

Supported Functions

This is a list of TeX functions supported by KaTeX. It is sorted into logical groups.

There is a similar [Support Table](#), sorted alphabetically, that lists both supported and unsupported functions.

Accents

a' a'	\tilde{a} <code>\tilde{a}</code>	\mathring{g} <code>\mathring{g}</code>
a'' a''	\widetilde{ac} <code>\widetilde{ac}</code>	\overline{AB} <code>\overline{AB}</code>
a' a'^{\prime}	\underline{AB} <code>\underline{AB}</code>	\underline{AB} <code>\underline{AB}</code>
\acute{a} <code>\acute{a}</code>	\vec{F} <code>\vec{F}</code>	\overrightarrow{AB} <code>\overrightarrow{AB}</code>
\bar{y} <code>\bar{y}</code>	\overleftarrow{AB} <code>\overleftarrow{AB}</code>	\overrightarrow{AB} <code>\overrightarrow{AB}</code>
\check{a} <code>\check{a}</code>	\overleftarrow{AB} <code>\overleftarrow{AB}</code>	\overrightarrow{AB} <code>\overrightarrow{AB}</code>
\breve{a} <code>\breve{a}</code>	\overleftarrow{ac} <code>\overleftarrow{ac}</code>	\overrightarrow{ac} <code>\overrightarrow{ac}</code>
\dot{a} <code>\dot{a}</code>	\overleftrightarrow{AB} <code>\overleftrightarrow{AB}</code>	\widehat{AB} <code>\widehat{AB}</code>
\ddot{a} <code>\ddot{a}</code>	\overleftarrow{AB} <code>\overleftarrow{AB}</code>	\underbrace{AB} <code>\underbrace{AB}</code>
\grave{a} <code>\grave{a}</code>	\overline{AB} <code>\overline{AB}</code>	\overline{AB} <code>\overline{AB}</code>
$\hat{\theta}$ <code>\hat{\theta}</code>	\underline{AB} <code>\underline{AB}</code>	\underline{AB} <code>\underline{AB}</code>
\widehat{ac} <code>\widehat{ac}</code>	\widecheck{ac} <code>\widecheck{ac}</code>	\underline{X} <code>\underline{X}</code>

Accent functions inside `\text{...}`

\acute{a} <code>\acute{a}</code>	\tilde{a} <code>\tilde{a}</code>	\grave{a} <code>\grave{a}</code>	\check{a} <code>\check{a}</code>
------------------------------------	------------------------------------	------------------------------------	------------------------------------

See also letters and unicodes.

Delimiters

Delimiter Sizing

$(AB) \left(\right)$

```
((((((( ( \big( \Big( \bigg( \Bigg(
```

\left	\big	\bigl	\bigm	\bigr
\middle	\Big	\Bigl	\Bigm	\Bigr
\right	\bigg	\biggl	\biggm	\biggr
	\Bigg	\Biggl	\Biggm	\Biggr

Try

Docs

v0.16.11

Users

GitHub

$c \quad d$	$c \ & \ d$ <code>\end{matrix}</code>	$c \quad d$	$c \ & \ d$ <code>\end{array}</code>
$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	<code>\begin{pmatrix} a & b \\ c & d \end{pmatrix}</code>	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	<code>\begin{bmatrix} a & b \\ c & d \end{bmatrix}</code>
$\begin{vmatrix} a & b \\ c & d \end{vmatrix}$	<code>\begin{vmatrix} a & b \\ c & d \end{vmatrix}</code>	$\begin{Vmatrix} a & b \\ c & d \end{Vmatrix}$	<code>\begin{Vmatrix} a & b \\ c & d \end{Vmatrix}</code>
$\left\{ \begin{array}{cc} a & b \\ c & d \end{array} \right\}$	<code>\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}</code>	$\begin{array}{ c c c } \hline a & b & c \\ \hline d & e & f \\ \hline g & h & i \\ \hline \end{array}$	<code>\def\arraystretch{1.5}\begin{array}{c:c:c} a & b & c \\ \hline d & e & f \\ \hline g & h & i \end{array}</code>
$\begin{cases} a & \text{if } b \\ c & \text{if } d \end{cases}$	<code>x = \begin{cases} a & \text{if } b \\ c & \text{if } d \end{cases}</code>	$\begin{cases} a & \text{if } b \\ c & \text{if } d \end{cases} \Rightarrow \dots$	<code>\begin{rcases} a & \text{if } b \\ c & \text{if } d \end{rcases} \Rightarrow \dots</code>
$\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}$	<code>\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}</code>	$\sum_{\substack{i \in \Lambda \\ 0 < j < n}}$	<code>\sum_{{\begin{array}{l} \begin{subarray}{l} i \in \Lambda \\ 0 < j < n \end{subarray} \end{array}}}</code>

The auto-render extension will render the following environments even if they are not inside math delimiters such as `$$...$$`. They are display-mode only.

$\begin{aligned} a &= b + c \\ &= e + f \quad (1) \end{aligned}$	<code>\begin{aligned} \begin{equation} \begin{aligned} \begin{split} a &= b + c \\ &= e + f \end{split} \end{aligned} \end{aligned}</code>	$\begin{aligned} a &= b + c \quad (2) \\ d + e &= f \quad (3) \end{aligned}$	<code>\begin{aligned} \begin{aligned} \begin{aligned} \begin{aligned} a &= b + c \\ d + e &= f \end{aligned} \end{aligned} \end{aligned}</code>
--	--	--	---

Try	Docs	v0.16.11	Users	GitHub
$a = b \quad (4)$ $e = b + c \quad (5)$	<pre>\begin{gathered} a=b \\ e=b+c \\ \end{gathered}</pre>	$10x+ 3y = 2 \quad (6)$ $3x+13y = 4 \quad (7)$	<pre>10&x+&3&y=2 \\ 3&x+&13&y=4</pre>	<pre>\begin{aligned} & \\ & \end{aligned}</pre>
$\begin{array}{ccc} A & \xrightarrow{a} & B \\ b \downarrow & & \uparrow c \\ C & \xlongequal{} & D \end{array}$	<pre>\begin{CD} A @>a>> B \\ @VbVV @AAcA \\ C @= D \end{CD}</pre>			

Other KaTeX Environments

Environments	How they differ from those shown above
<code>darray</code> , <code>dcases</code> , <code>drcases</code>	... apply <code>displaystyle</code>
<code>matrix*</code> , <code>pmatrix*</code> , <code>bmatrix*</code> <code>Bmatrix*</code> , <code>vmatrix*</code> , <code>Vmatrix*</code>	... take an optional argument to set column alignment, as in <code>\begin{matrix*}[r]</code>
<code>equation*</code> , <code>gather*</code> <code>align*</code> , <code>alignat*</code>	... have no automatic numbering. Alternatively, you can use <code>\nonumber</code> or <code>\notag</code> to omit the numbering for a specific row of the equation.
<code>gathered</code> , <code>aligned</code> , <code>alignedat</code>	... do not need to be in display mode. ... have no automatic numbering. ... must be inside math delimiters in order to be rendered by the auto-render extension.

Acceptable line separators include: `\backslash` , `\cr` , `\backslash[distance]` , and `\cr[distance]` . *Distance* can be written with any of the [KaTeX units](#).

The `{array}` environment supports `|` and `:` vertical separators.

The `{array}` environment does not yet support `\cline` or `\multicolumn` .

RAW HTML

The following "raw HTML" features are potentially dangerous for untrusted inputs, so they are disabled by default, and attempting to use them produces the command names in red (which you can configure via the `errorColor option`). To fully trust your LaTeX input, you need to pass an option of `trust: true`; you can also enable just some of the commands or for just some URLs via the `trust option`.

KATEX	<code>\href{https://katex.org/}{\KaTeX}</code>
https://katex.org/	<code>\url{https://katex.org/}</code>
	<code>\includegraphics[height=0.8em, totalheight=0.9em, width=0.9em, alt=KA logo] {https://katex.org/img/khan-academy.png}</code>
<code>xx......</code>	<code>\htmlId{bar}{x}</code>
<code>xx......</code>	<code>\htmlClass{foo}{x}</code>
<code>xx......</code>	<code>\htmlStyle{color: red;}{x}</code>
<code>xx...</ span>...</code>	<code>\htmlData{foo=a, bar=b}{x}</code>

`\includegraphics` supports `height`, `width`, `totalheight`, and `alt` in its first argument. `height` is required.

HTML extension (`\html`-prefixed) commands are non-standard, so loosening `strict` option for `htmlExtension` is required.

Letters and Unicode

Greek Letters

Direct Input: ΑΒΓΔΕΖΗΘΙΚΛΜΝΞΟΠΡΣΤΥΦΧΨΩ αβγδεζηθικλμνξοπρστυφχψωεθωρσφ

[Try](#)[Docs](#)[v0.16.11](#)[Users](#)[GitHub](#)

$N \backslash Nu$	$\Xi \backslash xi$	$O \backslash Omicron$	$\Pi \backslash Pi$
$P \backslash Rho$	$\Sigma \backslash Sigma$	$T \backslash Tau$	$\Upsilon \backslash Upsilon$
$\Phi \backslash Phi$	$X \backslash Chi$	$\Psi \backslash Psi$	$\Omega \backslash Omega$
$\Gamma \backslash varGamma$	$\Delta \backslash varDelta$	$\Theta \backslash varTheta$	$\Lambda \backslash varLambda$
$\Xi \backslash varXi$	$\Pi \backslash varPi$	$\Sigma \backslash varSigma$	$\Upsilon \backslash varUpsilon$
$\Phi \backslash varPhi$	$\Psi \backslash varPsi$	$\Omega \backslash varOmega$	
$\alpha \backslash alpha$	$\beta \backslash beta$	$\gamma \backslash gamma$	$\delta \backslash delta$
$\epsilon \backslash epsilon$	$\zeta \backslash zeta$	$\eta \backslash eta$	$\theta \backslash theta$
$\iota \backslash iota$	$\kappa \backslash kappa$	$\lambda \backslash lambda$	$\mu \backslash mu$
$\nu \backslash nu$	$\xi \backslash xi$	$\omicron \backslash omicron$	$\pi \backslash pi$
$\rho \backslash rho$	$\sigma \backslash sigma$	$\tau \backslash tau$	$\upsilon \backslash upsilon$
$\phi \backslash phi$	$\chi \backslash chi$	$\psi \backslash psi$	$\omega \backslash omega$
$\varepsilon \backslash varepsilon$	$\varkappa \backslash varkappa$	$\vartheta \backslash vartheta$	$\vartheta \backslash thetasym$
$\varpi \backslash varpi$	$\varrho \backslash varrho$	$\varsigma \backslash varsigma$	$\varphi \backslash varphi$
$F \backslash digamma$			

Other Letters

$i \backslash imath$	$\nabla \backslash nabla$	$\Im \backslash Im$	$\mathbb{R} \backslash Reals$	$\text{\OE} \backslash text{\OE}$
$j \backslash jmath$	$\partial \backslash partial$	$\mathfrak{I} \backslash image$	$\wp \backslash wp$	$\emptyset \backslash text{\o}$
$\aleph \backslash aleph$	$\beth \backslash Game$	$\mathbb{k} \backslash Bbbk$	$\wp \backslash weierp$	$\emptyset \backslash text{\O}$
$\alef \backslash alef$	$\Finv \backslash Finv$	$\mathbb{N} \backslash N$	$\mathbb{Z} \backslash z$	$\mathfrak{B} \backslash text{\ss}$
$\alef \backslash alefsym$	$\mathbb{C} \backslash cnums$	$\mathbb{N} \backslash natnums$	$\mathring{a} \backslash text{\aa}$	$\mathfrak{i} \backslash text{\i}$
$\beth \backslash beth$	$\mathbb{C} \backslash Complex$	$\mathbb{R} \backslash R$	$\AA \backslash text{\AA}$	$\mathfrak{j} \backslash text{\j}$
$\gimel \backslash gimel$	$\ell \backslash ell$	$\mathfrak{R} \backslash Re$	$\mathfrak{ae} \backslash text{\ae}$	

Direct Input: $\sigma \vee \neg \sigma$ — $\neg \sigma \vee \sigma$

ÀÁÂÃÄÅÆÇÈÉÊËÍÍÐÑÒÓÔÖÖÙÚÛÜÝþßàáâäåçèéêëííðñòóôöùúûüýþþ

+ = () 0123456789 abcdeghi jklmno prstuwxyzvβγδφχ
0123456789αεηικλμνορστυωχβγδφχ

Math-mode Unicode (sub|super)script characters will render as if you had written regular characters in a subscript or superscript. For instance, A^{2+3} will render the same as A^{2+3} .

Unicode Mathematical Alphanumeric Symbols

Item	Range	Item	Range
Bold	A-Z a-z 0-9	Double-struck	Ā-Ž ķ
Italic	<i>A-Z a-z</i>	Sans serif	A-Z a-z 0-9
Bold Italic	<i>A-Z a-z</i>	Sans serif bold	A-Z a-z 0-9
Script	<i>Ā-Ž</i>	Sans serif italic	<i>A-Z a-z</i>
Fraktur	<i>Ā-Ž a-ž</i>	Monospace	A-Z a-z 0-9
Bold Fraktur	Ā-Ž a-ž		

Unicode

The letters listed above will render properly in any KaTeX rendering mode.

In addition, Armenian, Brahmic, Georgian, Chinese, Japanese, and Korean glyphs are always accepted in text mode. However, these glyphs will be rendered from system fonts (not KaTeX-supplied fonts) so their typography may clash. You can provide rules for CSS classes `.latin_fallback`, `.cyrillic_fallback`, `.brahmic_fallback`, `.georgian_fallback`, `.cjk_fallback`, and `.hangul_fallback` to provide fallback fonts for these languages. Use of these glyphs may cause small vertical alignment issues: KaTeX has detailed metrics for listed symbols and most Latin, Greek, and Cyrillic letters, but other accepted glyphs are treated as if they are each as tall as the letter M in the current KaTeX font.

If the KaTeX rendering mode is set to `strict: false` or `strict: "warn"` (default), then KaTeX will accept all Unicode letters in both text and math mode. All unrecognized characters will be treated as if they appeared in text mode, and are subject to the same issues of using system fonts and possibly using incorrect vertical alignment.

For Persian composite characters, a user-supplied [plug-in](#) is under development.



Layout

Annotation

$\cancel{5}$	$\overbrace{a+b+c}^{\text{note}} \text{ \textbackslash overbrace\{a+b+c\}^{\{\text{note}\}}}$
$\bcancel{5}$	$\underbrace{a+b+c}_{\text{note}} \text{ \textbackslash underbrace\{a+b+c\}_{\{\text{note}\}}}$
\cancel{ABC}	$\not= \text{ \textbackslash not =}$
\abc	$\boxed{\pi = \frac{c}{d}} \text{ \textbackslash boxed\{\text{pi=\frac c d}\}}$
$a_{\overline{n}}$	$a_{\overline{n}} \text{ a_angln}$
$\underline{-78^\circ}$	$\text{phase\{-78^\circ\}}$

`\tag{hi} x+y^{2x}`

$$x + y^{2x} \quad (\text{hi})$$

`\tag*{hi} x+y^{2x}`

$$x + y^{2x} \quad \text{hi}$$

Line Breaks

KaTeX 0.10.0+ will insert automatic line breaks in inline math after relations or binary operators such as “=” or “+”. These can be suppressed by `\nobreak` or by placing math inside a pair of braces, as in `{F=ma}`. `\allowbreak` will allow automatic line breaks at locations other than relations or operators.

Hard line breaks are `\\` and `\newline`.

In display math, KaTeX does not insert automatic line breaks. It ignores display math hard line breaks when rendering option `strict: true`.

Vertical Layout

x_n	$\stackrel{!}{=} \text{ \textbackslash stackrel\{!\}}$	$\frac{a}{b} \text{ a \atop b}$
-------	--	---------------------------------

[Try](#)[Docs](#)

v0.16.11

[Users](#)[GitHub](#)

$\frac{o}{u}$	$\underset{!}{=}$ $\{=\}$	$a + \left(\frac{\frac{a}{b}}{c} \right) = + \left(\vcenter{ \hbox{ $ \frac{a}{b} $ } } \right)$
		$\sum_{\substack{0 < i < m \\ 0 < j < n}} \sum_{\substack{0 < i < m \\ 0 < j < n}}$

`\raisebox` and `\hbox` put their argument into text mode. To raise math, nest `$...$` delimiters inside the argument as shown above.

`\vcenter` can be written without an `\hbox` if the `strict` rendering option is *false*. In that case, omit the nested `$...$` delimiters.

Overlap and Spacing

\neq	$\{=\} \mathop{\mathllap{/}\!\!\!/}\ (x^2) \ \left(x^{\smash{2}} \right)$
\neq	$\mathop{\mathrlap{,}/}\!\!\!/ \sqrt{y} \ \sqrt{\smash[b]{y}}$

$$\sum_{1 \leq i \leq j \leq n} x_{ij} \ \sum_{\mathop{\mathclap{1 \leq i \leq j \leq n}}} x_{ij}$$

KaTeX also supports `\llap`, `\rlap`, and `\clap`, but they will take only text, not math, as arguments.

Spacing

Function	Produces	Function	Produces
<code>\,</code>	$\frac{3}{18}$ em space	<code>\kern{distance}</code>	space, width = <i>distance</i>
<code>\thinspace</code>	$\frac{3}{18}$ em space	<code>\mkern{distance}</code>	space, width = <i>distance</i>
<code>\></code>	$\frac{4}{18}$ em space	<code>\mskip{distance}</code>	space, width = <i>distance</i>
<code>\:</code>	$\frac{4}{18}$ em space	<code>\hskip{distance}</code>	space, width = <i>distance</i>
<code>\medspace</code>	$\frac{4}{18}$ em space	<code>\hspace{distance}</code>	space, width = <i>distance</i>
<code>\;</code>	$\frac{5}{18}$ em space	<code>\hspace*{distance}</code>	space, width = <i>distance</i>

Try	Docs	v0.16.11	Users	GitHub
\enspace	½ em space	\hphantom{content}	space the width of content	
\quad	1 em space	\vphantom{content}	a strut the height of content	
\quad	2 em space	\!	– ⅓ em space	
~	non-breaking space	\negthinspace	– ⅓ em space	
\<space>	space	\negmedspace	– ⅔ em space	
\nobreakspace	non-breaking space	\negthickspace	– ⅕ em space	
\space	space	\mathstrut	\vphantom{()}	

Notes:

`distance` will accept any of the [KaTeX units](#).

`\kern`, `\mkern`, `\mskip`, and `\hspace` accept unbraced distances, as in: `\kern1em`.

`\mkern` and `\mskip` will not work in text mode and both will write a console warning for any unit except `\mu`.

Logic and Set Theory

\forall \forall	\complement \complement	\therefore \therefore	\emptyset \emptyset
\exists \exists	\subset \subset	\because \because	\emptyset \emptyset
\exists \exist	\supset \supset	\mapsto \mapsto	\varnothing \varnothing
\nexists \nexists	\mid \mid	\rightarrow \to	\implies \implies
\in \in	\wedge \land	\leftarrow \gets	\Leftarrow \impliedby
\in \isin	\vee \lor	\leftrightarrow \leftrightarrow	\iff \iff

[Try](#)[Docs](#)

v0.16.11

[Users](#)[GitHub](#)

$\Set{ x \mid x < \frac{1}{2} }$	$\set{x x<5}$
----------------------------------	---------------

Direct Input: $\forall : \exists | \in \notin \subset \wedge \vee \mapsto \leftarrow \rightarrow \neg \in \mathbb{C} \mathbb{H} \mathbb{N} \mathbb{P} \mathbb{Q} \mathbb{R}$

Macros

$x^2 + x^2$	<code>\def\foo{x^2} \foo + \foo</code>
$y^2 + y^2$	<code>\gdef\foo#1{#1^2} \foo{y} + \foo{y}</code>
	<code>\edef\macro#1#2...{definition to be expanded}</code>
	<code>\xdef\macro#1#2...{definition to be expanded}</code>
	<code>\let\foo=\bar</code>
	<code>\futurelet\foo\bar x</code>
	<code>\global\def\macro#1#2...{definition}</code>
	<code>\newcommand\macro[numargs]{definition}</code>
	<code>\renewcommand\macro[numargs]{definition}</code>
	<code>\providecommand\macro[numargs]{definition}</code>

Macros can also be defined in the KaTeX [rendering options](#).

Macros accept up to nine arguments: #1, #2, etc.

Macros defined by `\gdef`, `\xdef`, `\global\def`, `\global\edef`, `\global\let`, and `\global\futurelet` will persist between math expressions. (Exception: macro persistence may be disabled. There are legitimate security reasons for that.)

KaTeX has no `\par`, so all macros are long by default and `\long` will be ignored.

Available functions include:

```
\char  \mathchoice  \TextOrMath  \@ifstar  \@ifnextchar  \@firstoftwo  \@secondoftwo
\relax  \expandafter  \noexpand
```

@ is a valid character for commands, as if `\makeatletter` were in effect.

$\sum \backslash sum$	$\prod \backslash prod$	$\otimes \backslash bigotimes$	$\vee \backslash bigvee$
$\int \backslash int$	$\coprod \backslash coprod$	$\oplus \backslash bigoplus$	$\wedge \backslash bigwedge$
$\iint \backslash iint$	$\inttop \backslash inttop$	$\odot \backslash bigodot$	$\cap \backslash bigcap$
$\iiint \backslash iiint$	$\smallint \backslash smallint$	$\uplus \backslash biguplus$	$\cup \backslash bigcup$
$\oint \backslash oint$	$\oiint \backslash oiint$	$\oiint \backslash oiint$	$\sqcup \backslash bigsqcup$

Direct Input: $\int \iiint \oint \prod \coprod \sum \bigvee \bigwedge \bigcap \bigcup \biguplus \bigcup \sqcup$

Binary Operators

$+$ $+$	$\cdot \backslash cdot$	$\gg \backslash gtrdot$	$x \pmod{a} \times \backslash pmod{a}$
$-$ $-$	$\cdot \backslash cdotp$	$\intercal \backslash intercal$	$x(a) \times \backslash pod{a}$
$/$ $/$	$\centerdot \backslash centerdot$	$\wedge \backslash land$	$\triangleright \backslash rhd$
$*$ $*$	$\circ \backslash circ$	$\leftthreetimes \backslash leftthreetimes$	$\rightthreetimes \backslash rightthreetimes$
\amalg \amalg	$\circledast \backslash circledast$	$\ldotp \backslash ldotp$	$\rtimes \backslash rtimes$
\And \And	$\circledcirc \backslash circledcirc$	$\vee \backslash lor$	$\setminus \backslash setminus$
\last \last	$\circleddash \backslash circleddash$	$\lessdot \backslash lessdot$	$\smallsetminus \backslash smallsetminus$
\barwedge \barwedge	$\Cup \backslash Cup$	$\lhd \backslash lhd$	$\sqcap \backslash sqcap$
\bigcirc \bigcirc	$\cup \backslash cup$	$\ltimes \backslash ltimes$	$\sqcup \backslash sqcup$
\bmod \bmod	$\curlyvee \backslash curlyvee$	$x \bmod{a} \times \backslash mod{a}$	$\times \backslash times$
\boxdot \boxdot	$\curlywedge \backslash curlywedge$	$\mp \backslash mp$	$\unlhd \backslash unlhd$
\boxminus \boxminus	$\div \backslash div$	$\odot \backslash odot$	$\unrhd \backslash unrhd$
\boxplus \boxplus	$\divideontimes \backslash divideontimes$	$\ominus \backslash ominus$	$\uplus \backslash uplus$

	$\backslash doublebarwedge$		
\Cap $\backslash Cap$	\doublecap $\backslash doublecap$	\oslash $\backslash oslash$	\wedge $\backslash wedge$
\cap $\backslash cap$	\doublecup $\backslash doublecup$	\pm $\backslash pm$ or $\backslash plusmn$	\wr $\backslash wr$

Fractions and Binomials

$\frac{a}{b}$	$\frac{a}{b}$	$\left(\frac{a}{a+1}\right)$
$\frac{a}{b}$	$\frac{a}{b}$	$\overline{\frac{a}{b+1}}$
a/b		$\frac{a}{1 + \frac{1}{b}}$

$\binom{n}{k}$	$\dbinom{n}{k}$	$\left\{ \begin{matrix} n \\ k \end{matrix} \right\}$
$\{n \choose k\}$	$\tbinom{n}{k}$	$\left[\begin{matrix} n \\ k \end{matrix} \right]$

Math Operators

arcsin \arcsin	cosec \cosec	deg \deg	sec \sec
arccos \arccos	cosh \cosh	dim \dim	sin \sin
arctan \arctan	cot \cot	exp \exp	sinh \sinh
arctg \arctg	cotg \cotg	hom \hom	sh \sh
arcctg \arcctg	coth \coth	ker \ker	tan \tan
arg \arg	csc \csc	lg \lg	tanh \tanh
ch \ch	ctg \ctg	ln \ln	tg \tg
cos \cos	cth \cth	log \log	th \th

			\varinjlim
arg min \argmin	lim \lim	plim \plim	<u>lim</u> \varliminf
det \det	lim inf \liminf	Pr \Pr	<u>lim</u> \varlimsup
gcd \gcd	lim sup \limsup	proj lim \projlim	<u>lim</u> \varprojlim
inf \inf	max \max	sup \sup	
f \operatorname{f}	f \operatorname{f}		

Functions in the bottom six rows of this table can take `\limits`.

\sqrt

\sqrt{x} \sqrt{x}

$\sqrt[3]{x}$ \sqrt[3]{x}

Relations

$\stackrel{!}{=}$ \stackrel{!}{=} \stackrel{!}{\equiv}

= =	\doteqdot	\lessapprox	\smile
< <	\eqcirc	\lesseqtr	\sqsubset
> >	\eqcolon or \minuscolon	\lesseqgtr	\sqsubseteq
: :	\Eqcolon or \minuscoloncolon	\lessgtr	\sqsupset
\approx	\eqqcolon or \equalscolon	\lesssim	\sqsupseteq

Try	Docs	v0.16.11	Users	GitHub
\approxcoloncolon				\sub
\approx \approxeq	\geqslantgtr	\lllless	\subseteq \subsetreq or \sube	
\asymp \asymp	\eqslantless	\lt	\subseteq \subsetreqq	
\backepsilon \backepsilon	\equiv \equiv	\mid \mid	\succ \succ	
\backsimeq \backsimeq	\fallingdotseq	\models \models	\succapprox \succapprox	
\backsimeq \backsimeq	\frown \frown	\multimap \multimap	\succcurlyeq \succcurlyeq	
\between \between	\geq \geq	\bullet\bullet \origof	\succeq \succeq	
\bowtie \bowtie	\geqq \geqq	\owns \owns	\succsim \succsim	
\bumpeq \bumpeq	\geqq \geqq	\parallel \parallel	\Supset \Supset	
\Bumpeq \Bumpeq	\geqslant \geqslant	\perp \perp	\supset \supset	
\circeq \circeq	\gg \gg	\pitchfork \pitchfork	\supseteqq \supseteqq or \supe	
\colonapprox \colonapprox	\ggg \ggg	\prec \prec	\supseteqqq \supseteqqq	
\colonapprox \Colonapprox or \coloncolonapprox	\gggtr \gggtr	\precapprox \precapprox	\thickapprox \thickapprox	
\colon \colon or \colonminus	\gt \gt	\preccurlyeq \preccurlyeq	\thicksim \thicksim	
\colon \colon or \coloncolonminus	\gtapprox \gtapprox	\preceq \preceq	\trianglelefted \trianglelefted	
\colon \colon or \colonequals	\gtreqless \gtreqless	\precsim \precsim	\triangleq \triangleq	
\colon \colon or \coloncolon>equals	\gtreqqless \gtreqqless	\propto \propto	\trianglerighteq \trianglerighteq	
\colon \colon or \coloncolononequals	\gtless \gtless	\risingdotseq \risingdotseq	\varpropto \varpropto	

Try	Docs	v0.16.11	Users	GitHub
-	-	\shortparallel	\vartriangleleft	
\curlyeqprec	\in or \isin	\sim	\triangleright	\vartriangleright
\curlyeqsucc	\Join	\sim:	\vcntcolon or \ratio	
\dashv	\leq	\sim::	\vdash	\vdash
:: \dblcolon or \colon\colon	\leq	\simeq	\vDash	\vDash
\doteq	\leqq	\smallfrown	\Vdash	\Vdash
\Doteq	\leqslant	\smallsmile	\VvDash	\VvDash

Negated Relations

\neq \not =

$\not\approx \backslash gnapprox$	$\not\equiv \backslash ngeqslant$	$\not\subseteq \backslash nsubseteq$	$\not\asymp \backslash precneqq$
$\not\geq \backslash gneq$	$\not> \backslash ngtr$	$\not\subset \backslash nsubseteqq$	$\not\sim \backslash precsim$
$\not\leq \backslash gneqq$	$\not\leq \backslash nleq$	$\not\prec \backslash nsucc$	$\not\subsetneq \backslash subsetneq$
$\not\geq \backslash gnsim$	$\not\leq \backslash nleqq$	$\not\preccurlyeq \backslash nsuccceq$	$\not\subsetneqq \backslash subsetneqq$
$\not\geq \backslash gvertneqq$	$\not\leq \backslash nleqslant$	$\not\supset \backslash nsupseteq$	$\not\approx \backslash succnapprox$
$\not\approx \backslash lnapprox$	$\not< \backslash nless$	$\not\supseteq \backslash nsupseteqq$	$\not\approx \backslash succneqq$
$\not\leq \backslash lneq$	$\not\models \backslash nmid$	$\not\triangleleft \backslash ntriangleleft$	$\not\sim \backslash succnsim$
$\not\leq \backslash lneqq$	$\not\models \backslash notin$	$\not\trianglelefteq \backslash ntrianglelefteq$	$\not\supset \backslash supsetneq$
$\not\approx \backslash lnsim$	$\not\models \backslash notni$	$\not\triangleright \backslash ntriangleright$	$\not\supset \backslash supsetneqq$

Try	Docs	v0.16.11	Users	GitHub
$\neq \backslash neq$	$\nshortmid \backslash nshortmid$	$\nDash \backslash nVDash$	$\nsubseteq \backslash varsupsetneqq$	
$\ngeq \backslash ngeq$	$\nshortparallel \backslash nshortparallel$	$\nDash \backslash nVdash$		
$\ngeqq \backslash ngeqq$	$\nsim \backslash nsim$	$\napprox \backslash precnapprox$		

Direct Input: ↕↑↓←→

卷之三

Arrows

\circlearrowleft \cicleararrowleft	\leftharpoonup \leftharpoonup	\Rightarrow \rArr
\circlearrowright \cicleararrowright	\leftleftarrows \leftleftarrows	\rightarrow \rarr
\curvearrowleft \curvearrowleft	\leftrightarrow \leftrightarrow	\restriction \restriction
\curvearrowright \curvearrowright	\Leftrightarrow \Leftrightarrow	\rightarrow \rightarrow
\Downarrow \Darr	\leftrightsquigarrow \leftrightsquigarrow	\Rightarrow \Rightarrow
\Downarrow \dArr	\leftrightharpoons \leftrightharpoons	\rightarrowtail \rightarrowtail
\downarrow \darr	\rightsquigarrow \rightsquigarrow	\rightarrow \rightarrow
\dashleftarrow \dashleftarrow	\Leftarrow \Leftarrow	\rightarrowtail \rightarrowtail
\dashrightarrow \dashrightarrow	\longleftarrow \longleftarrow	\rightleftarrows \rightleftarrows
\downarrow \downarrow	\Longleftarrow \Longleftarrow	\rightleftharpoons \rightleftharpoons
\Downarrow \Downarrow	\longleftrightarrow \longleftrightarrow	\rightrightarrows \rightrightarrows
\Downarrow \downdownarrows	\Longleftrightarrow \Longleftrightarrow	\rightsquigarrow \rightsquigarrow
\downharpoonleft \downharpoonleft	\longmapsto \longmapsto	\Rightarrow \Rightarrow
\downharpoonright \downharpoonright	\longrightarrow \longrightarrow	\Rsh \Rsh
\leftarrow \gets	\Longrightarrow \Longrightarrow	\searrow \searrow
\Leftrightarrow \Harr	\looparrowleft \looparrowleft	\swarrow \swarrow
\Leftrightarrow \hArr	\looparrowright \looparrowright	\rightarrow \rightarrow

\iff	<code>\iff</code>	\Lsh	<code>\Lsh</code>	\uparrow	<code>\uArr</code>
\Leftarrow	<code>\impliedby</code>	\mapsto	<code>\mapsto</code>	\uparrow	<code>\uarr</code>
\implies	<code>\implies</code>	\nearrow	<code>\nearrow</code>	\uparrow	<code>\uparrowarrow</code>
\Leftarrowtail	<code>\Leftarrowtail</code>	\nleftarrow	<code>\nleftarrow</code>	\uparrow	<code>\Uparrow</code>
\Downarrowtail	<code>\Downarrowtail</code>	\nLeftarrow	<code>\nLeftarrow</code>	\downarrow	<code>\updownarrow</code>
\Downarrow	<code>\Downarrow</code>	\nleftrightarrow	<code>\nleftrightarrow</code>	\Downarrow	<code>\Updownarrow</code>
\leadsto	<code>\leadsto</code>	\nLeftrightarrow	<code>\nLeftrightarrow</code>	\upharpoonleft	<code>\upharpoonleft</code>
\leftarrowtail	<code>\leftarrowtail</code>	\nrightarrow	<code>\nrightarrow</code>	\upharpoonright	<code>\upharpoonright</code>
\Leftarrowtail	<code>\Leftarrowtail</code>	\nRightarrow	<code>\nRightarrow</code>	\upuparrows	<code>\upuparrows</code>
\leftarrowtailtail	<code>\leftarrowtailtail</code>	\nwarrow	<code>\nwarrow</code>		
\leftharpoonondown	<code>\leftharpoonondown</code>	\Rarr	<code>\Rarr</code>		

Extensible Arrows

Extensible arrows all can take an optional argument in the same manner as `\xrightarrow[under]{over}`.

Special Notation

Bra-ket Notation

$\langle \phi $ <code>\bra{\phi}</code>	$ \psi \rangle$ <code>\ket{\psi}</code>	$\langle \phi \psi \rangle$ <code>\braket{\phi \psi}</code>
$\langle \phi $ <code>\Bra{\phi}</code>	$ \psi \rangle$ <code>\Ket{\psi}</code>	$\left\langle \phi \middle \frac{\partial^2}{\partial t^2} \middle \psi \right\rangle$ <code>\Braket{ \phi \frac{\partial^2}{\partial t^2} \psi }</code>

Style, Color, Size, and Font

Class Assignment

```
\mathbin  \mathclose  \mathinner  \mathop
\mathopen  \mathord  \mathpunct  \mathrel
```

Color

$F = \textcolor{blue}{ma}$ `\color{blue} F=ma`

Note that `\color` acts like a switch. Other color functions expect the content to be a function argument:

$F = \textcolor{blue}{ma}$ `\textcolor{blue}{F=ma}`
 $F = \textcolor{#228B22}{ma}$ `\textcolor{#228B22}{F=ma}`
 $F = \textcolor{aqua}{ma}$ `\colorbox{aqua}{$F=ma$}`
 $F = \textcolor{red}{ma}$ `\fcolorbox{red}{aqua}{$F=ma$}`

Note that, as in LaTeX, `\colorbox` & `\fcolorbox` renders its third argument as text, so you may want to switch back to math mode with `$` as in the examples above.

For color definition, KaTeX color functions will accept the standard HTML [predefined color names](#). They will also accept an RGB argument in CSS hexadecimal style. The "#" is optional before a six-digit specification.

Font

[Try](#)[Docs](#)

v0.16.11

Users

GitHub

<code>\text{AB}</code>	<code>\boldsymbol{AB}</code>	<code>\mathbb{AB}</code>
<code>\textnormal{AB}</code>	<code>\boldsymbol{AB}</code>	<code>\mathbb{AB}</code>
<code>\textit{AB}</code>	<code>\boldsymbol{\textit{AB}}</code>	<code>\mathbb{\textit{AB}}</code>
<code>\textmd{AB}</code>	<code>\boldsymbol{\textmd{AB}}</code>	<code>\mathbb{\textmd{AB}}</code>
<code>\texttt{AB}</code>	<code>\boldsymbol{\texttt{AB}}</code>	<code>\mathbb{\texttt{AB}}</code>
<code>\textit{AB}</code>	<code>\boldsymbol{\textit{AB}}</code>	<code>\mathbb{\textit{AB}}</code>
<code>\textbf{AB}</code>	<code>\boldsymbol{\textbf{AB}}</code>	<code>\mathbb{\textbf{AB}}</code>
<code>\textsf{AB}</code>	<code>\boldsymbol{\textsf{AB}}</code>	<code>\mathbb{\textsf{AB}}</code>
<code>\textit{AB}</code>	<code>\boldsymbol{\textit{AB}}</code>	<code>\mathbb{\textit{AB}}</code>
<code>\texttt{AB}</code>	<code>\boldsymbol{\texttt{AB}}</code>	<code>\mathbb{\texttt{AB}}</code>
<code>\textit{AB}</code>	<code>\boldsymbol{\textit{AB}}</code>	<code>\mathbb{\textit{AB}}</code>
<code>\textbf{AB}</code>	<code>\boldsymbol{\textbf{AB}}</code>	<code>\mathbb{\textbf{AB}}</code>
<code>\textsf{AB}</code>	<code>\boldsymbol{\textsf{AB}}</code>	<code>\mathbb{\textsf{AB}}</code>

One can stack font family, font weight, and font shape by using the `\textXX` versions of the font functions. So `\textsf{\textbf{H}}` will produce **H**. The other versions do not stack, e.g., `\mathsf{\mathbf{H}}` will produce **H**.

In cases where KaTeX fonts do not have a bold glyph, `\pmb` can simulate one. For example, `\pmb{\mu}` renders as : μ

Size

AB <code>\Huge AB</code>	AB <code>\normalsize AB</code>
AB <code>\huge AB</code>	AB <code>\small AB</code>
AB <code>\LARGE AB</code>	AB <code>\footnotesize AB</code>
AB <code>\Large AB</code>	AB <code>\scriptsize AB</code>
AB <code>\large AB</code>	AB <code>\tiny AB</code>

Style

[Try](#)[Docs](#)

v0.16.11

[Users](#)[GitHub](#)

<code>x \scriptstyle x</code>	(The size of a first sub/superscript)
<code>x \scriptscriptstyle x</code>	(The size of subsequent sub/superscripts)
<code>lim_x \lim\limits_x</code>	
<code>lim_x \lim\nolimits_x</code>	
<code>x^2 \verb!x^2!</code>	

`\text{...}` will accept nested `$...$` fragments and render them in math mode.

Symbols and Punctuation

<code>% comment</code>	<code>... \dots</code>	KATEX \KaTeX
<code>% \%</code>	<code>... \cdots</code>	LATEX \LaTeX
<code># \#</code>	<code>... \ddots</code>	TEX \TeX
<code>& \&</code>	<code>... \ldots</code>	∇ \nabla
<code>_ _</code>	<code>... \vdots</code>	∞ \infty
<code>_ \text{\textunderscore}</code>	<code>... \dotsb</code>	∞ \infin
<code>-- \text{\textendash}</code>	<code>... \dotsc</code>	✓ \checkmark
<code>— \text{\textendash}</code>	<code>... \dotsi</code>	\dagger \dag
<code>— \text{\textemdash}</code>	<code>... \dotsm</code>	\ddagger \dagger
<code>— \text{\textemdash}</code>	<code>... \dotso</code>	\ddagger \text{\textdagger}
<code>~ \text{\textasciitilde}</code>	<code>... \sdot</code>	\ddagger \ddag
<code>^ \text{\textasciicircum}</code>	<code>... \mathellipsis</code>	\ddagger \ddagger
<code>' \text{\textellipsis}</code>	<code>... \text{\textellipsis}</code>	\ddagger \text{\textdaggerdbl}
<code>' \text{\textquotleft}</code>	<code>\Box</code>	\ddagger \Dagger
<code>' \text{\textquotright}</code>	<code>\square</code>	\angle \angle

Try	Docs	v0.16.11	Users	GitHub
" "	▷ \triangleleft	⊥ \bot		
"\text{\textquotedblright}	▷ \triangleright	\$ \\$		
:	▽ \bigtriangledown	\$ \text{\textdollar}		
\backprime	△ \bigtriangleup	£ \pounds		
' \prime	▲ \blacktriangle	£ \mathsterling		
< \text{\textless}	▼ \blacktriangledown	£ \text{\textsterling}		
> \text{\textgreater}	◀ \blacktriangleleft	¥ \yen		
\text{\textbar}	▶ \blacktriangleright	✓ \surd		
\text{\textbardbl}	◊ \diamond	° \degree		
{ \text{\textbraceleft}	◊ \Diamond	° \text{\textdegree}		
} \text{\textbraceright}	◊ \lozenge	⌚ \mho		
\ \text{\textbackslash}	◆ \blacklozenge	＼ \diagdown		
¶ \text{\P} or \P	★ \star	／ \diagup		
§ \text{\S} or \S	★ \bigstar	♭ \flat		
§ \text{\sect}	♣ \clubsuit	♮ \natural		
(C) \copyright	♣ \clubs	# \sharp		
(R) \circledR	◊ \diamondsuit	♡ \heartsuit		
(R) \text{\textregistered}	◊ \diamonds	♡ \hearts		
(S) \circledS	♠ \spadesuit	♠ \spades		
(a) \text{\textcircled a}	⊗ \maltese	⊖ \minuso		

Direct Input: § ¶ £ ¥ ▽ ∞ · ∠ ∠ ◀ ♠ ♥ ◊ ♣ ♦ ♤ ✓ ... ; ... : .. ! !! ⊖

Units

[Try](#)[Docs](#)

v0.16.11

[Users](#)[GitHub](#)

em	CSS em	bp	1/72 inch × F × G
ex	CSS ex	pc	12 KaTeX pt
mu	1/18 CSS em	dd	1238/1157 KaTeX pt
pt	1/72.27 inch × F × G	cc	14856/1157 KaTeX pt
mm	1 mm × F × G	nd	685/642 KaTeX pt
cm	1 cm × F × G	nc	1370/107 KaTeX pt
in	1 inch × F × G	sp	1/65536 KaTeX pt

where:

$$F = (\text{font size of surrounding HTML text})/(10 \text{ pt})$$

$G = 1.21$ by default, because KaTeX font-size is normally $1.21 \times$ the surrounding font size.
This value [can be overridden](#) by the CSS of an HTML page.

The effect of style and size:

Unit	textstyle	scriptscript	huge
em or ex			
mu			
others			

[← FONT](#)[SUPPORT TABLE →](#)[Docs](#)[Community](#)[More](#)[Installation](#)[Who is using KaTeX?](#)[Contributors](#)[Usage](#)[GitHub Discussions](#)

Star 18,042



Try

Docs

v0.16.11

Users

GitHub



Licensed under MIT License

Copyright © 2024 Khan Academy and other contributors