Chapter 4.

Predicate Logic

Section 4.1. Introduction

In this chapter we shall expand treatment of the formal system of logic into the domain of ***first order predicate logic*** (often simply called "predicate logic" or "first order logic" or "quantifier logic"). We shall (1) discuss the notion of translation of English into first order predicate logic notation, (2) give an explicit formal system of first order logic, and (3) discuss some normal forms for formulas of first order logic. In Chapter 9 we will discuss what would be an appropriate semantics for first order logic akin to the truth table semantics we gave for propositional logic in the last chapter.

Section 4.2. Translation

In propositional logic, the smallest unit we considered was the atomic sentence. We investigated logical properties of sentences (and arguments) where these properties were dependent on the relationships that obtain between atomic sentences (such as the "if-then" relationship, the "and" relationship, etc.), but we did not consider any logical properties that a sentence (or argument) had in virtue of the relationship between subjects and predicates of atomic sentences. It is this topic that first order logic addresses. First order logic is built on top of propositional logic; all you have learned about propositional logic still holds here, but we will add some new things in order to accommodate the new information which we can represent because we have data about the relationship between subjects, objects and predicates. So: what does this new information amount to? Mostly it has to do with a new representation of atomic sentences. If you were to apply the translation method of the last chapter exactly as stated there, all you would have to do to obtain a predicate logic representation of the sentences would be to use a different scheme of abbreviation which breaks atomic sentences into their smaller components (subjects, predicates, etc.). Of course, this means that we need a somewhat different definition of a formula so that we can reflect this further breakdown. And when we come to the explicit formal system we shall need a few more rules of inference to tell us how to operate on these new atomic sentences. But all of the old rules of inference will remain valid.

It is traditional to break the topic of first order logic up into four segments: (a) propositional logic (which we discussed in the last chapter) (b) monadic (that is, 1-ary) predicate logic, (c) relations (that is, n-ary predicate logic), and (d), identity. We shall continue to use the notation we discussed in the last chapter, and start our discussion with (b). Once we have translations for monadic predicate logic under our belts, we shall move on to translation for the n-adic predicate logic and then discuss translations

for identity. In the next section we will give rules for operating in a formal system.

4.2.1. Monadic Predicate Logic

Consider a typical atomic sentence from the last chapter:

Kim danced

If we were going to break this atomic sentence up into parts, we would pick out *Kim*, the subject, and *danced*, the predicate. Intuitively the sentence asserts that the predicate is true of the subject. Similarly, if we had the atomic sentence

Jeff is a scholar

we would say the *Jeff* is the subject, and the sentence asserts that the predicate *is a scholar* is true of him, or that the characteristic or property of being a scholar can be truly asserted of Jeff. To fully represent this logical form (of these kinds of atomic sentences) we obviously need some kind of abbreviation for the subjects and predicates. We shall use lower case letters (usually with some association to the English name) to abbreviate the names. *Kim* would be represented as *k*, and *Jeff* as *j*. The predicates of the sentences, *danced* and *is a scholar*, will get represented by capital letters (again, normally with some association with the English); *danced* by *D*, and *is a scholar* by *S*. Finally, to represent the whole sentence, we wish to associate *k* with *D* and *j* with *S*. We do this simply by writing

D(k)

S(j)

(Later on, when we are more comfortable with the notation, we will often omit these parentheses when the context makes it clear how to restore them). It will be noted that the kind of predicates we are considering take one argument (the subject) to form an entire (atomic) sentence. This is why they are called "monadic", and it stands in contrast with such predicates (or relations) as *is taller than* which obviously require two arguments to form an atomic sentence. Quite often this information about how many arguments a predicate (or relation) takes to form an entire atomic sentence is given as a superscript. So we might have had D^1 stand for *dances*, S^1 stand for *is a scholar*, and maybe T^2 stand for *is taller than*. So the superscript says how many arguments are required for an atomic sentence. What, then, would one make of a "predicate" like A^0? Well - it requires zero names to form an atomic sentence. And what is that? That's just an atomic proposition, as we discussed in the last chapter. So you can see that propositional logic is just 0-ary predicate logic. The topic of this section is 1-ary (or monadic) predicate logic and the topic of the next section is n-ary predicate logic. Normally, when the context makes it clear, we will omit the use of superscripts.

In any case, one type of atomic sentence we can now represent more fully is the kind we have just discussed, where there is a name in the subject position and the rest of the sentence predicates some property of the object named by the subject term. But this is not the only kind of atomic sentence we want to represent in monadic predicate logic. Consider such sentences as

Every integer is a number

Some integers are primes

In such sentences we intuitively pick out a set (in these examples, the set indicated by 'integer') and assert that some predicate (*is a number* or *is a prime*) is true of every (or some) member of that set. To represent such sentences we obviously need some way of picking out sets, some way of talking about every or some members of the set, and a way of predicating a property of those members. The words *every* and *some* in these examples are *quantifiers*, and we have special symbols for them namely, $\forall$ and $\exists$ (respectively). These are called the *universal quantifier* and *existential quantifier*. To say that something is in the set of integers is to say that the predicate *is an integer* is true of it (so we can continue to use our predicates for the purposes of picking out sets). All that remains is to state a method for talking about members of the set when we don't have any specific ones in mind. For this purpose we shall employ *variables*. Stylistically we use x, y, z, w, etc (lower case letters near the end of the alphabet) for this and try to keep them separate from the lower case letters we use for names. So we say things like 'I(x)' to mean *x is an integer;* but since 'x' is a variable and does not name anything, the formula 'I(x)' is neither true nor false - it is instead an *open formula* (of one variable) which expresses an *open proposition* (of one variable). Similarly, if we let P stand for *is a prime*, then 'P(x)' is an open formula (of one variable) and '(I(x)∧P(x))' is also an open formula (of one variable). These will become sentences if the variable is replaced by a name. They will also become sentences if the variable is *bound* to a quantifier.

To bind a variable in a formula to a quantifier, one expands the quantifier into a *quantifier phrase* by adding a variable to the quantifier (the same one used in the open formula). For example, we might add 'x' to the quantifier '$\exists$', forming the quantifier phrase '$(\exists x)$' and then place it in front of the open formula '$(I(x)\wedge P(x))$' forming the "closed" formula

$$(\exists x)(I(x)\wedge P(x))$$

(It is the parentheses that tell what the *scope* of the quantifier phrase is - just as it is the 'BEGIN/END' structure of programs which tells what the scope of identifiers in a program is for a block structured, statically scoped language).) The formula above is different from

$$((\exists x)I(x)\wedge P(x))$$

In this last example, the scope of the '$(\exists x)$' extends only to the '$(I(x)$' and not the '$P(x)$'.

The quantifier phrase '$(\exists x)$' can be read "there exists an x" and the quantifier phrase '$(\forall x)$' can be read "for each x". Of course, the use of 'x' instead of, say, 'y' here is immaterial, because "there exists an x" and "there exists a y" say the same thing. The only reason to use one rather than another would be in sentences where we wish to distinguish two or more things. In these cases, the use of a different variable *allows* them to refer to different things (but does not require it). We shall shortly give examples of this. The quasi-English phrases "there exists an x" and "for each x" have many stylistic variants. The obvious ones are:

there exists an x: there is at least one x, something

for each x: for all x, for every x.

More stylistic variants become apparent after we give some other translation hints.

Given the apparatus so far developed we can translate a wide variety of atomic English sentences, besides the ones involving names and a predicate. There are simple ones involving only one predicate, such as those of I-IV in the table following. But we can also translate certain natural relationships between predicates, such as the

relationships indicated in V-VIII. (In each type we give a sampling of the stylistic variants of the quantifiers). In the translations we have omitted the superscripts and the parentheses around the variables, but this should raise no difficulties.

	Type of English sentence	*Translation*	*Examples*
I.	Everything is A Everything is an A	$(\forall x)Ax$	Everything is extended. Everything is a physical object
II.	Something is an A There is an A At least one A exists There are A's	$(\exists x)Ax$	Something is a buffalo. There is a positron At least one aardvark exists There are quasars
III.	Nothing is A A's don't exist	$\sim(\exists x)Ax$ or $(\forall x)\sim Ax$	Nothing is a chialligon Unicorns don't exist
IV.	Something is not A There is a non-A	$(\exists x)\sim Ax$	Something isn't dead There is a non-student
V.	Every A is a B Every A is B Every A B's Each A is a B All A's are B's A's are B's Any A is a B	$(\forall x)(Ax\longrightarrow Bx)$	Every dog is a mammal (A: x is a dog) Every computer is electronic Every professor talks Each senator is a citizen All women are people Students are smart Any table is a piece of furniture
VI.	No A is a B No A's are B's None of the A's are B's	$(\forall x)(Ax\longrightarrow\sim Bx)$ or $\sim(\exists x)(Ax\wedge Bx)$	No student is a professor No dogs are birds None of the senators is unemployed
VII.	Some A is a B At least one A is a B There exists an A which is a B There is an A B	$(\exists x)(Ax\wedge Bx)$	Some philosopher is a scientist At least one mathematician is a charlatan There is a computer which is cheap There is a tall man
VIII.	Some A is not a B Some A's are not B's Some A's are not B	$(\exists x)(Ax\wedge\sim Bx)$	Some professor is not a student Some scientists are not women Some philosophers are not serious

A number of observations should be made about these translations. The first has to do with the translation of the stylistic variants of "for each x" with a '$\longrightarrow$' relation holding between the subject (A) and predicate (B) while we translate the stylistic variants of "there is an x" with '$\wedge$' relating them. The easiest way to convince yourself that this is correct is to just try the alternatives with a few examples. For example, we say to translate *All boys are male* as $(\forall x)(Bx\longrightarrow Mx)$. Try reading these symbols back into "stilted English": you get *for each x, if x is a boy then x is a male*. Doesn't this sound as if it means the same as *All boys are male?* But if it were translated as $(\forall x)(Bx\wedge Mx)$, that is, by using an '$\wedge$' rather than an '$\longrightarrow$', then it would go back into "stilted English" as *for each x, x is both a boy and male.* But this just can't be right! This last would mean that *everything* (tables, computers, water, etc) is both a boy and male, whereas the original sentence only said that boys were male. The moral here is: when translating sentences using $(\forall x)$ -- that is, translating sentences using stylistic variants of "for each x" -- use the '$\longrightarrow$' to connect the subject of the sentence with the predicate of the sentence. We also translated *some dogs are* *terriers* as $(\exists x)(Dx \wedge Tx)$. Again, translate this back into "stilted English" and you get *there is an x which is a dog and a terrier* - which sounds right, doesn't it? But if we had translated it as $(\exists x)(Dx \longrightarrow Tx)$ we'd be in trouble. Remember that $(p \longrightarrow q)$ and $(\sim p \vee q)$ have the same truth table. This means that $(\exists x)(Dx \longrightarrow Tx)$ and $(\exists x)(\sim Dx \vee Tx)$ are the same. But the English of this last sentence is *something is either not a dog or else is a terrier*. That is made true much too easily - for example by a briefcase (which is not a dog, and so is something which either is not a dog or else is a terrier)! Rather, the original English sentence says that there is something which simultaneously is a dog and is a terrier.

For the Group VI sentences -- like *no whale is a fish* -- one can view it in two ways: either as saying, of all whales, that they are not fish; or as saying that it is false that some whale is a fish. The former way would yield $(\forall x)(Wx \longrightarrow \sim Fx)$. The latter way yields $\sim(\exists x)(Wx \wedge Fx)$. We shall later see that these are equivalent, and either can be used for translation. One is tempted to translate *everything is physical* as $(\forall x)(Tx \longrightarrow Px)$, where T:x is a thing. But this is not required, since the possible values of 'x' are all and only *things*; the antecedent "x is a thing" is already included in the $(\forall x)$ part, and so the sentence could just as well be translated as $(\forall x)Px$.

There are some other sentences whose translations depend on the specific predicates involved, and judgment must be used to determine which quantifier to use. Consider *primes are integers*. Such sentences should be translated with a universal quantifier: $(\forall x)(Px \longrightarrow Ix)$. But other, similar, sentences should be translated with an existential quantifier, such as *Dogs are barking:* $(\exists x)(Dx \wedge Bx)$. One needs to consult one's linguistic intuitions to discover whether the universal or the existential quantifier is appropriate.

Another fine point concerns the quantifier *any*. Often it is natural to translate it as a universal, as in our above list. But at other times, especially in the antecedent of a conditional, it is natural to translate it as an existential, as in *if any person can solve the equation, Terry can*, which would be translated as $((\exists x)(Px \wedge Sx) \longrightarrow S(t))$. It is worth noticing here in this example that we first recognized *if-then* as the main connective, yielding '$\longrightarrow$'. Then we translated *any person can solve the equation* as $(\exists x)(Px \wedge Sx)$ and the consequent as S(t).[2] So the hints given in the last chapter concerning how to translate truth-functionally complex sentences still hold -- the only difference now is our expanded definition of "atomic sentence".

You should keep the translations mentioned in Groups I-VIII of the previous table carefully in mind, as they will form the building blocks of most specific translations you will have to perform. For example, whenever you see a sentence like *All A's are B*, you will know that it is to be translated as $(\forall x)(Ax \longrightarrow Bx)$. The translation of the specific subject and of the specific predicate will go in place of the 'A' and 'B' in this translation scheme. Let us therefore turn to the question of how to translate various types of subjects and predicates. One important fact is that modification of a noun by an adjective or by a relative clause amounts to conjunction. For example, the subject *tall person* amounts to saying, of some *x*, that *x* is tall and *x* is a person. The subject *student who is taking computing* amounts to saying *x* is a student and *x* is taking computing. Similar remarks hold for adjectival modification and relative clause restriction within predicates. So to translate *Every tall student taking computing is an ambitious person who will go far:* we first recognize it as an *Every <subject> is a <predicate>* type, and hence start with $(\forall x)(\langle subject \rangle x \longrightarrow \langle predicate \rangle x)$. The

[2] However, if the sentence with *any* in its antecedent has a pronoun in its consequent which refers back to the *any*, then the hint given in this paragraph will yield incorrect results. For example, if the sentence is *if any person can solve the equation, then he will get an "A"* and we translate as suggested in the paragraph here, we get $((\exists x)(Px \wedge Sx) \longrightarrow Ax)$, which is incorrect because the *x* in the consequent is not bound to the quantifier in the antecedent. The correct translation for such sentences is $(\forall x)((Px \wedge Sx) \longrightarrow Ax)$.

<subject> has an adjective *(tall)* a noun *(student)* and a relative clause *(taking computing)*, and this is $(Tx \wedge Sx \wedge Cx)$. The predicate also has an adjective *(ambitious)*, a noun *(person)* and a relative clause *(who will go far)*, so it gets translated as $(Ax \wedge Px \wedge Gx)$. The whole sentence is therefore translated (again omitting internal parentheses)

$$(\forall x)((Tx \wedge Sx \wedge Cx) \longrightarrow (Ax \wedge Px \wedge Gx))$$

(Read this back into stilted English: for each *x*, if *x* is tall and a student and taking computing, then *x* is ambitious and a person and will go far. This should be close enough to the original to convince you that we have indeed translated it correctly).

As nice as it would be to have some sort of automatic translation procedure from English to first-order logic, none seems to be forthcoming. One must merely rely on one's linguistic intuitions - together with the kind of hints we have been giving. Two further hints we can pass on are these. Often, when English uses an explicit *and* in the subject, it should <u>NOT</u> be translated as an '$\wedge$'. You can tell this by doing the translation with '$\wedge$' and the re-translating it back into English and seeing whether it means the same as the original. Consider for example *Every man and woman can apply*. This appears to have the form $(\forall x)(\langle subject \rangle(x) \longrightarrow \langle predicate \rangle(x))$. And the <subject> appears to be *x* is a man and *x* is a woman. So you complete the translation, getting $(\forall x)((Mx \wedge Wx) \longrightarrow Cx)$. But this is wrong, as you discover in re-translating it to: for each *x*, if *x* is a man and *x* is a woman then *x* can apply. Note that, according to this last sentence, in order to be able to apply, *x* must be both a man and a woman! There are various solutions to this difficulty. Here are three (which of course are equivalent):

$$(\forall x)(M(x) \longrightarrow C(x)) \wedge (\forall x)(W(x) \longrightarrow C(x))$$

$$(\forall x)(\forall y)([M(x) \longrightarrow C(x)] \wedge [W(y) \longrightarrow C(y)])$$

$$(\forall x)([M(x) \vee W(x)] \longrightarrow C(x))$$

In the first, we have broken it into two sentences, one saying men can apply and the other that women can apply. In the second, we used two quantifiers, one for the men and one for the women, and we say of each variable that they can apply. In the third, we say of *x* that if it is a man *or* a woman then it can apply.

Another hint concerns *only*. In the last chapter we pointed out that when it is used in combination with *if*, it reverses the antecedent and the consequent of a conditional. Another use of *only* is as a universal quantifier (as in *Only integers are primes)*; but again it reverses the antecedent and consequent. This sentence should be translated as $(\forall x)(P(x) \longrightarrow I(x))$, and not the other way around (as it would be if the sentence were *All integers are primes).*

Finally, we should remark that it is important to distinguish truth-functionally complex sentences from others. For example, the sentence *Something is a square and something is a circle* is truth functionally complex and should be translated with '$\wedge$' as its main connective. So overall it is $((\exists x)S(x) \wedge (\exists x)C(x))$. But the sentence *Something is a square and a circle* is not truth functionally complex. Instead it gets translated as $(\exists x)(S(x) \wedge C(x))$, saying of one thing that it is both a square and a circle.

Let us look at the translation of some more interesting sentences. Here is a set of examples together with a few brief remarks on the process of translation, and a final translation. You should carefully go through these examples to make sure you could come up with the same answers.

(a) All cats will purr if their ears are rubbed.

(b) Only persons over 21 will be admitted.

(c) Any healthy baby is pleased if people sing to him or show him bright objects.

(d) No valuable diamonds are cracked or cloudy.

(e) If everything is mental, then nothing is physical unless something is both mental and physical.

(f) If a registered voter has not declared a party and only those who have declared a party can vote, then he or she cannot vote.

Sentence (a) is a type V statement with *cats* as subject and a complex if-then as predicate. So it should become

$$(a^*)\ (\forall x)(C(x) \longrightarrow (R(x) \longrightarrow P(x))$$

Sentence (b) uses *only* as a type of universal quantifier, which (you will remember) reverses the order of antecedent and consequent; so it translates as (P: x is a person, O: x is over 21)

$$(b^*)\ (\forall x)(A(x) \longrightarrow (P(x) \wedge O(x))$$

It is perhaps instructive to see why the various other translations one might think of are incorrect. The sentence $(\forall x)(P(x) \wedge O(x)) \longrightarrow A(x))$ is wrong because it says (wrongly) that *all* persons over 21 are admitted and does not say that no one else is admitted. The sentence $(\forall x)((P(x) \wedge A(x)) \longrightarrow O(x))$ says (correctly) that all people who are admitted are over 21, but does not also say (as the English does) that the only things that can gain admittance are people.

Sentence (c) should be recognized as using *any* as a universal quantifier. The subject is the conjunction of *healthy* and *baby*, and the predicate is an if-then statement with an "or" in the antecedent. The relevant predicates are: S: people sing to x, O: people show x bright objects,

$$(c^*)\ (\forall x)([H(x) \wedge B(x)] \longrightarrow [(S(x) \vee O(x)) \longrightarrow P(x)])$$

Sentence (d) is of our type VI. The subject is the conjunction of *valuable* and *diamonds*, while the predicate is the disjunction of *cracked* and *cloudy*. Since it is a type VI sentence, the translation hints give two ways to translate it

$$(d^*)\ (\forall x)([V(x) \wedge D(x)] \longrightarrow \sim[C(x) \vee L(x)])$$

$$(d^{**})\ \sim(\exists x)([V(x) \wedge D(x)] \wedge [C(x) \vee L(x)])$$

Sentence (e) should be recognized as having if-then as main connective, and having *unless* as the main connective of the consequent. The antecedent is our type I, and the consequent's parts can be recognized as type III and type VII respectively. So it would get translated as

$$(e^*)[(\forall x)M(x) \longrightarrow ((\forall x)\sim P(x) \vee (\exists x)(M(x) \wedge P(x)))]$$

Sentence (f) contains a bit of a trick. We can see the problem if we just try to translate it using the hints so far given. Using these hints, we would say that the main connective is if-then, and that the antecedent has a conjunction. The first conjunct looks like an existential statement: ***Some registered voter has not declared a party*** and the second conjunct is an "only" statement, becoming ***All who can vote have declared a party.*** The consequent uses the phrase ***he or she*** to refer to the previously-mentioned voter who has not declared a party. (One does not use a disjunction - the *he or she* just refers to the voter, x, regardless of sex). So we are ready to try to translate it, getting

$$\{((\exists x)[R(x) \wedge V(x) \wedge \sim D(x)] \wedge (\forall x)[C(x) \longrightarrow D(x)]) \longrightarrow \sim C(x)\}$$

But this translation is not quite right. In the consequent we have an "x" which is not attached to any quantifier phrase - the '$(\exists x)$' only extends to the first conjunct and the '$(\forall x)$' to the second conjunct of the antecedent. In technical jargon (which we shall carefully define later), this last occurrence of 'x' is ***free*** and ***not bound*** by any quantifier. (This is the same problem noted in the last footnote about ***any***). In any case, it does not necessarily refer to who we want it to, namely the registered voter who has not declared a party. A general trick which you will find useful is to recognize that when English uses a pronoun in a consequent to refer to some x introduced by a quantifier in the antecedent, what is *really* asserted by the sentence is that *every* thing that satisfies the antecedent also satisfies the consequent. We should therefore translate this sentence as

$$(f^*)\ (\forall x)\{([R(x) \wedge V(x) \wedge \sim D(x)] \wedge (\forall x)(C(x) \longrightarrow D(x)]) \longrightarrow \sim C(x)\}$$

You will now notice that the last 'x' is *in the scope of* the initial quantifier. But you might ask about the embedded '$(\forall x)$' quantifier in the antecedent - doesn't the fact that it uses the same variable as the main quantifier mean that there will be some confusion? The answer is no, and for more-or-less the same reasons that most programming languages do not get confused when you declare the same variable name inside the scope of another declaration -- it always uses the "nearest" declaration if it is still in its scope. Here, the x's in $[C(x) \longrightarrow D(x)]$ are bound by the closest '$(\forall x)$' and not the "outside" one. But it might be more clear to use a different variable, like this

$$(f^{**})\ (\forall x)\{([R(x) \wedge V(x) \wedge \sim D(x) \wedge (\forall y)[C(y) \longrightarrow D(y)]) \longrightarrow \sim C(x)\}$$

One final remark about translation has to do with the quantifiers and negation. If we say "Something is not F", that's the same as saying "not everything is F". That is, $(\exists x)\sim F(x)$ is the same as $\sim(\forall x)F(x)$. Similarly, "Nothing is F" is the same as "Everything is not F", so $\sim(\exists x)F(x)$ is the same as $(\forall x)\sim F(x)$. Generally, a negation can be "moved through" a quantifier phrase (in either direction) by changing the quantifier. These are called *the laws of quantifier negation.*

4.2.2. Translating Relations

The examples we have discussed thus far involve translating 0-ary predicates (i.e., propositions) and translating monadic predicates. In this section we discuss the full range of predicates, excepting the special case of identity (which we discuss in the next section).

Just as monadic translation builds on the propositional logic translation, so too the n-ary translations build on the monadic ones. What is new is the recognition of how to handle certain new kinds of English constructions.

Some predicates in English are obviously two-place, as for example "___ is larger than ___", "___flows into ___", "___is north of ___" and the like. These can be translated by such predicates as L^2, F^2, N^2; and to form a sentence, two names or variables are added; we can say $L^2(a,b)$ says "Alaska is larger than Belgium" and that $F^2(m,a)$ says "The MacKenzie River flows into the Arctic Ocean". Other predicates are three-place. *Line A extends from B to C* would be translated $E^3(a,b,c)$, and *Eight is a sum of five and three* could be translated $S^3(e,f,t)$. Still other ones are four-, five-, etc, place, and are translated with the appropriate superscript and the requisite number of names.

Adding quantifiers to this is easy. *Something is larger than Alberta* would be translated $(\exists x)T^2(x,a)$; *Nothing is north of the North Pole* would go as $\sim(\exists x)N^2(x,n)$. The issue is slightly more complicated when instead of having *something, nothing, everything* we have a quantified noun, such as *some man, every number, no physical object.* For these types of subjects one needs to recall the hints in the last subsection.

Some state is larger than Alberta

Every number is larger than zero

No person knows more than Ken

would be translated, respectively, as (we omit superscripts when no ambiguity results)

$$(\exists x)(S(x)\wedge L(x,a))$$

$$(\forall x)(N(x)\longrightarrow L(x,z))$$

$$\sim(\exists x)(P(x)\wedge K(x,k))$$

which is exactly what you would expect from the previous subsection.

Reflexive pronouns, like *itself, himself, themselves* are translated as follows: if the grammatical antecedent is a name, just re-use the name. *Sammy hates himself* gets translated as $H^2(s,s)$. If the grammatical antecedent is a quantified term, re-use the

variable. *No person hates himself* becomes $\sim(\exists x)(P(x)\wedge H(x,x))$ or $(\forall x)(P(x)\longrightarrow\sim H(x,x))$.

If a formula has a sequence of quantifiers all of the same type (all universals or all existentials), then the order of the quantifiers is immaterial. $(\forall x)(\forall y)(\forall z)F^3(x,y,z)$ is the same as $(\forall y)(\forall z)(\forall x)F^3(x,y,z)$ - and the same as any other order of quantifiers. But when the quantifiers are mixed, the story is different. Consider

Everything is caused by something

There is something which everything is caused by

These are obviously quite different in meaning, and translating them we see that the difference shows up in the order of quantifiers. They are, respectively

$$(\forall x)(\exists y)C^2(x,y)$$

$$(\exists y)(\forall x)C^2(x,y)$$

where C^2:x is caused by y. Sometimes it is difficult to see exactly what the difference is, but let's consider G^3 x gave y to z.

$$(\forall x)(\forall z)(\exists y)G(x,y,z)$$

says that everyone gave everyone something (or other). As remarked before, this is equivalent to

$$(\forall z)(\forall x)(\exists y)G(x,y,z)$$

On the other hand

$$(\forall x)(\exists y)(\forall z)G(x,y,z)$$

says that for each person there is a specific gift he gave to everyone. That is, for each person you can find some particular gift which he gave to everyone. On the third hand

$$(\exists y)(\forall x)(\forall z)G(x,y,z)$$

[which is equivalent to $(\exists y)(\forall z)(\forall x)G(x,y,z)$] says that there is some one thing and everyone gave it to everyone.

Genitives and possessives indicate ownership or possession. In English this might be explicitly indicated with *owns* or *has*, but more commonly by the possessive case. *Jeff's computer is slow* should be paraphrased as *there is a computer which Jeff owns* *that is slow* and therefore translated $(\exists x)(C(x) \wedge O(j,x) \wedge S(x))$. Sometimes the possessive case indicates a different relationship. *Jeff's brother* probably ought to be translated by B^2:x is a brother of y. So, *Jeff's brother's computer is slow* should be translated as $(\exists x)(C(x) \wedge (\exists y)[B(y,j) \wedge O(y,x)] \wedge S(x))$ - which says that there is a computer and a brother of Jeff who owns it, and it is slow.

English uses prepositional phrases to express relationships, and often one wishes to use existential quantifiers to capture the meaning.

Jeff bought a computer from a woman with a dog

would be translated, in stages

$$(\exists x)(x \textit{ is a computer} \wedge j \textit{ bought } x \textit{ from a woman with a dog})$$

$$(\exists x)(C(x) \wedge (\exists y)[y \textit{ is a woman with a dog} \wedge j \textit{ bought } x \textit{ from } y])$$

$$(\exists x)(C(x) \wedge (\exists y)[(\exists z(y \textit{ is a woman} \wedge z \textit{ is a dog} \wedge y \textit{ owns } z) \wedge B(j,x,y)])$$

$$(\exists x)(C(x) \wedge (\exists y)[(\exists z)(W(y) \wedge D(z) \wedge O(y,z)) \wedge B(j,x,y)])$$

Here are some examples translated for you. You should try to translate them yourself and see if you come up with equivalent tránslations.

(a) Everyone who used Jeff's computer is sought by Sally.

(b) Everything which will convince somebody of sound judgement will convince everyone.

(c) Any number such that all numbers less than it are interesting is interesting.

(d) No number is prime unless it is divisible only by 1 and itself

(e) A teacher has no scruples if he assigns a problem that has no solution.

We here provide a translation of these sentences (we translate *everyone* in (a) as *every person*). The scheme of abbreviation for these is

P: x is a person
C^1: x is a computer
O: x owns y
U: x used y
S: x is sought by y
J: x has sound judgment
C^2: x convinces y
N: x is a number
L: x is less than y
I: x is interesting
R: x is prime
D: x is divisible by y
E: x is equal to y
T: x is a teacher
B: x is a problem

A: x assigns y
L: x is a solution to y
H: x has scruples
j: Jeff
s: Sally
o: 1

$$(a^*)\ (\forall x)(P(x) \wedge (\exists y)(C^1(y) \wedge O(j,y) \wedge U(x,y)) \longrightarrow S(x,s))$$

$$(b^*)\ (\forall x)((\exists y)[P(y) \wedge J(y) \wedge C^2(x,y)] \longrightarrow (\forall z)(P(z) \longrightarrow C^2(x,z)))$$

$$(c^*)\ (\forall x)(N(x) \wedge (\forall y)(N(y) \wedge L(y,x) \longrightarrow I(y)) \longrightarrow I(x))$$

$$(d^*)\ (\forall x)(N(x) \longrightarrow (\sim R(x) \vee (\forall y)(N(y) \wedge D(x,y) \longrightarrow (E(y,o) \vee E(y,x))))$$

The last clause of (d*) says "Every number that divides x is either equal to 1 or to x". When we encounter identity in the next section we shall use that instead.

$$(e^*)\ (\forall x)(T(x) \wedge (\exists y)(B(y) \wedge A(x,y) \wedge \sim(\exists z)L(z,y)) \longrightarrow \sim H(x))$$

4.2.3. Identity

The final area of first-order logic that we are interested in is that of *identity*. This is that relation which holds between anything and itself, and not to another thing (no matter how similar). Being a relation, it *could* be translated by some two-place predicate as we did with an example in the last section. But it also has some special properties which set it apart from other relations, and for that reason it is usually given a special symbol, '='.

Sentences which can be translated using identity are:

1. Simple identities such as *Mark twain is Samual Clemens* and the stylistic variants using *is the same as, is none other than, is identical to* and the like, would be translated as 'm=s'

2. Negations of simple identities, such as *Mark Twain is not Walter Scott* and its stylistic variants using *is not the same as, is different from, is other than,* etc, would be translated as '$\sim m = w$', or, commonly as '$m \neq w$'.

3. *Exceptives.* These are sentences where someone is explicitly excluded from consideration. A sentence like *John is taller than everyone else* tells us to gather together all people except for John, and that a comparison of John with this group will show that John is taller than each of them. So, using P: x is a person, T: x is taller than y, we would translate it as

   $$(\forall x)(P(x) \wedge x \neq j \longrightarrow T(x,j))$$

   A sentence like *John is taller than everyone except Len* explicitly excludes Len from the group of people that John is taller than. So, in addition to excluding John from the comparison group, as above, we also wish to exclude Len by adding another clause to the antecedent, giving

$$(\forall x)(P(x)\wedge x\neq j\wedge x\neq l\longrightarrow T(j,x))$$

Stylistic variants of *except* are: *else but, but, other than*. Another type of exceptive involves "only". Consider

Len is the only person smarter than John

Only Len is a smarter person than John

(stylistic variants of each other). Such sentences say three things: that Len is a person, that Len *is* smarter than John, and that no other person is.

$$P(l)\wedge S(l,j)\wedge(\forall x)(P(x)\wedge x\neq l\longrightarrow S(j,x))$$

To negate this sentence, one should be careful. The "literal negation" is

$$\sim[P(l)\wedge S(l,j)\wedge(\forall x)(P(x)\wedge x\neq l\longrightarrow S(j,x))]$$

which by DeMorgan's laws is

$$\sim P(l)\vee\sim S(l,j)\vee\sim(\forall x)(P(x)\wedge x\neq l\longrightarrow S(j,x))$$

You should ask yourself whether you think that *Len is not the only person smarter than Len* should be translated this way - i.e., whether it should be translated as saying that either Len isn't a person or he isn't smarter than John or someone else is. If not, you have to be careful where you place negations in the translation. Most likely you would want the negation to have "narrow scope", yielding

$$P(l)\wedge S(l,j)\wedge\sim(\forall x)(P(x)\wedge x\neq l\longrightarrow S(j,x))$$

wherein Len is a person, is smarter than John, but is not the only such person.

4. *Superlatives:* A sentence like *Len is the tallest person in town* is a superlative. A superlative sentence says that some object has a property to the highest degree. Superlatives are to be translated using the comparative, like *taller than*, and identity. So generally speaking, we never have a predicate which means "is the F-est", but rather only relations like "is F-er than". For the present example let's use I^2: x is in y, P: x is a person, T^2: x is taller than y, l: Len, and t: the town. The translation is

$$(\forall x)[P(x)\wedge I(x,t)\wedge x\neq l\longrightarrow T(l,x)]\ \wedge P(l)\wedge I(l,t)$$

Intuitively we say, "for any person in the town other than Len, Len is taller than him; and furthermore Len is a person in the town." It is important to put the $x\neq l$ clause in, otherwise you will find Len amongst the x's, and he

will turn out to be taller than himself! Note that the $P(l)$ and $I(l,t)$ are outside the scope of the quantifier.

5. *Numerical statements.* For any finite natural number, careful use of identity gives a way of saying that there are that many things of a certain type. For example, let P: x is a professor. We can say that there are *at least n* professors in the following way.

at least	formula
1	$(\exists x)Px$
2	$(\exists x)(\exists y)(P(x)\wedge P(y)\wedge x\neq y)$
3	$(\exists x)(\exists y)(\exists z)(P(x)\wedge P(y)\wedge P(z)\wedge x\neq y\wedge x\neq z\wedge y\neq z)$
.	.
.	.
.	.

You need to add the non-identities because merely using different variables does not ensure distinctness of objects - they might be the same. You can also say *at most n* professors

at most	formula
1	$(\forall x)(\forall y)(P(x)\wedge P(y)\longrightarrow x=y)$
2	$(\forall x)(\forall y)(\forall z)(P(x)\wedge P(y)\wedge P(z)\longrightarrow x=y\vee x=z\vee y=z)$
3	$(\forall x)(\forall y)(\forall z)(\forall w)(P(x)\wedge P(y)\wedge P(z)\wedge P(w)\longrightarrow$ $x=y\vee x=z\vee x=w\vee y=z\vee y=w\vee z=w)$
.	.
.	.
.	.

(The idea here is: try to pick out more than *n* professors and you will discover that there is some identity happening.) To say there is *exactly n* professors you could conjoin the formula for at least *n* professors with the one for 'at most *n* professors. For example, to say there are exactly two professors

$$(\exists x)(\exists y)(P(x)\wedge P(y)\wedge x\neq y)\wedge(\forall x)(\forall y)(\forall z)(P(x)\wedge P(y)\wedge P(z)\longrightarrow x=y\vee x=z\vee y=z)$$

There are certain shorter formulas which are equivalent to this. One strategy is this: start out by saying that there are at least *n*, but before closing these parentheses, add that any professor you choose is identical to one of the first *n*. So the exactly two professors case becomes

$$(\exists x)(\exists y)(P(x)\wedge P(y)\wedge x\neq y\wedge(\forall z)(P(z)\longrightarrow x=z\vee y=z))$$

As a special case, to say exactly one professor exists, there is another commonly-used formula

$$(\exists x)(\forall y)(P(y) \leftrightarrow x = y)$$

which is equivalent to the other ways of saying that there's exactly one professor. (And of course to say that there are no professors we can use $\neg(\exists x)P(x)$.)

Section 4.3. Definition of Formula and Related Items

We have so far discussed translation into a formal system on an intuitive basis without giving an explicit definition of what strings of symbols count as well-formed formulas. Here is a definition of our system.

I. A constant or a variable is a *term*

II. $\forall$ and $\exists$ are *quantifiers*

III. If P^n is an n-place predicate and $x_1,...x_n$ are n terms, then

$$P^n(x_1, \ldots, x_n)$$

is a *formula* (an atomic formula)

IV. If x_1 and x_2 are terms, then

$$x_1 = x_2$$

is a *formula* (an atomic formula)

V. If ϕ is a formula, then

$$\neg\phi$$

is a *formula*

VI. If ϕ and ψ are formulas, then

$$(\phi \wedge \psi)$$

$$(\phi \vee \psi)$$

$$(\phi \rightarrow \psi)$$

$$(\phi \leftrightarrow \psi)$$

are *formulas*

VII. If Π is a quantifier and α is a variable and ϕ is a formula, then

$$(\Pi\alpha)\phi$$

is a *formula*

We retain our conventions about dropping parentheses and altering their style, and we introduce the conventions that $\neg x_1 = x_2$ can be written as $x_1 \neq x_2$ and that we can drop the superscript on a predicate.

The definition admits certain formulas which have no intuitive English counterpart, such as $(\forall x)P^o$, $(\forall x)(\forall x)F^1(x)$ and the like. These are harmless, and to eliminate them would mean needless complication in the definition of formula.

Intuitively speaking, an occurrence of a variable in a formula ϕ is a *free occurrence in* ϕ if that variable is in ϕ but is not in the scope (or domain) of any quantifier phrase using that variable. It is a *bound occurrence in* ϕ if it is in ϕ but not a free occurrence in ϕ. The problem with this as a definition is that it uses "scope" to define bound/free; but the definition of "scope" has not been given. So, although the intuitive meaning of these terms is clear, there is need of a precise definition. Here is one for *free occurrence in* ϕ

1. Any occurrence of a variable in an atomic formula is a free occurrence in that formula

2. If an occurrence of a variable is free in formula ϕ, then it is a free occurrence in $\neg\phi$

3. If an occurrence of a variable is free in formula ϕ or in formula ψ, then it is a free occurrence in

$$(\phi \wedge \psi)$$

$$(\phi \vee \psi)$$

$$(\phi \rightarrow \psi)$$

$$(\phi \leftrightarrow \psi)$$

4. If an occurrence of a variable is free in formula ϕ, and it is not the variable α, then it is a free occurrence in

$$(\forall \alpha)\phi$$

$$(\exists \alpha)\phi$$

5. No other occurrences of variables in a formula are free in that formula.

An occurrence of a variable in a formula is bound if and only if it is not free. To discover what quantifier phrase an occurrence of a variable is bound by, one needs to investigate how the formula was "put together". Intuitively, whenever one uses rule VII to construct a sub-formula of a given formula, one binds every occurrence of the variable which occurs in the quantifier phrase. They are *bound by* (and in the scope of) that quantifier phrase. Further detail could be given here, but we trust that the concept is clear enough.

A formula with a free occurrence of a variable intuitively has no fixed meaning. It is akin to

$x + 3 = 7$

He is tall

Where 'x' and 'he' function as free occurrences of variables. You cannot tell whether these formulas are true or false until you know what 'x' and 'he' stand for. Note that this is not the case when we quantify these variables:

$(\exists x)(x + 3 = 7)$

Everything is tall

Here we *do* have a formula which is either true or false. Such formulas -- ones that have no free occurrences of variables -- are called *sentences*, and are the kinds of formulas we are interested in describing truth and falsity for. (As we will in Chapter 9).

Section 4.4. A Formal System for First-Order Logic with Identity

We shall continue our practice of giving introduction and elimination rules for each connective; however, for ease of constructing proofs we shall also give the Quantifier Negation and Bound Variable Substitution rules as if they were primitive. (Actually they can be derived from the others). Most of these rules require the notation of *proper free variable substitution*. The idea of this is clear, although the formal statement is a bit complicated. Suppose we had the formula

$$(\forall x)(F(x,y)\rightarrow[(\exists y)G(x,y,z)\wedge H(y)])$$

and you wished to convert this to a formula which "said the same thing" but used different variables. Obviously, if w is a variable, then

$$(\forall x)(F(x,w)\rightarrow[(\exists y)G(x,y,z)\wedge H(w)])$$

will do the job by replacing the first and last occurrences of y by w. We have altered none of the formula's structure - every variable which was bound by a quantifier is still bound by that same quantifier, no new variables become bound by a quantifier, and all the free occurrences of y were uniformly replaced. But if we were to also alter the y in the middle to w, yielding

$$(\forall x)(F(x,w)\rightarrow[(\exists y)G(x,w,z)\wedge H(w)])$$

we no longer have a formula which "says the same thing" - the old position used to be bound by $(\exists y)$ but no longer is. Similarly, we could not change the y's to x, since then the occurrences would be bound by the $(\forall x)$. One normally says that these restrictions avoid "collision and confusion" of variables in substitution. One hitch: suppose we were to change the z in our first formula to y. Then the result no longer "says the same", and we can tell this by noting that we cannot perform a proper substitution on this result to get back to our original formula. Even though the resulting formula does not "say the same thing" as the original, we wish to allow this to be a proper free variable substitution. Basically, a proper free variable substitution comes about when, in a formula ϕ, you take all free occurrences of some variable α and replace them by the variable β in such a way that none of the β's become bound. It should be emphasized that free variable substitution is *not* a rule of our system. Rather it is a concept which is used in stating the things that *are* rules of inference. In fact, the unrestricted use of this as if it were a rule of inference would lead from truths to falsehoods. So it is used in the very specific circumstances indicated in certain rules.

The notion of *bound variable substitution* is similar, except that we wish to change a quantifier phrase and all the variables it binds. So we wish to go from

$$(\forall x)(F(x,y)\rightarrow(\exists y)G(x,y,z))$$

to

$$(\forall w)(F(w,y)\rightarrow(\exists y)G(w,y,z))$$

(by changing the bound x's to w) and also to

$$(\forall x)(F(x,y)\rightarrow(\exists w)G(x,w,z))$$

(by changing the bound y's to w). We cannot change x's to y for two reasons: the original occurrence of y would become bound, and the original last occurrence of z would become bound by a different quantifier. As we shall see shortly, unlike the free variable substitution, bound variable substitution *is* a rule of inference in our system.

We are now in a position to give our rules. With the definitions of variable substitution before us, the rules of Quantifier Negation, Bound Variable Substitution, ∀-elimination, and ∃-introduction are quite straightforward and intuitively very plausible. The other two rules, ∀-introduction and ∃-elimination, require some discussion.

QUANTIFIER NEGATION (QN)

.	.	.	.
$\sim(\forall x)\phi$	$\sim(\exists x)\phi$	$(\forall x)\sim\phi$	$(\exists x)\sim\phi$
.	.	.	.
$(\exists x)\sim\phi$	$(\forall x)\sim\phi$	$\sim(\exists x)\phi$	$\sim(\forall x)\phi$

BOUNDED VARIABLE SUBSTITUTION (BVS)

| .
.
.
ϕ
.
.
.
ϕ'

where ϕ' comes from ϕ by proper bound variable substitution on some subformula of ϕ

A universally quantified formula - that is, one where the universal quantifier is the main connective - says that the formula inside the quantifier is true for any value substituted for the variable which has been quantified over. So we ought to be able to replace these occurrences by any term we wish - so long as we avoid "confusion and collision" of variables. We shall use $\phi\frac{t}{x}$ to mean "the result of proper free substitution of t for x in the formula ϕ".

UNIVERSAL QUANTIFIER ELIMINATION ($\forall - E$)

| .
.
.
$(\forall x)\phi$
.
.
.
$\phi\frac{t}{x}$

for any t

A formula with a name (or variable) in it intuitively says that the object designated by the name has a certain (possibly complex) property true of it. So if you knew this you would be justified in asserting that *something* has that property. And this is precisely what existential quantifier introduction says.

EXISTENTIAL QUANTIFIER INTRODUCTION ($\exists - I$)

| .
.
.
$\phi\frac{t}{x}$
.
.
.
$(\exists x)\phi$

It is important to note what ($\exists - I$) says: it is a legitimate operation if you could have done a proper free variable substitution from the formula you end up with to get back to the original. This has the consequence of not requiring that the addition of the ($\exists x$) quantifier phrase "capture" all of the occurrences of t in the original formula. For example we can pass from $F(t,y,t)$ to $(\exists x)F(x,y,t)$ or to $(\exists x)F(t,y,x)$ as well as to $(\exists x)F(x,y,x)$ by the rule of ($\exists - I$).

Here are some examples using the rules we have so far discussed.

We derive $(\exists y)P(y)$ from $(\forall x){\sim}{\sim}P(x)$

1.	$(\forall x){\sim}{\sim}P(x)$	
2.	${\sim}{\sim}P(x)$	1, $\forall - E$
3.	$P(x)$	2, $\sim E$
4.	$(\exists y)P(y)$	3, $\exists - I$

We derive $(\exists w)(\exists y)F(w,y)$ from $(\forall x)(\forall y)(G(x,y) \longrightarrow F(x,y))$ and G(a,b)

1.	$(\forall x)(\forall y)(G(x,y) \longrightarrow F(x,y))$	
2.	$G(a,b)$	
3.	$(\forall y)(G(a,y) \longrightarrow F(a,y)$	1, $\forall - E$
4.	$G(a,b) \longrightarrow F(a,b)$	3, $\forall - E$
5.	$F(a,b)$	2,4 $\longrightarrow E$
6.	$(\exists y)F(a,y)$	5, $\exists - I$
7.	$(\exists w)(\exists y)F(w,y)$	6, $\exists - I$

We derive $(\forall z){\sim}P(z)$ from $(\forall y){\sim}F(y,a)$ and $({\sim}(\exists x)P(x) \vee (\exists w)F(w,a)$

1.	$(\forall y){\sim}F(y,a)$	
2.	${\sim}(\exists x)P(x) \vee (\exists w)F(w,a)$	
3.	${\sim}(\exists x)P(x) \vee (\exists y)F(y,a)$	2, BVS
4.	${\sim}(\exists y)F(y,a)$	1, QN
5.	${\sim}(\exists x)P(x)$	3,4 $\vee - E$
6.	$(\forall x){\sim}P(x)$	5, QN
7.	$(\forall z){\sim}P(z)$	6, BVS

The remaining two rules, Existential Quantifier Elimination ($\exists - E$) and Universal Quantifier Introduction ($\forall - I$), require the notion of an ***arbitrary variable***. Let us start with ($\exists - E$). Intuitively speaking, when you are given an existentially quantified formula, what you have been told is that *something* satisfies the formula following the quantifier phrase. But you are *not* told *what* that something is. Given then that we don't know what the object in question is, how can we tell what further formulas can be derived from it? The answer is: the formulas that follow from an existentially quantified formula are exactly those formulas which would follow no matter what name you were to use in place of the existentially quantified variable. Our method of finding this out involves the notion of an arbitrary variable. An arbitrary variable is first a variable, but importantly is a method of preventing us from illicitly using any information we might have about that variable. What we do is this: we have an existentially quantified formula in our derivation; we pick an "arbitrary instance" of it by using some variable and introducing this instance as a subsidiary assumption; we mark the scope line of this subsidiary assumption with the variable we have chosen, and we do not allow reiteration to be performed from outside this scope line to within it if the formula being reiterated has a free occurrence of the "arbitrary variable" we chose. Finally, anything which can be derived from this arbitrary instance can be derived from the existentially quantified formula outside it ***as long as the formula thus derived has no free occurrences of the "arbitrary variable"***. So here is the pattern

EXISTENTIAL QUANTIFIER ELIMINATION ($\exists - E$)

$\mid (\exists x)\phi$		
$\mid$.		
$\mid$.		
$\mid$.		
$\mid \quad \mid \phi\frac{u}{x}$		
$\mid \quad u\mid$ ———	where:(a) $\phi\frac{u}{x}$ comes from ϕ by proper free variable substitution of u for x	
$\mid \quad \mid$.		
$\mid \quad \mid$.		
$\mid \quad \mid$.		
$\mid \quad \mid \psi$		
$\mid \quad \mid$.		
$\mid \quad \mid$.	(b) no formula with a free occurrence of u was reiterated to within the u-scope line	
$\mid \quad \mid$.		
$\mid$.		
$\mid$.	(c) the formula ψ (which has been "popped outside") has no free occurrence of u	
$\mid$.		
$\mid \psi$		

The annotation for the line ψ is: the number of the line $(\exists x)\phi$, and the line numbers of the subproof, plus '$\exists - E$'. It is the (b) restriction here which was our original motivation for requiring all formulas to which a rule of inference is to be applied to be in the scope level where the result of the rule was to appear, and to require that the rule of Reiteration be applied to ensure this. Our way of preventing certain illicit inferences in the predicate logic is to require the Reiteration and to make Reiteration impossible when a formula with a free variable must go through a scope line which mentions that variable. This same restriction will come into play in our rule $\forall - I$ also (see below).

Some examples:

From $(\forall x)(P(x)\longrightarrow Q(a))$ we prove $((\exists y)P(y)\longrightarrow Q(a))$

1.	$\mid (\forall x)(P(x)\longrightarrow Q(a))$	
2.	$\mid \quad \mid (\exists y)P(y)$	
3.	$\mid \quad \mid \quad \mid P(u)$ (u)	
4.	$\mid \quad \mid \quad \mid (\forall x)(P(x)\longrightarrow Q(a))$	1,R
5.	$\mid \quad \mid \quad \mid P(u)\longrightarrow Q(a)$	4, $\forall - E$
6.	$\mid \quad \mid \quad \mid Q(a)$	3,5 $\longrightarrow E$
7.	$\mid \quad \mid Q(a)$	2,3-6 $\exists - E$
8.	$\mid [(\exists y)P(y)\longrightarrow Q(a)]$	2-7 $\longrightarrow I$

From $(\exists x)(P(x)\longrightarrow(\forall x)Qx)$ we prove $((\forall x)P(x)\longrightarrow(\forall x)Qx)$

1.	$\mid (\exists x)(P(x)\longrightarrow(\forall x)Q(x))$	
2.	$\mid \quad \mid (\forall x)P(x)$	
3.	$\mid \quad \mid (\exists x)(P(x)\longrightarrow(\forall x)Q(x))$	1,R
4.	$\mid \quad \mid \quad u\mid P(u)\longrightarrow(\forall x)Q(x)$	
5.	$\mid \quad \mid \quad \mid (\forall x)P(x)$	2,R
6.	$\mid \quad \mid \quad \mid P(u)$	5, $\forall - E$
7.	$\mid \quad \mid \quad \mid (\forall x)Q(x)$	4,6 $\longrightarrow E$
8.	$\mid \quad \mid (\forall x)Q(x)$	3,4-7 $\exists - E$
9.	$\mid ((\forall x)P(x)\longrightarrow(\forall x)Q(x))$	2-8 $\longrightarrow I$

From $(\exists x)(\forall y)(\forall z)F(x,y,z)$ we prove $(\exists x)F(x,x,x)$

1.	$\mid (\exists x)(\forall y)(\forall z)F(x,y,z)$	
2.	$\mid \quad u\mid (\forall y)(\forall z)F(u,y,z)$	
3.	$\mid \quad \mid (\forall z)F(u,u,z)$	2, $\forall - E$
4.	$\mid \quad \mid F(u,u,u)$	3, $\forall - E$
5.	$\mid \quad \mid (\exists x)F(x,x,x)$	4, $\exists - I$
6.	$\mid (\exists x)F(x,x,x)$	1,2-5 $\exists - E$

The last of our quantifier rules (there are still two identity rules) is Universal Quantifier Introduction ($\forall - I$). This rule follows the principle that if you can show that a formula is true of a "completely arbitrary" object, then you are justified in asserting that the formula holds of everything. Again we shall use our scope lines with variables to ensure that we've really proven something of an arbitrary object. Again, we cannot reiterate any formula with a free variable through any scope line which is flagged with that variable. The general form of this rule of inference is

UNIVERSAL QUANTIFIER INTRODUCTION ($\forall - I$)

.	
.	
.	
u | .	where: (a) $\phi\frac{u}{x}$ could come from ϕ by proper free variable substitution of u for x
| .	
| .	
| .	
| $\phi\frac{u}{x}$	
| .	(b) no formula with a free occurrence of u is reiterated to within the u-scope line
| .	
| .	
$(\forall x)\phi(x)$	

(Note that, unlike $\exists - E$, we have made no assumption here. Rather, we simply start a scope line which is labelled with u.)
Here are some examples

We prove $((\forall x)P(x) \longrightarrow (\forall x)Q(x))$ from $(\forall x)(P(x) \longrightarrow Q(x))$

1.	$(\forall x)(P(x) \longrightarrow Q(x))$	
2.	| $(\forall x)P(x)$	
3.	| u | $(\forall x)(P(x) \longrightarrow Q(x))$	1,R
4.	| | $P(u) \longrightarrow Q(u)$	3, $\forall - E$
5.	| | $(\forall x)P(x)$	2, R
6.	| | $P(u)$	5, $\forall - E$
7.	| | $Q(u)$	4,6 $\longrightarrow E$
8.	| $(\forall x)Q(x)$	3-7 $\forall - I$
9.	$(\forall x)P(x) \longrightarrow (\forall x)Q(x))$	2-8 $\longrightarrow I$

We prove $(\exists x)P(x) \longrightarrow (\forall x)Q(x)$ from $(\forall x)((\exists y)P(y) \longrightarrow Q(x))$

1.	$(\forall x)((\exists y)P(y) \longrightarrow Q(x))$	
2.	| $(\exists x)P(x)$	
3.	| u | $(\forall x)((\exists y)P(y) \longrightarrow Q(x))$	1,R
4.	| | $(\exists y)P(y) \longrightarrow Q(u)$	3, $\forall - E$
5.	| | $(\exists x)P(x)$	2,R
6.	| | $(\exists y)P(y)$	5, BVS
7.	| | $Q(u)$	4,6 $\longrightarrow E$
8.	| $(\forall x)Q(x)$	3-7 $\forall - I$
9.	$(\exists x)P(x) \longrightarrow (\forall x)Q(x)$	2,8 $\longrightarrow I$

We are now in a position to give the rules of identity. Unlike our other rules, Identity Introduction (=I) requires no premises. What it says is that everything is self-identical; that we are allowed to write $t = t$ (for any constant or variable t) anywhere we wish in a proof.

IDENTITY INTRODUCTION (=I)

.
.
.
$t = t$

Here are a few examples using =I

From $(b = b \longrightarrow (\forall x)G(x))$ and $(G(a) \longrightarrow H(a))$ we prove $(\exists x)H(x)$

1.	$b = b \longrightarrow (\forall x)G(x)$	
2.	$G(a) \longrightarrow H(a)$	
3.	$b = b$	=I
4.	$(\forall x)G(x)$	1,3 $\longrightarrow E$
5.	$G(a)$	4, $\forall - E$
6.	$H(a)$	2,5 $\longrightarrow E$
7.	$(\exists x)H(x)$	6, $\exists - I$

We prove $(\forall x)(\exists y)x = y$ from no premises

1.	u | $u = u$	=I
2.	| $(\exists y)u = y$	1, $\exists - I$
3.	$(\forall x)(\exists y)x = y$	1-2, $\forall - I$

The last rule is Identity Elimination (=E), which is often called Leibniz's Law. Intuitively stated, (=E) says that given a sentence in which a name occurs, you can replace that name by a different name of the same object without changing the truth or falsity of the sentence. Thus, if you know that 2x is even, and that 2x = y, then you know that y is even. Here's a formal statement of the rule. Note that the relative position of the identity and the formula being substituted into is irrelevant. So there are really four versions of this rule. Also, one needn't substitute for every occurrence of the constant or variable.

IDENTITY ELIMINATION (=E)

```
| .              | .
| .              | .
| .              | .
| a=b            | b=a
| .              | .
| .              | .
| .              | .
| φ a/b          | φ a/b
| .              | .
| .              | .
| φ              | φ
```

where: $\phi\frac{a}{b}$ could come from ϕ by proper free variable substitution of a for b

Here are some derivations using all our rules

From P(a) we derive $(\exists x)(x=a \wedge P(x))$

```
1. | P(a)
   |------
2. | a=a                     =I
3. | a=a ∧ P(a)              1,2 ∧I
4. | (∃x)(x=a ∧ P(x))        3, ∃-I
```

From $P(a)$ and $\sim P(b)$ we prove $a \neq b$

```
1. | P(a)
2. | ~P(b)
   |------
3. |   | a=b
   |   |-----
4. |   | P(a)                1,R
5. |   | P(b)                3,4 =E
6. |   | ~P(b)               2 R
7. | a≠b                     3-6 ~I
```

We prove $(\forall x)(\forall y)(x=y \longrightarrow y=x)$ [Symmetry of '='] from no premises.

```
1. | u|  v|  | u=v
   |  |   |  |-----
2. |  |   |  | v=v                      =I
3. |  |   |  | v=u                      1,2 =E
4. |  |   | u=v ⟶ v=u                   1-3 ⟶I
5. |  | (∀y)(u=y ⟶ y=u)                 1-4 ∀-I
6. | (∀x)(∀y)(x=y ⟶ y=x)               1-5 ∀-I
```


We prove $(\forall x)(\forall y)(\forall z)(x=y \wedge y=z \longrightarrow x=z)$ [Transitivity of '='] from no premises.

```
1. | u|  v|  w|  | u=v ∧ v=w
   |  |   |   |  |----------
2. |  |   |   |  | u=v                          1, ∧E
3. |  |   |   |  | v=w                          1, ∧E
4. |  |   |   |  | u=w                          2,3 =E
5. |  |   |   | u=v ∧ v=w ⟶ u=w                 1-4 ⟶I
6. |  |   | (∀z)(u=v ∧ v=z ⟶ u=z)               1-5 ∀-I
7. |  | (∀y)(∀z)(u=y ∧ y=z ⟶ u=z)               1-6 ∀-I
8. | (∀x)(∀y)(∀z)(x=y ∧ y=z ⟶ u=z)             1-7 ∀-I
```

From $P(a)$ and $(\exists x)(\forall y)x=y$ we derive $(\exists x)(\forall y)(P(y) \longleftrightarrow x=y)$,

```
1.  | P(a)
2.  | (∃x)(∀y)x=y
    |-------------
3.  |   x| (∀y)x=y
    |    |-------------
4.  |    | z|  | P(z)
    |    |  |  |------
5.  |    |  |  | (∀y)x=y                      3, R
6.  |    |  |  | x=z                          5 ∀-E
7.  |    |  | P(z) ⟶ x=z                      4-6 ⟶I
    |    |  |
8.  |    |  |  | x=z
    |    |  |  |------
9.  |    |  |  | (∀y)x=y                      3, R
10. |    |  |  | x=a                          9, ∀E
11. |    |  |  | a=z                          8,10 =E
12. |    |  |  | P(a)                         1, R
13. |    |  |  | P(z)                         11,12 =E
14. |    |  | x=z ⟶ P(z)                      8-13 ⟶I
15. |    |  | P(z) ⟷ x=z                      7,14 ⟷I
16. |    | (∀y)(P(y) ⟷ x=y)                   4-15 ∀-I
17. |    | (∃x)(∀y)(P(y) ⟷ x=y)               16,∃-I
18. | (∃x)(∀y)(P(y) ⟷ x=y)                    2, 3-17 ∃-E
```

A relation R^2 is said to be *reflexive* if and only if everything bears R^2 to itself, that is

$(\forall x)R^2(x,x)$

It is said to be *symmetric* if and only if, anything which bears R^2 to another has R^2 borne back to it by that other object, i.e.,

$(\forall x)(\forall y)(R^2(x,y) \longrightarrow R^2(y,x))$

It is said to be *transitive* if and only if, whenever three objects are such that the first bears R^2 to the second and the second bears R^2 to the third, then the first bears R^2 to the third. Symbolically,

$(\forall x)(\forall y)(\forall z)(R^2(x,y) \wedge R^2(y,z) \longrightarrow R^2(x,z))$

It is said to be *non-isolated* if and only if everything bears R^2 to something, i.e.,

$(\forall x)(\exists y)R^2(x,y)$

(We will present these concepts again in a set-theoretic framework when we present more set theory in Chapter 5). Here we prove that any relation which is non-isolated, symmetric, and transitive must also be reflexive, that is

$\{(\forall x)(\exists y)R(x,y), (\forall x)(\forall y)(R(x,y) \longrightarrow R(y,x)), (\forall x)(\forall y)(\forall z)(R(x,y) \wedge R(y,z) \longrightarrow R(x,z))\}$
$\vdash (\forall x)R(x,x)$

```
1.   | (∀x)(∃y)Rxy
2.   | (∀x)(∀y)(Rxy → Ryx)
3.   | (∀x)(∀y)(∀z)(Rxy ∧ Ryz → Rxz)
     |____________
4.   |  u | (∀x)(∃y)Rxy                        1,R
5.   |    | (∃y)Ruy                            4, ∀ − E
6.   |    |  v | Ruv
     |    |    |____
7.   |    |    | (∀x)(∀y)(Rxy → Ryx)           2,R
8.   |    |    | (∀y)(Ruy → Ryu)               7, ∀ − E
9.   |    |    | (Ruv → Rvu)                   8, ∀ − E
10.  |    |    | Rvu                           6,9 → E
11.  |    |    | Ruv ∧ Rvu                     6,11 ∧ I
12.  |    |    | (∀x)(∀y)(∀z)(Rxy ∧ Ryz → Rxz) 3,R
13.  |    |    | (∀y)(∀z)(Ruy ∧ Ryz → Ruz)     12, ∀ − E
14.  |    |    | (∀z)(Ruv ∧ Rvz → Ruz)         13, ∀ − E
15.  |    |    | (Ruv ∧ Rvu → Ruu)             14, ∀ − E
16.  |    |    | Ruu                           11,15 → E
17.  |    | Ruu                                5,6-16, ∃ − E
18.  | (∀x)Rxx                                 4-17 ∀ − I
```

We close our examples with a rather long and difficult problem. Given that there are at most two things, that *a* and *b* are both P's, and they're different, prove that everything is a P.

```
1.   (∃x)(∃y)(∀z)(z=x ∨ z=y)
2.   P(a) ∧ P(b)
3.   a≠b
     ____________
4.    u|  |~P(u)
      |   |________
5.    |   |(∃x)(∃y)(∀z)(z=x ∨ z=y)                 1, R
6.    |   | x|(∃y)(∀z)(z=x ∨ z=y)
      |   |  |_________
7.    |   |  | y|(∀z)(z=x ∨ z=y)
      |   |  |  |_______
8.    |   |  |  | a=x ∨ a=y                         7 ∀ − E
```



```
9.   |  |  |  |  | b=x ∨ b=y                    7 ∀ − E
10.  |  |  |  |  | u=x ∨ u=y                    7 ∀ − E
11.  |  |  |  |  | | u=x
     |  |  |  |  | |_____
12.  |  |  |  |  | | a=x ∨ a=y                  8, R
13.  |  |  |  |  | | a=u ∨ a=y                  11,12 = E
14.  |  |  |  |  | | | a=u
     |  |  |  |  | | |_____
15.  |  |  |  |  | | | ~P(u)                    4, R
16.  |  |  |  |  | | | P(a) ∧ P(b)              2, R
17.  |  |  |  |  | | | P(a)                     16, ∧ E
18.  |  |  |  |  | | | ~P(a)                    14,15 = E
19.  |  |  |  |  | | a≠u                        14-18 ~I
20.  |  |  |  |  | | a≠x                        11,19 = E
21.  |  |  |  |  | | a=x ∨ a=y                  8, R
22.  |  |  |  |  | | a=y                        21,20 ∨ E
23.  |  |  |  |  | | a≠b                        3, R
24.  |  |  |  |  | | y≠b                        22,23 = E
25.  |  |  |  |  | | b=x ∨ b=y                  9, R
26.  |  |  |  |  | | b=x                        24,25 ∨ E
27.  |  |  |  |  | | b=u                        11,26 = E
28.  |  |  |  |  | | P(a) ∧ P(b)                2, R
29.  |  |  |  |  | | P(b)                       28, ∧ E
30.  |  |  |  |  | | ~P(u)                      4, R
31.  |  |  |  |  | | P(u)                       27,29 = E
32.  |  |  |  |  | u≠x                          11-31 ~I
33.  |  |  |  |  | u=y                          10,32 ∨ E
34.  |  |  |  |  | a=x ∨ a=u                    8,33 = E
35.  |  |  |  |  | | a=u
     |  |  |  |  | |_____
36.  |  |  |  |  | | ~P(u)                      4, R
37.  |  |  |  |  | | P(a) ∧ P(b)                2, R
38.  |  |  |  |  | | P(a)                       37, ∧ E
39.  |  |  |  |  | | P(u)                       35,38 = E
40.  |  |  |  |  | a≠u                          35-39 ~I
41.  |  |  |  |  | a=x                          34,40 ∨ E
42.  |  |  |  |  | a≠b                          3, R
43.  |  |  |  |  | x≠b                          41,42 = E
44.  |  |  |  |  | b=y                          9,43 ∨ E
45.  |  |  |  |  | b=u                          33,44 = E
46.  |  |  |  |  | P(a) ∧ P(b)                  2, R
47.  |  |  |  |  | P(b)                         46 ∧ E
48.  |  |  |  |  | P(u)                         45,47 = E
49.  |  |  |  |  | ~P(u)                        4, R
50.  |  |  |  |  | P(u) ∧ ~P(u)                 48,49 ∧ I
51.  |  |  |  | P(u) ∧ ~P(u)                   6, 7-50 ∃ − E
52.  |  |  | P(u) ∧ ~P(u)                      5, 6-51 ∃ − E
53.  |  |  | P(u)                              52 ∧ E
54.  |  |  | ~P(u)                             52 ∧ E
55.  |  | ~~P(u)                              4-54 ~I
56.  |  | P(u)                                55 ~E
57.  | (∀x)P(x)                               4-56 ∀ − I
```