

Introduction

Set up

Prepare the app

Deploy the app

View logs

Define a Procfile

Scale the app

Declare app dependencies

Run the app locally

Push local changes

Provision add-ons

Start a one-off dyno

Define config vars

Use a database

Next steps

Start a one-off dyno

You can run a command, typically scripts and applications that are part of your app, in a one-off dyno using the `heroku run` command. It can also be used to launch a REPL process attached to your local terminal for experimenting in your app's environment:

```
> heroku run jirb
Running `rails console` attached to terminal... up, run.1594
Loading production environment (Rails 4.1.4)
irb(main):001:0>
```

If you receive the error, "Error connecting to process", then you may need to configure your firewall.

When the console starts, it has your entire app loaded. For example, you can type `puts Cowsay.say("hi")` and an animal saying "hi" will be displayed. Type `exit` to quit the console.

```
irb(main):001:0> require 'cowsay'
require 'cowsay'
=> true
irb(main):002:0> puts Cowsay.say("hi")
puts Cowsay.say("hi")

  __
 | hi |
 ----
   \
    \
      .--.
     |o_o |
     |:/_ |
    //  \ \
   (|    |)
  /'\_  _/'\
   \__)=(_)/

=> nil
irb(main):003:0> exit
```

To get a real feel for how dynos work, you can create another one-off dyno and run the `bash` command, which opens up a shell on that dyno. You can then execute commands there. Each dyno has its own ephemeral filesystem, populated with your app and its dependencies. When the command completes (in this case, `bash`), the dyno is removed.

```
> heroku run bash
Running `bash` attached to terminal... up, run.1421
~ $ ls
app  config db  Gemfile.lock  log  public  README.rdoc  tmp
bin  config.ru  Gemfile  lib          Procfile  Rakefile  test  vendor
~ $ exit
exit
```

Don't forget to type `exit` to exit the shell and terminate the dyno.

[Log in to report a problem \(/login?back_to=%2Farticles%2Fgetting-started-with-jruby%23start-a-one-off-dyno\)](/login?back_to=%2Farticles%2Fgetting-started-with-jruby%23start-a-one-off-dyno)

I created a one-off dyno that ran a console ([start-a-one-off-dyno](#))

([Log in \(/login?back_to=%2Farticles%2Fgetting-started-with-jruby%23start-a-one-off-dyno\)](#) to save and track your progress)