

Curso Elixir

Sección 15

Documentación en línea

El comentario en línea de Elixir está denotado con #

```
# Salida 'Hello, World'
IO.puts "Hello, " <> "World"
```

Documentación de Módulos

El anotador @moduledoc permite agregar documentación al nivel de módulo. Ejemplo:

```
defmodule Races.Race do
  @moduledoc """
  Módulo que contiene la funcionalidad de CRUD sobre la tabla reces
  """
```

Podemos acceder a la documentación por medio del comando **h**

```
$ iex -S mix
iex> alias Races.Race
iex> h Race

              Races.Race

Módulo que contiene la funcionalidad de CRUD sobre la tabla reces
```

Documentación de Funciones

El anotador @doc permite agregar documentación a nivel de función. Ejemplo:

```
@doc """
  get_race.

  Regresa un registro de la tabla reces

  ## Parameters

  - id: valor entero de la llave (raceId).
```

```
## Examples

iex> alias Races.Race
iex> race = Race.get_race(1000)
iex> race.name
"Hungarian Grand Prix"

"""
def get_race(id) do
  ...
```

Podemos acceder a la documentación por medio del comando `h`

```
$ iex -S mix
iex> alias Races.Race
iex> h Race.get_race

def get_race(id)

get_race.

Regresa un registro de la tabla races

## Parameters

  • id: valor entero de la llave (raceId).

## Examples

iex> alias Races.Race
iex> race = Race.get_race(1000)
iex> race.name
"Hungarian Grand Prix"
```

ExDoc

ExDoc es un proyecto oficial de Elixir que produce HTML y documentación en línea para proyectos Elixir (https://hexdocs.pm/ex_doc/0.22.1/readme.html).

Para poder usar este proyecto necesitamos instalar la siguiente dependencia en `mix.exs`:

```
{:ex_doc, "~> 0.21", only: :dev, runtime: false}
```

y corremos el comando `$ mix deps.get`

Después de instalar la librería corremos el comando `mix docs`

Generating docs...

View "html" docs at "doc/index.html"

View "epub" docs at "doc/app.epub"