6.1.s. Summary

6.1.1. We move beyond truth-functional logic by recognizing complete expressions other than sentences and operations other than connectives. Our additions are motivated by a traditional description of grammatical subjects and predicates. The new complete expressions are individual terms, whose function is to name. Given this idea, we can define a predicate as an operation that forms a sentence from one or more individual terms.

6.1.2. A predicate corresponds to an English sentence with blanks that might be filled by terms. These blanks are the predicate's places and the operation of filling them is predication. We will maintain something analogous to truth-functionality by requiring that predicates be extensional. This means that all places of a predicate must be referentially transparent (rather than referentially opaque): when judging the truth value of a sentence formed by the predicate, we must be able see through the terms filling these places to what those terms refer to. Thus, just as a connective expresses a truth function, a predicate expresses a function that takes reference values as input and issues truth values as output. Such a function may be called an attribute—or, more specifically, a property if it has one place and a relation if it has 2 or more. In symbolic notation, it takes the form $\sigma = \tau$ and, in English notation, it takes the form σ `is` τ.

6.1.3. While recognizing quite a variety of non-logical vocabulary in our analyses, we recognize only one new item of logical vocabulary, the predicate identity. This is a 2-place predicate that forms an equation, which is true when its component terms have the same reference value.

6.1.4. Lambda abstraction provides notation for linking the places of a predicate to blanks in an English sentence. An expression formed using it—which will have the general form $\lambda x_1 \ldots x_n$ (... x_1 ... x_n ...)—is an abstract (in this use, a predicate abstract); it consists of a lambda operator applied to a parenthesized body. In English notation, a predicate abstract takes the form `the attribute of` $x_1 \ldots x_n$ `that` ... x_1 ... x_n ... . Variables in the body of an abstract are bound to the lambda operator. Expressions that establish the same patterns of binding using different variables are alphabetic variants. They may be thought of as pronouns whose antecedent is the lambda operator. An expression (such as the body of an abstract) that has variables not bound to lambda operators, is not a sentence in the strict sense, but it does count as a formula. Formulas have many of the syntactic properties of sentences; in particular, they can be built from other formulas using connectives. And we can distinguish as atomic formulas not only unanalyzed sentences but all formulas that are predictions. (Indeed, unanalyzed sentences can be thought of as predications of zero-place predicates.)

6.1.5. In our symbolic notation, we use lower case letters to stand for unanalyzed individual terms, the equal sign for identity, and capital letters to stand for non-logical predicates. Non-logical predicates, both capital letters and predicate abstracts are written in front of the terms they apply to (with a predicate abstract enclosed in brackets), and = is written between the terms to which it applies. In English notation, predications other than equations are written as θ `fits` τ_1, ..., `'n` τ_n.

6.1.x. Exercise questions

1. Analyze each of the following sentences in as much detail as possible.

a. *Ann introduced Bill to Carol.*

b. *Ann gave the book to either Bill or Carol.*

c. *Ann gave the book to Bill and he gave it to Carol.*

d. *Tom had the package sent to Sue, but it was returned to him.*

e. *Georgia will see Ed if she gets to Denver before Saturday.*

f. *If the murderer is either the butler or the nephew, then I'm Sherlock Holmes.*

g. *Neither Ann nor Bill saw Tom speak to either Mike or Nancy.*

h. *Tom will agree if each of Ann, Bill, and Carol asks him.*

2. Synthesize idiomatic English sentences that express the propositions associated with the logical forms below by the intensional interpretations that follow them.

a. Wci ∧ Scl
[S: λxy (x *is south of* y); W: λxy (x *is west of* y); c: *Crawfordsville*; i: *Indianapolis*; l: *Lafayette*]

b. Mab → Mba
[M: λxy (x *has met* y); a: *Ann*; b: *Bill*]

c. Iacb ∧ Iadb
[I: λxyz (x *introduced* y *to* z); a: *Alice*; b: *Boris*; c: *Clarice*; d: *Doris*]

d. Wab ∧ Kabab
[K: λxyzw (x *asked* y *to write* z *about* w); W: λxy (x *wrote to* y); a: *Alice*; b: *Boris*]

e. g = c → (f = s ∧ p = t)
[c: *the city*; f: *football*; g: *Green Bay*; p: *the Packers*; s: *the sport*; t: *the team*]

Homework assigned Wed 10/20 and due Fri 10/22

(i) Analyze: *If Al was the winner, he called Carol and told her about the prize*

(ii) Use derivations to show the following: A ∨ (B → C) ⇒ B → (A ∨ C)