

rails c

github.com/przprz/rails_console

- CRUD operations
- Monkey-patching
- Configuration
- Keyboard shortcuts
- Tips
- Links

CRUD operations

- documents https://guides.rubyonrails.org/active_record_querying.html
- cheat-sheet <https://makandracards.com/makandra/42641-different-ways-to-set-attributes-in-activerecord>

Playing around

```
class Pinger
  def ping!(patch)
    # if patch
    #   puts :pong
    #   return
    # end
    puts :ping
  end
end
```

```
Pinger.new.ping!
```

- searching for methods

```
ClaimPayout.first.methods.grep /_at$/
```

```
file, line=ClaimPayout.first.method(:currency).source_location
y IO.readlines(file)[line-1, 10]
```

Configuration

- put it in your .irbrc

```
# ~/.irbrc
def ta
  [1, 2, :three, 'four']
end

def ha
  {a: 1, :b => nil, 'three' => false, 4 => 4}
end
```

- example: lets copy `awesome!` that AH consoles have

We'll need to look for `awesome_print` in `AirHelp/dockerfiles` repo

Keyboard shortcuts

Most *readline* shortcuts are supported

TIP: many other applications also support some of these (try in your browser)

- tab - autocompletion
- ctrl+l - `clear` screen
- ctrl+p/ctrl+n - show previous/next command (but it acts weirdly via ssh)
- ctrl+a/ctrl+e - go to beginning/end of line
- alt+b/alt+f - go to previous/next word
- ctrl+b/ctrl+f - go to previous/next character
- ctrl+w/alt+d - delete previous/next word

Keyboard shortcuts

- `ctrl+u/ctrl+k` - yank text from current position to the beginning/end of line
- `ctrl+y` - paste yanked text
- `ctrl+r` - recursive search
- `ctrl+j/ctrl+m` - insert new line (acts as ENTER key)

Tips

`$ rails console -e production --sandbox` - when you quit your session everything is rolled back!

`$ rails console -- --nomultiline` - use nomultiline if you need to paste some long code (e.g. when monkey-patching)

`reload!` - loads latest version of code (clears monkey-patches)

`_` - stores result of previous command

`ap Claim.last` - pretty prints

`y Caim.last` - "yaml" prints (display content by serializing it to YAML)

`ActiveRecord::Base.connection.tables` - list all tables in application

Links

https://guides.rubyonrails.org/command_line.html#bin-rails-console

<https://hackernoon.com/meeting-the-query-interface-in-ruby-on-rails-9bu3yec>

https://guides.rubyonrails.org/active_record_querying.html

<https://pragmaticstudio.com/tutorials/rails-console-shortcuts-tips-tricks>

<https://medium.com/better-programming/rails-console-magic-tricks-da1fdd657d32>

<https://www.bounga.org/tips/2018/10/23/rails-console-tips/>