

# Logic RQ – Basic One

## *Mnemonic Character Map*

¬	n: <i>not</i>	Negation	⊢	s: <i>sequence</i>	Proves, yields; reducible; sequence
~	~: <i>similar</i>	Negation (glyph variant of ¬)	⊣	k: <i>'k'on-seq.</i>	Does not yield – or – proofs/yields right-to-left; sequence to the left
∧	a: <i>and</i>	Conjunction	⊥	Æ: <i>ligature</i>	Equal, mutually reducible
∨	o: <i>or</i>	Disjunction, adjunction	⊥	S: <i>shift-s</i>	Does not yield; not reducible
→	i: <i>implies</i>	Implication, subjuction, conditional	⊥	K: <i>shift-k</i>	Does not proof/yield right-to-left
←	r: <i>replies</i>	Replication	⊥	c: <i>consequence</i>	True, tautology, satisfies, results in
↔	e: <i>equal</i>	Equivalence, bijunction, biconditional	⊥	b: <i>before c</i>	True, tautology etc. right-to-left
⊃	x: <i>exclusive</i>	XOR, contravalance, disjunction	⊥	Æ: <i>upcase æ</i>	Equal, mutually satisfying etc.
↑	u: <i>up</i>	NAND, exclusion, negat-ad/disjunction	⊥	C: <i>shift-c</i>	Not true, no tautology etc.
	: <i>just a bar</i>	NAND, Sheffer bar (glyph variant of ↑)	⊥	B: <i>shift-b</i>	Not true, no tautology right-to-left
↓	v: <i>like a v</i>	NOR, Peirce arrow, negat-conjunction	∴	m: <i>arbitrary</i>	Therefore (old-style)
			∴	M: <i>upcase m</i>	Because (old-style)
⊖	â: <i>not-a</i>	NAND (glyph variant of ↑)	t	t: <i>true</i>	True (value) – English
∇	ô: <i>not-o</i>	NOR (glyph variant of ↓)	w	w: <i>wahr</i>	Wahr (value) – German
△	ã: ~ + a	<i>Undefined (included for completeness)</i>	f	f: <i>false, falsch</i>	False, falsch (value)
∇	ô: ~ + o	XOR (glyph variant of ⊃)	T	T: <i>shift-t</i>	True, verum (operator)
⇒	I or î: <i>not-i</i>	Negat-implication	⊥	F: <i>shift-f</i>	False, falsum (operator)
⇐	R: <i>shift-i</i>	Negat-replication	✓	V: <i>like a V</i>	Correct (proofreader's sign)
⇔	ê: <i>not-e</i>	Negat-equivalence (glyph variant of ⊃)	×	X: <i>like a X</i>	Error (proofreader's sign)
∧	á: <i>accent-a</i>	Conjunction with dot (usage varies)	φ ψ	p q: <i>usage</i>	Phi and psi (formula variables)
∨	ó: <i>accent-o</i>	Dis/adjunction with dot (usage varies)	ℳ ℒ	J L: <i>similar</i>	Interpretation; formal language
⇒	í: <i>accent-i</i>	Implication with dot (usage varies)	∈	€: <i>similar</i>	Element of
⇐	®: <i>from r</i>	Replication with dot (usage varies)	∅	∅: <i>similar</i>	Empty set
↔	é: <i>accent-e</i>	Equivalence with dot (usage varies)	=	=: <i>normal</i>	Equal (general)
⋈	ä: <i>dbl dot a</i>	Repeated conjunction	≠	≠: <i>normal</i>	Not equal
w	ö: <i>dbl dot o</i>	Repeated dis/adjunction	≈	≈: <i>normal</i>	Similar etc.
⊗	â: <i>a with o</i>	Repeated con- or dis/adjunction resp.	≈	≈: <i>similar</i>	Tilde with double dot (special usage)
⇒	z: <i>arbitrary</i>	Right arrow from bar (usage varies)	≡	#: <i>well, ...</i>	Identical
⇐	y: <i>before z</i>	Left arrow from bar (usage varies)	≈	\$: <i>near to #</i>	Special mutually reducible etc.
⇒	h: <i>arbitrary</i>	Double right arrow (usage varies)	≈	€: <i>like \$</i>	Special mutually reducible etc.
⇐	g: <i>before h</i>	Double left arrow (usage varies)	()[]{}	<i>like usual</i>	Parens like in ordinary fonts
↔	d: <i>before e</i>	Double left right arrow (usage varies)	⟨⟩	<>: <i>similar</i>	Angle brackets
⇒	H: <i>shift-h</i>	Double right arrow with stroke	┌┐	≤ ≥: <i>paren-like</i>	Corner brackets
⇐	G: <i>shift-g</i>	Double left arrow with stroke	■	Z: <i>last letter</i>	End of poof
↔	D: <i>shift-d</i>	Double left right arrow with stroke	■	Y: <i>before Z</i>	End of proof (glyph variant)
∀	A: <i>for all</i>	For all quantifier			
∃	E: <i>exists</i>	There exists quantifier			
∄	Ê: <i>not-E</i>	Negated there exists quantifier			
∩	U: <i>unify</i>	Unificator			
□	N: <i>necessary</i>	It is necessary; <i>sometimes</i> : end of proof			
◇	P: <i>possible</i>	It is possible			
○	O: <i>like a O</i>	<i>Undefined (included for completeness)</i>			
λ	l: <i>lambda</i>	Lambda operator			
ι	j: <i>like iota</i>	Turned iota operator			

The punctuation glyphs use more or less their ordinary shapes (.,:; ; ? ! ' " " " > < > < ... --- • \* etc). The figures offer index numbers: <sup>0123456789</sup>, to which one may add <sub>m</sub> (at μ), <sub>n</sub> (at ñ) and <sub>i</sub> (at ι) as well as <sub>def</sub> (at %) for use with =<sub>def</sub> and ' prime and " second (at ' and "). And there is some minor stuff placed ad free character places here and there. *Roman Eisele*