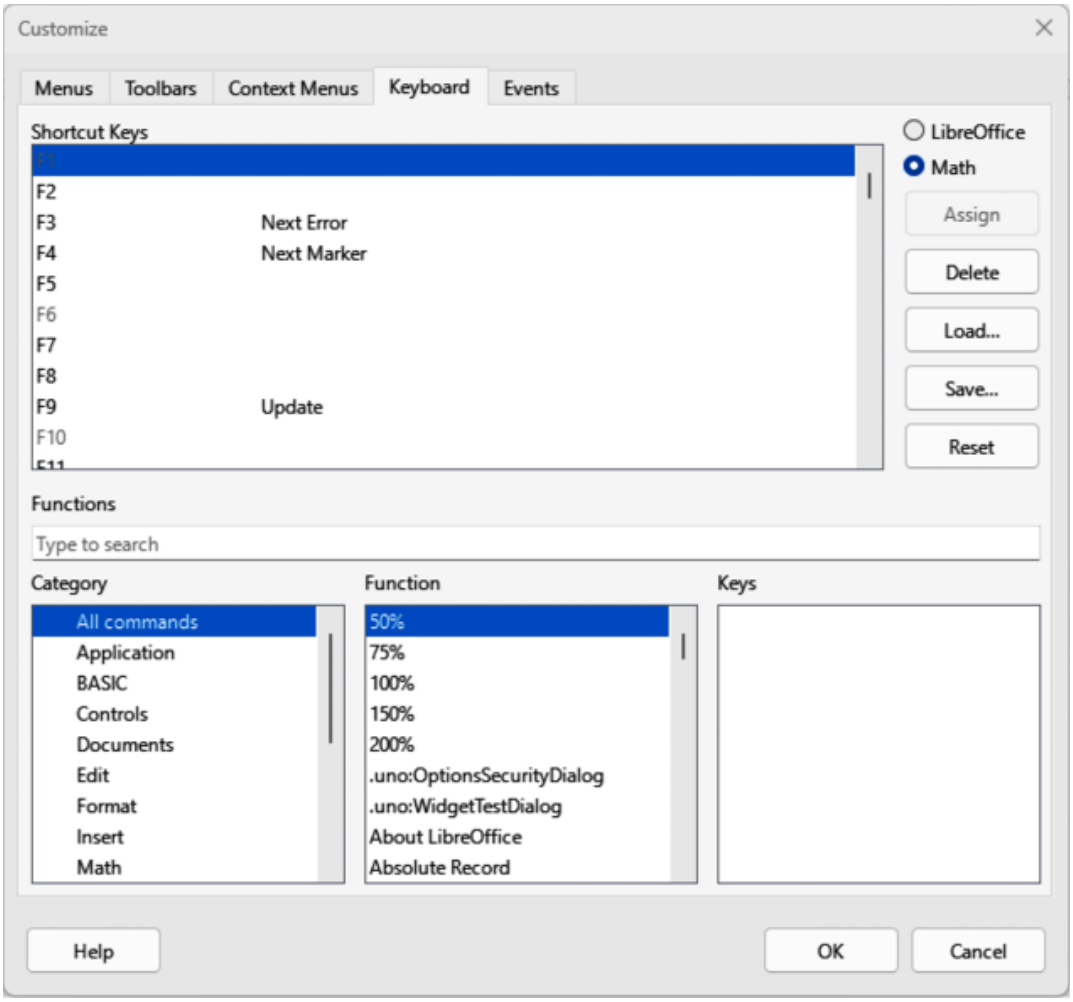


Figure 26: Customize dialog — Keyboard page

5) In **Keys**, select the key or keyboard combination required for the new keyboard shortcut.
6) Select **Modify** and the new keyboard shortcut appears in **Keys**.
7) If required, continue to add keyboard shortcuts using Steps 1) to 6).
8) Click **OK** to save the keyboard shortcuts and close the Customize dialog.

Customize dialog options for keyboard shortcuts

LibreOffice

Displays shortcut keys that are common to all LibreOffice modules.

Math

Displays the keyboard shortcuts available in Math.

Assign

Assigns the keyboard shortcut to an action.

Delete

Deletes the selected element or elements without requiring confirmation.

Load

Replaces a shortcut key configuration with one that has been saved.

Save

Saves the current shortcut key configuration, so that you can load it later.

Reset

Resets any modified keyboard shortcut to the default setting.

Shortcut Keys

Lists the available keyboard shortcuts and any associated commands. To assign or modify a keyboard shortcut select a listed keyboard shortcut has to be selected first.

Functions

Type a function name in the text field to search for the function in the keyboard shortcuts.

Category

Lists the available categories for functions.

Function

Lists the available function categories for keyboard shortcuts.

Keys

Displays the keyboard shortcut keys that is assigned to a selected function.

Catalog customization

If a symbol is regularly used and is not available in Math, it can be added to the Symbols dialog (Figure 27) using the Edit Symbols dialog (Figure 28). Symbols can be modified, have names assigned, or redefined.

Adding symbols

1) Go to **Tools > Symbols** on the Menu bar or click on **Symbols** on the Tools toolbar to open the Symbols dialog (Figure 27).
2) In **Symbol set** select a font to use for the new symbol from the available options in the drop-down list.
3) Click on **Edit** to open Edit Symbols dialog (Figure 28).
4) Select the symbol character required and it appears in the preview box. If necessary, scroll down in the preview box to locate the required symbol.
5) In **Symbol**, enter a memorable name for the required symbol.
6) In **Symbol set**, select a symbol set in the drop-down for the selected symbol, or create a new name for the selected symbol.

Notes

When a new symbol is added to the catalog, a percentage sign (%) followed by the name of the symbol is typed into the markup language in the Formula Editor and the symbol appears in the formula. Symbol names are case sensitive, for example, `%prime` is a different symbol to `%Prime`.

There are numerous free fonts available that contain several symbols if the required symbol is not available on the computer. For example, the STIX font was developed for use in mathematical and technical texts. Also, DejaVu and Lucida fonts have a wide range of symbols.

By default, only user defined symbols that are used in the document are stored with the document. It is useful to embed all user defined symbols so that when a document is transferred to another computer it is available for editing. Go to **Tools > Options > LibreOffice Math > Settings** (macOS **LibreOffice > Preferences > LibreOffice Math > Settings**) on the Menu bar and uncheck *Embed only used symbols (smaller file size)* in **Miscellaneous Options**.

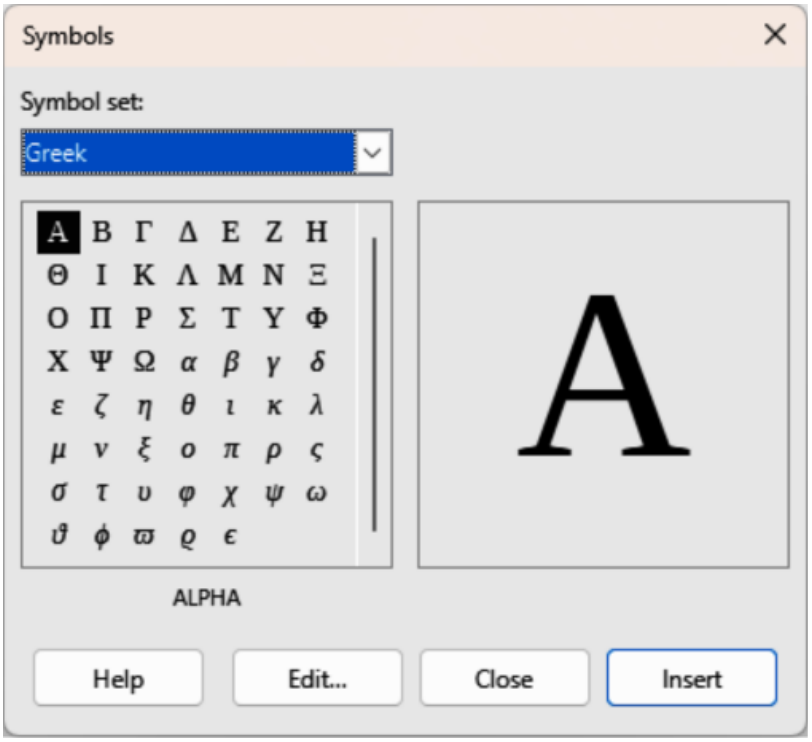


Figure 27: Symbols dialog — Greek page

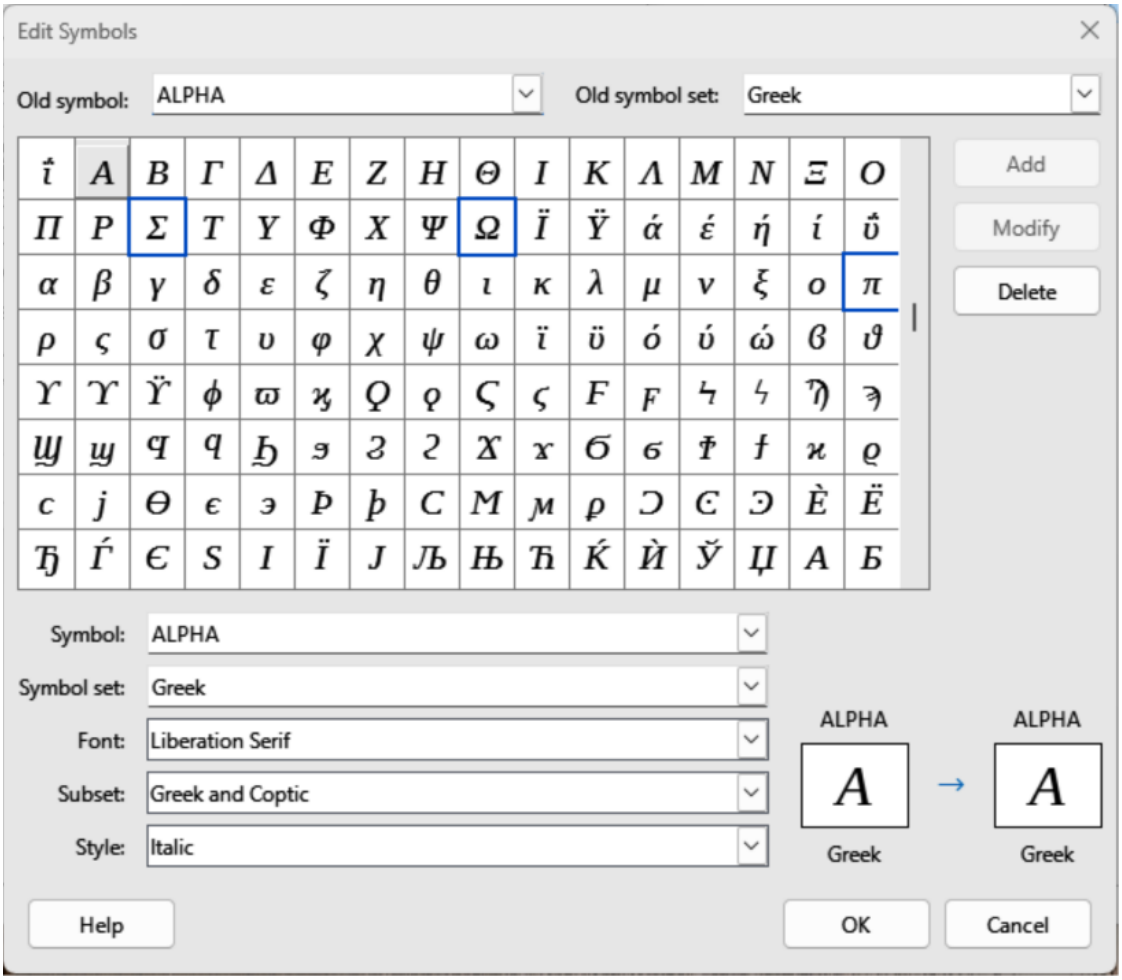


Figure 28: Edit Symbols dialog

7) In **Font** and if required, select a font for the selected symbol.
8) In **Subset**, select a subset from the available options for the selected symbol.
9) In **Style** and if required, select a font style from the available options for the selected symbol.
10) Click on **Add**, to add the symbol to Math.
11) If required, click on **Modify** after making any changes to an existing symbol.
12) If required, click on **Delete** if an existing symbol is no longer required.
13) Click on **OK** to close the Edit Symbols dialog. The new symbol and, if created, new symbol set are now available for use.

Editing symbols

Changing symbol names

1) In the Edit Symbols dialog (Figure 28), select the symbol name that is to be changed in the **Old symbol** drop-down list. The symbol appears in the left preview box at the bottom of the dialog.
2) Type a new name for the symbol in the **Symbol** text box, or select a new name from the **Symbol** drop-down list. The new symbol name appears in the right preview pane.
3) Click **Modify** to change the symbol name, click **OK** to close the Edit Symbols dialog.

Moving symbols

1) In **Old symbol set**, select the symbol set from the options available in the drop-down list in the Edit Symbols dialog (Figure 28).
2) Select a symbol from options available in the **Old symbol** drop-down list. In the left preview box of the dialog, the name, symbol, and symbol set are displayed.
3) In **Symbol**, select a symbol name from the drop-down list.
4) In **Symbol set**, select the required symbol set from the drop-down list. In the right preview box at the bottom of the dialog, the name, symbol, and symbol set are displayed.
5) If required, select an option from the drop-down lists in **Font**, **Subset**, and **Style**.
6) Click on **Modify** and the symbol is moved to the new symbol set.
7) Click on **OK** to close the Edit Symbols dialog.

Deleting symbols

1) In **Old symbol set**, select the symbol set from the options available in the drop-down list in the Edit Symbols dialog (Figure 28).
1) In **Old symbol**, select the symbol for deletion from the drop-down list. The selected symbol appears in the left preview pane at the bottom of the Edit Symbols dialog.
2) Click on **Delete** and the symbol is deleted from the symbol set without any confirmation.
3) Click on **OK** to close the Edit Symbols dialog.

Note

To delete a symbol set all of the symbols in the selected set have to be deleted. The last symbol is deleted, the symbol set is also deleted.

Symbol options

Old symbol

Select the name of a symbol from the drop-down list. The symbol, symbol name, and symbol set are displayed in the left preview box at the bottom of the dialog.

Old symbol set

Displays the name of the current symbol set. A different symbol set can be selected from the options in the drop-down list.

Symbol

Displays the name of the selected symbol set. Select a name from the options available in the drop-down list or type a name for a newly added symbol when modifying or deleting symbols.

Symbol set

The drop-down list displays the names of the available symbol sets.

Font

Displays the name of the selected font for symbols. When changing fonts for symbols, select from the options in the drop-down list.

Subset

Displays the name of the selected subset for symbol fonts. When changing font subset for symbols, select from the options in the drop-down list.

Style

Displays the name of the selected style for symbol fonts. When changing font style for symbols, select from the options in the drop-down list.

Add

Click on **Add** to save and add the symbol displayed in the right preview box to the selected symbol set. The Symbol or Symbol name displayed in the Symbol list must be unique before the **Add** option becomes available. Symbol and Symbol Set names can only be used once.

Modify

Click on **Modify** to replace the symbol displayed in the left preview box with the new name after carrying out any modifications to a symbol.

Delete

Click on **Delete** to delete the symbol displayed in the left preview box from the selected symbol set. There is no confirmation of deletion. Deleting the last symbol in a symbol set also deletes the symbol set.

Formula spacing

The grave accent (`) inserts an additional small space and the tilde (~) inserts an additional large space into formulas. However, in a LibreOffice basic installation, these symbols are ignored when they occur at the end of a formula. When working with running text in a formula, it may be necessary to also include spacing at the end of formulas. This customization is only required when working with a Math document and is not required when inserting a formula into another LibreOffice module.

To add spacing at the end of formula in Math, go to **Tools > Options > LibreOffice Math > Settings** (macOS **LibreOffice > Preferences > LibreOffice Math > Settings**) on the Menu bar and uncheck *Ignore ~ and ` at the end of the line* in **Miscellaneous Options**.

Scaling of code

By default, the code in the Formula Editor input window is scaled to 100%. To change the scaling factor go to **Tools > Options > LibreOffice Math > Settings** (macOS **LibreOffice > Preferences > LibreOffice Math > Settings**) and change the percentage in the *Scaling code input window* in **Miscellaneous Options**.

For example, to change the scaling to 150%, the code in the input window appears larger. This setting affects the Formula Editor in all LibreOffice modules. However, the LibreOffice Math options can only be accessed when using LibreOffice Math. It is not available for editing formulas in other LibreOffice modules.

Extensions

Creating formulas frequently in documents, LibreOffice can be customized by adding extensions that are designed to help in the creation of formulas. Extensions are installed using the Extension Manager. For more information on how to install extensions, see the *Getting Started Guide*.

A commonly used extension is **Formatting of All Math Formulas**. This extension allows formatting of formulas in Writer, Calc, Draw, or Impress documents. Using this extension the font names and sizes of all formulas used in LibreOffice documents. For more information on this extension, go to https://extensions.libreoffice.org/en/extensions/show/formatting-of-all-math-formulas.

LibreOffice

Math Guide 26.2

Chapter 5,
Exporting and Importing



MathML format

In addition to exporting documents in PDF format, LibreOffice export formulas in the MathML format. This allows formulas to be inserted into documents that were created using other software, for example, Microsoft Office or an internet browser.

> **Note**
>
> Some internet browsers do not fully support the MathML format and a formula may not display correctly when imported.

When working on a formula in Math, go to **File > Save as** on the Menu bar, or use the keyboard combination *Ctrl+Shift+S* (macOS ⌘+*Shift+S*) to open the Save as dialog. Select *MathML 2.0 (*.mml)* from the list of available file formats in **File type** to save the formula as MathML.

When working in another LibreOffice module, right-click on the formula object and select **Save Copy as** as from the context menu to open the Save as dialog. Select *MathML 2.0 (*.mml)* from the list of available file formats in **File type** to save the formula as MathML.

In Math, formulas can be imported as in MathML format. Use **Tools > Import MathML from Clipboard** on the Menu bar.

Microsoft file formats

To control how formulas in Microsoft format are imported and exported using LibreOffice, go to **Tools > Options > Load/Save > Microsoft Office** (macOS **LibreOffice > Preferences > Load/Save > Microsoft Office**) on the Menu bar, then select or deselect one or both options for *MathType to LibreOffice Math or reverse*.

- *[L]: Load and convert the object*
- *[S]: Convert and save the object*

[L]: Load and convert the object

Select this option if Microsoft OLE objects are to be converted into the specified LibreOffice OLE objects when a Microsoft document is opened in LibreOffice. For formulas, any embedded MathType objects must not exceed the MathType 3.1 specifications to be successfully loaded and converted. Information on MathType format can be found at https://docs.wiris.com/en/mathtype/office_tools/microsoft_office.

If a document containing OMML formulas has been saved in `.docx` format and then converted to the older `.doc` format, then any OMML objects are converted into graphics, which are displayed in LibreOffice as graphics.

[S]: Convert and save the object

Select this option if LibreOffice OLE objects are to be converted and saved in Microsoft file format. LibreOffice converts any formulas into a format that can be read and modified by Microsoft Equation Editor and MathType.

When this option is not selected, the formula is treated as an OLE object on conversion into a `.doc` format and remains linked to LibreOffice. A double-click on the object in Microsoft Office opens LibreOffice.

LibreOffice

Math Guide 26.2

Appendix A,
Command Reference



Introduction

This appendix lists the functions and commands that are available for use in the Formula Editor for creating formulas.

Formula commands

The more common formula functions and commands are entered by clicking on the appropriate icon in the **Elements** panel on the Sidebar. When no icon displayed in the **Elements** panel, then the function or command has to be entered into the Formula Editor using markup language.

> **Note**
>
> In the markup language displayed in the following tables, replace the place holder <? > with the value required in the formula being created.

Unary/binary operators

To access the regularly used unary/binary operators when entering markup language into the Formula Editor, select *Unary/Binary Operators* in the **Elements** panel on the Sidebar (Figure 29). Refer to Table 8 for a complete list of unary/binary operators.

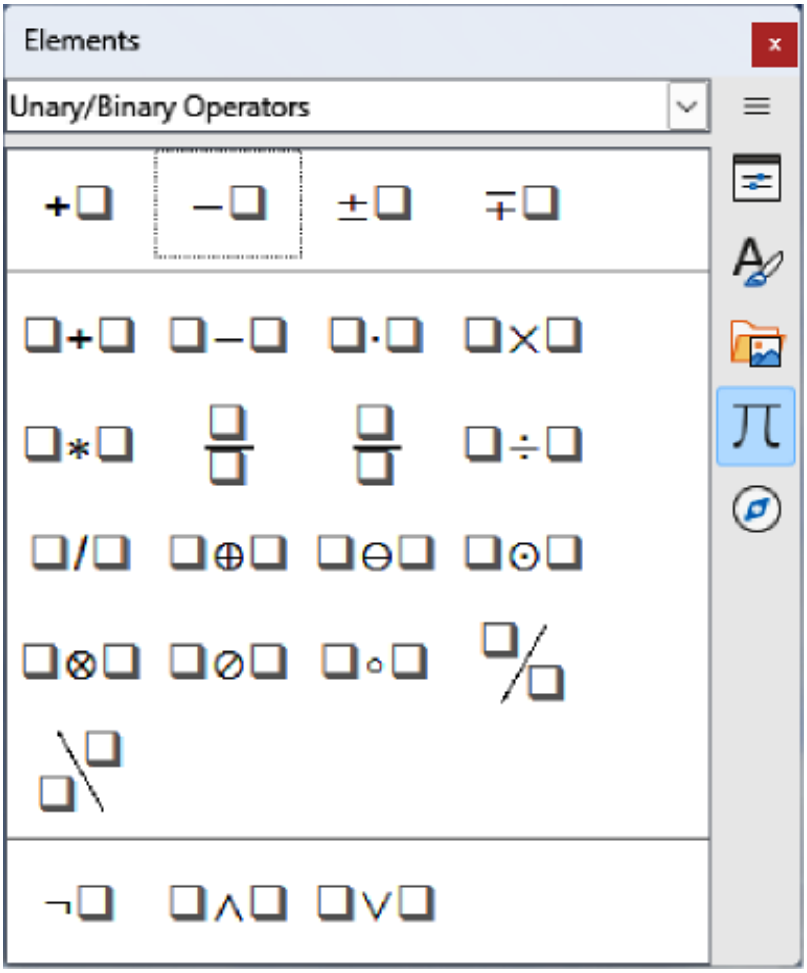


Figure 29: Sidebar Elements panel — Unary/Binary Operators page

Table 8: Unary/binary operators

Unary/binary operator	***Markup language***	***Example formula***
Plus	+	+1

Unary/binary operator	***Markup language***	***Example formula***
Minus	`-`	-1
Plus/minus	`+-` or `plusminus`	± 1
Minus/plus	`-+` or `minusplus`	∓ 1
Addition	`<?> + <?>`	$A+B$
Subtraction	`<?> - <?>`	$A-B$
Multiplication (Dot)	`<?> cdot <?>`	$A \cdot B$
Multiplication	`<?> times <?>`	$A \times B$
Multiplication (*)	`<?> * <?>`	$A * B$
Division (Fraction)	`{<?>} over {<?>}`	$\frac{A}{B}$
Frac (Fraction)	`frac {<?>} {<?>}`	$\frac{A}{B}$
Division	`<?> div <?>`	$A \div B$
Division (Slash)	`<?> / <?>` or `<?> slash <?>`	A/B
Division (Wideslash)	`{<?>} wideslash {<?>}`	$A \mathbin{\big/} B$
Division (Counter Wideslash)	`<?> widebslash <?>`	$A \mathbin{\big\backslash} B$
Concatenate	`<?> circ <?>`	$A \circ B$
Boolean NOT	`neg <?>`	$\neg A$
Boolean AND	`<?> and <?>` or `<?> & <?>`	$A \wedge B$
Boolean OR	`<?> or <?>`	$A \vee B$
Backslash	`<?> bslash <?>`	$A \setminus B$
Slash in circle	`<?> odivide <?>`	$A \oslash B$
Small multiply symbol in circle	`<?> odot <?>`	$A \odot B$
Subtract symbol in circle	`<?> ominus <?>`	$A \ominus B$
Add symbol in circle	`<?> oplus <?>`	$A \oplus B$
Multiply symbol in circle	`<?> otimes <?>`	$A \otimes B$
User defined binary operator	`<?> boper ????? <?>`	$A \; binOp \; B$
User defined unary operator	`uoper ????? <?>`	$unOp \; B$

Relations

To access the regularly used relations when entering markup language into the Formula Editor, select *Relations* in the **Elements** panel on the Sidebar (Figure 30). Refer to Table 9 for a complete list of relations.

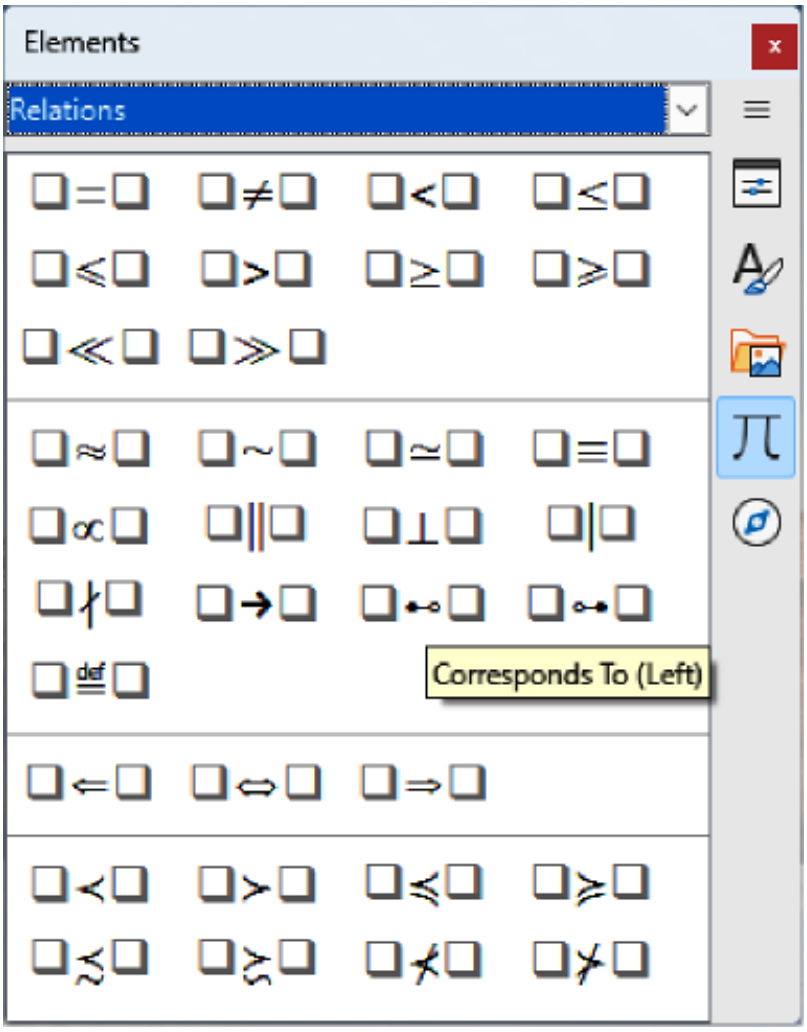


Figure 30: Elements panel in Sidebar — Relations page

Table 9: Relations

Relation	***Markup language***	***Example formula***
Is equal	`<?> = <?>`	$A = B$
Is not equal	`<?> <> <?>` or `<?> neq <?>`	$A \neq B$
Is less than	`<?> < <?>` or `<?> lt <?>`	$A < B$
Is less than or equal to	`<?> <= <?>`	$A \leq B$
Is less than or equal to	`<?> leslant <?>`	$A \leqslant B$
Is greater than	`<?> > <?>` or `<?> gt <?>`	$A > B$
Is greater than or equal to	`<?> >= <?>`	$A \geq B$
Is greater than or equal to	`<?> geslant <?>`	$A \geqslant B$
Is much less than	`<?> << <?>` or `<?> ll <?>`	$A \ll B$
Is much greater than	`<?> >> <?>` or `<?> gg <?>`	$A \gg B$
Is approximately equal	`<?> approx <?>`	$A \approx B$

Relation	*Markup language*	*Example formula*
Is similar to	`<?> sim <?>`	$A \sim B$
Is similar to or equal	`<?> simeq <?>`	$A \simeq B$
Is congruent to	`<?> equiv <?>`	$A \equiv B$
Is proportional to	`<?> prop <?>`	$A \propto B$
Is parallel to	`<?> parallel <?>`	$A \parallel B$
Is orthogonal to	`<?> ortho <?>`	$A \perp B$
Divides	`<?> divides <?>`	$A \mid B$
Does not divide	`<?> ndivides <?>`	$A \nmid B$
Toward	`<?> toward <?>`	$A \rightarrow B$
Double arrow left	`<?> dlarrow <?>`	$A \Leftarrow B$
Double arrow left and right	`<?> dlrarrow <?>`	$A \Leftrightarrow B$
Double arrow right	`<?> drarrow <?>`	$A \Rightarrow B$
Precedes	`<?> prec <?>`	$A \prec B$
Succeeds	`<?> succ <?>`	$A \succ B$
Precedes or equal to	`<?> preccurlyeq <?>`	$A \preccurlyeq B$
Succeeds or equal to	`<?> succcurlyeq <?>`	$A \succcurlyeq B$
Precedes or equivalent to	`<?> precsim <?>`	$A \precsim B$
Succeeds or equivalent to	`<?> succsim <?>`	$A \succsim B$
Not precedes	`<?> nprec <?>`	$A \not\prec B$
Not succeeds	`<?> nsucc <?>`	$A \not\succ B$
Definition	`<?> def <?>`	$A \stackrel{\text{def}}{=} B$
Corresponding symbol image of	`<?> transl <?>`	$A \multimap B$
Corresponding symbol original of	`<?> transr <?>`	$A \circ\!\!- B$

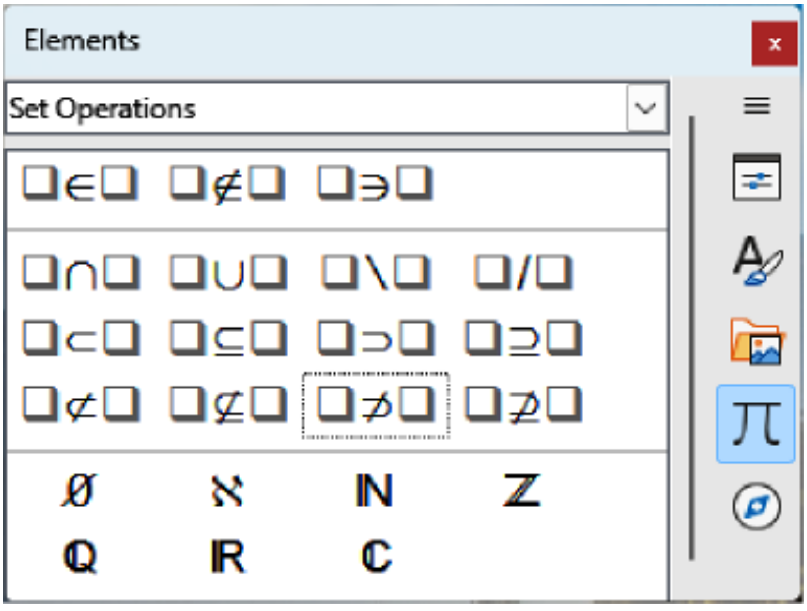


Figure 31: Elements panel in Sidebar — Set Operations page

Set operations

To access the regularly used set operations when entering markup language into the Formula Editor, select *Set Operations* in the **Elements** panel on the Sidebar (Figure 31). Refer to Table 10 for a complete list of set operations.

Table 10: Set operations

Set operation	***Markup language***	***Example formula***
Is in	`<?> in <?>`	$A \in B$
Is not in	`<?> notin <?>`	$A \notin B$
Owns	`<?> owns <?>` or `<?> ni <?>`	$A \ni B$
Intersection	`<?> intersection <?>`	$A \cap B$
Union	`<?> union <?>`	$A \cup B$
Difference	`<?> setminus <?>` or `<?> bslash <?>`	$A \setminus B$
Quotient set (slash) between characters	`<?> slash <?>`	A / B
Subset	`<?> subset <?>`	$A \subset B$
Subset or equal to	`<?> subseteq <?>`	$A \subseteq B$
Superset	`<?> supset <?>`	$A \supset B$
Superset or equal to	`<?> supseteq <?>`	$A \supseteq B$
Not subset	`<?> nsubset <?>`	$A \not\subset B$
Not subset or equal to	`<?> nsubseteq <?>`	$A \not\subseteq B$
Not superset	`<?> nsupset <?>`	$A \not\supset B$
Not superset or equal to	`<?> nsupseteq <?>`	$A \not\supseteq B$
Empty set	`emptyset`	$\emptyset$
Aleph (cardinal numbers)	`aleph`	$\aleph$
Natural numbers set	`setN`	$\mathbb{N}$
Integers set	`setZ`	$\mathbb{Z}$
Set of rational numbers	`setQ`	$\mathbb{Q}$
Real numbers set	`setR`	$\mathbb{R}$
Complex numbers set	`setC`	$\mathbb{C}$

Functions

To access the regularly used functions when entering markup language into the Formula Editor, select *Functions* in the **Elements** panel on the Sidebar (Figure 32). Refer to Table 11 for a complete list of functions.

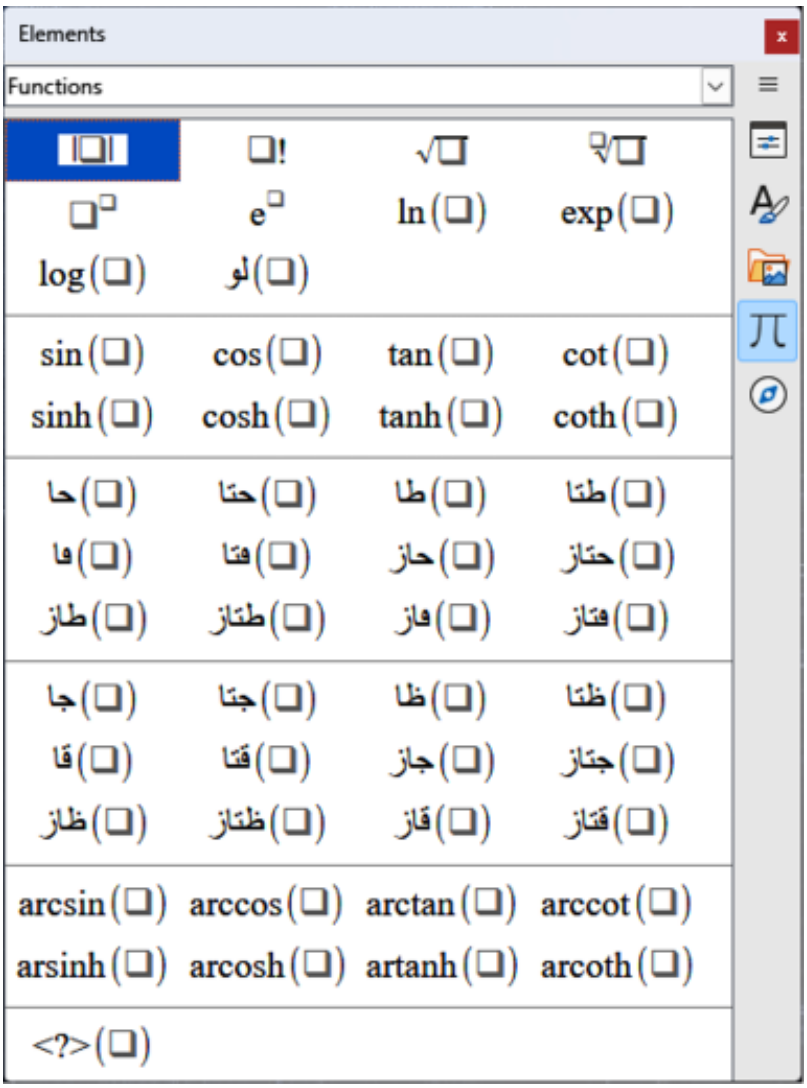


Figure 32: Elements panel in Sidebar — Functions page

Table 11: Functions

Operation	***Markup language***	***Example formula***
Absolute value	`abs{<?>}`	$\lvert A \rvert$
Factorial	`fact{<?>}`	$A!$
Square root	`sqrt{<?>}`	$\sqrt{A}$
Nth root	`nroot{<?>}{<?>}`	$\sqrt[A]{B}$
Power	`<?>^{<?>}`	A^{B}
Natural exponential function	`func e^{<?>}`	e^{A}
Natural logarithm	`ln(<?>)`	$\ln(A)$
Exponential function	`exp(<?>)`	$\exp(A)$
Logarithm	`log(<?>)`	$\log(A)$
Sine	`sin(<?>)`	$\sin(A)$
Cosine	`cos(<?>)`	$\cos(A)$
Tangent	`tan(<?>)`	$\tan(A)$

Operation	*Markup language*	*Example formula*
Cotangent	`cot(<?>)`	$\cot(A)$
Hyperbolic sine	`sinh(<?>)`	$\sinh(A)$
Hyperbolic cosine	`cosh(<?>)`	$\cosh(A)$
Hyperbolic tangent	`tanh(<?>)`	$\tanh(A)$
Hyperbolic cotangent	`coth(<?>)`	$\coth(A)$
Inverse sine or arcsine	`arcsin(<?>)`	$\arcsin(A)$
Inverse cosine or arccosine	`arccos(<?>)`	$\arccos(A)$
Inverse tangent or arctangent	`arctan(<?>)`	$\arctan(A)$
Inverse cotangent or arccotangent	`arccot(<?>)`	$\operatorname{arccot}(A)$
Inverse or area hyperbolic sine	`arsinh(<?>)`	$\operatorname{arsinh}(A)$
Inverse or area hyperbolic cosine	`arcosh(<?>)`	$\operatorname{arcosh}(A)$
Inverse or area hyperbolic tangent	`artanh(<?>)`	$\operatorname{artanh}(A)$
Inverse or area hyperbolic cotangent	`arcoth(<?>)`	$\operatorname{arcoth}(A)$
Back epsilon	`backepsilon`	$\ni$
Subscript	`<?> sub <?>`	A_B

Operators

To access the regularly used operators when entering markup language into the Formula Editor, select *Operators* in the **Elements** panel on the Sidebar (Figure 33). Refer to Table 12 for a complete list of operators.

Note

For customized operators, use the command `oper` followed by the custom operator.

For example, entering `oper OP from 0 to 1 A` will create the formula $\overset{1}{\underset{0}{OP}}\,A$.

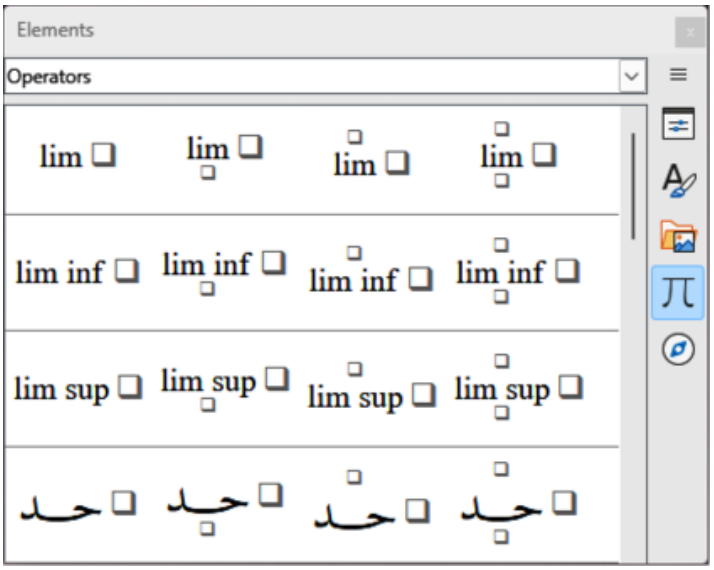


Figure 33: Elements panel in sidebar — Operators page

Table 12: Operators

Operation	***Markup language***	***Example formula***
Limes	`lim <?>`	$\lim A$
Limes subscript bottom	`lim from{<?>} <?>`	$\lim_{A} B$
Limes superscript top	`lim to{<?>} <?>`	$\overset{A}{\lim} B$
Limes sup/sub script	`lim from{<?>} to{<?>} <?>`	$\lim_{A}^{B} C$
Limes inferior	`liminf <?>`	$\liminf A$
Limes superior	`limsup <?>`	$\limsup A$
Sum	`sum <?>`	$\sum a$
Sum subscript bottom	`sum from{<?>} <?>`	$\sum_{A} B$
Sum superscript top	`sum to{<?>} <?>`	$\overset{A}{\sum} B$
Sum sup/sub script	`sum from{<?>} to{<?>} <?>`	$\sum_{A}^{B} C$
Product	`prod <?>`	$\prod A$
Product subscript bottom	`prod from{<?>} <?>`	$\prod_{A} B$
Product superscript top	`prod to{<?>} <?>`	$\overset{A}{\prod} B$
Product sup/sub script	`prod from{<?>} to{<?>} <?>`	$\prod_{A}^{B} C$
Coproduct	`coprod <?>`	$\coprod A$
Coproduct subscript bottom	`coprod from{<?>} <?>`	$\coprod_{A} B$
Coproduct superscript top	`coprod to{<?>} <?>`	$\overset{A}{\coprod} B$
Coproduct sup/sub script	`coprod from{<?>} to{<?>} <?>`	$\coprod_{A}^{B} C$
Integral	`int <?>`	$\int A$
Integral subscript bottom	`int from{<?>} <?>`	$\int_{A} B$
Integral superscript top	`int to{<?>} <?>`	$\overset{A}{\int} B$
Integral sup/sub script	`int from{<?>} to{<?>} <?>`	$\int_{A}^{B} C$
Double integral	`iint <?>`	$\iint A$

Operation	*Markup language*	*Example formula*
Double integral subscript bottom	`iint from{<?>} <?>`	$\iint_{A} B$
Double integral superscript top	`iint to{<?>} <?>`	$\overset{A}{\iint} B$
Double integral sup/sub script	`iint from{<?>} to{<?>} <?>`	$\overset{B}{\underset{A}{\iint}} C$
Triple integral	`iiint <?>`	$\iiint A$
Triple integral subscript bottom	`iiint from{<?>} <?>`	$\underset{A}{\iiint} B$
Triple integral superscript top	`iiint to{<?>} <?>`	$\overset{A}{\iiint} B$
Triple integral sup/sub script	`iiint from{<?>} to{<?>} <?>`	$\overset{B}{\underset{A}{\iiint}} C$
Curve integral	`lint <?>`	$\oint A$
Curve integral subscript bottom	`lint from{<?>} <?>`	$\underset{A}{\oint} B$
Curve integral superscript top	`lint to{<?>} <?>`	$\overset{A}{\oint} B$
Curve integral sup/sub script	`lint from{<?>} to{<?>} <?>`	$\overset{B}{\underset{A}{\oint}} C$
Double curve integral	`llint <?>`	$\oiint A$
Double curve integral subscript bottom	`llint from{<?>} <?>`	$\underset{A}{\oiint} B$
Double curve integral superscript top	`llint to{<?>} <?>`	$\overset{A}{\oiint} B$
Double curve integral sup/sub script	`llint from{<?>} to{<?>} <?>`	$\overset{B}{\underset{A}{\oiint}} C$
Triple curve integral	`lllint <?>`	$\oiiint A$
Triple curve integral subscript bottom	`lllint from{<?>} <?>`	$\underset{A}{\oiiint} B$
Triple curve integral superscript top	`lllint to{<?>} <?>`	$\overset{A}{\oiiint} B$
Triple curve integral sup/sub script	`lllint from{<?>} to{<?>} <?>`	$\overset{B}{\underset{A}{\oiiint}} C$

Attributes

To access the regularly used attributes when entering markup language into the Formula Editor, select *Attributes* in the **Elements** panel on the Sidebar (Figure 34). Refer to Table 13 for a complete list of attributes.

Note

The font command changes the font type. The first placeholder is replaced with the name of the font and the second placeholder is replaced with the selected formula values or text. The default fonts are Serif, Sans, or Fixed. If custom fonts have been added to Math (see Chapter 1, Creating and Editing Formulas), then replace the first placeholder with the custom font name.

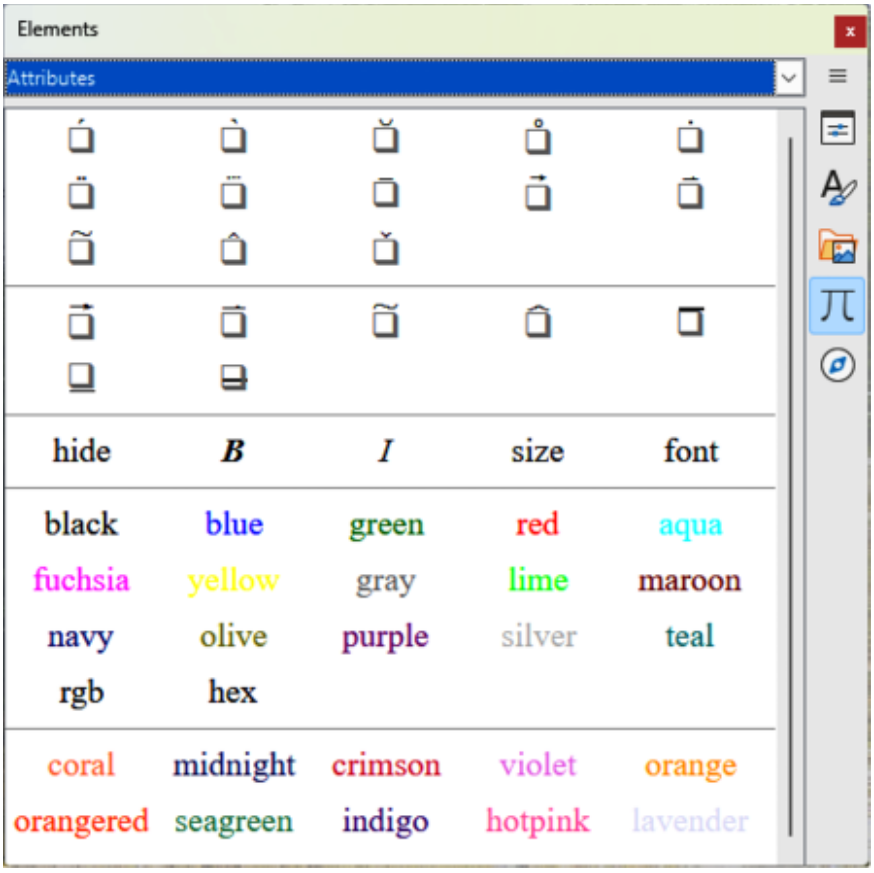


Figure 34: Elements panel in Sidebar — Attributes page

Table 13: Attributes

Operation	***Markup language***	***Example formula***
Acute accent	`acute <?>`	$\acute{A}$
Grave accent	`grave <?>`	$\grave{A}$
Breve	`breve <?>`	$\breve{A}$
Circle	`circle <?>`	$\mathring{A}$
Dot	`dot <?>`	$\dot{A}$
Double dot	`ddot <?>`	$\ddot{A}$
Triple dot	`dddot <?>`	$\dddot{A}$
Line above	`bar <?>`	$\bar{A}$
Vector arrow	`vec <?>`	$\vec{A}$
Harpoon	`harpoon <?>`	$\overset{\frown}{A}$
Tilde	`tilde <?>`	$\tilde{A}$

Operation	*Markup language*	*Example formula*
Circumflex	`hat <?>`	$\hat{A}$
Reverse circumflex	`check <?>`	$\check{A}$
Large vector arrow	`widevec {<?>}`	$\overrightarrow{AB}$
Large harpoon	`wideharpoon <?>`	$\overrightarrow{AB}$
Large tilde	`widetilde {<?>}`	$\widetilde{AB}$
Large circumflex	`widehat {<?>}`	$\widehat{AB}$
Line over	`overline {<?>}`	$\overline{AB}$
Line below	`underline {<?>}`	$\underline{AB}$
Line through	`overstrike {<?>}`	~~AB~~
Transparent (blank placeholder to create space)	`phantom {<?>}`	$A \quad B$
Bold font	`bold <?>`	$\boldsymbol{AB}$
Not bold font	`nbold <?>`	AB
Italic font	`ital <?> or italic <?>`	AB
Not italic font	`nitalic <?>`	AB
Resize	`size <?> {<?>}`	AB
Change font	`font <?> {<?>}`	AB
Color black	`color black {<?>}`	AB
Color blue	`color blue {<?>}`	AB
Color green	`color green {<?>}`	AB
Color red	`color red {<?>}`	AB
Color cyan	`color cyan {<?>}`	AB
Color aqua (same as cyan)	`color aqua {<?>}`	AB
Color magenta	`color magenta {<?>}`	AB
Color fuchsia (same as magenta	`Color fuchsia {<?>}`	AB
Color yellow	`color yellow {<?>}`	AB
Color gray	`color gray {<?>}`	AB
Color lime	`color lime {<?>}`	AB
Color maroon	`color maroon {<?>}`	AB
Color navy	`color navy {<?>}`	AB
Color olive	`color olive {<?>}`	AB
Color purple	`color purple {<?>}`	AB
Color silver	`color silver {<?>}`	AB
Color teal	`color teal {<?>}`	AB
RGB colors	`color rgb R G B {<?>}`	AB
Hexadecimal colors	`color hex 000000 {<?>}`	AB

Operation	Markup language	Example formula
Color coral	`color coral {<?>}`	$\color{coral}{AB}$
Color midnight	`color midnightblue {<?>}`	$\color{midnightblue}{AB}$
Color crimson	`color crimson {<?>}`	$\color{crimson}{AB}$
Color violet	`color violet {<?>}`	$\color{violet}{AB}$
Color orange	`color orange {<?>}`	$\color{orange}{AB}$
Color orangered	`color orangered {<?>}`	$\color{orangered}{AB}$
Color seagreen	`color seagreen {<?>}`	$\color{seagreen}{AB}$
Color indigo	`color indigo {<?>}`	$\color{indigo}{AB}$
Color hotpink	`color hotpink {<?>}`	$\color{hotpink}{AB}$
Color lavender	`color lavender {<?>}`	$\color{lavender}{AB}$

Brackets

To access the regularly used brackets when entering markup language into the Formula Editor, select *Brackets* in the **Elements** panel on the Sidebar (Figure 35). Refer to Table 14 for a complete list of brackets.

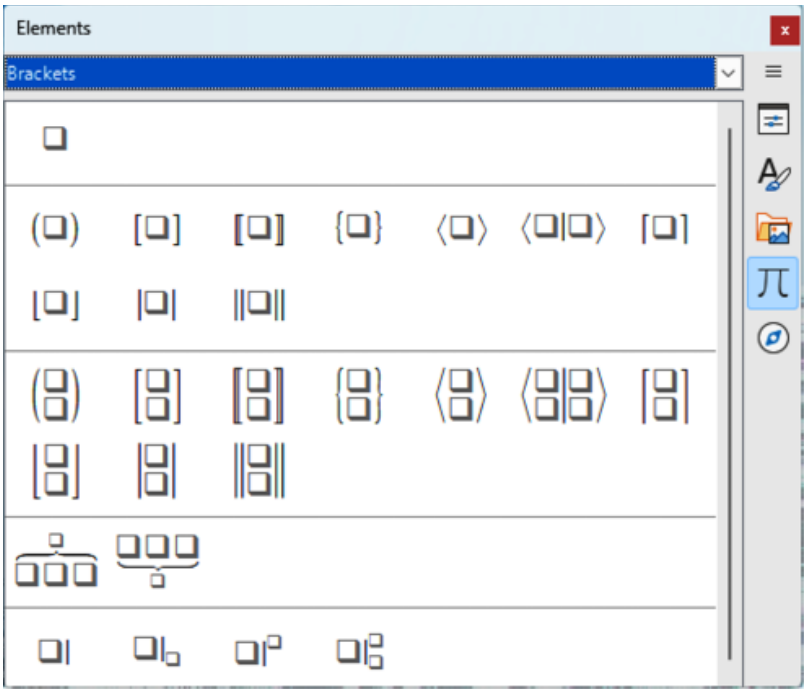


Figure 35: Elements panel in Sidebar — Brackets page

Table 14: Brackets

Operation	Markup language	Example formula
Group brackets (not displayed in formula)	`{<?>}`	AB
Round brackets	`(<?>)`	(AB)
Left round bracket only	`\(<?>`	$(A$
Right round bracket only	`<?>\)`	$A)$
Square brackets	`[<?>]`	$[AB]$
Left square bracket only	`\[<?>`	$[A$

Operation	***Markup language***	***Example formula***
Right square bracket only	`<?>\]`	$A\,]$
Double square brackets	`ldbracket <?> rdbracket`	$[\![AB]\!]$
Left double square bracket only	`\ldbracket <?>`	$[\![A$
Right double square bracket only	`<?> \rdbracket`	$A\,]\!]$
Braces	`lbrace <?> rbrace`	$\{AB\}$
Left brace only	`\lbrace <?>`	$\{A$
Right brace only	`<?> \rbrace`	$A\}$
Angle brackets	`langle <?> rangle`	$\langle AB \rangle$
Left angle bracket only	`\langle <?>`	$\langle A$
Right angle bracket only	`<?> \rangle`	$A\rangle$
Operator brackets	`langle <?> mline <?> rangle`	$\langle AB \vert CD \rangle$
Ceiling (upper half square brackets)	`lceil <?> rceil`	$\lceil AB \rceil$
Left ceiling only	`\lceil <?>`	$\lceil A$
Right ceiling only	`<?> \rceil`	$A\,]$
Floor (lower half square brackets)	`lfloor <?> rfloor`	$\lfloor AB \rfloor$
Left floor only	`\lfloor <?>`	$\lfloor A$
Right floor only	`<?> \rfloor`	$A\rfloor$
Single lines	`lline <?> rline`	$\vert AB \vert$
Left single line only	`\lline <?>`	$\vert A$
Right single line only	`<?> \rline`	$A\vert$
Double lines	`ldline <?> rdline`	$\Vert AB \Vert$
Left double line only	`\ldline <?>`	$\Vert A$
Right double line only	`<?> \rdline`	$A\Vert$
Scalable round brackets	`left ( <?> right )`	$\left(\frac{A}{B} \right)$
Scalable left round bracket only	`left ( <?> right none`	$\left(\frac{A}{B}$
Scalable right round bracket only	`left none <?> right )`	$\frac{A}{B} \right)$
Scalable square brackets	`left [ <?> right ]`	$\left[\frac{A}{B} \right]$
Scalable left square bracket only	`left [ <?> right none`	$\left[\frac{A}{B}$
Scalable right square bracket only	`left none <?> right ]`	$\frac{A}{B} \right]$

Operation	*Markup language*	*Example formula*
Scalable double square brackets	`left ldbracket <?> right rdbracket`	$\left[\!\left[\frac{A}{B} \right]\!\right]$
Scalable left double square bracket only	`left ldbracket <?> right none`	$\left[\!\left[\frac{A}{B} \right.$
Scalable right double square bracket only	`left none <?> right rdbracket`	$\left. \frac{A}{B} \right]\!\right]$
Scalable braces	`left lbrace <?> right rbrace`	$\left\{ \frac{A}{B} \right\}$
Scalable left brace only	`left lbrace {<?>} right none`	$\left\{ \frac{A}{B} \right.$
Scalable right brace only	`left none {<?>} right rbrace`	$\left. \frac{A}{B} \right\}$
Scalable angle brackets	`left langle <?> right rangle`	$\left\langle \frac{A}{B} \right\rangle$
Scalable left angle bracket only	`left langle {<?>} right none`	$\left\langle \frac{A}{B} \right.$
Scalable right angle bracket only	`left none {<?>} right rangle`	$\left. \frac{A}{B} \right\rangle$
Scalable operator brackets	`left langle <?> mline <?> right none`	$\left\langle \frac{A}{B} \middle\vert \frac{C}{D} \right\rangle$
Scalable left operator bracket	`left langle <?> mline <?> right none`	$\left\langle \frac{A}{B} \middle\vert \frac{C}{D} \right.$
Scalable right operator bracket	`left none <?> mline <?> right rangle`	$\left. \frac{A}{B} \middle\vert \frac{C}{D} \right\rangle$
Scalable ceiling (upper half square brackets)	`left lceil <?> right rceil`	$\left\lceil \frac{A}{B} \right\rceil$
Scalable left ceiling	`left lceil <?> right none`	$\left\lceil \frac{A}{B} \right.$
Scalable right ceiling	`left none <?> right rceil`	$\left. \frac{A}{B} \right\rceil$
Scalable floor (lower half square brackets)	`left lfloor <?> right rfloor`	$\left\lfloor \frac{A}{B} \right\rfloor$
Scalable left floor	`left lfloor <?> right none`	$\left\lfloor \frac{A}{B} \right.$
Scalable right floor	`left none <?> right rfloor`	$\left. \frac{A}{B} \right\rfloor$
Scalable single lines	`left lline <?> right rline`	$\left\vert \frac{A}{B} \right\vert$

Operation	Markup language	Example formula
Scalable left single line	`left lline <?> right none`	$\left\lvert \frac{A}{B} \right.$
Scalable right single line	`left none <?> right rline`	$\left. \frac{A}{B} \right\rvert$
Scalable double lines	`left ldline <?> right rdline`	$\left\lVert \frac{A}{B} \right\rVert$
Scalable left double line	`left ldline <?> right none`	$\left\lVert \frac{A}{B} \right.$
Scalable right double line	`left none <?> right rdline`	$\left. \frac{A}{B} \right\rVert$
Scalable top brace	`{<?>} overbrace {<?>}`	$\overbrace{ABC}^{D}$
Scalable bottom brace	`{<?>} underbrace {<?>}`	$\underbrace{ABC}_{D}$

Formats

To access the regularly used formats when entering markup language into the Formula Editor, select *Formats* in the **Elements** panel on the Sidebar (Figure 36). Refer to Table 15 for a complete list of formats.



Figure 36: Elements panel in Sidebar — Formats page

Note

By default, text characters are aligned to the center in a formula. Use the `alignl` and `alignr` commands to align text characters in a formula to the left or right when a formula has more than one line. Also, use the align commands in conjunction with the stack commands to align multi-line formulas on a specific formula element, for example, the equals sign (=).

Table 15: Formats

Operation	Markup language	Example formula
Power (superscript right)	`<?>^{<?>}` or `<?>sup{<?>}` or `<?>rsup{<?>}`	20^{10}

Operation	*Markup language*	*Example formula*
Subscript right	`<?>_{<?>}` or `<?>sub{<?>}` or `<?>rsub{<?>}`	20_{10}
Superscript left	`<?>lsup{<?>}`	$^{10}20$
Subscript left	`<?>lsub{<?>}`	$_{10}20$
Superscript center	`<?>csup{<?>}`	$\overset{10}{20}$
Subscript center	`<?>csub{<?>}`	$\underset{10}{20}$
New line	`newline`	$123 \\ CD$
Small gap	`` ` ``	$123\ 456$
Blank or large gap	`~`	$123\ \ 456$
No space (suppresses space between elements)	`nospace {<?>}`	$12+4+5$
Align left	`alignl <?>`	$ABC \\ D$
Align center	`alignc <?>`	$ABC \\ \ \ D$
Align right	`alignr <?>`	$ABC \\ \ \ \ D$
Vertical stack (2 elements)	`binom{<?>}{<?>}`	$AB \\ 12$
Vertical stack (more than 2 elements)	`stack{<?> # <?> # <?>}`	$AB \\ 12 \\ CD$
Matrix stack	`matrix{<?> # <?> ## <?> # <?>}`	$AB \quad 12 \\ 34 \quad CD$

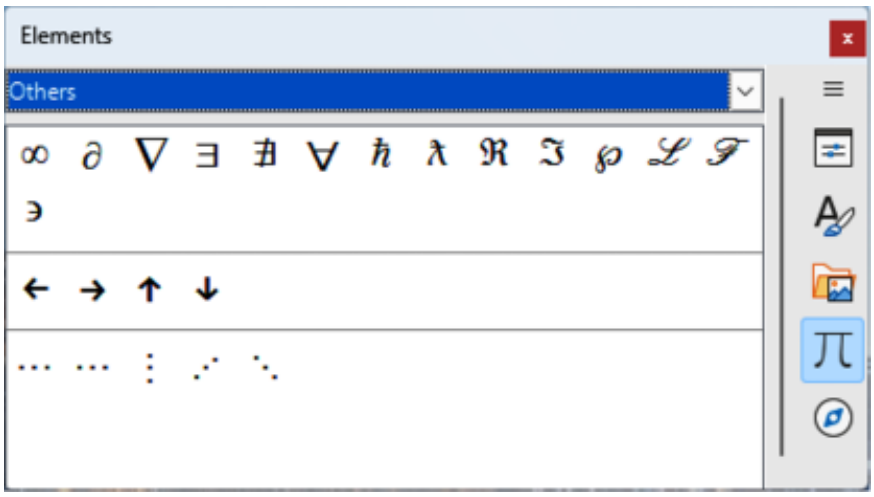


Figure 37: Elements panel in Sidebar — Others page

Others

To access the regularly used other commands when entering markup language into the Formula Editor, select *Others* in the **Elements** panel on the Sidebar (Figure 37). Refer to Table 16 for a complete list of other commands.

Table 16: Others

Operation	***Markup language***	***Example formula***
Placeholder	`<?>`	
Infinity	`infinity` or `infty`	∞
Partial derivative or set margin	`partial`	∂
Nabla vector operator	`nabla`	∇
Existential quantifier, there is at least one	`exists`	$\exists$
Existential quantifier, there does not exist	`notexists`	$\nexists$
Universal quantifier, for all	`forall`	$\forall$
H-bar constant	`hbar`	$\hbar$
Lambda bar	`lambdabar`	λ
Real part of a complex number	`re`	$\Re$
Imaginary part of a complex number	`im`	$\Im$
Weierstrass p function	`wp`	$\wp$
Laplace transformation	`laplace`	$\mathscr{L}$
Left arrow	`leftarrow`	$\leftarrow$
Right arrow	`rightarrow`	$\rightarrow$
Up arrow	`uparrow`	$\uparrow$
Down arrow	`downarrow`	$\downarrow$
Ellipsis	`dotslow`	$\cdots$
Math axis ellipsis	`dotsaxis`	$\cdots$
Vertical ellipsis	`dotsvert`	$\vdots$
Upward diagonal ellipsis	`dotsup` or `dotsdiag`	$\iddots$
Downward diagonal ellipsis	`dotsdown`	$\ddots$

Greek characters

Greek and italic Greek characters are normally entered into a formula using the Symbols dialog (Figure 38 and Figure 39). However, if the Greek character is known, type a percentage sign (%) followed by the Greek character name into the Formula Editor.

- For uppercase Greek characters, type the name using uppercase characters, for example %BETA B . See Table 17 for a list of uppercase Greek characters.
- For lowercase Greek characters, type the name using lowercase characters, for example %beta β . See Table 17 for a full list of lowercase Greek characters.
- Italic Greek characters, type the character i after the percentage sign (%) and before the name of the Greek character, for example, %iPI $\mathit{\Pi}$ for uppercase and %ipi for lowercase π . See Table 18 for a full list of italic Greek characters.

Note

In localized versions of LibreOffice, the names of Greek characters are also localized. If a document is not localized to the same language, then the names of Greek characters in the following tables may not work correctly. If this happens, use the Symbols dialog to select the required Greek character. When selected in the Symbols dialog, the name of the Greek character is displayed at the bottom of the Symbols dialog in the correct language.

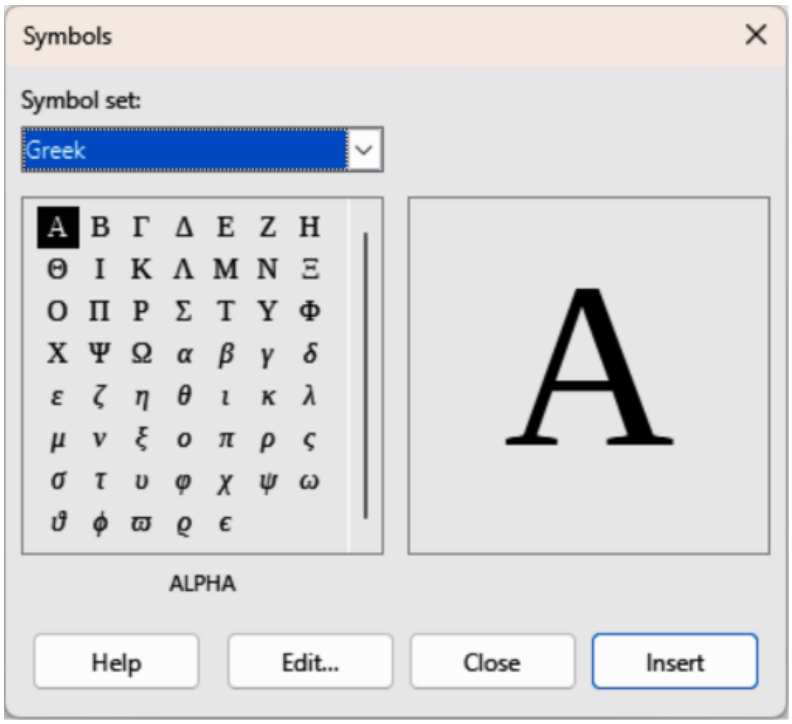


Figure 38: Symbols dialog — Greek characters

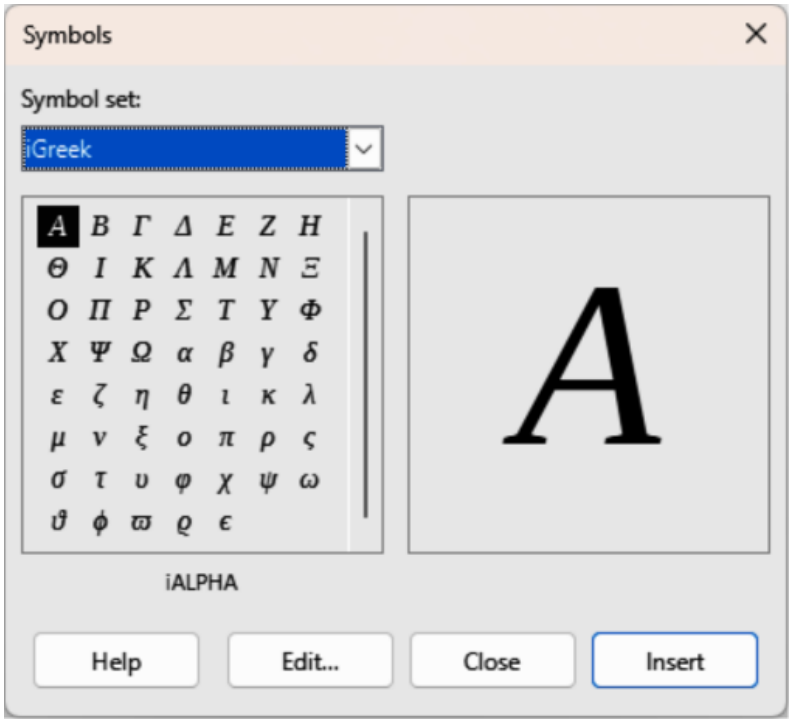


Figure 39: Symbols dialog — Italic Greek characters

Table 17: Greek characters

Markup language	***Uppercase Character***	***Markup language***	***Lowercase Character***
%ALPHA	Α	%alpha	α
%BETA	Β	%beta	β
%GAMMA	Γ	%gamma	ɣ
%DELTA	Δ	%delta	δ
%EPSILON	Ε	%epsilon	ε
%ZETA	Ζ	%zeta	ζ
%ETA	Η	%eta	η
%THETA	Θ	%theta	θ
%IOTA	Ι	%iota	ι
%KAPPA	Κ	%kappa	κ
%LAMBDA	Λ	%lambda	λ
%MU	Μ	%mu	μ
%NU	Ν	%nu	ν
%XI	Ξ	%xi	ξ
%OMICRON	Ο	%omicron	ο
%PI	Π	%pi	π
%RHO	Ρ	%rho	ρ
%SIGMA	Σ	%sigma	σ
%TAU	Τ	%tau	τ
%UPSILON	Υ	%upsilon	υ
%PHI	Φ	%phi	φ
%CHI	Χ	%chi	χ
%PSI	Ψ	%psi	ψ
%OMEGA	Ω	%omega	ω

Table 18: Italic Greek characters

Markup language	***Uppercase Character***	***Markup language***	***Lowercase Character***
%iALPHA	A	%ialpha	α
%iBETA	B	%ibeta	β
%iGAMMA	Γ	%igamma	γ
%iDELTA	Δ	%idelta	δ
%iEPSILON	E	%iepsilon	ϵ
%iZETA	Z	%izeta	ζ
%iETA	H	%ieta	η
%iTHETA	Θ	%itheta	θ
%iIOTA	I	%iiota	ι
%iKAPPA	K	%ikappa	κ
%iLAMBDA	Λ	%ilambda	λ

Markup language	Uppercase Character	Markup language	Lowercase Character
%iMU	M	%imu	μ
%iNU	N	%inu	ν
%iXI	Ξ	%ixi	ξ
%iOMICRON	O	%iomicron	o
%iPI	Π	%ipi	π
%iRHO	P	%irho	ρ
%iSIGMA	Σ	%isigma	σ
%iTAU	T	%itau	τ
%iUPSILON	Y	%iupsilon	υ
%iPHI	Φ	%iphi	ϕ
%iCHI	X	%ichi	χ
%iPSI	Ψ	%ipsi	ψ
%iOMEGA	Ω	%iomega	ω

Special characters

Special characters are normally entered into a formula using the Symbols dialog (Figure 38 and Figure 39). However, if the name of the special character is known, type a percentage sign (%) followed by the special character name into the Formula Editor. Table 19 Shows the full list of special characters that are available in LibreOffice.

Table 19: Special characters

Markup language	Character	Markup language	Character
%perthousand	‰	%tendto	→
%noelement	∉	%infinite	∞
%angle	∢	%and	∧
%or	∨	%notequal	≠
%identical	≡	%strictlylessthan	≪
%strictlygreaterthan	≫		

Note

In localized versions of LibreOffice, the names of special characters are also localized. If a document is not localized to the same language, then the special character names in Table 19 may not work correctly. If this happens, use the Symbols dialog to select the required special character. When selected in the Symbols dialog, the name of the special character is displayed at the bottom of the Symbols dialog in the correct language.

Reserved words

A reserved word is a word that is used by LibreOffice as a command in the Math markup language and cannot be used as a variable name or entered into a formula as text. To use a reserved word as text in a formula then a reserved word must be placed between double quotes.

For example, consider that to use the word “purple” in a formula and all the formula elements are not changed to the color purple. If `A purple B` is typed into the Formula Editor, the resulting formula is $A\,¿\,B$, which is incorrect and the error is shown by the inverted question mark. To correct this error, type `A "purple" B` in the Formula Editor using double quotes each side of the word purple and the resulting correct formula is $A\,\text{purple}\,B$.

If the *italic* version of the word “purple” is required, enter `A italic{ "purple" } B` into the Formula Editor, which creates $A\,purple\,B$ in the formula.

LibreOffice

LibreOffice Documentation Team

Math Guide

Using the formula editor

About this book:

Math is the formula editor module in LibreOffice and provides the layout options for displaying mathematical, chemical, electrical, or scientific formulas quickly in standard written notation.

Math is most commonly used in Writer when creating text documents, but Math can also be used with other types of documents or stand-alone.

If you have never used LibreOffice before, or require an introduction to all LibreOffice modules, please refer to the Getting Started Guide first.

About the authors:

This book was written by volunteers from the LibreOffice community. Profits from the sales of the printed edition are used to benefit the community.

A PDF version of this book can be downloaded free from:
https://documentation.libreoffice.org/en/

About LibreOffice:

LibreOffice is the free, libre, and open source personal productivity suite from The Documentation Foundation. It runs on Windows, macOS, and GNU/Linux. Support and documentation is free from a large, dedicated community of users, contributors, and developers.

Get involved as a volunteer and work in many areas: development, quality assurance, documentation, translation, user support, and more.

Download LibreOffice for free from:
https://libreoffice.org/downloads/

For commercial use:

Work with certified LibreOffice professionals for any business needs, from deployment or conversion from other office suits, staff training, or custom development needs.

For information on professional support services, go to
https://www.libreoffice.org/get-help/professional-support/

26.2

Fantastic Community - Fun Project - Free Office Suite