The `physics-patch` package

Improved version of the physics package

Willie Shen (Willie169)

Version 2.4

Last update: Feb. 17, 2025

Contents

1 Preface

Since version 2.0, the `physics-patch` package has evolved from merely patching the `physics` package to fully replacing it, covering all its commands. While preserving the original goal—simplifying mathematical and physics typesetting for greater readability and efficiency—this package refines the design by addressing unconventional behaviors, extending commands, and introducing additional macros.

Like the original, this package provides commands with intuitive names and well-defined shorthands, ensuring both clarity and ease of recall.

This package resolved the unintuitive definitions and behaviors in `physics` without changing the command names and intended behaviors. For instance, in the original package, suffix parentheses and their contents in expressions like `\dv{f}{x}(\typical)` are ignored.

Beyond refining existing functionality, this package extends commands for broader applicability—such as enabling `\xmat` to support ellipses—and introduces entirely new macros, such as `\omat`.

2 Usage

2.1 Required packages

The `physics-patch` package requires `amsmath`, `etoolbox`, `xcolor`, `xparse`, and `xstring` package to work properly in your LaTeX document. If you are unsure whether you've had them installed, you can either install it again using your local package manager (comes with most distributions) or by visiting the CTAN online package database, or even just try to use `physics-patch` package without worrying about it. Many modern LaTeXcompilers will locate and offer to download missing required packages for you.

2.2 Using physics-patch in your LaTeX document

To use `physics-patch` in your LaTeX document, simply insert `\usepackage{physics-patch}` in the preamble of your document, before `\begin{document}` and after `\documentclass{class}`:

```
\documentclass{class}
...
\usepackage{physics-patch}
...
\begin{document}
content...
\end{document}
```

- `physics-patch` has covered all commands in `physics` since version 2.0, so there's no need to load `physics`.
- It is ok to load `physics` before this package. This package will silently overrides macros in `physics` with an improved version. To use the original version provided by `physics`, load `physics` before this package and use the `nooverride` option (not recommended). `nooverride` falls back to `override` if physics is not loaded.

- This package pretends that `physics` package is loaded so that this package won't be overriden if loading `physics` is called afterward and packages loaded afterward that checks whether `physics` is loaded to determine its behavior (e.g. `siunitx`) work correctly. To disable this, use the `nopretend` option (not recommended).
- If `siuitx` is loaded before this package, this package will define `\ITquantity` and `\ITqty` as the integration of the improved definition of `physics`'s `\qty` (in `\PHquantity` and `\PHqty`) and `siuitx`'s `\SI`. You can optionally set `siintegrate` option to override `\PTquantity` and `\PTqty` with `\ITqty` (not recommended). `siintegrate` falls back to `nosiintegrate` if `siunitx` is not loaded.

3 Communication Channels

- **Bug tracker:** https://github.com/Willie169/physics-patch/issues.
- **Announcements:** https://github.com/Willie169/physics-patch/releases.
- **Repository:** https://github.com/Willie169/physics-patch.

4 License and Credit

- This package is released under the **LaTeX Project Public License (LPPL) 1.3c.**
  See https://www.latex-project.org/lppl/lppl-1-3c for the details of that license.
- Many parts of this package are modified from the `physics` package, created by **Sergio C. de la Barrera** and licenced under **LPPL 1.3**.
  See https://ctan.org/pkg/physics for the details of that package.

5 List of Commands

In the commands listed below, the left column is long-form names with non-default alternate names (if any), the middle column is default shorthand commands with detailed syntaxes and explanations.
Commands that have different definitions come with `PT` in the beginning of their name (e.g. `\PTmqty`). If `nooverride` is not used or the `physics` package is not loaded before this package, the commands without `PT` will be silent overriden to be the same as the ones with `PT`.

5.1 Automatic bracing

`\PTquantity` or `\PHquantity` or `\PHqty`	`\PTqty(\typical)` → (🟦)	automatic ( ) braces
	`\PTqty(\tall)` → (🟦)	
	`\PTqty(\grande)` → (🟦)	
	`\PTqty[\typical]` → [🟦]	automatic [ ] braces
	`\PTqty|\typical|` → |🟦|	automatic | | braces

	`\PTqty{\typical}` → {🟦}	automatic { } braces
	`\PTqty\big{}` → $\big\{\big\}$	manual sizing (works with any of the above bracket types)
	`\PTqty\Big{}` → $\Big\{\Big\}$	
	`\PTqty\bigg{}` → $\bigg\{\bigg\}$	
	`\PTqty\Bigg{}` → $\Bigg\{\Bigg\}$	
	`\pqty{}` ↔ `\PTqty()`	alternative syntax; robust and more LaTeX-friendly
	`\bqty{}` ↔ `\PTqty[]`	
	`\vqty{}` ↔ `\PTqty||`	
	`\Bqty{}` ↔ `\PTqty{}`	
`\absolutevalue`	`\abs{a}` → $\lvert a\rvert$	automatic sizing; equivalent to `\PTqty|a|`
	`\abs\Big{a}` → $\Big\lvert a\Big\rvert$	inherits manual sizing syntax from `\PTqty`
	`\abs*{\grande}` → |🟦|	star for no resize
`\norm`	`\norm{a}` → $\lVert a\rVert$	automatic sizing
	`\norm\Big{a}` → $\Big\lVert a\Big\rVert$	manual sizing
	`\norm*{\grande}` → ‖🟦‖	star for no resize
`\evaluated`	`\eval{x}_0^\infty` → $x\Big\rvert_0^\infty$	vertical bar for evaluation limits
	`\eval(x|_0^\infty` → $\left(x\right\rvert_0^\infty$	alternate form
	`\eval[x|_0^\infty` → $\left[x\right\rvert_0^\infty$	alternate form
	`\eval[\venti|_0^\infty` → [🟦|$_0^\infty$	automatic sizing
	`\eval*[\venti |_0^\infty` → [🟦|$_0^\infty$	star for no resize
`\order`	`\order{x^2}` → $\mathcal{O}\left(x^2\right)$	order symbol; automatic sizing and space handling
	`\order\Big{x^2}` → $\mathcal{O}\Big(x^2\Big)$	manual sizing
	`\order*{\grande}` → $\mathcal{O}($🟦$)$	star for no resize
`\commutator`	`\comm{A}{B}` → $[A, B]$	automatic sizing
	`\comm\Big{A}{B}` → $\Big[A, B\Big]$	manual sizing
	`\comm*{A}{\grande}` → $[A,$ 🟦$]$	star for no resize
`\anticommutator` or `\acommutator`	`\acomm{A}{B}` → $\{A, B\}$	same as `\poissonbracket`
`\poissonbracket`	`\pb{A}{B}` → $\{A, B\}$	same as `\anticommutator`

5.2 Vector notation

The default del symbol ∇ used in `physics-patch` vector notation can be switched to appear with an arrow $\vec{\nabla}$ by including the option `arrowdel` in the document preamble:

```
\usepackage[arrowdel]{physics-patch}
```

`\vectorbold`	`\vb{a}` → $\mathbf{a}$ `\vb*{a}`, `\vb*{\theta}` → $\boldsymbol{a}$, $\boldsymbol{\theta}$	upright/no Greek italic/Greek
`\vectorarrow`	`\va{a}` → $\vec{\mathbf{a}}$ `\va*{a}`, `\va*{\theta}` → $\vec{\boldsymbol{a}}$, $\vec{\boldsymbol{\theta}}$	upright/no Greek italic/Greek
`\vectorunit`	`\vu{a}` → $\hat{\mathbf{a}}$ `\vu*{a}`, `\vu*{\theta}` → $\hat{\boldsymbol{a}}$, $\hat{\boldsymbol{\theta}}$	upright/no Greek italic/Greek
`\dotproduct`	`\vdot` → $\cdot$ as in $\mathbf{a}\cdot\mathbf{b}$	note: `\dp` is a protected TeX primitive
`\crossproduct`	`\cross` → $\times$ as in $\mathbf{a}\times\mathbf{b}$ `\cp` → $\times$ as in $\mathbf{a}\times\mathbf{b}$	alternate name shorthand name
`\gradient`	`\grad` → ∇ `\grad{\Psi}` → $\nabla\Psi$ `\grad(\Psi+\tall)` → $\nabla\left(\Psi+\blacksquare\right)$ `\grad[\Psi+\tall]` → $\nabla\left[\Psi+\blacksquare\right]$	 default mode long-form (like `\PTqty` but also handles spacing)
`\divisionsymbol`	`\divisionsymbol` → $\div$	
`\divergence`	`\divg` → $\nabla\cdot$ `\divg{\vb{a}}` → $\nabla\cdot\mathbf{a}$ `\divg(\vb{a}+\tall)` → $\nabla\cdot\left(\mathbf{a}+\blacksquare\right)$ `\divg[\vb{a}+\tall]` → $\nabla\cdot\left[\mathbf{a}+\blacksquare\right]$	note: if `nooriginaldiv` option is used, `\div` will be overriden as $\nabla\cdot$ too (not recommended) default mode long-form
`\curl`	`\curl` → $\nabla\times$ `\curl{\vb{a}}` → $\nabla\times\mathbf{a}$ `\curl(\vb{a}+\tall)` → $\nabla\times\left(\mathbf{a}+\blacksquare\right)$ `\curl[\vb{a}+\tall]` → $\nabla\times\left[\mathbf{a}+\blacksquare\right]$	 default mode long-form
`\laplacian`	`\laplacian` → ∇^2 `\laplacian{\Psi}` → $\nabla^2\Psi$	 default mode

	`\laplacian(\Psi+\tall)` $\to \nabla^2\left(\Psi + \blacksquare\right)$ `\laplacian[\Psi+\tall]` $\to \nabla^2\left[\Psi + \blacksquare\right]$	long-form

5.3 Operators

The standard set of trig functions is redefined in `physics-patch` to provide automatic braces that behave like `\PTqty()`. In addition, an optional power argument is provided. This behavior can be switched off by including the option `notrig` in the preamble:

```
\usepackage[notrig]{physics-patch}
```

Example trig redefinitions:		
`\sin`	`\sin(\grande)` $\to \sin\left(\blacksquare\right)$	automatic braces; old `\sin` renamed `\sine`
	`\sin[2](x)` $\to \sin^2(x)$	optional power
	`\sin x` $\to \sin x$	can still use without an argument

The full set of available trig functions in `physics-patch` includes:

`\sin(x)`	`\sinh(x)`	`\arcsin(x)`	`\asin(x)`
`\cos(x)`	`\cosh(x)`	`\arccos(x)`	`\acos(x)`
`\tan(x)`	`\tanh(x)`	`\arctan(x)`	`\atan(x)`
`\csc(x)`	`\csch(x)`	`\arccsc(x)`	`\acsc(x)`
`\sec(x)`	`\sech(x)`	`\arcsec(x)`	`\asec(x)`
`\cot(x)`	`\coth(x)`	`\arccot(x)`	`\acot(x)`

⇒

$$\begin{array}{cccc}
\sin(x) & \sinh(x) & \arcsin(x) & \mathrm{asin}(x) \\
\cos(x) & \cosh(x) & \arccos(x) & \mathrm{acos}(x) \\
\tan(x) & \tanh(x) & \arctan(x) & \mathrm{atan}(x) \\
\csc(x) & \mathrm{csch}(x) & \mathrm{arccsc}(x) & \mathrm{acsc}(x) \\
\sec(x) & \mathrm{sech}(x) & \mathrm{arcsec}(x) & \mathrm{asec}(x) \\
\cot(x) & \coth(x) & \mathrm{arccot}(x) & \mathrm{acot}(x)
\end{array}$$

The standard trig functions (plus a few that are missing in `amsmath`) are available without any automatic bracing under a new set of longer names:

`\sine`	`\hypsine`	`\arcsine`	`\asine`
`\cosine`	`\hypcosine`	`\arccosine`	`\acosine`

`\tangent`	`\hyptangent`	`\arctangent`	`\atangent`
`\cosecant`	`\hypcosecant`	`\arccosecant`	`\acosecant`
`\secant`	`\hypsecant`	`\arcsecant`	`\asecant`
`\cotangent`	`\hypcotangent`	`\arccotangent`	`\acotangent`

Similar behavior has also been extended to the following functions:

`\exp(\tall)`	exp(🟦)	old definitions ⇒	`\exponential`
`\log(\tall)`	log(🟦)		`\logarithm`
`\ln(\tall)`	ln(🟦)		`\naturallogarithm`
`\det(\tall)`	det(🟦)		`\determinant`
`\Pr(\tall)`	Pr(🟦)		`\Probability`

New operators:		
`\trace` or `\tr`	`\tr\rho` → tr ρ also `\tr(\tall)` → tr(🟦)	trace; same bracing as trig functions
`\Trace` or `\Tr`	`\Tr\rho` → Tr ρ	alternate
`\rank`	`\rank M` → rank M	matrix rank
`\erf`	`\erf(x)` → erf(x)	Gauss error function
`\Res`	`\Res[f(z)]` → Res[$f(z)$]	residue; same bracing as trig functions
`\principalvalue`	`\pv{\int f(z) \dd{z}}` → $\mathcal{P}\int f(z)\,\mathrm{d}z$ `\PV{\int f(z) \dd{z}}` → P.V. $\int f(z)\,\mathrm{d}z$	Cauchy principal value alternate

5.4 Quick quad text

This set of commands produces text in math-mode padded by `\quad` spacing on either side. This is meant to provide a quick way to insert simple words or phrases in a sequence of equations. Each of the following commands includes a starred version which pads the text only on the right side with `\quad` for use in aligned environments such as `cases`.

General text:		
`\qqtext`	`\qq{}` `\qq{word or phrase}` →␣word or phrase␣	general quick quad text with argument normal mode; left and right `\quad`

	`\qq*{word or phrase}` → word or phrase␣	starred mode; right `\quad` only

Special macros:	
`\qcomma` or `\qc` →,␣	right `\quad` only
`\qcc` →␣c.c.␣	complex conjugate; left and right `\quad` unless starred `\qcc*` → c.c.␣
`\qif` →␣if␣	left and right `\quad` unless starred `\qif*` → if␣

Similar to `\qif`:

`\qthen`, `\qelse`, `\qotherwise`, `\qunless`, `\qgiven`, `\qusing`, `\qassume`, `\qsince`, `\qlet`
`\qfor`, `\qall`, `\qeven`, `\qodd`, `\qinteger`, `\qand`, `\qor`, `\qas`, `\qin`

5.5 Derivatives

The default differential symbol d which is used in `\differential` and `\derivative` can be switched to an italic form *d* by including the option `italicdiff` in the preamble:

```
\usepackage[italicdiff]{physics-patch}
```

`\differential`	`\dd` → d	
	`\dd x` → $\mathrm{d}x$	no spacing (not recommended)
	`\dd{x}` → ␣$\mathrm{d}x$␣	automatic spacing based on neighbors
	`\dd[3]{x}` → d^3x	optional power
	`\dd(\cos\theta)` → $\mathrm{d}(\cos\theta)$	long-form; automatic braces
`\PTderivative`	`\PTdv{x}` → $\frac{\mathrm{d}}{\mathrm{d}x}$	one argument
	`\PTdv{f}{x}` → $\frac{\mathrm{d}f}{\mathrm{d}x}$	two arguments
	`\PTdv[n]{f}{x}` → $\frac{\mathrm{d}^n f}{\mathrm{d}x^n}$	optional power
	`\PTdv{x}(\grande)` → $\frac{\mathrm{d}}{\mathrm{d}x}$(🟦)(🟦)	long-form; automatic braces, spacing
	`\PTdv*{f}{x}` → $\mathrm{d}f\big/\mathrm{d}x$	inline form using `\flatfrac`
`\PTpartialderivative` or `\PTpderivative`	`\PTdv{f}{x}(\grande)` → $\frac{\mathrm{d}f}{\mathrm{d}x}$(🟦)	note: in original physics package, `\dv{f}{x}(\grande)` → $\frac{\mathrm{d}f}{\mathrm{d}x}$
	`\PTpdv{x}` → $\frac{\partial}{\partial x}$	shorthand name
	`\PTpdv{f}{x}` → $\frac{\partial f}{\partial x}$	two arguments
	`\PTpdv[n]{f}{x}` → $\frac{\partial^n f}{\partial x^n}$	optional power
	`\PTpdv{x}(\grande)` → $\frac{\partial}{\partial x}$(🟦)	long-form

	`\PTpdv{f}{x}{y}` → $\frac{\partial^2 f}{\partial x \partial y}$ `\PTpdv*{f}{x}` → $\partial f \big/ \partial x$ `\PTpdv{f}{x}(\grande)` → $\frac{\partial f}{\partial x}\left(\blacksquare\right)$	mixed partial inline form using `\flatfrac` note: in original `physics` package, `\pdv{f}{x}(\grande)` → $\frac{\partial f}{\partial x}$
`\variation`	`\var{F[g(x)]}` → $\delta F[g(x)]$ `\var(E-TS)` → $\delta(E-TS)$	functional variation (works like `\dd`) long-form
`\functionalderivative`	`\fdv{g}` → $\frac{\delta}{\delta g}$ `\fdv{F}{g}` → $\frac{\delta F}{\delta g}$ `\fdv{V}(E-TS)` → $\frac{\delta}{\delta V}(E-TS)$ `\fdv*{F}{x}` → $\delta F \big/ \delta x$	functional derivative (works like `\PTdv`) long-form inline form using `\flatfrac`

5.6 Dirac bra-ket notation

The following collection of macros for Dirac notation contains two fundamental commands, `\bra` and `\ket`, along with a set of more specialized macros which are essentially combinations of the fundamental pair. The fundamental commands are designed to contract with one another algebraically when appropriate and are thus suggested for general use. For instance, the following code renders correctly[1]

$$\texttt{\bra\{\phi\}\ket\{\psi\}} \rightarrow \langle\phi|\psi\rangle \quad \text{as opposed to} \quad \langle\phi|\psi\rangle$$

whereas a similar construction with higher-level macros will not contract in a robust manner

$$\texttt{\bra\{\phi\}\dyad\{\psi\}\{\xi\}} \rightarrow \langle\phi|\,|\psi\rangle\!\langle\xi|\,.$$

On the other hand, the correct output can be generated by sticking to the fundamental commands,

$$\texttt{\bra\{\phi\}\ket\{\psi\}\bra\{\xi\}} \rightarrow \langle\phi|\psi\rangle\,\langle\xi|$$

allowing the user to type out complicated quantum mechanical expressions without worrying about bra-ket contractions. That being said, the high-level macros do have a place in convenience and readability, as long as the user is aware of rendering issues that may arise due to an absence of automatic contractions.

`\ket`	`\ket{\tall}` → $\left\vert\blacksquare\right\rangle$ `\ket*{\tall}` → $\vert\blacksquare\rangle$	automatic sizing no resize
`\bra`	`\bra{\tall}` → $\left\langle\blacksquare\right\vert$ `\bra*{\tall}` → $\langle\blacksquare\vert$	automatic sizing no resize

---

[1] Note the lack of a space between the bra and ket commands. This is necessary in order for the bra to find the corresponding ket and form a contraction.

	`\bra{\phi}\ket{\psi}` → $\langle\phi|\psi\rangle$	automatic contraction
	`\bra{\phi}\ket{\tall}` → $\left\langle\phi\middle|\blacksquare\right\rangle$	contraction inherits automatic sizing
	`\bra{\phi}\ket*{\tall}` → $\langle\phi|\blacksquare\rangle$	a star on either term in the contraction prohibits resizing
	`\bra*{\phi}\ket{\tall}` → $\langle\phi|\blacksquare\rangle$	
	`\bra*{\phi}\ket*{\tall}` → $\langle\phi|\blacksquare\rangle$	
`\innerproduct`	`\braket{a}{b}` → $\langle a|b\rangle$	two-argument braket
	`\braket{a}` → $\langle a|a\rangle$	one-argument (norm)
	`\braket{a}{\tall}` → $\left\langle a\middle|\blacksquare\right\rangle$	automatic sizing
	`\braket*{a}{\tall}` → $\langle a|\blacksquare\rangle$	no resize
	`\ip{a}{b}` → $\langle a|b\rangle$	shorthand name
`\outerproduct`	`\dyad{a}{b}` → $|a\rangle\langle b|$	two-argument dyad
	`\dyad{a}` → $|a\rangle\langle a|$	one-argument (projector)
	`\dyad{a}{\tall}` → $\left|a\right\rangle\left\langle\blacksquare\right|$	automatic sizing
	`\dyad*{a}{\tall}` → $|a\rangle\langle\blacksquare|$	no resize
	`\ketbra{a}{b}` → $|a\rangle\langle b|$	alternative name
	`\op{a}{b}` → $|a\rangle\langle b|$	shorthand name
`\expectationvalue`	`\expval{A}` → $\langle A\rangle$	implicit form
	`\expval{A}{\Psi}` → $\langle\Psi|A|\Psi\rangle$	explicit form
	`\ev{A}{\Psi}` → $\langle\Psi|A|\Psi\rangle$	shorthand name
	`\ev{\grande}{\Psi}` → $\langle\Psi|\blacksquare|\Psi\rangle$	default sizing ignores middle argument
	`\ev*{\grande}{\tall}` → $\langle\blacksquare|\blacksquare|\blacksquare\rangle$	single star does no resizing whatsoever
	`\ev**{\grande}{\Psi}` → $\left\langle\Psi\middle|\blacksquare\middle|\Psi\right\rangle$	double star resizes based on all parts
`\matrixelement`	`\matrixel{n}{A}{m}` → $\langle n|A|m\rangle$	requires all three arguments
	`\mel{n}{A}{m}` → $\langle n|A|m\rangle$	shorthand name
	`\mel{n}{\grande}{m}` → $\langle n|\blacksquare|m\rangle$	default sizing ignores middle argument
	`\mel*{n}{\grande}{\tall}` → $\langle n|\blacksquare|\blacksquare\rangle$	single star does no resizing whatsoever

	`\mel**{n}{\grande}{m}` → $\left\langle n \middle| \blacksquare \middle| m \right\rangle$	double star resizes based on all parts

5.7 Matrix macros

Note: `\mqty` and `\smqty` in `physics` uses `\mathord`, while `\PTmqty` and `\PTsmqty` in `physics-patch` don't.

```
\begin{pmatrix}
\imat{2} \\ a & b
\end{pmatrix}
```

$$\Rightarrow \begin{pmatrix}1&0\\0&1\\a&b\end{pmatrix}$$

The following matrix macros produce unformatted rows and columns of matrix elements for use as separate matrices as well as blocks within larger matrices. For example, the command `\identitymatrix{2}` which has also has the shortcut `\imat{2}` produces the elements of a 2 × 2 identity matrix $\begin{smallmatrix}1&0\\0&1\end{smallmatrix}$ without braces or grouping. This allows the command to also be used within another matrix, as in: To specify elements on the right of left sides of our `\imat{2}` sub-matrix we use the grouping command `\PTmatrixquantity` or `\PTmqty` to effectively convert `\imat{2}` into a single matrix element of a larger matrix:

```
\begin{pmatrix}
\PTmqty{\imat{2}} & \PTmqty{a\\b} \\ \PTmqty{c & d} &
    e
\end{pmatrix}
```

$$\Rightarrow \begin{pmatrix}1&0&a\\0&1&b\\c&d&e\end{pmatrix}$$

The extra `\PTmqty` groups were required in this case in order to get the a and b elements to behave as a single element, since `\PTmqty{\imat{2}}` also acts like a single matrix element (the same can be said of the grouped c and d elements). Finally, the outermost `pmatrix` environment could have also been replaced with the `physics-patch` macro `\PTmqty()`, allowing the above example to be written on one line:

```
\PTmqty(\PTmqty{\imat{2}} & \PTmqty{a\\b} \\ \PTmqty{
    c & d} & e)
```

$$\Rightarrow \begin{pmatrix}1&0&a\\0&1&b\\c&d&e\end{pmatrix}$$

`\PTmatrixquantity`	`\PTmqty{a & b \\ c & d}` → $\begin{matrix}a&b\\c&d\end{matrix}$	groups a set of matrix elements into a single object
	`\PTmqty(a & b \\ c & d)` → $\begin{pmatrix}a&b\\c&d\end{pmatrix}$	parentheses
	`\PTmqty*(a & b \\ c & d)` → $\begin{pmatrix}a&b\\c&d\end{pmatrix}$	alternate parentheses
	`\PTmqty[a & b \\ c & d]` → $\begin{bmatrix}a&b\\c&d\end{bmatrix}$	square brackets
	`\PTmqty|a & b \\ c & d|` → $\begin{vmatrix}a&b\\c&d\end{vmatrix}$	vertical bars

	`\omqty{}` ↔ `\PTmqty{}`
	`\pmqty{}` ↔ `\PTmqty()`
	`\PTpmqty{}` ↔ `\PTmqty()`
	`\PTpmqty*{}` ↔ `\PTmqty*{}`
	`\Pmqty{}` ↔ `\PTmqty*()`
	`\bmqty{}` ↔ `\PTmqty[]`
	`\vmqty{}` ↔ `\PTmqty||`
		alternative syntax; robust and more LaTeX-friendly

`\PTsmallmatrixquantity`	`\PTsmqty{a & b \\ c & d}` → $\begin{smallmatrix}a&b\\c&d\end{smallmatrix}$	the `smallmatrix` form of `\PTmqty`
	`\PTsmqty()` or `\spmqty{}` or `\PTspmqty{}`	small version of `\PTmqty()`

	`\PTsmqty*()` or `\sPmqty{}` or `\PTspmqty*{}`
	`\PTsmqty[]` or `\sbmqty{}`
	`\PTsmqty||` or `\svmqty{}`
		small version of `\PTmqty*()`
		small version of `\PTmqty[]`
		small version of `\PTmqty||`

`\matrixdeterminant`	`\mdet{a & b \\ c & d}` → $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$ `\smdet{a & b \\ c & d}` → $\left\lvert\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right\rvert$	matrix determinant small matrix determinant
`\identitymatrix`	`\imat{n}` `\PTmqty(\imat{3})` → $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	elements of $n \times n$ identity matrix formatted with `\PTmqty` or `\PTsmqty`

`\PTxmatrix`	`\PTxmat{x}{n}{m}`
	`\PTmqty(\PTxmat{x}{3}{3})` → $\begin{pmatrix} x & x & x \\ x & x & x \\ x & x & x \end{pmatrix}$
	`\PTmqty(\PTxmat{x}{}{3})` → $\begin{pmatrix} x & x & x \end{pmatrix}$
	`\PTmqty(\PTxmat{x}{3}{})` → $\begin{pmatrix} x \\ x \\ x \end{pmatrix}$
	`\PTxmat*{x}{n}{m}`
	`\PTmqty(\PTxmat*{x}{3}{3})` → $\begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}$
	`\PTmqty(\PTxmat*{x}{1}{3})` → $\begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}$
	`\PTmqty(\PTxmat*{x}{3}{1})` → $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$
	`\PTxmat{x}{n}{m}[p]`
	`\PTmqty(\PTxmat{x}{5}{3}[3])` → $\begin{pmatrix} x & x & x \\ \vdots & \vdots & \vdots \\ x & x & x \end{pmatrix}$
	`\PTmqty(\PTxmat{x}{3}{3}[3])` → $\begin{pmatrix} x & x & x \\ x & x & x \\ x & x & x \end{pmatrix}$
	`\PTmqty(\PTxmat{x}{}{3}[3])` → $\begin{pmatrix} x & x & x \\ \vdots & \vdots & \vdots \\ x & x & x \end{pmatrix}$
	`\PTxmat{x}{n}{m}[p][q]`
	`\PTmqty(\PTxmat{x}{5}{5}[3][3])` → $\begin{pmatrix} x & \ldots & x \\ \vdots & \ddots & \vdots \\ x & \ldots & x \end{pmatrix}$
	`\PTmqty(\PTxmat{x}{5}{3}[3][3])` → $\begin{pmatrix} x & x & x \\ \vdots & \vdots & \vdots \\ x & x & x \end{pmatrix}$
		elements of $n \times m$ matrix filled with x, if not provided, 1 is used
		star for element indices, skip row/column indices $n = 1/m = 1$
		only show p rows (including `\vdots` row) with skipped rows indicated by `\vdots`. If n isn't provided, p is used
		only show p rows (including `\vdots` row) and q columns (including `\ldots` column) with skipped rows indicated by `\vdots`, skipped columns indicated by `\ldots`, intersection of `\vdots` row and `\ldots` column being `\ddots`. If n/m isn't provided, p/q is used. No indices will be added for ellipses even if star is given

	`\PTmqty(\PTxmat{x}{3}{3}[3][3])`
	$\to \begin{pmatrix} x & x & x \\ x & x & x \\ x & x & x \end{pmatrix}$
	`\PTmqty(\PTxmat{x}{5}{}[3][3])`
	$\to \begin{pmatrix} x & \ldots & x \\ \vdots & \ddots & \vdots \\ x & \ldots & x \end{pmatrix}$
	`\PTmqty(\PTxmat{x}{}{}[3][3])`
	$\to \begin{pmatrix} x & \ldots & x \\ \vdots & \ddots & \vdots \\ x & \ldots & x \end{pmatrix}$	

	`\PTxmat*{x}{n}{m}{g}`
	`\PTmqty(\PTxmat*{x}{3}{3}{A})`
	$\to \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{A1} & x_{A2} & x_{A3} \end{pmatrix}$
	`\PTmqty(\PTxmat*{x}{5}{5}[3][3]{A})`
	$\to \begin{pmatrix} x_{11} & \cdots & x_{15} \\ \vdots & \ddots & \vdots \\ x_{A1} & \cdots & x_{A5} \end{pmatrix}$
		customize last row's element indices to g

	`\PTxmat*{x}{n}{m}{g}{h}`
	`\PTmqty(\PTxmat*{x}{3}{3}{A}{B})`
	$\to \begin{pmatrix} x_{11} & x_{12} & x_{1B} \\ x_{21} & x_{22} & x_{2B} \\ x_{A1} & x_{A2} & x_{AB} \end{pmatrix}$
	`\PTmqty(\PTxmat*{x}{5}{5}[3][3]{A}{B})`
	$\to \begin{pmatrix} x_{11} & \cdots & x_{1B} \\ \vdots & \ddots & \vdots \\ x_{A1} & \cdots & x_{AB} \end{pmatrix}$
		customize last row's element indices to g and last column's element indices to h

	`\PTxmat[0 or 1 or 2]{x}{n}{m}[p][q]`
	`\PTmqty(\PTxmat[0]{x}{5}{5}[3][3])`
	$\to \begin{pmatrix} x & x & \ldots \\ x & x & \ldots \\ \vdots & \vdots & \ddots \end{pmatrix}$
	`\PTmqty(\PTxmat[1]{x}{5}{5}[3][3])`
	$\to \begin{pmatrix} x & \ldots & x \\ x & \ldots & x \\ \vdots & \ddots & \vdots \end{pmatrix}$
	`\PTmqty(\PTxmat[2]{x}{5}{5}[3][3])`
	$\to \begin{pmatrix} x & x & \ldots \\ \vdots & \vdots & \ddots \\ x & x & \ldots \end{pmatrix}$
		Change the `\vdots` row/`\ldots` column from the second last one to last one, 0 for both, 1 for row only, 2 for column only. Only work when corresponding p/q is provided and do not change the behavior of element indices

`\zeromatrix`	`\zmat{n}{m}` `\PTmqty(\zmat{2}{2})` $\to \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ `\PTmqty(\zmat{2})` $\to \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$	$n \times m$ matrix filled with zeros, equivalent to `\xmat{0}{n}{m}`. If m isn't provided, n is used
`\paulimatrix`	`\pmat{n}` `\PTmqty(\pmat{0})` $\to \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ `\PTmqty(\pmat{1})` $\to \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	n^{th} Pauli matrix $n \in \{0, 1, 2, 3 \text{ or } x, y, z\}$

	`\PTmqty(\pmat{2})` → $\begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}$ `\PTmqty(\pmat{3})` → $\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}$	
`\diagonalmatrix`	`\dmat{a,b,c,...}` `\PTmqty(\dmat{1,2,3})` → $\begin{pmatrix}1 & & \\ & 2 & \\ & & 3\end{pmatrix}$ `\PTmqty(\dmat[0]{1,2})` → $\begin{pmatrix}1 & 0\\ 0 & 2\end{pmatrix}$ `\PTmqty(\dmat{1,2&3\\4&5})` → $\begin{pmatrix}1 & & \\ & 2 & 3\\ & 4 & 5\end{pmatrix}$	specify up to eight diagonal or block diagonal elements optional argument to fill spaces enter matrix elements for each block as a single diagonal element
`\antidiagonalmatrix`	`\admat{a,b,c,...}` `\PTmqty(\admat{1,2,3})` → $\begin{pmatrix} & & 1\\ & 2 & \\ 3 & & \end{pmatrix}$	same as syntax as `\dmat`

5.8 Symbols

`\lparen` → (	
`\rparen` →)	
`\ordersymbol` → $\mathcal{O}$	
`\typical` → 🟦	
`\tall` → 🟦	
`\grande` → 🟦	
`\venti` → 🟦	
`\calE`→`\mathcal{E}`	
`\bbR`→`\mathbb{R}`	
`\bbC`→`\mathbb{C}`	
`\bbQ`→`\mathbb{Q}`	
`\bbN`→`\mathbb{N}`	
`\bbZ`→`\mathbb{Z}`	
`\Vtextvisiblespace[width]` → ␣	a visible space character, where the optional argument, defaulting to `.3em`, sets the width of the horizontal rule

5.9 Shorthands for Greek alphabet

If option `shortgreek` is used, the following shorthands will be defined for every uppercase and lowercase Greek letter:

`\tgAlpha` → `\text{\textAlpha}}`	accept an optional argument argument in `{}` that is simply skipped
`\vAlpha` → `\varAlpha`	
`\uAlpha` → `\upAlpha`	
`\uvAlpha` → `\upvarAlpha`	

Note that these don't ensure those commands are defined.

5.10 Shorthands for mathrm alphabet and chemical element symbols

If option `shortmathrm` is used, the following shorthands will be defined for every uppercase and lowercase English letter and every chemical element symbols (`A\verb` for example):

`\rmA_a^b` → A_a^b	work in both math mode and text mode

which are implemented with:

```
\DeclareDocumentCommand{\rmA}{e{_^}}{\ensuremath{\mathrm{A}\IfValueTF{#1}{_{#1}}{}\IfValueTF{#2}{^{#2}}{}}}
```

5.11 Shorthands for textnormal alphabet

If option `shorttext` is used, the following shorthands will be defined for every uppercase and lowercase English letter (`A\verb` for example):

`\txA` → `\textnormal{A}`

5.12 Others

`\mathcolorbox`	`\mcbox{color}{content}` `\mcbox{cyan}{\phantom{A}}` → 🟦	`\colorbox` for math environment, applying to all four levels of math styles
`\autocolorbox` or `\acbox`	`\cbox{color}{content}`	calls `\colorbox` when in text mode, calls `\mathcolorbox` when in math mode
`\tentothepowerof`	`\tenpow{n}` → 10^n	work in both math mode and text mode
`\scientificnotation`	`\scinote{3.00}{8}` → 3.00×10^8	work in both math mode and text mode