

The background of the slide is a grayscale, semi-transparent image of a workspace. It features an iMac on the left, a MacBook Pro on the right, a white keyboard in the center, and a tablet in the foreground. The tablet screen displays a Polish website with the heading 'Księgujemy za Ciebie bezproblemowo'.

mix tasks

Iván Jaimes

@iver14

ivan@iver.mx



\$ mix tasks

¿Qué es?

Mix.Task behaviour

Proporciona comodidades para crear, cargar y manipular tareas Mix.

Convenciones

```
defmodule Mix.Tasks.Ecto.Create do
  use Mix.Task
  import Mix.Ecto

  @shortdoc "Creates the repository storage"

  @switches [
    quiet: :boolean,
    repo: [:string, :keep],
    no_compile: :boolean,
    no_deps_check: :boolean
  ]

  @aliases [
    r: :repo,
    q: :quiet
  ]
end
```

- Se debe crear un módulo cuyo nombre empiece con **Mix.Tasks**.

Convenciones

```
@doc false
def run(args) do
  repos = parse_repo(args)
  {opts, _} = OptionParser.parse! args, strict: @s

  Enum.each repos, fn repo ->
    ensure_repo(repo, args)
    ensure_implements(repo.__adapter__, Ecto.Adapter
                                     "create st

  case repo.__adapter__.storage_up(repo.config)
  do
    :ok ->
      unless opts[:quiet] do
        Mix.shell().info "The database for #{ins
      end
    {:error, :already_up} ->
      unless opts[:quiet] do
        Mix.shell().info "The database for #{ins
      end
    {:error, term} when is_binary(term) ->
```

- Se debe crear un módulo cuyo nombre empiece con **Mix.Tasks**.
- Debe contener una función **run/1**

Convenciones

```
deps/ecto/lib/mix/  
├── ecto.ex  
└── tasks  
    ├── ecto.create.ex  
    ├── ecto.drop.ex  
    ├── ecto.ex  
    └── ecto.gen.repo.ex
```

- Se debe crear un módulo cuyo nombre empiece con **Mix.Tasks**.
- Debe contener una función **run/1**
- Los módulos de tareas se encuentran en el directorio **lib/mix/tasks** (por lo general)

Convenciones

```
deps/ecto/lib/mix/  
├── ecto.ex  
└── tasks  
    ├── ecto.create.ex  
    ├── ecto.drop.ex  
    ├── ecto.ex  
    └── ecto.gen.repo.ex
```

- Se debe crear un módulo cuyo nombre empiece con **Mix.Tasks**.
- Debe contener una función **run/1**
- Los módulos de tareas se encuentran en el directorio **lib/mix/tasks** (por lo general)
- Los nombres de archivo usan separadores de puntos en lugar de guiones bajos.

¿COMO FUNCIONA?

```
# lib/mix/tasks/echo.ex
defmodule Mix.Tasks.Echo do
  @moduledoc "Printed when the user requests `mix help echo`"
  @shortdoc "Echoes arguments"

  use Mix.Task

  @impl Mix.Task
  def run(args) do
    Mix.shell().info(Enum.join(args, " "))
  end
end
```

Mix Task behaviour

<https://hexdocs.pm/mix/Mix.Task.html>

¿COMO FUNCIONA?

USA UN COMPORTAMIENTO (BEHAVIOUR)

Los comportamientos tienen dos roles importantes:

- Definir un conjunto de funciones que deben ser implementadas
- Revisar que ese conjunto de funciones haya efectivamente sido implementado.

FUNCIÓN

- `run([binary]) :: any`

ATRIBUTOS

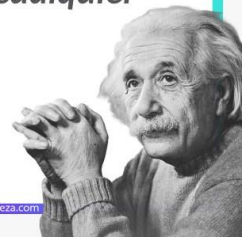
- **@shortdoc:** hace pública la tarea con una breve descripción que aparece
- **@recursive:** ejecuta la tarea de forma recursiva en proyectos umbrella

¿DONDE LO USO?

Donde más te sirva!

*"La lógica te llevará
desde A hasta B.
La imaginación te
llevará a cualquier
parte."*

—Albert
Einstein



mayneza.com

¿DONDE LO USO?

ALGUNOS EJEMPLOS SON:

- mix ecto.migrations
- mix phx.server
- mix hex





¿Preguntas?

Iván Jaimes
@iver14

Código
https://github.com/iver/meetups/tree/master/elixir/hello_tasks



Gracias

Iván Jaimes

@iver14

ivan@iver.mx