`principia.sty`
A LaTeX 2ε Package for Typesetting Whitehead and Russell's *Principia Mathematica* (Version 3.2)

Landon D. C. Elkind `landon.elkind@wku.edu`

January 30, 2025

The `principia` package is designed for typesetting the Peanese notation of *Principia Mathematica*. "Peanese" is something of a misnomer: Whitehead and Russell invented much of the notations used in *Principia Mathematica* even while borrowing from many others.

`principia`'s style has antecedents in Kevin C. Klement's excellent *Tractatus* typesetting, to which we owe the device of adding 'd's and 't's to typeset further square dots. The device of beginning all `principia` commands with '`\pm`' derives from the `begriff` package, a style understandably mimicked in both the `frege` package and the `Grundgesetze` package.

In *Principia Mathematica* some symbols occur with an argument and sometimes that same symbol occurs without an argument. For example, '$(\exists x)$' occurs in some formulas, but sometimes '$\exists$' occurs in the text when they talk about the symbol itself. `principia` is designed to accommodate these different occurrences of symbols. When a symbol is to occur without an argument, capitalize the first letter following the '`\pm`' part of the command. E.g. `\pmsome{x}` produces $(\exists x)$ and `\pmSome` produces '$\exists$'. Note the former command requires an argument and the latter command does not. Not all commands in the `principia` package admit of such dual use because some symbols in *Principia Mathematica* never occur without an argument or do not take an argument in the usual sense. For example, the propositional connectives do not take an 'argument' in the way singular or plural descriptions do.

Version 2.0 of `principia` is adequate to typeset all notations throughout Volumes I-III of *Principia*. Version 3.0 includes commands that greatly ease typesetting of the appendices of Volume I and some minor fixes (especially to square dots). To help the user get the hang of the package, below is a table with commands for Volume I.

`principia`'s dependencies are `amsmath`, `amssymb`, `pifont`, and `graphicx`. To load `principia`, type `\usepackage{principia}` in the document's preamble (as with any TeX package).

Symbol	**TEX command**	**Notes**
$\vdash$	`\pmthm`	Theorem.
$\ast$	`\pmast`	As in $\ast 1$.
$\cdot$	`\pmcdot`	As in, $\ast 1\cdot 1$.
Pp	`\pmpp`	Primitive proposition. Note the indentation.
$=$	`\pmiddf`	Identity for definitions ('=' differs in spacing).
Df	`\pmdf`	Definition. Note the indentation.
Dem.	`\pmdem`	This symbol begins a proof.
$\left[\frac{p}{q}\right]$, $\left[\frac{p,\ \ r}{q,\ \ s}\right]$, $\left[\frac{p,\ \ r,\ \ t}{q,\ \ s,\ \ u}\right]$, ... $\left[\text{Add}\ \frac{p}{q}\right]$, ...	`\pmsub{p}{q}`, `\pmsubb{p}{q}{r}{s}`, `\pmsubbb{p}{q}{r}{s}{t}{u}`, ... `\pmSub{\text{Add}}{p}{q}`	Substitution into theorems. Add 'b's to the end of `\pmsub` to increase the number of substitutions (up to four 'b's). Each extra 'b' adds two arguments. To substitute and specify the theorem as well, capitalize the 's' in `\pmsub`.
., :, :., ::, ::., :::	`\pmdot`, `\pmdott`, `\pmdottt`, ...	Add 't's to the end of `\pmdot` to increase the number of dots (up to six 't's).
., :, :., ::, ::., :::	`\pmand`, `\pmandd`, `\pmanddd`, ...	Add 'd's to the end of `\pmand` command to increase the number of dots (up to six 'd's).
$\vee$	`\pmor`	Disjunction.
$\sim$	`\pmnot`	Negation. Note its spacing differs from `\sim`.
$\supset$	`\pmimp`	Material implication.
$\equiv$	`\pmiff`	Material biconditional.
$\supset_x$, $\supset_{x,y}$	`\pmimp_x`, `\pmimp_{x,y}`	And so on for more subscripts.
$\equiv_x$, $\equiv_{x,y}$	`\pmiff_x`, `\pmiff_{x,y}`	And so on for more subscripts.
$\hat{x}$	`\pmhat{x}`	This command requires one argument. It can be embedded in other commands. E.g., `\pmpf{\phi}{\pmhat{x}}` renders '$\phi\hat{x}$'.
ϕx	`\pmpf{\phi}{x}`	This command requires two arguments.
$\phi(x, y)$	`\pmpff{\phi}{x}{y}`	This command requires three arguments.
$\phi(x, y, z)$	`\pmpfff{\phi}{x}{y}{z}`	This command requires four arguments.
(x)	`\pmall{x}`	Universal quantifier.
$(\exists x)$, $\exists$	`\pmsome{x}`, `\pmSome`	Existential quantifier.
!	`\pmshr`	The predicative propositional functions.
$\phi!x$	`\pmpred{\phi}{x}`	This command requires two arguments.
$\phi!(x, y)$	`\pmpredd{\phi}{x}{y}`	This command requires three arguments.
$\phi!(x, y, z)$	`\pmpreddd{\phi}{x}{y}{z}`	This command requires four arguments.

$=$, $\neq$	`=`, `\pmnid`	Identity and its negation.
(℩x)	`\pmdsc{x}`	Definite description.
E!	`\pmexists`	Existence.
$\hat{z}(\psi z)$	`\pmcls{z}{\psi z}`	The class of zs satisfying ψ.
ϵ	`\pmcin`	The class membership symbol.
Cls^n, Cls	`\pmClsn{n}`, `\pmCls`	The class of classes of individuals.
Cl‘α, Cl	`\pmscl{\alpha}`, `\pmsCl`	The subclasses of a class α.
Rl‘R, Rl	`\pmsrl{R}`, `\pmsRl`	The sub-relations of a relation R.
V	`\pmcuni`	The universal class.
Λ	`\pmcnull`	The null class.
∃!	`\pmcexists`	The existence of a class.
$-\alpha$	`\pmccmp{\alpha}`	This command requires one argument.
$\alpha - \beta$	`\pmcmin{\alpha}{\beta}`	This command requires two arguments.
$\cup$	`\pmccup`	Class union.
$\cap$	`\pmccap`	Class intersection.
$\subset$	`\pmcinc`	Class inclusion.
$\hat{x}\hat{y}\phi(x,y)$	`\pmrel{x}{y}{\phi(x,y)}`	The relation in extension given by ϕ.
$a\{\hat{x}\hat{y}R(x,y)\}b$	`\pmrele{a}{x}{y}{R}{b}`	This command requires five arguments.
$a\{R\}b$	`\pmrelep{a}{R}{b}`	This command requires three arguments.
ϵ	`\pmrin`	The relation membership symbol.
Rel^n, Rel	`\pmReln{n}`, `\pmRel`	The class of relations (n-many ‘of relations’).
$\dot{\mathrm{V}}$	`\pmruni`	The universal relation.
$\dot{\Lambda}$	`\pmrnull`	The null relation.
$\dot{\exists}$!	`\pmrexists`	This symbol prefixes relations.
$\dot{-}R$	`\pmrcmp{\alpha}`	This command requires one argument.
$R \dot{-} S$	`\pmcmin{R}{S}`	This command requires two arguments.
$\dot{\cup}$	`\pmrcup`	Relation union.
$\dot{\cap}$	`\pmrcap`	Relation intersection.
$\dot{\subset}$	`\pmrinc`	Relation inclusion.
$\breve{R}$	`\pmcrel{R}`	The converse of a relation.
Cnv	`\pmCnv`	The command for ‘Cnv’.
R‘x	`\pmdscf{R}{x}`	A singular descriptive function.
R‘‘β	`\pmdscff{R}{\beta}`	A plural descriptive function.
R‘‘‘κ	`\pmdscfff{R}{\kappa}`	A plural descriptive function.
E!! R‘‘β	`\pmdscfe{R}{\beta}`	The existence of a plural descriptive function.

R_{ϵ}‘x, ‘R_{ϵ}’	`\pmdscfr{R}{x},`
	`\pmdscfR{R}`
		The relation of R_{ϵ}‘β to β.

D‘R, D	`\pmdm{R}`, `\pmDm`	The domain of a relation R.
ꓷ‘R, ꓷ	`\pmcdm{R}`, `\pmCdm`	The converse domain of a relation R.
C‘R, C	`\pmcmp{R}`, `\pmCmp`	The campus of a relation R.
F‘R, F	`\pmfld{R}`, `\pmFld`	The field of a relation R.
$\overrightarrow{R}$‘x, $\overrightarrow{R}$	`\pmrrf{R}{x}`, `\pmRrf{R}`	The referents of a given relation.
$\overleftarrow{R}$‘x, $\overleftarrow{R}$	`\pmrrl{R}{x}`, `\pmRrl{R}`	The relata of a given relation.
sg‘R, sg	`\pmsg{R}`, `\pmSg`	
gs‘R, gs	`\pmgs{R}`, `\pmGs`	
$R \mid S$, $\mid$	`\pmrprd{R}{S}`, `\pmrprd`	The relative product of R and S.
R^n	`\pmrprdn{R}{n}`	The nth relative product of R.
$R \,\Vert\, S$, $\Vert$	`\pmrprdd{R}{S}`, `\pmrprdd`	The double relative product of R and S.
$\alpha \upharpoonleft R$	`\pmrld{\alpha}{R}`	The limitation of R's domain to α.
$R \upharpoonright \beta$	`\pmrld{R}{\beta}`	The limitation of R's converse domain to β.

$\alpha \upharpoonleft R \upharpoonright \beta$	`\pmrldcd{\alpha}`
	`{R}{\beta}`
		The limitation of R's field to α and β, resp.

$P \upharpoonleft\!\!\upharpoonright \alpha$	`\pmrlf{\alpha}{R}{\beta}`	The limitation of P's field to α.
$\alpha \uparrow \beta$	`\pmrl{\alpha}{\beta}`	The relation made of all xs in α and ys in β.
♀	`\pmop`	The operation symbol.
$\alpha \underset{,,}{♀} y$	`\pmopc{\alpha}{y}`	The relation of xs in α taken to y by ♀.
p‘α	`\pmccprd{\alpha}`	The product of a class of classes.
s‘α	`\pmccsum{\alpha}`	The sum of a class of classes.
$\dot{p}$‘α	`\pmcrprd{\alpha}`	The product of a class of relations.
$\dot{s}$‘α	`\pmcrsum{\alpha}`	The sum of a class of relations.
I, J	`\pmrid`, `\pmrdiv`	The relations of identity and diversity.
ι‘x, ι	`\pmcunit{x}`, `\pmcUnit`	The unit class.
$\breve{\iota}$‘α	`\pmcunits{\alpha}`	The sum of unit classes of α's elements.
$\dot{n}$	`\pmrn{n}`	The ordinal number n.
$\dot{n}$	`\pmdn{n}`	The class of relations equal to an n-tuple.
$x \downarrow y$	`\pmoc{x}{y}`	The ordinal number restricted to $R = (x, y)$.
t‘x, t^{n}‘x	`\pmrt{x}`, `\pmrti{n}{x}`	The relative type of x (n-many ‘type of’s).
t_{n}‘α	`\pmrtc{n}{\alpha}`	The relative type of α (n-many ‘type of’s).

t^{n}‘R, t_{n}‘R	`\pmrtri{n}{R},`
	`\pmrtrc{n}{R}`
		The relative type of (with n-many ‘type of’s) R from individuals to individuals, or from classes to classes. ‘nm’ can replace ‘n’.

$^{n}t_{m}‘R$, $t_{n}^{m}‘R$	`\pmrtric{n}{R}`, `\pmrtrci{n}{R}`	The relative type of R from individuals to classes, or from classes to individuals.
α_{x}, $R_{(x,y)}$	`\pmrtdi{\alpha}{x}`, `\pmrtdri{R}{(x,y)}`	The result of determining that the members of α (R) belong to the relative type of x (in the domain, and of y in the converse domain).
$\alpha(x)$, $R(x,y)$	`\pmrtdc{\alpha}{x}`, `\pmrtdrc{R}{x,y}`	The result of determining that the members of α (R) belong to the relative type of $t‘x$ (in the domain, and of $t‘y$ in the converse domain).
$\alpha \rightarrow \beta$	`\pmrdc{\alpha}{\beta}`	The class of relations R with domain contained in α and converse domain in β.
$1 \rightarrow 1$, $1 \rightarrow \mathrm{Cls}$, $\mathrm{Cls} \rightarrow 1$	`\pmoneone`, `\pmonemany`, `\pmmanyone`	The class of one-one, or one-many, or many-one, relations. Note `\pmrdc` can be used here.
sm, $\overline{\mathrm{sm}}$	`\pmsm`, `\pmsmbar`	The similarity relation.
$P_{\Delta}‘\kappa$, P_{Δ}	`\pmselp{\kappa}`, `\pmSelp`	The P-selections from κ
$\boldsymbol{\epsilon}_{\Delta}‘\kappa$, $\boldsymbol{\epsilon}_{\Delta}$	`\pmsele{\kappa}`, `\pmSele`	The $\boldsymbol{\epsilon}$-selections from κ
$F_{\Delta}‘\kappa$, F_{Δ}	`\pmself{\kappa}`, `\pmSelf`	The F-selections from κ
$\mathrm{Cls}^{2}\,\mathrm{excl}$	`\pmexc`	The class of pairwise-disjoint classes.
$\mathrm{Cls}\,\mathrm{ex}^{2}\,\mathrm{excl}$	`\pmexcn`	The class of pairwise-disjoint non-null classes.
$\mathrm{Cl}\,\mathrm{excl}‘\gamma$	`\pmexcc{\gamma}`	A class of mutually exclusive classes in γ.
$P ↧ y$	`\pmselc{P}{y}`	The class of couples $(y, P‘y)$.
$\mathrm{Cls}^{2}\,\mathrm{Mult}$	`\pmmultc`	The class of multipliable classes.
Rel Mult	`\pmmultr`	The class of multipliable relations.
Mult ax	`\pmmultax`	The multiplicative axiom.
R_{*}, $\breve{R}_{*}$	`\pmanc{R}`, `\pmancc{R}`	The ancestral and its converse.
R_{st}, R_{ts}	`\pmrst{R}`, `\pmrts{R}`	The powers of the ancestral and its converse.
$\min_{P}$, $\max_{P}$	`\pmmin{P}`, `\pmmax{P}`	The minimum and maximum under P.
$\mathrm{Pot}‘R$, $\mathrm{Potid}‘R$	`\pmpot{R}`, `\pmpotid{R}`	The products (strict and not) of an ancestral.
R_{po}	`\pmpo{R}`	The product of a class of ancestrals R.
B	`\pmB`	The relation of beginning under P.
$\mathrm{gen}‘P$	`\pmgen{P}`	The generation of P.
$P \ast Q$	`\pmefr{P}{Q}`	The equi-factor relation.
$I_{R}‘x$	`\pmipr{R}{x}`	The non-distinct posterity of x under R.
$J_{R}‘x$	`\pmjpr{R}{x}`	The distinct posterity of x under R.
$\overleftrightarrow{R}‘x$	`\pmfr{R}{x}`	The ancestry and posterity of x under R.
$\mathrm{Nc}‘\kappa$, Nc	`\pmnc{\kappa}`, `\pmNc`	The cardinal number of κ.