Navigating Ruby Files with Vim - The Cheat Sheet

Precision motions for Ruby

Look up Ruby motions by running `:help ruby-motion`.

command	effect
]m	go to start of next method definition
]M	go to end of next (or current) method definition
[m	go to start of previous method definition
[M	go to end of previous method definition
]]	go to start of next module or class definition.
[[	go to start of previous module or class definition.
%	jump between matchin keyword pairs

The `%` command is provided by the matchit plugin. All of the other motions are implemented by vim-ruby.

Text objects for working with Ruby

Look up Ruby motions by running `:help ruby-text-objects`.

command	effect
ir	inside the current rubyblock
ar	around the current rubyblock
im	inside the current method definition
am	around the current method definition
iM	inside the current class or module
aM	around the current class or module

The `ir` and `ar` text objects are supplied by the textobj-rubyblock plugin. All of the other text objects are implemented by vim-ruby.

Jump to filenames

When your `path` option is properly configured, you can use these commands to navigate your project:

command	effect
gf	jump to the filename under the cursor
:find {file}	jump to the specified {file} in path

Jump to definitions

When your codebase has been indexed with ctags, you can use these commands to navigate your project:

command	effect
<C-]>	jump to the first tag that matches the word under the cursor
:tag {keyword}	jump to the first tag that matches {keyword}
g<C-]>	prompt user to select from multiple matches for the word under the cursor. If only one match exists, jump to it without prompting.
:tselect {keyword}	prompt user to select from multiple matches for {keyword}. If only one match exists, jump to it without prompting.

Tab completion at Vim’s command-line

At Vim’s command line, you can use tab completion:

command	effect
<Tab>	expand commandline to use next match
<S-Tab>	expand commandline to use previous match
<C-d>	show a list of all possible matches

For example, if you type:

```
:tag assert
```

Then press `<C-d>`, Vim will reveal a list of all `assert_*` methods. Press `<Tab>` to expand the command line to use the next match from that list.