Propositional Logic Rules	Coq tactic(s), commands, rules, etc.
/\-e1	destruct
/\-e2	destruct
/\-i	split
\/-e	destruct
\/-i1	left
\/-i2	right
->-i	intro / intros
->-e	apply
_|_-i followed by _|_ -e	contradiction
p \/ ~p (law of excluded middle)	generalize (classic p); intro.
~~p |-- p (double negation elimination)	apply NNPP.
_|_ |-- p (bottom elimination)	elimtype False.
make-box	assert
Simple automatic uses of some of above	trivial, auto
Inductive Definitions and Proofs	
Make a standard (non-recursive) definition	Definition
Make an inductive/recursive definition	Inductive (for sets) or Fixpoint (for functions)
Take cases	destruct
Do proof by induction	induction
Equivalence of (P = Q) and (P <-> Q)	apply prop_ext
To go from p->q to p |-- q (reverse ->-i)	revert
Predicate (first-order) logic	
=-i (equality introduction)	reflexivity
=-e (equality elimination)	rewrite (from left to right) rewrite <- (built in shortcut for from right to left) subst, congruence (powerful auto-rewrite tactics!)
∀-i	intro / intros
∀-e	apply (sometimes “generalize” is useful too)
∃-i	exists
∃-e	destruct