Harold's Mathematical Logic
Cheat Sheet

18 November 2026

The 7 Basic Logical Symbols

Operator	Symbol	Example	English
1) Conjunction	∧, **⋀**, ⩓, ⋀, ⋀ •	p ∧ q	• similar to ⋂ intersection for sets • p and q • p, but q • despite the fact that p, q • even though p, q • although p, q • overlap
2) Disjunction	∨, **⋁**, ⩔, ⋁, ⋁	p ∨ q	• similar to ⋃ union for sets • p or q • inclusive or • both combined
3) Negation	¬, ⌐, ~	¬p	• not p
4) Conditional	→, ➙, 🡒, ⟶, ⇒, ⟹, ⊃	p → q	• if p then q • if p, q • q if p • p implies q • p only if q • q in case that p • p is sufficient for q • q is necessary for p
5) Biconditional	↔, ⟷, **⟷**, ⇔, ⟺	p ⟷ q	• p iff q • p if and only if q • p is necessary and sufficient for q • if p then q, and conversely • if not p then not q, and conversely
(Equivalence)	≡, ≡, ≡	$expression_1$ ≡ $expression_2$	• is identical to • is logically equivalent to • the two expressions have the same truth value under every interpretation
6) Universal Quantifier	∀*x*, (*x*)	∀*x p(x)*	• for all • for any • for each
7) Existential Quantifier	∃*x*	∃*x p(x)*	• there exists • there is at least one

"… the structure of all mathematical statements can be understood using these symbols, and all mathematical reasoning can be analyzed in terms of the proper use of these symbols."
Source: "How to Prove It: A Structured Approach", 3rd Edition, p. 75.

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Logical Truth Tables

p	q	Conjunction (AND) $\wedge$	NAND $\overline{\wedge}$	Disjunction (OR) $\vee$	NOR $\overline{\vee}$	XOR $\veebar, \oplus$	XNOR $\odot$	Negation (NOT) $\neg p$
F	**F**	F	T	F	T	F	T	T
F	**T**	F	T	T	F	T	F	T
T	**F**	F	T	T	F	T	F	F
T	**T**	T	F	T	F	F	T	F

p	q	Material Implication (If … Then) $\rightarrow$	Biconditional (Iff) $\leftrightarrow$	Tautology (True) $\top$	Contradiction (False) $\bot$
F	**F**	T	T	T	F
F	**T**	T	F	T	F
T	**F**	F	F	T	F
T	**T**	T	T	T	F

Blank Truth Tables

Inputs				Output		
p	q	r	s	x	y	z
F	**F**	**F**	**F**			
F	**F**	**F**	**T**			
F	**F**	**T**	**F**			
F	**F**	**T**	**T**			
F	**T**	**F**	**F**			
F	**T**	**F**	**T**			
F	**T**	**T**	**F**			
F	**T**	**T**	**T**			
T	**F**	**F**	**F**			
T	**F**	**F**	**T**			
T	**F**	**T**	**F**			
T	**F**	**T**	**T**			
T	**T**	**F**	**F**			
T	**T**	**F**	**T**			
T	**T**	**T**	**F**			
T	**T**	**T**	**T**			

Inputs			Output	
p	q	r	x	y
F	**F**	**F**		
F	**F**	**T**		
F	**T**	**F**		
F	**T**	**T**		
T	**F**	**F**		
T	**F**	**T**		
T	**T**	**F**		
T	**T**	**T**		

Inputs		Output
p	q	x
F	**F**	
F	**T**	
T	**F**	
T	**T**	

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Logical Conditional Connective Laws

Law or Statement	Logical Expression	Is Equivalent To (≡)	Description
Antecedent / Consequent	If <Antecedent> then <Consequent> . <Consequent> if <Antecedent> .		The Antecedent immediately follows the "if" statement. a.k.a. <Hypothesis> / <Conclusion>
Conditional Laws	$p \rightarrow q$	$\neg p \vee q$ $\neg(p \wedge \neg q)$ Logical Equivalences: $p \vee q \equiv \neg p \rightarrow q$ $p \wedge q \equiv \neg(p \rightarrow \neg q)$ $\neg(p \rightarrow q) \equiv p \wedge \neg q$ $(p \rightarrow q) \wedge (p \rightarrow r) \equiv p \rightarrow (q \wedge r)$ $(p \rightarrow q) \vee (p \rightarrow r) \equiv p \rightarrow (q \vee r)$ $(p \rightarrow r) \wedge (q \rightarrow r) \equiv (p \vee q) \rightarrow r$ $(p \rightarrow r) \vee (q \rightarrow r) \equiv (p \wedge q) \rightarrow r$	Conditional, If ... Then, Implication
Biconditional Laws	$p \leftrightarrow q$	$(p \rightarrow q) \wedge (q \rightarrow p)$ $(p \rightarrow q) \wedge (\neg p \rightarrow \neg q)$ $(p \wedge q) \vee (\neg p \wedge \neg q)$ $\neg p \leftrightarrow \neg q$ Logical Equivalences: $\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$	Bi-conditional, If and only If, iff, XNOR
Sufficient Condition	p is a sufficient condition for q	The truth of p suffices to guarantee the truth of q.	
Necessary Condition	q is a necessary condition for p	For p to be true, q must also be true. $\neg q \rightarrow \neg p$	
(Equivalence)	$p \leftrightarrow q$ (can be true for one assignment)	$p \equiv q$ (must be true for every assignment)	$p \equiv q$ exactly when $p \leftrightarrow q$ is a tautology
Contrapositive	$p \rightarrow q$	$\equiv \neg q \rightarrow \neg p$	True
Converse*	$p \rightarrow q$	$\not\equiv q \rightarrow p$	Not logically equivalent in general.
Inverse*	$p \rightarrow q$	$\not\equiv \neg p \rightarrow \neg q$	

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Rules of Implication
(Inference with Propositions)

Rule Name	Rule Logic	Example
Hypothesis	Givens. First lines of a proof.	It is raining today. You live in McKinney, Texas.
Therefore	$\therefore$	Therefore. In conclusion.
1) Modus Ponens (MP)	$\begin{array}{l} p \\ \underline{p \to q} \\ \therefore q \end{array}$	It is raining today. If it is raining today, I will not ride my bike to school. Therefore, I will not ride my bike to school.
2) Modus Tollens (MT)	$\begin{array}{l} \neg q \\ \underline{p \to q} \\ \therefore \neg p \end{array}$	If Sam studied for his test, then Sam passed his test. Sam did not pass his test. Therefore, Sam did not study for his test.
3) Hypothetical Syllogism (HS) (Transitivity)	$\begin{array}{l} p \to q \\ \underline{q \to r} \\ \therefore p \to r \end{array}$	If you are mad, then you will yell. If you yell, then you will wake the baby. Therefore, if you are mad, then you will wake the baby.
4) Disjunctive Syllogism (DS) (Elimination)	$\begin{array}{l} p \vee q \\ \underline{\neg p} \\ \therefore q \end{array}$	Sam studied for his test, or Sam took a nap. Sam did not study for his test. Therefore, Sam took a nap.
5) Constructive Dilemma (CD)	$\begin{array}{l} p \vee q \\ \underline{(p \to r) \wedge (q \to s)} \\ \therefore r \vee s \end{array}$	Oscar is either a dog or a cat. If Oscar is a dog, then you'll have fleas, and if Oscar is a cat, then you'll have fur balls. Therefore, you'll have either fleas or fur balls.
6) Simplification (Simp) (Specialization)	$\begin{array}{l} \underline{p \wedge q} \\ \therefore p \end{array}$	It is rainy today, and it is windy today. Therefore, it is rainy today.
7) Conjunction (Conj)	$\begin{array}{l} p \\ \underline{q} \\ \therefore p \wedge q \end{array}$	Sam studied for his test. Sam passed his test. Therefore, Sam studied for his test, and Sam passed his test.
8) Addition (Add) (Generalization)	$\begin{array}{l} p \\ \hline \therefore p \vee q \end{array}$	It is raining today. Therefore, it is either raining today or snowing today or both.
9) Resolution	$\begin{array}{l} p \vee q \\ \underline{\neg p \vee r} \\ \therefore q \vee r \end{array}$	Your shirt is red, or your pants are blue. Your shirt is not red, or your shoes are white. Therefore, your pants are blue, or your shoes are white.

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

10) Proof by Division into Cases	$\begin{array}{l} p \vee q \\ p \rightarrow r \\ q \rightarrow r \\ \hline \therefore r \end{array}$	It is raining, or it is Monday. It is raining, so it is wet. It is Monday, so it is wet. It is wet.
11) Contradiction Rule	$\begin{array}{l} \neg p \rightarrow F \\ \hline \therefore p \end{array}$	If "it is not raining" is false, then it is raining.

Rules of Replacement

(Logical Connective Laws / Equivalences / Inference)

Law	Disjunction Example	Conjunction Example
12) Identity Laws	p ∨ F ≡ p	p ∧ T ≡ p
13) Domination or Null (Universal Bound Laws)	p ∨ T ≡ T	p ∧ F ≡ F
14) Idempotent Laws	p ∨ p ≡ p	p ∧ p ≡ p
15) Double Negations (DN) (Involution Law)	¬ ¬p ≡ p	
16) Negation or Complement (Complementary Laws)	p ∨ ¬p ≡ T ¬F ≡ T	p ∧ ¬p ≡ F ¬T ≡ F
17) Commutative Laws (Com)	p ∨ q ≡ q ∨ p	p ∧ q ≡ q ∧ p
18) Associative Laws (Assoc)	(p ∨ q) ∨ r ≡ p ∨ (q ∨ r)	(p ∧ q) ∧ r ≡ p ∧ (q ∧ r)
19) Distributive Laws (Dist)	p ∧ (q ∨ r) ≡ (p ∧ q) ∨ (p ∧ r)	p ∨ (q ∧ r) ≡ (p ∨ q) ∧ (p ∨ r)
20) Uniting Laws	(p ∧ q) ∨ (p ∧ ¬q) ≡ p	(p ∨ q) ∧ (p ∨ ¬q) ≡ p
21) Absorption Laws	p ∨ (p ∧ q) ≡ p	p ∧ (p ∨ q) ≡ p
22) De Morgan's Laws (DM) (Propositional Logic)	p ∨ q ≡ ¬(¬p ∧ ¬q) ¬(p ∨ q) ≡ ¬p ∧ ¬q (p ∨ ¬q) → r ≡ ¬r → (¬p ∧ q)	p ∧ q ≡ ¬(¬p ∨ ¬q) ¬(p ∧ q) ≡ ¬p ∨ ¬q
23) Multiplication and Factoring Laws	(p ∨ q) ∧ (¬p ∨ r) ≡ (p ∧ r) ∨ (¬p ∧ q)	(p ∧ q) ∨ (¬p ∧ r) ≡ (p ∨ r) ∧ (¬p ∨ q)
24) Consensus Laws	(p ∧ q) ∨ (q ∧ r) ∨ (¬p ∧ r) ≡ (p ∧ q) ∨ (¬p ∧ r)	(p ∨ q) ∧ (q ∨ r) ∧ (¬p ∨ r) ≡ (p ∨ q) ∧ (¬p ∨ r)
25) Tautology Laws (⊤)	p ∨ (⊤) ≡ ⊤ p ∨ ¬p ≡ ⊤ (True)	p ∧ (⊤) ≡ p
	¬(⊤) = ⊥	
26) Contradiction Laws (⊥)	p ∨ (⊥) ≡ p	p ∧ (⊥) ≡ ⊥ p ∧ ¬p ≡ ⊥ (False)
	¬(⊥) ≡ ⊤	
27) Exclusive Or Laws (⊕)	p ⊕ q ≡ (p ∨ q) ∧ ¬(p ∧ q)	p ⊕ q ≡ (¬p ∧ q) ∨ (p ∧ ¬q)

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Proof Methods

Method	Definition
Direct (DP)	• Assume the hypothesis (antecedent) is true, then use logical steps to arrive at the conclusion (consequent). • Assume p is true, then conclude q.
Indirect (IP) (Contradiction)	• Assume the opposite of what you want to prove, then show this leads to a contradiction. • To prove p, assume ¬p and derive a contradiction, such as ¬q ∧ q. • If some statement is assumed true, and a logical contradiction occurs, then the statement must be false. • Technique: Assume ¬(p → q), which is equivalent to p ∧ ¬q. • Assumption for Indirect Proof (AIP)
Conditional (CP)	• Assume a hypothesis (antecedent) is temporarily true to derive a conclusion (consequent). • After discharging an assumption, statements that depend on it cannot be used outside that subproof. • Assume p, derive q; conclude p → q. • The goal is not to prove p is true in reality, but to prove that **if** p were true, then q would necessarily follow. • Assumption for Conditional Proof (ACP)
Contrapositive	• Similar to Modus Tollens. • To prove p → q, assume ¬q, derive ¬p, and discharge the assumption to obtain ¬q → ¬p; then use Transposition.
Construction (Example)	• The construction of a concrete example with a property to show that something having that property exists. • AKA proof by example.
Exhaustion / By Cases	• The conclusion is established by dividing it into a finite number of cases and proving each one separately.
Induction	• A single "base case" is proved, and an "induction rule" is proved that establishes that any arbitrary case implies the next case. • Prove $P(n_0)$. Then prove $P(k) \rightarrow P(k+1)$ for an arbitrary integer $k \geq n_0$. Conclude $P(n)$ for every integer $n \geq n_0$.

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Logical Predicates

Definition	Logical Expression	Definition / Explanation	Plain English
Universe of Discourse	$\boldsymbol{U}$	All possible inputs in a given range	• Universe of Discourse • Universal Set • Universe
Domain of Discourse	$\mathbb{D}$	All possible inputs in a given range	• Domain of Discourse • Universe of Discourse
Proposition or Logical Statement	*p: "Roxy is a mammal."*	*p*	• Must be True or False • Cannot be a question • Cannot be a command
Predicate	*P(x): "x is a mammal"*	*P(x)*	• A logical statement whose truth value is a function of one or more variables • Truth depends upon the input variable *x* • P(x) ≠ a number • P(5) is a proposition provided 5 belongs to the domain
Example Statements	*q:* $\forall x \in \mathbb{D}$, *P(x)*: "x is a mammal"	*"For all x in the domain of discourse, P(x) is true."*	• Is either True or False • Binding all free variables produces a statement with a definite truth value
	T(x, y)	"*x* is a twin of *y*."	Predicate with two input variables
Truth Set (Single Free Variable)	$T = \{x \in \mathbb{D} \mid P(x)\}$	$T = \{a \mid P(a)\}$ $T = \{a \in A \mid P(a)\}$ $a \in T$	The set of all values of x that make the statement *p(x)* true
	Example:	$P(x_1)$, $P(x_2)$, and $P(x_3)$ are True	
Truth Set (Ordered Pair)	$T = \{(x, y) \in \mathbb{D}^2 \mid P(x, y)\}$	$\{(a, b) \in A \times B \mid P(a, b)\}$ $(a, b) \in T$	Cross product truth set
	Examples:	$\{(p, n) \in P \times \mathbb{N} \mid$ the person p has n children$\}$ = {(John, 2), …} $\{(p, c, n) \in P \times C \times \mathbb{N} \mid$ the person p has lived in the city c for n years$\}$	

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Logical Quantifiers

Definition	Logical Expression	Is Equivalent To (≡)	Plain English
Universal Quantifier (∀)	$\forall x\ P(x)$ $\forall x \in \mathbb{D},\ P(x)$ $\forall x$, *if* x *is in* $\mathbb{D}$ *then* $P(x)$	"For all x in the domain, $P(x)$ is true" $\forall x \in A\ P(x) \equiv \forall x\ (x \in A \rightarrow P(x))$ For the finite set domain of discourse $\{a_1, a_2, \ldots, a_k\}$, $\forall x\ P(x) \equiv P(a_1) \wedge P(a_2) \wedge \ldots \wedge P(a_k)$	• for all • all elements • for each member • any • anyone • anything • every • everyone • everybody • everything • x could be anything at all • whoever
Existential Quantifier (∃)	$\exists x\ P(x)$ $\exists x \in \mathbb{D},\ P(x)$	"There exists x in the domain such that $P(x)$ is true" For the finite set domain of discourse $\{a_1, a_2, \ldots, a_k\}$, $\exists x\ P(x) \equiv P(a_1) \vee P(a_2) \vee \ldots \vee P(a_k)$ $\{x \in \mathbb{D}: P(x)\} \neq \emptyset$	• there exists an x • there is • some • someone • somebody • something • at least one value of x • there is at least one x • the truth set is not equal to ∅
Uniqueness Quantifier (∃!)	$\exists! x\ P(x)$	"Exactly one x in the domain satisfying $P(x)$" $\exists x\ (P(x) \wedge \neg\exists y\ (P(y) \wedge y \neq x))$ $\exists x\ (P(x) \wedge \forall y\ (P(y) \rightarrow y = x))$ $\exists x\ \forall y\ (P(y) \leftrightarrow y = x)$ $\exists x\ P(x) \wedge \forall y\ \forall z((P(y) \wedge P(z)) \rightarrow y = z)$	• unique • there is a unique x • there exists exactly one • there is exactly one x such that $P(x)$
Negated Quantifiers	$\neg\ [\exists x\ P(x)]$	$\forall x\ \neg P(x)$	• nobody • no one • not one • there does not exist
	$\neg\ [\forall x\ P(x)]$	$\exists x\ \neg P(x)$	• not everyone
Order of Precedence	PEMDAS for Logic: 1. Parenthesis 2. Logical NOT 3. Quantifiers 4. Logical AND 5. Logical OR 6. Logical Conditional 7. Logical Biconditional	Symbol: (), {}, [] ¬ ∀, ∃ ∧ ∨ → ↔	As used in this sheet. Use parentheses for scope. Example : $\forall x\ P(x) \wedge Q(x) \equiv$ $(\forall x\ P(x)) \wedge Q(x)$

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Quantifier Logic Examples

Quantifier	Symbolic Translations	English Example
Everyone	$\forall x\ P(x)$	• Everyone is <something>.
	$\forall x\ \forall y\ P(x, y)$ NOTE: includes (x = y)	• Everyone <did something> to everyone.
Everyone Else	$\forall x\ (x \neq a \rightarrow P(x)) \land \neg P(a)$	• Everyone except <someone> is <something>.
	$\forall x\ \forall y\ (x \neq y \rightarrow P(x, y))$ NOTE: excludes (x = y)	• Everyone <did something> to everyone else.
Not Every	$\neg\forall x\ (A(x) \rightarrow B(x))$	• It is not the case that every <something> <did something>.
	$\exists x\ \neg P(x)$	• Not every <something> <did something>.
Someone Else	$\exists x\ ((x \neq a) \land P(x))$	• Someone other than <someone> is <something>.
	$\forall x\ \exists y\ ((x \neq y) \land P(x, y))$ NOTE: excludes (x = y)	• Everyone <did something> to <someone> else.
Exactly One	$\exists !x\ P(x)$	• Exactly one person <did something>.
	$\exists x\ [P(x) \land \forall y\ ((x \neq y) \rightarrow \neg P(y))]$	
No One	$\neg\exists x\ P(x)$	• No one <did something>.
	$\forall x\ \neg P(x)$	

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Rules of Inference with Quantifiers

Rule Name	Rule Logic	English Example
Variables	***x***: Quantified variable	The domain is the set of all integers.
Elements	***c***, ***d***: Elements of the domain, arbitrary or particular	*c* is a particular integer. Element definition.
Universal Instantiation (UI)	*c* is an element (arbitrary or particular) $\forall x\, P(x)$ $\therefore P(c)$	Sam is a student in the class. Every student in the class completed the assignment. Therefore, Sam completed his assignment.
Universal Generalization (UG)	*c* is an arbitrary element $P(c)$ $\therefore \forall x\, P(x)$ (The derivation must not depend on any undischarged assumption specific to *c*.)	All psychiatrists are doctors. All doctors are college graduates. Therefore, all psychiatrists are college graduates. 1. $\forall x\,(P(x) \rightarrow D(x))$ Given 2. $\forall x\,(D(x) \rightarrow C(x))$ Given 3. $P(x) \rightarrow D(x)$ 1, UI 4. $D(x) \rightarrow C(x)$ 2, UI 5. … 6. $P(x) \rightarrow C(x)$ 7. $\therefore \forall x\,(P(x) \rightarrow C(x))$ 6, UG
Existential Instantiation (EI)	$\exists x\, P(x)$ $\therefore P(c)$ (*c* is a fresh name for an element satisfying *P*)	All attorneys are college graduates. Some attorneys are golfers. Therefore, some golfers are college graduates. i.e., If an object is known to exist, then that object can be given a name.
Existential Generalization (EG)	*c* is an element (arbitrary or particular) $P(c)$ $\therefore \exists x\, P(x)$	All tenors are singers. Andrea Bocelli is a tenor. Therefore, there is at least one singer. 1. $\forall x\,(T(x) \rightarrow S(x))$ Given 2. $T(a)$ Given 3. $T(a) \rightarrow S(a)$ 1, UI 4. $S(a)$ 2,3, MP 5. $\therefore \exists x\, S(x)$ 4, EG

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Quantifier Translation Hints

Statement Form	Symbolic Translation	English Example
A B	$A(x) \wedge B(x)$	Pretty girl.
Either A or B	$A(x) \vee B(x)$	Rachel is a journalist or a newscaster.
	$(A(x) \wedge \neg B(x)) \vee (\neg A(x) \wedge B(x))$	Rachel is either a journalist or a newscaster, but not both. (XOR)
Neither A nor B	$\neg A(x) \wedge \neg B(x)$	Neither Wordsworth nor Shelley was Irish. $\neg I(w) \wedge \neg I(s)$
A is B	$S(e)$	Eli is a student.
Anything is A	$\forall x\, A(x)$	Anything is conceivable.
All A are B	$\forall x\, (A(x) \rightarrow B(x))$	All maples are trees. Sea lions are mammals.
Some A are B	$\exists x\, (A(x) \wedge B(x))$	Some grapes are sour.
Some A are not B	$\exists x\, (A(x) \wedge \neg B(x))$	Some grapes are not sour.
A exist	$\exists x\, A(x)$	Tigers exist.
A do not exist	$\neg\exists x\, A(x)$ $\forall x\, \neg A(x)$	Unicorns do not exist.
No A are B	$\forall x\, (A(x) \rightarrow \neg B(x))$ $\neg\exists x\, (A(x) \wedge B(x))$	No novels are biographies.
Not a single A did B	$\forall x\, (A(x) \rightarrow \neg B(x))$ $\neg\exists x\, (A(x) \wedge B(x))$	Not a single psychologist attended the convention.
Whoever is A is B	$\forall x\, (A(x) \rightarrow B(x))$	Whoever is a socialite is vain.
Some A B iff C	$\exists x\, (A(x) \wedge (B(x) \leftrightarrow C(x)))$	Some dogs bite if and only if they are teased.
Some A B are C	$\exists x\, [(A(x) \wedge B(x)) \wedge C(x)]$	Some French restaurants are exclusive.
A B are C	$\forall x\, [(A(x) \wedge B(x)) \rightarrow C(x)]$	Ripe apples are delicious.
A and B are C D	$\forall x\, [(A(x) \vee B(x)) \rightarrow (C(x) \wedge D(x))]$	Violins and cellos are stringed instruments.
Only i is F.	$F(i) \wedge \forall x\, [F(x) \rightarrow (x = i)]$	Only Sally is running.
The only F that is G is i.	$F(i) \wedge G(i) \wedge \forall x\, [(F(x) \wedge G(x)) \rightarrow (x = i)]$	The only student who is running is Sally.
No F except i is G.		No student except Sally is running.
All F except i are G.	$F(i) \wedge \neg G(i) \wedge \forall x\, [(F(x) \wedge (x \neq i)) \rightarrow G(x)]$	All students except Billy are on time.
i is the F that is most so-and-so.	$F(i) \wedge \forall x\, [(F(x) \wedge (x \neq i)) \rightarrow$ i is more so-and-so than x]	Rex is the dog that is more loved than the rest.
There is at most one F.	$\forall x\, \forall y\, [(F(x) \wedge F(y)) \rightarrow (x = y)]$	There is at most one moon.
There are at least two F's.	$\exists x\, \exists y\, [F(x) \wedge F(y) \wedge (x \neq y)]$	There are at least two moons.
There are exactly two F's.	$\exists x\, \exists y\, \{ F(x) \wedge F(y) \wedge (x \neq y) \wedge \forall z\, [F(z) \rightarrow ((z = x) \vee (z = y))] \}$	There are exactly two moons.
The F is G.	$\exists x\, [F(x) \wedge \forall y\, (F(y) \rightarrow (y = x)) \wedge G(x)]$	The moon is bright.

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Quantifier Laws

Definition	Logical Expression	Is Equivalent To (≡)		English Example
Abbreviation	$\forall x\ (x \in A \rightarrow P(x))$	≡	$(\forall x \in A)\ P(x)$	Simplification
	$\exists x\ (x \in A \bullet \neg P(x))$	≡	$(\exists x \in A)\ \neg P(x)$	
Expanding Abbreviation	$\forall x \in A\ P(x)$	≡	$\forall x\ (x \in A \rightarrow P(x))$	Complication
1) Quantifier Negation Laws (QN)	$\forall x\ P(x)$	≡	$\neg\exists x\ \neg P(x)$	• Everyone is perfect. • No one is imperfect.
	$\exists x\ P(x)$	≡	$\neg\forall x\ \neg P(x)$	• Somebody is perfect. • Not everyone is imperfect.
	$\neg\forall x\ P(x)$	≡	$\exists x\ \neg P(x)$	• Not everyone is perfect. • Someone is imperfect.
	$\neg\exists x\ P(x)$	≡	$\forall x\ \neg P(x)$	• No one is perfect. • Everybody is imperfect.
2) Conditional Law	$x \in A \rightarrow P(x)$	≡	$x \notin A \vee P(x)$	$p \rightarrow q \equiv \neg p \vee q$
3) Membership Negation Law	$x \in A$	≡	$\neg(x \notin A)$	Negate, then swap $\in$ with $\notin$, or vice versa
4) De Morgan's Law (Quantifier Negation)	$\neg\forall x\ P(x)$	≡	$\exists x\ \neg P(x)$	De Morgan's Law for a single quantifier
	$\neg\exists x\ P(x)$	≡	$\forall x\ \neg P(x)$	
	$\neg\forall x\ \forall y\ P(x, y)$	≡	$\exists x\ \exists y\ \neg P(x, y)$	De Morgan's Law for nested quantifiers
	$\neg\forall x\ \exists y\ P(x, y)$	≡	$\exists x\ \forall y\ \neg P(x, y)$	
	$\neg\exists x\ \forall y\ P(x, y)$	≡	$\forall x\ \exists y\ \neg P(x, y)$	
	$\neg\exists x\ \exists y\ P(x, y)$	≡	$\forall x\ \forall y\ \neg P(x, y)$	
5) Nested / Multiple-Quantified Statements	$\forall x\ \forall y\ P(x, y)$	≡	$\forall y\ \forall x\ P(x, y)$	• for all objects x and y, …
	$\exists x\ \exists y\ P(x, y)$	≡	$\exists y\ \exists x\ P(x, y)$	• there are objects x and y such that …
	$\forall x\ \exists y\ P(x, y)$	$\not\equiv$	$\exists x\ \forall y\ P(x, y)$	Not equivalent in general. Counterexample for $x, y \in \mathbb{Z}$: $\forall x\ \exists y\ (x + y = 0) \equiv$ True $\exists x\ \forall y\ (x + y = 0) \equiv$ False
	$\neg\ (\forall x\ \exists y\ P(x, y))$	≡	$\exists x\ \forall y\ \neg P(x, y)$	Negation of multiple-quantified statements
	$\neg(\exists x\ \forall y\ P(x, y))$	≡	$\forall x\ \exists y\ \neg P(x, y)$	
6) Moving Quantifiers	$\forall x\ (P(x) \rightarrow \exists y\ Q(x, y))$	≡	$\forall x\ \exists y\ (P(x) \rightarrow Q(x, y))$	You can move a quantifier left subject to the condition stated below.

Note: Moving a quantifier requires the variable not to occur **free** in the crossed formula and must avoid variable capture.

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Valid Quantifier Formulas

A	Direction	B
$\forall x\,(P(x) \wedge Q(x))$	$\equiv$	$\forall x\,P(x) \wedge \forall x\,Q(x)$
$\exists x\,(P(x) \wedge Q(x))$	$\rightarrow$	$\exists x\,P(x) \wedge \exists x\,Q(x)$
$\forall x\,(P(x) \vee Q(x))$	$\leftarrow$	$\forall x\,P(x) \vee \forall x\,Q(x)$
$\exists x\,(P(x) \vee Q(x))$	$\equiv$	$\exists x\,P(x) \vee \exists x\,Q(x)$
$\forall x\,(P(x) \rightarrow Q(x))$	$\leftarrow$	$\exists x\,P(x) \rightarrow \forall x\,Q(x)$
$\exists x\,(P(x) \rightarrow Q(x))$	$\equiv$	$\forall x\,P(x) \rightarrow \exists x\,Q(x)$
$\forall x\,\neg P(x)$	$\equiv$	$\neg\exists x\,P(x)$
$\exists x\,\neg P(x)$	$\equiv$	$\neg\forall x\,P(x)$
$\forall x\,\exists y\,T(x, y)$	$\leftarrow$	$\exists y\,\forall x\,T(x, y)$
$\forall y\,\exists x\,T(x, y)$	$\leftarrow$	$\exists x\,\forall y\,T(x, y)$
$\forall x\,\forall y\,T(x, y)$	$\equiv$	$\forall y\,\forall x\,T(x, y)$
$\exists x\,\exists y\,T(x, y)$	$\equiv$	$\exists y\,\exists x\,T(x, y)$
$\forall x\,(P(x) \vee R)$	$\equiv$	$\forall x\,P(x) \vee R$
$\exists x\,(P(x) \wedge R)$	$\equiv$	$\exists x\,P(x) \wedge R$
$\forall x\,(P(x) \rightarrow R)$	$\equiv$	$\exists x\,P(x) \rightarrow R$
$\exists x\,(P(x) \rightarrow R)$	$\rightarrow$	$\forall x\,P(x) \rightarrow R$
$\forall x\,(R \rightarrow Q(x))$	$\equiv$	$R \rightarrow \forall x\,Q(x)$
$\exists x\,(R \rightarrow Q(x))$	$\rightarrow$	$R \rightarrow \exists x\,Q(x)$
$\forall x\,R$	$\leftarrow$	R
$\exists x\,R$	$\rightarrow$	R

Note: The above formulas are valid in classical first-order logic, assuming that *x* does not occur free in *R*. The arrows → and ← are the directions of the displayed valid implications.

Invalid Quantifier Formulas

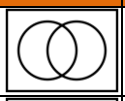
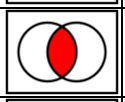
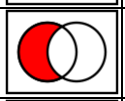
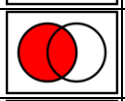
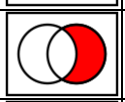
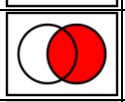
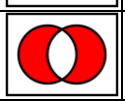
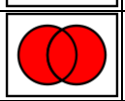
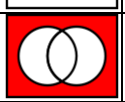
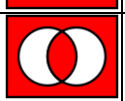
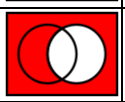
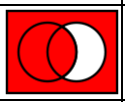
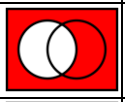
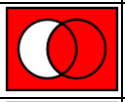
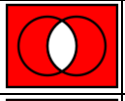
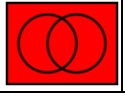
A		B	Counterexample
$\exists x\,(P(x) \wedge Q(x))$	$\leftarrow$	$\exists x\,P(x) \wedge \exists x\,Q(x)$	D = {*a*, *b*}, M = {*P*(*a*), *Q*(*b*)}
$\forall x\,(P(x) \vee Q(x))$	$\rightarrow$	$\forall x\,P(x) \vee \forall x\,Q(x)$	D = {*a*, *b*}, M = {*P*(*a*), *Q*(*b*)}
$\forall x\,(P(x) \rightarrow Q(x))$	$\rightarrow$	$\exists x\,P(x) \rightarrow \forall x\,Q(x)$	D = {*a*, *b*}, M = {*P*(*a*), *Q*(*a*)}
$\forall x\,\exists y\,T(x, y)$	$\rightarrow$	$\exists y\,\forall x\,T(x, y)$	D = {*a*, *b*}, M = {*T*(*a*, *b*), *T*(*b*, *a*)}
$\exists x\,(P(x) \rightarrow R)$	$\leftarrow$	$\forall x\,P(x) \rightarrow R$	D = Ø, M = {*R*}
$\exists x\,(R \rightarrow Q(x))$	$\leftarrow$	$R \rightarrow \exists x\,Q(x)$	D = Ø, M = Ø
$\forall x\,R$	$\rightarrow$	R	D = Ø, M = Ø
$\exists x\,R$	$\leftarrow$	R	D = Ø, M = {*R*}

Notes:

- M lists the true atomic statements; unlisted atomic statements are false.
- The first four implication domains are nonempty.
- The last four implications fail only when empty domains are allowed. They are valid when the domain is nonempty.

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

The Sixteen Logical Operations on Two Variables (**red** indicates true)

#	Venn	Sym	Logical Notation(s)	Name(s)
0000		$\bot$	0	Contradiction; falsehood; antilogy; constant 0
0001		$\wedge$	$x \wedge y, \quad xy, \quad x \,\&\, y$	Conjunction; AND
0010		$\supsetneq$	$x \wedge \bar{y}, x \not\supset y, [x > y], x \div y$	Nonimplication; difference; but not
0011		$\llcorner$	x	Left projection
0100		$\subsetneq$	$\bar{x} \wedge y, x \not\subset y, [x < y], y \div x$	Converse nonimplication; not ... but
0101		R	y	Right projection
0110		$\oplus$	$x \oplus y, \quad x \not\equiv y, \quad x \hat{} y$	Exclusive disjunction; nonequivalence; XOR
0111		$\vee$	$x \vee y, \quad x \mid y$	(Inclusive) disjunction; and/or; OR
1000		$\bar{\vee}$	$\bar{x} \wedge \bar{y}, \overline{x \vee y}, x \,\bar{\vee}\, y, x \downarrow y$	Nondisjunction; joint denial; neither... NOR
1001		$\equiv$	$x \equiv y, \quad x \leftrightarrow y, \quad x \Leftrightarrow y$	Equivalence; if and only if; IFF
1010		$\bar{\mathrm{R}}$	$\bar{y}, \quad \neg y, \quad !\, y, \quad \sim y$	Right complementation; NOT
1011		$\subset$	$x \vee \bar{y}, x \subset y, x \Leftarrow y, [x \geq y], x^y$	Converse implication; IF
1100		$\bar{\llcorner}$	$\bar{x}, \quad \neg x, \quad !\, x, \quad \sim x$	Left complementation; NOT
1101		$\supset$	$\bar{x} \vee y, x \supset y, x \Rightarrow y, [x \leq y], y^x$	Implication; only if; if ... then
1110		$\bar{\wedge}$	$\bar{x} \vee \bar{y}, \overline{x \wedge y}, x \,\bar{\wedge}\, y$	Nonconjunction; not both ... and; NAND
1111		$\top$	1	Affirmation; validity; tautology; constant 1

Donald E. Knuth (January 2008). 7.1.1 Boolean Basics, *The Art of Computer Programming*, Pre-fascicle 0B: The sixteen logical operations in two variables. See also Wikipedia, Truth function.

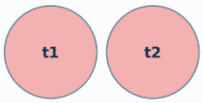
Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Twelve SQL Query and Join Patterns

T-SQL • Match condition: t1.fk = t2.id, except where indicated

Light red marks selected regions; Venn diagrams are conceptual, not row-count diagrams.

Separate SELECT statements

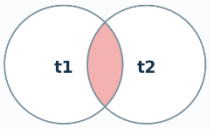



```
SELECT * FROM Table1;

SELECT * FROM Table2;
```

Two independent result sets; this is not a join.

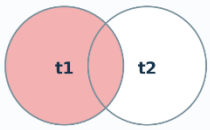
INNER JOIN




```
SELECT *
FROM Table1 AS t1
INNER JOIN Table2 AS t2
  ON t1.fk = t2.id;
```

Returns matching row pairs.

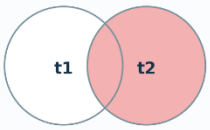
LEFT OUTER JOIN




```
SELECT *
FROM Table1 AS t1
LEFT JOIN Table2 AS t2
  ON t1.fk = t2.id;
```

All t1 rows; unmatched t2 columns are NULL.

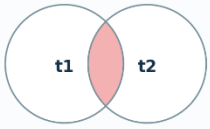
RIGHT OUTER JOIN




```
SELECT *
FROM Table1 AS t1
RIGHT JOIN Table2 AS t2
  ON t1.fk = t2.id;
```

All t2 rows; unmatched t1 columns are NULL.

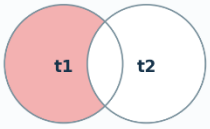
LEFT SEMI JOIN (EXISTS)




```
SELECT t1.*
FROM Table1 AS t1
WHERE EXISTS (
  SELECT 1 FROM Table2 AS t2
  WHERE t1.fk = t2.id
);
```

Returns only t1 rows having at least one match.

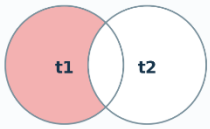
LEFT ANTI SEMI JOIN (NOT EXISTS)




```
SELECT t1.*
FROM Table1 AS t1
WHERE NOT EXISTS (
  SELECT 1 FROM Table2 AS t2
  WHERE t1.fk = t2.id
);
```

Returns only t1 rows having no match.

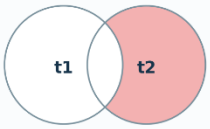
LEFT JOIN • unmatched rows




```
SELECT t1.*
FROM Table1 AS t1
LEFT JOIN Table2 AS t2
  ON t1.fk = t2.id
WHERE t2.id IS NULL;
```

Anti-join pattern; see the NULL note below.

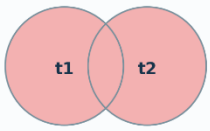
RIGHT JOIN • unmatched rows




```
SELECT t2.*
FROM Table1 AS t1
RIGHT JOIN Table2 AS t2
  ON t1.fk = t2.id
WHERE t1.fk IS NULL;
```

Anti-join pattern; see the NULL note below.

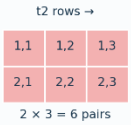
FULL OUTER JOIN




```
SELECT *
FROM Table1 AS t1
FULL OUTER JOIN Table2 AS t2
  ON t1.fk = t2.id;
```

Matching pairs plus unmatched rows from both tables.

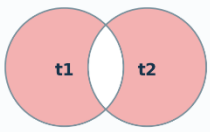
CROSS JOIN • Cartesian product




```
SELECT *
FROM Table1 AS t1
CROSS JOIN Table2 AS t2;
```

Every t1 row pairs with every t2 row: m × n rows.

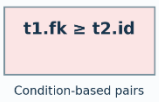
FULL OUTER JOIN • unmatched rows




```
SELECT *
FROM Table1 AS t1
FULL OUTER JOIN Table2 AS t2
  ON t1.fk = t2.id
WHERE t1.fk IS NULL
   OR t2.id IS NULL;
```

Unmatched rows from either table; excludes matches.

NON-EQUI INNER JOIN




```
SELECT *
FROM Table1 AS t1
INNER JOIN Table2 AS t2
  ON t1.fk >= t2.id;
```

Returns every pair satisfying the inequality.

NULL AND MULTIPLICITY NOTES

- NOT EXISTS and the unmatched-row patterns are not unconditional replacements for NOT IN.
  NOT IN may evaluate to UNKNOWN when its compared values include NULL; WHERE discards UNKNOWN.
- The IS NULL tests above detect unmatched rows for the displayed equality join; NULL never matches by =.
- Joins can multiply rows when several rows match. EXISTS keeps each qualifying t1 row once, preserving input duplicates.
- SQL uses TRUE, FALSE, and UNKNOWN. These diagrams summarize matching, not classical set operations.

Adapted and redrawn from Steve Stedman's SQL join-pattern chart.

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor

Sources

- SNHU MAT 230 - Discrete Mathematics, zyBooks.
- https://byjus.com/maths/set-theory-symbols/
- https://en.wikipedia.org/wiki/List_of_logic_symbols
- https://en.wikipedia.org/wiki/Truth_function#Table_of_binary_truth_functions
- https://nokyotsu.com/qscripts/2014/07/distribution-of-quantifiers-over-logic-connectives.html
- Knuth, Donald E. (January 2008). 7.1.1 Boolean Basics, *The Art of Computer Programming*, Pre-fascicle 0B: The sixteen logical operations in two variables.
- Stedman, Steve (2025). TSQL Join Types, version 22.03. https://stedmansolutions.kit.com/1296307228

Acknowledgements

- Proofread with assistance from ChatGPT 6.1 Sol, https://chatgpt.com/

See Also

- Harold's Logic (Mathematics) Cheat Sheet
- Harold's Logic (Philosophy) Cheat Sheet
- Harold's Boolean Algebra Cheat Sheet
- Harold's Proofs Cheat Sheet
- Harold's Sets Cheat Sheet

Copyright © 2021-2026 by Harold Toomey, Wyzant Tutor