Useful Links

Main site: `https://coq.inria.fr/`
Install: `https://github.com/coq/platform`
Sources: `https://github.com/coq/coq`

Installing `coq` with `opam`

```
opam init
eval $(opam env)
opam switch create with-coq 4.05.0
opam pin add coq ${VERSION}
opam install coqide
opam repo add coq-released https://coq.inria.fr/opam/released
opam install coq-sudoku
```

Commands

`Check` *formula*`.`	Check the type of a formula
`Locate "_ <= _".`	Find defintion of identifier
`Compute` *expression*`.`	Compute an expression
`Definition` *ident*`:=` *exp*`.`	Definition
`Definition` *f args*`:=` *exp*`.`	Function definition
`Reset` *ident*`.`	Forget definition
`Require Import` *Library*`.`	Import a library
`SearchPattern` *pattern*`.`	Search for a pattern (type)
`Search` *patterns*`.`	Search a combination of patterns
`Print` *ident*`.`	Print more information on *ident*
`Fixpoint` *f args*`:=` *exp*`.`	Recursive definition requires *structural recursion*

Expressions

`True, False`	`Prop`
`1`	`nat`
`1,1`	`nat * nat`
`1=1 / textbackslash1<=2`	`Prop`
`nat -> Prop`	`Type`
`fun x :  nat => x = 3`	`nat -> Prop`
`forall x:nat, x<3 \/ exists y:nat, x=y+3`	`Prop`
`let x := 1 in x+x`	`nat`
`A -> B -> C`	`A` implies that (`B` implies `C`)
`if cond then e1 else e2`	Conditional
`(match e with 0 => true` `| S p => false end)`	Pattern-matching

Libraries

`Bool`	Booleans
`Arith`	Natural Numbers: `0`, `1 = S 0`
`List`	Lists: `nil`, `1::nil`, `[1;2]`, `map f l`, `l++l`

BBB
CCC

AAA
AAA
AAA