



# **Programação Funcional com Elixir**



**Mostrando um Menu**

# Mostrando um Menu

- Comece criando uma Struct que servirá para montar o menu.

```
# friends_app/lib/friends_app/cli/menu.ex
defmodule FriendsApp.CLI.Menu do
  defstruct label: nil, id: nil
end
```

- Todo menu terá uma **label** (descrição do que será mostrado) e uma **id** que o identificará.

# Mostrando um Menu

- Agora vamos usar a Struct para informar quais os itens do menu.

```
#friends_app/lib/friends_app/cli/menu_items.ex
defmodule FriendsApp.CLI.MenuItems do
  alias FriendsApp.CLI.Menu

  def all, do: [
    %Menu{ label: "Cadastrar um amigo", id: :create },
    %Menu{ label: "Listar amigos", id: :read },
    %Menu{ label: "Atualizar um amigo", id: :update },
    %Menu{ label: "Excluir um amigo", id: :delete }
  ]
end
```

# Mostrando um Menu

- Perceba que não usamos o “**as:**” em conjunto com o alias.
- Isso ocorre visto que quando não informamos o “as:” o alias será o último nome do módulo em questão, ou seja...

```
alias FriendsApp.CLI.Menu, as: Menu
```

# é o mesmo que:

```
alias FriendsApp.CLI. Menu
```

# Mostrando um Menu

- Na sequência, vamos criar e mostrar de fato as possíveis escolhas do menu...

```
#friends_app/lib/friends_app/cli/menu_choice.ex
defmodule FriendsApp.CLI.MenuChoice do
  alias Mix.Shell.IO, as: Shell

  def start do
    Shell.cmd("clear")
    Shell.info("Escolha uma opção:")

    ...
  end
end
```

# Mostrando um Menu

```
...
  FriendsApp.CLI.MenuItems.all()
  |> Enum.map(&(&1.label))
  |> display_options()
end

defp display_options(options) do
  options
  |> Enum.with_index(1)
  |> Enum.each(fn {option, index} ->
    Shell.info("#{index} - #{option}")
  end)
end
end
end
```

# Mostrando um Menu

- Daí agora é só ajustar a nossa Main...

```
#friends_app/lib/friends_app/cli/main.ex
defmodule FriendsApp.CLI.Main do
  alias Mix.Shell.IO, as: Shell

  def start_app do
    Shell.cmd("clear")
    welcome_message()
    Shell.prompt("Pressione ENTER para continuar...")
    starts_menu_choice()
  end

  ...
end
```



# Mostrando um Menu

...

```
defp welcome_message do
  Shell.info("===== Friends App =====")
  Shell.info("Seja bem-vindo à sua agenda pessoal!")
  Shell.info("=====")
end
```

```
defp starts_menu_choice do
  FriendsApp.CLI.MenuChoice.start()
end
end
```

# Mostrando um Menu

- Agora é só fazer um novo teste...

```
mix start
===== Friends App =====
Seja bem-vindo à sua agenda pessoal!
=====
Pressione ENTER para continuar...
```

```
Escolha uma opção:
1 - Cadastrar um amigo
2 - Pesquisar um amigo
3 - Atualizar um amigo
4 - Excluir um amigo
```