

The Polyglot

LIFESTYLE

Johnny Winn



@johnny_rugger



hashrocket.com

The Polyglot

LIFESTYLE

Johnny Winn



@johnny_rugger



hashrocket.com



ELIXIR FOUNTAIN

FOLLOW
ON TWITTER



Sign up for the latest Elixir news every week

Sign Up

We won't sign you up for credit cards or sell your address or anything like that

 HASHROCKET

SUGGESTIONS

Send news to
info@elixir-fountain.com









Family

Why

is he sharing this?



Learning



Learning

**Knowledge is constructed
rather than acquired**

constructivism

**Gather resources
to support the
learning process**

scaffolding

What

**does this have to do
with programming?**



Needs more
scaffolding

How

**do we live the
Polyglot Lifestyle?**

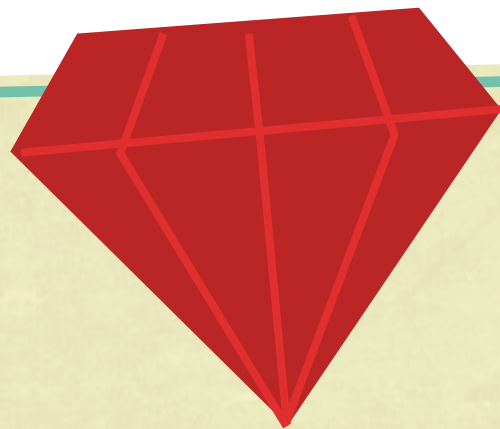
How

**do we live the
Polyglot Lifestyle?**

Just Enough

Elixir

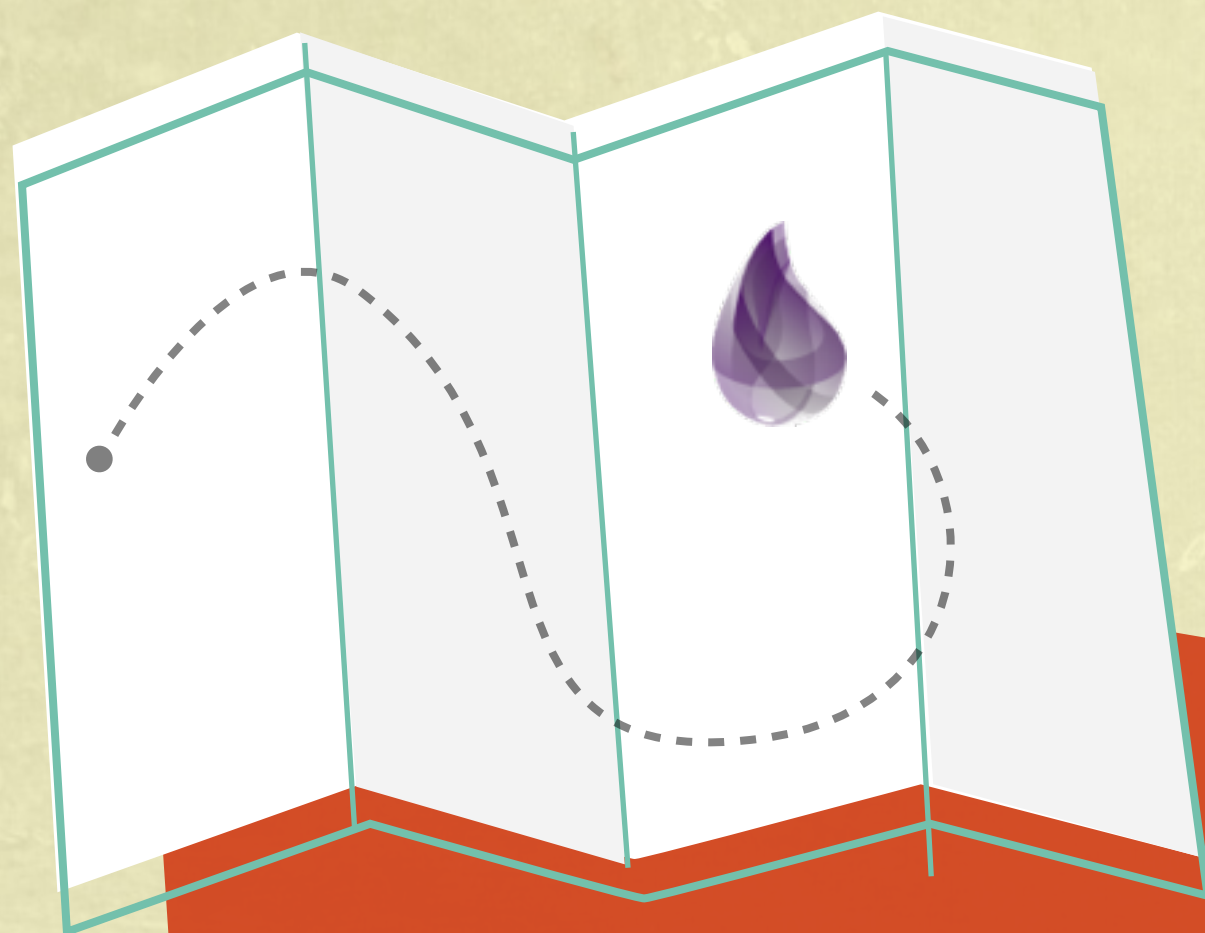
a touch of



Ruby

MIX It

ALL Together



Just
Enough

Elixir

The fun

IN Function


```
johnny@MAXX:~$ iex
```

```
Erlang R16B02 (erts-5.10.3) [source] [64-bit] [smp:4:4] [async-threads:10] [hipe] [kernel-poll:false]
```

```
Interactive Elixir (0.12.5-dev) - press Ctrl+C to exit (type h() ENTER for help)
```

```
iex(1)> █
```



```
johnny@MAXX:~$ irb  
2.0.0p195 :001 > █
```



```
johnny@MAXX:~$ irb  
2.0.0p195 :001 > █
```



```
johnny@MAXX:~$ iex
```

```
Erlang R16B02 (erts-5.10.3) [source] [64-bit] [smp:4:4] [async-threads:10] [hipe] [kernel-poll:false]
```

```
Interactive Elixir (0.12.5-dev) - press Ctrl+C to exit (type h() ENTER for help)
```

```
iex(1)> █
```


default values \\true

variables

:atoms

No, period

elixir-lang.org/crash-course.html

[illegible]

All

Recursion

```
1 def sum(list, acc=0)
2   if list.length == 0
3     puts acc
4   else
5     i = list.pop
6     sum(list, acc + 1)
7   end
8 end
9
10 sum((1...50000).to_a)
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
```

redo.rb

10,14

Top

"redo.rb" 54L, 348C written

Recursion


```
johnny@MAXX:~/projects/ruby$ ruby redo.rb  
redo.rb:2: stack level too deep (SystemStackError)  
johnny@MAXX:~/projects/ruby$ █
```

Recursion

```
1 define_method(:acc) do |list, n|
2   if list.length == 0
3     puts n
4   else
5     i = list.pop
6     n = i + n
7     redo
8   end
9 end
10
11 def sum(list)
12   acc(list, 0)
13 end
14
15 sum (1...500000).to_a
```

2 2 2 2 2 2 2 2 2 2 2

redo.rb

1,1

All

"redo.rb" 15L, 180C

Recursion


```
johnny@MAXX:~/projects/ruby$ ruby redo.rb  
124999750000  
johnny@MAXX:~/projects/ruby$ █
```

Recursion

Throw IT

In The mix


```

1 defmodule Mix.Tasks.Phoenix.New do
2   use Mix.Task
3
4   @shortdoc "Creates Phoenix application"
5
6   @template_dir "template"
7
8   def run([]) do
9     Mix.shell.info """
10     Supply application name and destination path.
11
12     e.g.
13       mix phoenix.new photo_blog /home/johndoe/
14     """
15   end
16
17   @doc """
18   Creates Phoenix application.
19   """
20   def run([name, path]) do
21     application_name = Mix.Utils.underscore(name)
22     application_module = Mix.Utils.camelize(name)
23     project_path = make_project_path(path, application_name)
24
25     bindings = [application_name: application_name,
26                 application_module: application_module]
27
28     Mix.Generator.create_directory(project_path)
29
30     lc source_path inlist template_files do
31       destination_path = make_destination_path(project_path, source_path, application_name)
32
33       if File.dir?(source_path) do

```

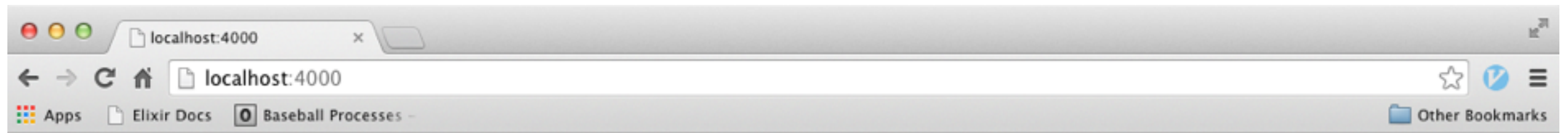
```
johnny@MAXX:~/projects/elixir/phoenix(master)$ mix phoenix.new erlang_factory ~/projects/erlang_factory
```



```
johnny@MAXX:~/projects/elixir/phoenix(master)$ mix phoenix.new erlang_factory ~/projects/erlang_factory
* creating /Users/johnny/projects/erlang_factory
* creating /Users/johnny/projects/erlang_factory/.gitignore
* creating /Users/johnny/projects/erlang_factory/README.md
* creating /Users/johnny/projects/erlang_factory/lib
* creating /Users/johnny/projects/erlang_factory/lib/erlang_factory
* creating /Users/johnny/projects/erlang_factory/lib/erlang_factory.ex
* creating /Users/johnny/projects/erlang_factory/lib/erlang_factory/controllers
* creating /Users/johnny/projects/erlang_factory/lib/erlang_factory/controllers/pages.ex
* creating /Users/johnny/projects/erlang_factory/lib/erlang_factory/router.ex
* creating /Users/johnny/projects/erlang_factory/lib/erlang_factory/supervisor.ex
* creating /Users/johnny/projects/erlang_factory/mix.exs
* creating /Users/johnny/projects/erlang_factory/mix.lock
* creating /Users/johnny/projects/erlang_factory/test
* creating /Users/johnny/projects/erlang_factory/test/erlang_factory_test.exs
* creating /Users/johnny/projects/erlang_factory/test/test_helper.exs
johnny@MAXX:~/projects/elixir/phoenix(master)$ cd ~/projects/erlang_factory/
johnny@MAXX:~/projects/erlang_factory$ mix do deps.get, compile
```



```
Compiled lib/phoenix/config/app.ex
Compiled lib/phoenix/config/config.ex
Compiled lib/phoenix/config/fallback.ex
Compiled lib/phoenix/plugs/code_reloader.ex
Compiled lib/phoenix/plugs/error_handler.ex
Compiled lib/phoenix/controller/controller.ex
lib/phoenix/plugs/logger.ex:15: warning: variable conn is unused
Compiled lib/phoenix/plugs/logger.ex
Compiled lib/phoenix/router/errors.ex
Compiled lib/phoenix/router/console_formatter.ex
Compiled lib/phoenix/router/options.ex
Compiled lib/phoenix/examples/controller.ex
Compiled lib/phoenix/router/resources_context.ex
Compiled lib/phoenix/router/mapper.ex
Compiled lib/phoenix/router/path.ex
lib/phoenix/router/router.ex:3: warning: unused alias Controller
Compiled lib/phoenix/router/router.ex
Compiled lib/phoenix/supervisor.ex
Compiled lib/phoenix/router/stack.ex
Compiled lib/phoenix/router/scope_context.ex
Compiled lib/phoenix/examples/router.ex
Generated phoenix.app
Compiled lib/erlang_factory/controllers/pages.ex
Compiled lib/erlang_factory.ex
Compiled lib/erlang_factory/supervisor.ex
Compiled lib/erlang_factory/router.ex
Generated erlang_factory.app
johnny@MAXX:~/projects/erlang_factory$ mix run -e 'ErlangFactory.Router.start' --no-halt mix.exs
```

Hello world

```
Compiled lib/phoenix/config/fallback.ex
Compiled lib/phoenix/plugs/code_reloader.ex
Compiled lib/phoenix/plugs/error_handler.ex
Compiled lib/phoenix/controller/controller.ex
lib/phoenix/plugs/logger.ex:15: warning: variable conn is unused
Compiled lib/phoenix/plugs/logger.ex
Compiled lib/phoenix/router/errors.ex
Compiled lib/phoenix/router/console_formatter.ex
Compiled lib/phoenix/router/options.ex
Compiled lib/phoenix/examples/controller.ex
Compiled lib/phoenix/router/resources_context.ex
Compiled lib/phoenix/router/mapper.ex
Compiled lib/phoenix/router/path.ex
lib/phoenix/router/router.ex:3: warning: unused alias Controller
Compiled lib/phoenix/router/router.ex
Compiled lib/phoenix/supervisor.ex
Compiled lib/phoenix/router/stack.ex
Compiled lib/phoenix/router/scope_context.ex
Compiled lib/phoenix/examples/router.ex
Generated phoenix.app
Compiled lib/erlang_factory/controllers/pages.ex
Compiled lib/erlang_factory.ex
Compiled lib/erlang_factory/supervisor.ex
Compiled lib/erlang_factory/router.ex
Generated erlang_factory.app
johnny@MAXX:~/projects/erlang_factory$ mix run -e 'ErlangFactory.Router.start' --no-halt mix.exs
>> Running Elixir.ErlangFactory.Router with Cowboy with [port: 4000, ssl: true]
```



```
johnny@MAXX:~/projects/erlang_factory$ mix compile
```


THE RISE OF

PHOENIX

```
1 defmodule ErlangFactory.Config.Router do
2   use Phoenix.Router, port: 4000
3
4   get "/", ErlangFactory.Controllers.Pages, :index, as: :page
5 end
```



```
1 defmodule ErlangFactory.Config.Router do
2   use Phoenix.Router, port: 4000, dispatch: [
3     { :_, [
4       {"/stylesheets/...", :cowboy_static, {:dir, "public/stylesheets"}},
5       {"/javascript/...", :cowboy_static, {:dir, "public/javascript"}},
6       {:_ , Plug.Adapters.Cowboy.Handler, { __MODULE__, [] }}
7     ]}
8   ]
9
10  get "/", ErlangFactory.Controllers.Pages, :index, as: :page
11 end
```

```
lib/erlang_factory/config/router.ex
```

All

```
johnny@MAXX:~/projects/elixir/erlang_factory(master)$ mix phoenix.routes
```


Calliope

THE Muse
the muse


```

1 defmodule Phoenix.Controller do
2   import Plug.Connection
3
4   defmacro __using__(_options) do
5     quote do
6       import Plug.Connection
7       import unquote(__MODULE__)
8     end
9   end
10
11   def haml(conn, haml, args//[]) do
12     html(conn, Calliope.render(haml, conn.params ++ args))
13   end
14
15   def json(conn, json), do: json(conn, 200, json)
16   def json(conn, status, json) do
17     send_response(conn, status, "application/json", json)
18   end
19
20   def html(conn, html), do: html(conn, 200, html)
21   def html(conn, status, html) do
22     send_response(conn, status, "text/html", html)
23   end
24
25   def text(conn, text), do: text(conn, 200, text)
26   def text(conn, status, text) do

```

deps/phoenix/lib/phoenix/controller/controller.ex [+]

1,1

Top

4 more lines

```
1 defmodule Calliope.Safe do
2
3   @doc """
4   Calliope.Safe is a utility for evaluating code and escaping HTML tag
5   characters.
6   """
7
8   @html_escape [
9     { "&", "&amp;" },
10    { "<", "&lt;" },
11    { ">", "&gt;" },
12    { "\"", "&quot;" },
13    { "'", "&#39;" },
14  ]
15
16  def eval_safe_script(<< "Safe.script", script :: binary >>, args) do
17    evaluate_script(args, script)
18  end
19  def eval_safe_script(script, args) do
20    clean args |> evaluate_script script
21  end
22
23  def evaluate_script(args, script) do
24    { result, _ } = Code.string_to_quoted!(script) |> Code.eval_quoted(args)
25    result
26  end
27 end
```


remember

THE Ecto

```
1 defmodule ErlangFactory.Mixfile do
2   use Mix.Project
3
4   def project do
5     [ app: :erlang_factory,
6       version: "0.0.1",
7       elixir: "~> 0.12.2",
8       deps: deps ]
9   end
10
11  def application do
12    [mod: { ErlangFactory, [] }]
13  end
14
15  defp deps do
16    [
17      { :phoenix, github: "nvrugger07/phoenix", branch: "render_haml_with_calliope"},
18      { :postgrex, github: "ericmj/postgrex"},
19      { :ecto, github: "elixir-lang/ecto"}
20    ]
21  end
22 end
```

ecto deps



~
~
~
~

mix.exs

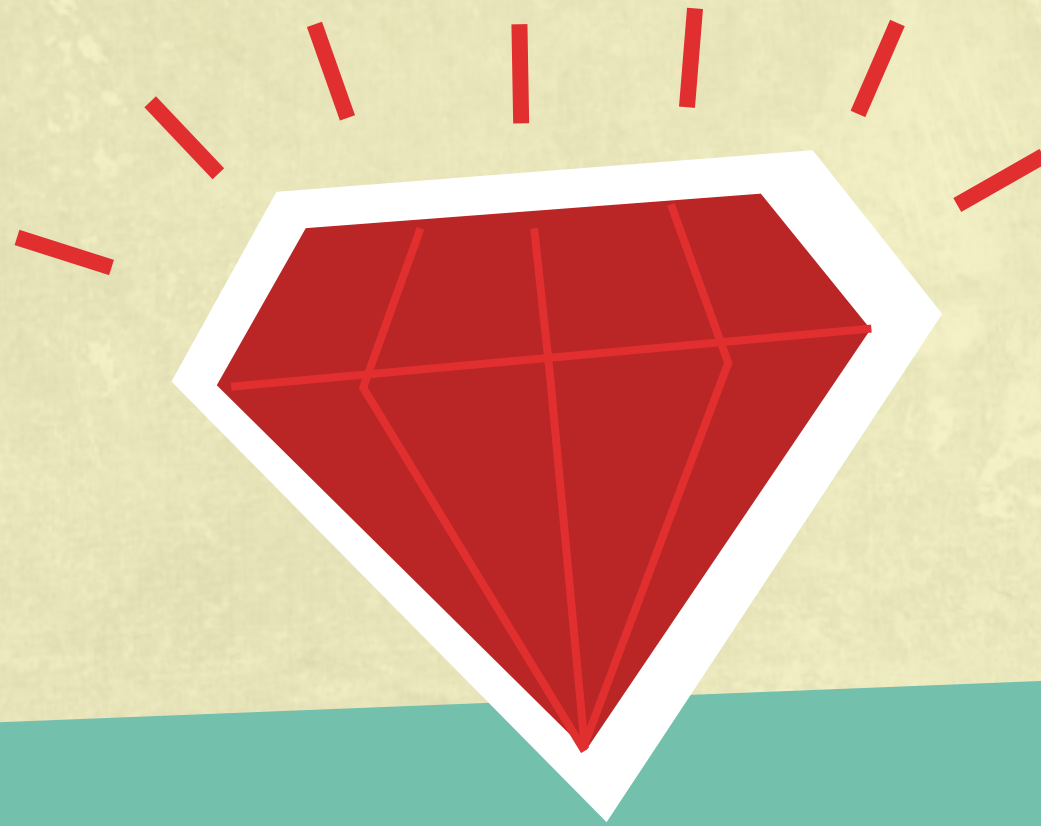
1,1

All

"mix.exs" 22L, 450C

now what's

Missing?



a touch of
Ruby

where ruby

Shines

~~~~~

1,1

"Gemfile" 4L, 55C written

talk gems



# SASS

In **sassy**





Behavior-driven

---

Development

~~~~~

1,1

All

testing gems


~~~~~

"test/test\_helper.exs" 2L, 47C

\_\_\_\_\_









## Maintaining Database

```
create_table "locations",  
  force: true do |t|  
    t.string "street_1", null: false  
    t.string "city",      null: false  
    t.string "state",     null: false  
    t.string "zip",       null: false  
    t.string "street_2"
```

# Schema

2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

All



```
1 #!/usr/bin/env rake
2
3 require 'active_record'
4
5 Dir['test/features/support/models/*'].each do |file|
6   require File.expand_path(file)
7 end
8
9 namespace :db do
10   PG_SPEC = {
11     :adapter => 'postgresql',
12     :host    => 'localhost',
13     :database => 'blog_dev',
14     :username => 'postgres',
15   }
16
17   def connection(config)
18     ActiveRecord::Base.establish_connection(config)
19     ActiveRecord::Migration.verbose = false
20   end
21
22   def run_migrations(direction, config)
23     connection(config)
24     ActiveRecord::Migration.descendants.each { |m| m.send(direction) }
25   end
26
```

```
1 #!/usr/bin/env rake
2
3 require 'active_record'
4
5 Dir['test/features/support/models/*'].each do |file|
6   require File.expand_path(file)
7 end
8
9 namespace :db do
10   PG_SPEC = {
11     :adapter => 'postgresql',
12     :host    => 'localhost',
13     :database => 'blog_dev',
14     :username => 'postgres',
15   }
16
17   def connection(config)
18     ActiveRecord::Base.establish_connection(config)
19     ActiveRecord::Migration.verbose = false
20   end
21
22   def run_migrations(direction, config)
23     connection(config)
24     ActiveRecord::Migration.descendants.each { |m| m.send(direction) }
25   end
26
```

```
26
27 task :migrate do
28   run_migrations(:up, PG_SPEC)
29 end
30
31 task :rollback do
32   run_migrations(:down, PG_SPEC)
33 end
34
35 namespace :test do
36   def config
37     PG_SPEC.merge({:database => 'blog_test'})
38   end
39
40   task :prepare do
41     run_migrations(:up, config)
42   end
43
44   task :rollback do
45     run_migrations(:down, config)
46   end
47 end
48
49 end
```



```
1 class CucumberPosts < ActiveRecord::Migration
2
3   def self.up
4     execute <<-SQL
5       CREATE TABLE IF NOT EXISTS posts (
6         id serial PRIMARY KEY,
7         headline varchar(255) NOT NULL,
8         content text NOT NULL,
9         user_id integer NOT NULL,
10        published_on date);
11   SQL
12   end
13
14   def self.down
15     execute "DROP TABLE IF EXISTS posts;"
16   end
17
18 end
```

~  
~  
~  
~  
~  
~  
~  
~  
~  
~

test/features/support/models/posts.rb

1,1

All

"test/features/support/models/posts.rb" 18L, 369C written

MIGRATE!

```
johnny@MAXX:~/projects/elixir/erlang_factory(master)$ rake db:migrate db:test:prepare
```

**MIGRATE!**



Fireworks!





```

1 class CucumberPosts < ActiveRecord::Migration
2   attr_accessor :headline, :content, :published_on, :user_id
3
4   def initialize(args)
5     args = Hash[args.map{ |k, v| [k.gsub(" ", "_").downcase.to_sym, v] }]
6     args.each do |key, val|
7       instance_variable_set("@#{key}", val)
8     end
9   end
10
11   def user_id
12     @user_id || 1
13   end
14
15   def self.insert(post)
16     execute <<-SQL
17       INSERT INTO posts (headline, content, published_on, user_id)
18       VALUES ('#{post.headline}', '#{post.content}',
19               '#{post.published_on}', #{post.user_id});
20     SQL
21   end
22
23   def self.clear
24     execute %q{DELETE FROM posts;}
25   end
26

```

test/features/support/models/posts.rb [+]

1,1

Top

entity support

```
1 Given "the following posts:" do |table|
2   table.hashes.map do |hash|
3     CucumberPosts.insert CucumberPosts.new(hash)
4   end
5 end
6
7 When "I am on the home page" do
8   visit "/"
9 end
10
11 Then "I should see the published posts" do
12   page.should have_content "Hell Is Other Robots"
13   page.should have_content "A crummy world of plot holes and spelling errors!"
14   page.should have_content "When Aliens Attack"
15
16   page.should_not have_content "A Pharaoh to Remember"
17 end
```

~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~

test/features/step\_definitions/posts\_steps.rb [+]

1,1

All

4 more lines

connect me





```
johnny@MAXX:~/projects/elixir/erlang_factory(master*)$ mix test
```

entity support

time to Mix it

together

---



~~~~~

All

the repo


```
1 defmodule Post do
2   use Ecto.Model
3
4   queryable "posts" do
5     field :headline, :string
6     field :content, :string
7     field :published_on, :datetime
8   end
9 end
10
11 defmodule PostQueries do
12   import Ecto.Query
13
14   alias ErlangFactory.Repo
15
16   def published do
17     Enum.filter_map(all, &is_published?(&1), &{ &1.headline, &1.content })
18   end
19
20   def is_published?(post), do: post.published_on < :erlang.date
21
22   defp all do
23     query = from p in Post,
24             select: p
25     Repo.all(query)
26   end
27 end
```

lib/erlang_factory/models/post.ex

1,1

Top

post entity


```
1 defmodule ErlangFactory.Controllers.Pages do
2   use Phoenix.Controller
3
4   @views "lib/erlang_factory/views"
5
6   def index(conn) do
7     haml conn, File.read!("#{@views}/pages/index.haml"), posts: PostQueries.published
8   end
9 end
```

lib/erlang_factory/controllers/pages.ex 1,1 All
"lib/erlang_factory/controllers/pages.ex" 9L, 225C

the controller

```

1  !!!
2  %html{lang: "en-US"}
3  %head
4    %title Erlang Factory 2014
5    %meta{charset: "UTF-8"}
6    <meta content="width=device-width", maximum-scale=1.0, user-scalable=0, name="viewport">
7    /[if IE]
8      <script src="/javascripts/html5.js" type="text/javascript"></script>
9    %link{href: "/stylesheets/app.css", media: "screen", rel: "stylesheet", type: "text/css"}
10   %link{href: "http://fonts.googleapis.com/css?family=Raleway:800,200|Merriweather:400,400italic", rel
: "stylesheet", type: "text/css"}
11
12   %body
13     %header
14       .container
15         %h1
16           %a{href: "#"} Erlang Factory 2014
17       .container
18         %p An Erlang Conference with an Elixir Problem
19
20     .container
21       %section
22         - lc { headline, content } inlist posts do
23           %article
24             %h2= headline
25             %p= content

```

lib/erlang_factory/views/pages/index.html 1,1 Top

the template

```
johnny@MAXX:~/projects/elixir/erlang_factory(master)$ mix test
```

passing tests

wrapping up

So now what?


```
johnny@MAXX:~/projects/elixir/erlang_factory(master*)$ mix test
```

the refactor

The Polyglot

LIFESTYLE

Johnny Winn



@johnny_rugger



hashrocket.com

Want TO LEARN
More about

Elixir?

Elixir => <http://elixir-lang.org/>

Elixir Sips => <http://elixirsips.com/>

Phoenix => phoenixframework/phoenix

Calliope => nurugger07/calliope

Ecto => elixir-lang/ecto