coffeescript

Source code of released version | Source code of development version

---

CoffeeScript is a little language that compiles into JavaScript. It provides a simple syntax without lots of braces and parentheses. The code compiles one-to-one into the equivalent JS, and there is no interpretation at runtime.

CoffeeScript is supported on both the client and the server. Files ending with `.coffee`, `.litcoffee`, or `.coffee.md` are automatically compiled to JavaScript.

Namespacing and CoffeeScript

Here's how CoffeeScript works with Meteor's namespacing.

- Per the usual CoffeeScript convention, CoffeeScript variables are file-scoped by default (visible only in the `.coffee` file where they are defined.)

- When writing a package, CoffeeScript-defined variables can be exported like any other variable (see Writing Packages). Exporting a variable pulls it up to package scope, meaning that it will be visible to all of the code in your app or package (both `.js` and `.coffee` files).

- Package-scope variables declared in `.js` files are visible in any `.coffee` files in the same app or project.

- There is no way to make a package-scope variable from a `.coffee` file other than exporting it. We couldn't figure out a way to make this fit naturally inside the CoffeeScript language. If you want to use package-scope variables with CoffeeScript, one way is to make a short `.js` file that declares all of your package-scope variables. They can then be read, mutated, and extended in `.coffee` files.

- If you want to share variables between `.coffee` files in the same package, and don't want to separately declare them in a `.js` file, we have an experimental feature that you may like. An object called `share` is visible in CoffeeScript code and is shared across all `.coffee` files in the same package. So, you can write `share.foo` for a value that is shared between all CoffeeScript code in a package, but doesn't escape that package.

Modules and CoffeeScript

See Modules » CoffeeScript Syntax.

Testing This Package

Follow the instructions to check out the Meteor repo and run `test-packages`. Once you can do that successfully, to test the `coffeescript` package run:

```
./meteor test-packages packages/non-core/coffeescript
```