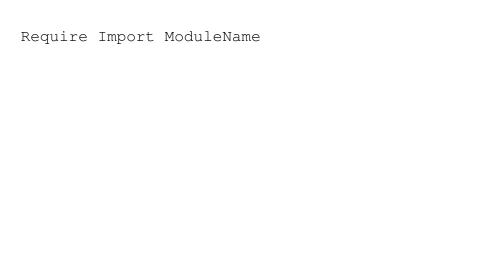
Import a module.



What is a *term*?

What are its subdivisions?

It's a well-formed expression.

Expressions are terms that can be thought of as programs. Types are terms that determine if a type is well-formed and obeys accompanying constraints.

Coq commands all end with what?



In Coq scopes are explicitly ...

... opened and closed, with the most recently opened scopes taking precedence.

Open a scope.

Close a scope.

Open Scope scope. Close Scope scope.

### Determine which interpretations are available for a notation.

Locate "something or other".

Use underscores for identifiers in the string. The bottom interpretation takes precedence.

What is a delimiting key?

Give its syntax.

A name associated with a scope used to explicitly provide the interpretation scope for a term.

(term) %key

Get everything Coq knows about a scope.

#### Print Scope scope

What is the Check command used for?

Give its syntax.

#### Determining if a term is well-formed and what its type is.

Check term.

## Where can decimal notation be used for natural numbers?

# In nat\_scope, whose delimiting key is nat.

## Where can decimal notation be used for all integers?

#### In Z\_scope whose delimiting key is Z.

It is provided in ZArith.

Why can't a nat be used as a Z.

How might this be simulated?

Coq has no type coercion.

There is however the possibility of defining such a conversion using implicit coercions.

What are the two forms of simple types?

- Atomic types, represented by single identifiers.
- Arrow types, of the form Type1 -> Type2.