**Harold’s Logic (Philosophy)**

**Cheat Sheet**

1 October 2025

**The 7 Basic Logical Symbols**

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Symbol** | **Example** | **English** |
| **1) Intersection (AND)** | ⋂,• | A • B | * Conjunction
* A and B
* A, but B
* despite the fact that A, B
* even though A, B
* although A, B
* overlap
 |
| **2) Union (OR)** | ⋃, ∨ | A ∨ B | * Disjunction
* A or B
* inclusive or
* both combined
 |
| **3) Negation (NOT)** | ~, $\overbar{A}$ | ~A | * not A
 |
| **4) Conditional** | →, ⊃ | A ⊃ B | * if A then q
* if A, B
* B if A
* A implies B
* A only if B
* B in case that A
* A is sufficient for B
* B is necessary for A
 |
| **5) Biconditional** | **↔,** ⟷, **↔**, ⇔, ⟺ | p ⟷q | * A iff B
* A if and only if B
* A is necessary and sufficient for B
* if A then B, and conversely
* if not A then not B, and conversely
 |
| **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
 |
| **Equivalence**(See Biconditional) | $$≡$$ | expression1 ≡ expression2 | * is identical to
* is equivalent to
* is defined as
* the two expressions always have the same truth value
 |
| “… *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](https://ia800501.us.archive.org/7/items/how-to-prove-it-a-structured-approach-daniel-j.-velleman/How%20to%20Prove%20It%20A%20Structured%20Approach%20%28Daniel%20J.%20Velleman%29.pdf)”, 3rd Edition, p. 75. |

**Logical Truth Tables**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **AND**• | **NOT AND**~• | **OR**∨ | **NOT OR****~**∨ | **XOR**⊻,⊕ | **NOT XOR**⊙ | **NOT**~(~A) | **If … Then**⊃ | **Iff**$$≡$$ | **Taut-ology****(True)**⊤ | **Contra-diction****(False)**F |
| **F** | **F** | F | T | F | T | F | T | T | T | T | T | F |
| **F** | **T** | F | T | T | F | T | F | T | T | F | T | F |
| **T** | **F** | F | T | T | F | T | F | F | F | F | T | F |
| **T** | **T** | T | F | T | F | F | T | f | T | T | T | F |

**Blank Truth Tables**

|  |  |
| --- | --- |
| **Inputs** | **Output** |
| **A** | **B** | **C** | **D** | **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** |
| **A** | **B** | **C** | **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** |
| **A** | **B** | **X** |
| **F** | **F** |  |
| **F** | **T** |  |
| **T** | **F** |  |
| **T** | **T** |  |

**Precedence Rules (PEMDAS for Logic)**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Operator** | **Symbol** | **Precedence** |
| **1** | **Parenthesis** | ( ) | Highest precedence |
| **2** | **NOT** | ~ |  |
| **3** | **Quantifiers** | (x)*,* (∃x) |  |
| **4** | **AND** | • | Applied Left to Right |
| **5** | **OR** | ∨ |  |
| **6** | **Conditional** | ⊃ |  |
| **7** | **Biconditional** | $$≡$$ | Lowest precedence |

**Logical Conditional Connective Laws**

|  |  |  |  |
| --- | --- | --- | --- |
| **Law or Statement** | **Logical Expression** | **Is Equivalent To****(≡)** | **Description** |
| **Conditional Laws** | p ⊃ q | ~p ∨ q~(p • ~q)Logical Equivalences:p ∨ q ≡ ~p ⊃ qp • q ≡ ~(p ⊃ ~q)~(p ⊃ q) ≡ p • ~q(p ⊃ q) • (p ⊃ r) ≡ p ⊃ (q • r)(p ⊃ q) ∨ (p ⊃ r) ≡ p ⊃ (q ∨ r)(p ⊃ r) • (q ⊃ r) ≡ (p • q) ⊃r (p ⊃ r) ∨ (q ⊃ r) ≡ (p ∨ q) ⊃r | Conditional, If ... Then, Implication |
| **Biconditional Laws**(Equivalence) | p ≡ qp **↔** q | (p ⊃ q) • (q ⊃ p)(p ⊃ q) • (~p ⊃ ~q)(p • q) ∨ (~p • ~q)~p **↔** ~qLogical Equivalences:~ (p **↔** q) ≡ p **↔** ~q | Bi-conditional, If and only If, iff, XNORIs equivalent to |
| **Converse\*** | p ⊃ q | ≢ q ⊃ p | False |
| **Inverse\*** | p ⊃ q | ≢ ~p ⊃ ~q | False |

**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. In conclusion. |
| **1) Modus Ponens (MP)** | $$\frac{\begin{matrix}p\\p⊃q\end{matrix}}{∴q}$$ | 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)** | $$\frac{\begin{matrix}\~q\\p⊃q\end{matrix}}{∴\~p}$$ | 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) | $$\frac{\begin{matrix}p⊃q\\q⊃r\end{matrix}}{∴p⊃r}$$ | 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) | $$\frac{\begin{matrix}p ∨ q\\\~p\end{matrix}}{∴q}$$ | 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)** | $$\frac{\begin{matrix} p ∨ q\\ \left(p⊃r\right)•(q⊃s)\end{matrix}}{∴r ∨s}$$ | 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) | $$\frac{p • q}{∴p}$$ | It is rainy today and it is windy today.Therefore, it is rainy today. |
| **7) Conjunction (Conj)** | $$\frac{\begin{matrix}p\\q\end{matrix}}{∴p • q}$$ | Sam studied for his test. Sam passed his test. Sam studied for his test and passed his test. |
| **8) Addition (Add)**(Generalization) | $$\frac{p}{∴p ∨ q}$$ | It is raining today. Therefore, it is either raining today or snowing today or both. |
| **Resolution** | $$\frac{\begin{matrix} p ∨ q\\\~p ∨ q\end{matrix}}{∴q ∨ r}$$ | Your shirt is red or your pants are blue. Your shirt is not red or your pants are blue. Therefore, your pants are blue or your shoes are white. |
| **Proof by Division into Cases** | $$\frac{\begin{matrix}\begin{matrix} p ∨ q\\p⊃r\end{matrix}\\q⊃r\end{matrix}}{∴r}$$ | It is raining or it is Monday.It is raining, so it is wet.It is Monday, so it is wet.It is wet. |
| **Contradiction Rule** | $$\frac{\~p⊃F}{∴p}$$ | If it is not raining is a false statement; then it is raining. |

**Rules of Replacement**

(Logical Connective Laws / Equivalences)

|  |  |  |
| --- | --- | --- |
| **Law** | **Or Example** | **And Example** |
| **9. De Morgan’s rule (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(p • ~q) ⊃ r ≡ ~r ⊃ (~p ∨ q) |
| **10. Commutative (Com)** | p ∨ q ≡ q ∨ p | p • q ≡ q • p |
| **11. Associative (Assoc)** | (p ∨ q) ∨ r ≡ p ∨ (q ∨ r) | (p • q) • r ≡ p • (q • r) |
| **12. Distributive (Dist)** | p • (q ∨ r) ≡ (p • q) ∨ (p • r) | p ∨ (q • r) ≡ (p ∨ q) • (p ∨ r) |
| **13. Double Negations (DN)**(Involution Law) | ~ ~p ≡ p |
| **14. Transposition (Trans)**(Contrapositive) | (p ⊃ q) ≡ (~q ⊃~p) |
| **15. Material Implication (Impl)** | (p ⊃ q) ≡ (~p ∨ q) |
| **16. Material Equivalence (Equiv)** | (p ≡ q) ≡ [(p • q) ∨ (~p • ~q)] | (p ≡ q) ≡ [(p ⊃ q) • (q ⊃ p)] |
| **17. Exportation (Exp)** | [(p ∨ q) ⊃ r] ≡ [(p ⊃ r) ∨ (q ⊃ r)] | [(p • q) ⊃ r] ≡ [p ⊃ (q ⊃ r)] |
| **18. Tautology (Taut)**(Idempotent) | p ≡ (p ∨ p) | p ≡ (p • p) |
| **Contradiction**(Identity) | p ∨ F ≡ p | p • T ≡ p |
| **Domination, Null**(Universal Bound Laws) | p ∨ T ≡ T | p • F ≡ F |
| **Negation, Complement**(Complementary Laws) | p ∨ ~p ≡ T~F ≡ T | p • ~p ≡ F~T ≡ F |
| **Uniting** | (p • q) ∨ (p • ~q) ≡ p | (p ∨ q) • (p ∨ ~q) ≡ p |
| **Absorption** | p ∨ (p • q) ≡ p | p • (p ∨ q) ≡ p |
| **Multiplying and Factoring Laws** | (p ∨ q) • (~p ∨ r) ≡ (p • r) ∨ (~p • q) | (p • q) ∨ (~p • r) ≡ (p ∨ r) • (~p ∨ q) |
| **Consensus** | (p • q) ∨ (q • r) ∨ (~p • r) ≡ (p • q) ∨ (~p • r) | (p ∨ q) • (q ∨ r) • (~p ∨ r) ≡ (p ∨ q) • (~p ∨ r) |
| **Exclusive Or (**⊕**)** | p ⊕ q ≡ (p ∨ q) ∨ ~(p • q) | p ⊕ q ≡ (~p • q) ∨ (p ∨ ~q)  |

**Proof Methods**

|  |  |
| --- | --- |
| **Method** | **Definition** |
| **Direct** | * The conclusion is established by logically combining the axioms, definitions, and earlier theorems.
* When given *P* ⊃ *Q*, assume *P* is true, then prove *Q*.
 |
| **Indirect**(Contradiction) | * If some statement is assumed true, and a logical contradiction occurs, then the statement must be false.
* Or assume that the theorem is false and then show that some logical inconsistency arises as a result of the assumption, such as r • ~r.
* Indirect proof.
* Can also be a proof by counterexample. E.g., Assume ~(p ⊃ q), which is equivalent to p • ~q.
 |
| **Conditional** | * A **conditional proof** is a structured argument that assumes the **antecedent** (*p*) of a conditional statement and then shows that this assumption logically leads to the **consequent** (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.
 |
| **Contrapositive** | * Infers the statement p⊃ *q* by establishing the logically equivalent contrapositive statement: *¬*q⊃ *~*p.
* When given p⊃ *q*, assume *~*q is true, then prove *~*p.
* We prove that if the negation of the original conclusion is false, then the negation of the initial theorem is false.
* Relies on De Morgen's Law.
* Modus tollens.

|  |  |  |  |
| --- | --- | --- | --- |
| **p** | **q** | **If** ⊃ **Then** | **Technique** |
| **F** | **F** | T | Modus Tollens |
| **F** | **T** | T |  |
| **T** | **F** | F |  |
| **T** | **T** | T | Modus Ponens |

* A proof by contrapositive is a special case of a proof by contradiction (indirect).
 |
| **Construction** | * 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.
 |

**Logical Quantifiers**

|  |  |  |  |
| --- | --- | --- | --- |
| **Definition** | **Logical Expression** | **Is Equivalent To (≡)** | **Plain English** |
| **Universal Quantifier** **(x)** | (*x*) *P(x)*(*x*) ∈ *P(x)*(*x*) ∈ 𝔻*, P(x)*(*x*)*, if x is in* 𝔻 *then P(x)* | “For all *x* in the domain, *P(x)* is true”(*x*) ∈ *A P(x)* ≡(*x*) *(x* ∈ *A* ⊃ *P(x))*For the finite set domain of discourse {a1, a2, …, ak}, (*x*) *P(x)* ≡ *P(a1)* • *P(a2)* • *…* • *P(ak)* | * for all
* all elements
* for each member
* any
* every
* everyone
* everybody
* everything
* x could be anything at all
 |
| **Existential Quantifier****(∃x)** | (∃*x*) *P(x)*(∃*x*) ∈ *P(x)*(∃*x*) ∈𝔻*, P(x)* | “There exists x in the domain, such that *P(x)* is true”For the finite set domain of discourse {a1, a2, …, ak}, (∃*x*) *P(x)* ≡ *P(a1)* ∨ *P(a2)* ∨ *…* ∨ *P(ak)**P(x)* ≠ ∅ | * there exists an x
* there is
* some
* someone
* somebody
* at least one value of x
* there is at least one x
* it is the case that
* the truth set is not equal to ∅
 |
| **Uniqueness Quantifier****(∃!)** | ∃!*x P(x)* | there is a unique x in *P(x)* such that …(∃*x*) (*P(x)* • ~(*y*) *(P(y)* • *y* ≠ *x))*(∃*x*) (*P(x)* • (*y*) *(P(y)* ⊃ *y* = *x))*​(∃*x*) (*y*) *(P(y)* ≡ *y* = *x*)​(∃*x*) *P(x)* • (*y*) (*z*) *((P(y)* • *P(z))* ⊃ *y* = *z*) | * unique
* there is a unique x
* there exists exactly one
* there is exactly one x such that *P(x)*
 |
| **Negated Existential Quantifier** | ~ [(∃*x*) *P(x)*] | (*x*)~*P(x)* | * nobody
* no one
* not one
* there does not exist
 |
| ~ [(*x*) *P(x)*] | (∃*x*)~*P(x)* |

**Rules of Inference with Quantifiers**

|  |  |  |
| --- | --- | --- |
| **Rule Name** | **Rule Logic** | **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** | c is an element (arbitrary or particular)(x) P(x)∴ P(c) | Sam is a student in the class.Every student in the class completed the assignment.Therefore, Sam completed his assignment. |
| **Universal Generalization** | c is an arbitrary elementP(c) .∴(x) P(x) | Let c be an arbitrary integer.c ≤ c2Therefore, every integer is less than or equal to its square. |
| **Existential Instantiation\*** | (∃x) P(x)∴ (c is a particular element) • P(c) | There is an integer that is equal to its square. Therefore, c2 = c, for some integer c.i.e., If an object is known to exist, then that object can be given a name. |
| **Existential Generalization** | c is an element (arbitrary or particular)P(c) .∴(∃x) P(x) | Sam is a particular student in the class.Sam completed the assignment.Therefore, there is a student in the class who completed the assignment. |

**Quantifier Laws**

|  |  |  |  |
| --- | --- | --- | --- |
|  **Definition** | **Logical Expression** | **Is Equivalent To (≡)** | **Plain English** |
| **Abbreviation** | (∃x) (x ∈ A • ~*P(x)*) | (∃x) ∈ A ~*P(x)* | Simplification |
| **Expanding Abbreviation** | (x) ∈ A *P(x)* | (x) (x ∈ A ⊃ *P(x)*) | Complication |
| **Quantifier Negation Laws** | (x) ~*P(x)* | ~(∃x) *P(x)* | * nobody’s perfect
 |
| ~(x) *P(x)* | (∃x) ~*P(x)* | * not everyone is perfect
* someone is imperfect
 |
| **Conditional Law** | x ∈ A ⊃ *P(x)* | x ∉ A ∨ *P(x)* | p ⊃ q ≡ ~p ∨ q |
| **Subset Negation Law** | x ∈ A | ~(x ∉ A) | Swap ∈ with ∉, or vice versa |
| **De Morgan’s Law (Quantifier Negation)** | ~(x) *P(x)* ≡ (∃x)~*P(x)*~(∃x) *P(x)* ≡(*x*) ~*P(x)*~(x) (y) *P(x, y)* ≡ (∃*x*)(∃*y*)~*P(x, y)*~(x) (∃x) *P(x, y)* ≡ (∃*x*)(y)~*P(x, y)*~(∃x) (y) *P(x, y)* ≡ (x)(∃*y*)~*P(x, y)*~(∃x) (∃y) *P(x, y)* ≡ (x)(y)~*P(x, y)* | De Morgan’s Law for single and nested quantifiers |
| **Nested / Multiple- Quantified Statements** | (x) (y) | (y) (x) | * for all objects x and y, …
 |
| (∃x) (∃y) | (∃y) (∃x) | * there are objects x and y such that …
 |
| (x) (∃y) *P(x, y)* ≢ (∃x) (y) *P(x, y)* | FalseCounterexample for x, y ∈ ℤ: (x) (∃y) (*x + y = 0*) ≡ True(∃x) (y) (*x + y = 0*) ≡ False |
| ~((x) (∃y) *P(x, y)*) | (∃x) (y) ~*P(x, y)* | Negation of multiply-quantified statements |
| ~((∃x) (y) *P(x, y)*) | (x) (∃y) ~*P(x, y)* |
| **Moving Quantifiers** | (x) (*P(x)* ⊃ (∃y) *Q(x, y)*) ≡(x) (∃y) (*P(x)* ⊃ *Q(x, y)*) | You can move a quantifier left if the variable is not used yet |

**Quantifier Logic Examples**

|  |  |  |
| --- | --- | --- |
| **Action** | **Logical Statement** | **Plain English** |
| **Everyone** | (x) (y) *P(x, y)* NOTE: includes (x = y) | * everyone <did something> to everyone
 |
| **Everyone Else** | (x) (y) (*x* ≠ *y*) ⊃ *P(x, y)* NOTE: excludes (x = y) | * everyone <did something> to everyone else
 |
| **Someone Else** | (x) (∃y) ((*x* ≠ *y*) • *P(x, y)*) NOTE: excludes (x = y) | * everyone <did something> to someone else
 |
| **Exactly One** | (∃x) (*P(x)* • (y) ((*x* ≠ *y*) ⊃ ~*P(y)*)) ≡ ∃!*x P(x)* | * exactly one person <did something>
 |
| **No One** | ~(∃x) *P(x)* | * no one <did something>
 |

**Valid Quantifier Formulas**

|  |  |  |
| --- | --- | --- |
| **A** |  | **B** |
| (x) (*P(x)* • *Q(x)*) | ≡ | ((x) *P(x)* • (x) *Q(x)*) |
| (∃x) (*P(x)* • *Q(x)*) | **→** | ((∃x) *P(x)* • (∃x) *Q(x)*) |
| (x) (*P(x)* ∨ *Q(x)*) | **←** | ((x) *P(x)* ∨ (x) *Q(x)*) |
| (∃x) (*P(x)* ∨ *Q(x)*) | ≡ | ((∃x) *P(x)* ∨ (∃x) *Q(x)*) |
| (x) (*P(x)* ⊃ *Q(x)*) | **←** | ((∃x) *P(x)* ⊃ (x) *Q(x)*) |
| (∃x) (*P(x)* ⊃ *Q(x)*) | ≡ | ((x) *P(x)* ⊃ (∃x) *Q(x)*) |
| (x) ~*P(x)* | ≡ | ~(∃x) *P(x)* |
| (∃x) ~*P(x)* | ≡ | ~(x) *P(x)* |
| (x) (∃y) *T(x, y)* | **←** | (∃y) (x) *T(x, y)* |
| (x) (y) *T(x, y)* | ≡ | (y) (x) *T(x, y)* |
| (∃x) (∃y) *T(x, y)* | ≡ | (∃y) (∃x) *T(x, y)* |
| (x) (*P(x)* ∨ *R*) | ≡ | ((x) *P(x)* ∨ *R*) |
| (∃x) (*P(x)* • *R*) | ≡ | ((∃x) *P(x)* • *R*) |
| (x) (*P(x)* ⊃ *R*) | ≡ | ((∃x) *P(x)* ⊃ *R*) |
| (∃x) (*P(x)* ⊃ *R*) | **→** | ((x) *P(x)* ⊃ *R*) |
| (x) (*R* ⊃ *Q(x)*) | ≡ | (*R* ⊃ (x) *Q(x)*) |
| (∃x) (*R* ⊃ *Q(x)*) | **→** | (*R* ⊃ (∃x) *Q(x)*) |
| (x) *R* | **←** | *R* |
| (∃x) *R* | **→** | *R* |

**Note**: The above formulas are valid in classical [first-order logic](https://en.wikipedia.org/wiki/First-order_logic), assuming that *x* does not occur free in *R*.

**Invalid Quantifier Formulas**

|  |  |  |  |
| --- | --- | --- | --- |
| **A** |  | **B** | **Counterexample** |
| (∃x) (*P(x)* • *Q(x)*) | **←** | ((∃x) *P(x)* • (∃x) *Q(x)*) | D = *{a, b}*, M = {*P(a)*, *Q(b)*} |
| (x) (*P(x)* ∨ *Q(x)*) | **→** | ((x) *P(x)* ∨ (x) *Q(x)*) | D = *{a, b}*, M = {*P(a)*, *Q(b)*} |
| (x) (*P(x)* ⊃ *Q(x)*) | **→** | ((∃x) *P(x)* ⊃ (x) *Q(x)*) | D = *{a, b}*, M = {*P(a)*, *Q(a)*} |
| (x) (∃y) *T(x, y)* | **→** | (∃y) (x) *T(x, y)* | D = *{a, b}*, M = {*T(a, b)*, *T(b, a)*} |
| (∃x) (*P(x)* ⊃ *R*) | **←** | ((x) *P(x)* ⊃ *R*) | D = Ø, M = {*R*} |
| (∃x) (*R* ⊃ *Q(x)*) | **←** | (*R* ⊃ (∃x) *Q(x)*) | D = Ø, M = Ø |
| (x) *R* | **→** | *R* | D = Ø, M = Ø |
| (∃x) *R* | **←** | *R* | D = Ø, M = {*R*} |

**Note**: if empty domains are not allowed, then the last four implications above are in fact valid.

**Sources**

* Hurley, Patrick J. (2024). [A Concise Introduction to Logic](https://www.amazon.com/Concise-Introduction-Logic-Patrick-Hurley/dp/0357798686/ref%3Dsr_1_1), 14th Edition, Cengage Learning, Inc.
* Wikipedia (2025).
	+ <https://en.wikipedia.org/wiki/List_of_logic_symbols>
	+ <https://en.wikipedia.org/wiki/Truth_function#Table_of_binary_truth_functions>

**See Also**

* [Harold’s Logic Cheat Sheet](https://www.toomey.org/tutor/harolds_cheat_sheets/Harolds_Logic_Cheat_Sheet.pdf)
* [Harold’s Logic (Philosophy) Cheat Sheet](https://www.toomey.org/tutor/harolds_cheat_sheets/Harolds_Logic_%28Philosophy%29_Cheat_Sheet.pdf)
* [Harold’s Sets Cheat Sheet](https://www.toomey.org/tutor/advanced_math.html)
* [Harold’s Boolean Algebra Cheat Sheet](https://www.toomey.org/tutor/discrete_math.html)
* [Harold’s Proofs Cheat Sheet](https://www.toomey.org/tutor/advanced_math.html)