

 免费电子书

学习

phoenix-framework

Free unaffiliated eBook created from
Stack Overflow contributors.

#phoenix-
framework

.....	1
1: phoenix-framework	2
.....	2
.....	2
Examples.....	4
.....	4
.....	5
.....	5
OSXElixir / Phoenix.....	6
.....	7
2: Ecto	8
.....	8
Examples.....	8
.....	8
ecto.....	8
3:	9
Examples.....	9
.....	9
.....	10

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [phoenix-framework](#)

It is an unofficial and free phoenix-framework ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official phoenix-framework.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: phoenix-framework

phoenix-framework。

◦ phoenix-framework。

0.1.1	2014-05-01
0.2.0	2014-05-01
0.2.1	2014-05-01
0.2.2	201465
0.2.3	201455
0.2.10	2014522
0.2.11	2014-06-30
0.3.0	201471
0.3.1	201475
0.4.0	2014831
0.4.1	201499
0.5.0	20141014
0.6.0	
0.6.1	
0.6.2	2014128
0.7.0	
0.7.1	
0.7.2	
0.8.0	2015111
0.9.0	2015212
0.10.0	201538

0.11.0	2015-04-09
0.12.0	201551
0.13.0	○
0.13.1	2015517
0.14.0	
0.15.0	2015727
0.16.0	201586
0.16.1	201586
0.17.1	2015827
1.0.0	2015828
1.0.1	201593
1.0.2	201597
1.0.3	2015929
1.0.4	20151215
1.1.0	2015916
1.1.1	2015927
1.1.2	201619
1.1.3	2016120
V1.2.0-rc.0	2016429
V1.2.0-RC.1	2016525
1.2.0	2016623
1.2.2	2017314
1.2.3	2017315
1.2.4	2017516
1.3.0 RC.1	2017315
1.3.0 rc.2	2017516

Examples

[PhoenixElixir](#)[ElixirErlang](#)[Erlang VM](#)◦ [phoenix](#)◦ [phoenix](#)

1.[Elixir](#) ◦ [ElixirElixir](#) ◦

2.[Hex](#)◦ [HexPhoenix](#)◦

```
$ mix local.hex
```

[Hex](#)◦

3.[Erlang](#) ◦ [ErlangElixir](#)[ElixirErlangVM](#)◦ [ElixirErlangElixirErlang](#)◦ [DebianErlang](#)◦

```
$ wget https://packages.erlang-solutions.com/erlang-solutions_1.0_all.deb && sudo dpkg -i
erlang-solutions_1.0_all.deb
$ sudo apt-get update
$ sudo apt-get install esl-erlang
```

4.[phoenix](#) ◦ [ElixirErlangPhoenix Mix](#)◦ [MixZipBEAM](#)◦ ◦ ◦ [Phoenix](#)

```
$ mix archive.install https://github.com/phoenixframework/archives/raw/master/phoenix_new.ez
```

◦ [Phoenix](#)

```
mix archive.install /path/to/local/phoenix_new.ez
```

5 [PlugCowboyEctoPhoenixmixmixPhoenix](#)◦ [mix](#)◦

6.[Node.js v5.0.0](#)◦ ◦ [Node.jsbrunch.io](#)◦ [Brunch.iojavascriptcss](#)◦

[node.js](#)◦[Phoenix5.0.0](#)◦

[Mac OS Xnode.js](#)◦

[io.jsNode.jsnpmPhoenix](#)◦

[Debian / Ubuntu](#)

```
sh: 1: node: not found
npm WARN This failure might be due to the use of legacy binary "node"
```

[DebianSO](#)

[Ubuntu](#)

[nodejs-legacy](#)

```
$ apt-get install nodejs-legacy
```

```
$ ln -s /usr/bin/nodejs /usr/bin/node
```

7 [PostgreSQL](#) ◦ [Phoenix--database mysql](#) [MySQL](#) ◦ [PostgreSQL wiki](#) ◦

[PostgrexPhoenix](#) ◦ [Phoenix Postgres](#) ◦

8 [inotify-tools](#) [linuxLinuxPhoenix](#) ◦ [Mac OS XWindows](#) ◦

[Linux](#) ◦ [inotify-tools wiki](#) ◦

◦ [follow](#) ◦

```
mix phoenix.new web --no-brunch --no-ecto
```

[ElixirErlangHexMixPhoenix](#)

[Phoenix ElixirErlangHexPhoenix](#) ◦ [PostgreSQLnode.js](#) ◦

◦ [phoenix.newmix](#) ◦ [hello_phoenix_world](#)

```
$ mix phoenix.new hello_phoenix_world
```

[mix phoenix.newPhoenix](#) ◦ [Phoenix](#)

```
$ mix phoenix.new /Users/username/work/elixir-projects/hello_phoenix_world
```

```
mix phoenix.new hello_phoenix_world
* creating hello_phoenix_world/config/config.exs
* creating hello_phoenix_world/config/dev.exs
* creating hello_phoenix_world/config/prod.exs
...
* creating hello_phoenix_world/web/views/layout_view.ex
* creating hello_phoenix_world/web/views/page_view.ex

Fetch and install dependencies? [Yn]
```

[Phoenix](#) ◦ [Mix](#) ◦ [""](#) ◦

```
Fetch and install dependencies? [Yn] Y
* running mix deps.get
* running npm install && node node_modules/brunch/bin/brunch build
```

◦

Move into your new project folder:

```
$cd hello_phoenix_world
```

postgres postgres username postgres config/dev.exs

```
# config/dev.exs
config :hello_phoenix_world, HelloPhoenixWorld.Repo,
  adapter: Ecto.Adapters.Postgres,
  username: "postgres",
  password: "postgres",
  database: "hello_phoenix_world_dev",
  hostname: "localhost",
  pool_size: 10
```

Now, create the database with the ecto mix task:

```
$ mix ecto.create
```

We have a working application! Run your Phoenix application:

```
$ mix phoenix.server
```

You can also run your app inside IEx (Interactive Elixir) as:

```
$ iex -S mix phoenix.server
```

Load `http://localhost:4000` into your browser and you will see the default landing page of your application.

Phoenix hello world web/templates/page/index.html.eex

```
<h2>Hello World</h2>
```

“Hello World”。

CRUD。

ctrl-c ctrl-c control key。

OSX Elixir / Phoenix

Elixir / Phoenix

Homebrew

```
/usr/bin/ruby -e "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

brew install elixir Elixir - Erlang。

mix local.hex mix。

Phoenix

```
mix archive.install https://github.com/phoenixframework/archives/raw/master/phoenix_new.ez
```


Node.js

NVMNode.js◦ [nvm](#)

```
curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.31.4/install.sh | bash
```

```
curlbrew install curlbrew install curl ◦
```

```
nvm install node
```

Node.js.

[Postgres.app](#)◦ [Phoenix](#)config/dev.exs -

```
config :myphoenixapp, MyPhoenixApp.Repo,  
  adapter: Ecto.Adapters.Postgres,  
  database: "myphoenixapp_dev",  
  hostname: "localhost",  
  pool_size: 10
```

CRUDRails[phoenix.gen.html](#) [mix](#)

```
mix phoenix.gen.html Book books title note:text pages:integer author_id:references:authors
```

Book books title note pages author_id **Author**belongs_to◦

[phoenix-framework](#) <https://riptutorial.com/zh-CN/phoenix-framework/topic/4996/phoenix-framework>

2: Ecto

ecto。

Examples

```
username password_hash email_id created_at updated_at typejson
```

```
mix phoenix.gen.json User users username:string email_id:string password_hash:string  
timestamps()
```

ecto

```
mix phoenix.gen.htmlmix phoenix.gen.json priv -> repo -> migrationspriv -> repo -> migrations。
```

```
mix ecto.migrate。
```

```
mix ecto.gen migrations <model_name>mix ecto.gen migrations <model_name>
```

```
mix ecto.gen migrations <model_name> -r <repo_name>
```

Ecto <https://riptutorial.com/zh-CN/phoenix-framework/topic/10890/ecto>

3:

Examples

- resource
- MyApp.ErrorHelpers.error_tag
- Phoenix.Router MyApp.Router.Helpers.*_path ◦

◦ PhoenixElixir@doc@moduledoc◦

- //
-
- MyApp.RepoEcto.Repo

ex_docmix.exs

```
# config/mix.exs

def deps do
  [[:ex_doc, "~> 0.11", only: :dev]]
end
```

Elixir @doc@moduledoc Markdown◦

mix deps.getmix docsmix docs ◦ [Elixir Docs](#) ◦

mix docs --output priv/static/docmy_app_url_or_ip/doc/index.html ◦

- [ex_doc](#)
- [Elixir.Version](#)

[Elixir Recipes](#) ◦

<https://riptutorial.com/zh-CN/phoenix-framework/topic/5868/>

S. No		Contributors
1	phoenix-framework	Community , helcim , penguin , Steve Pallen , SURAJ KUMAR , Svilen
2	Ecto	Faizan Ali
3		toraritte