



Programação Funcional com Elixir



Refatorando algumas funções

Refatorando algumas funções

- Vamos alterar a função **start** do arquivo

`friends_app/lib/friends_app/cli/menu_choice.ex`
deixando-a assim:

```
alias FriendsApp.CLI.MenuItems
```

```
def start do
```

```
  Shell.cmd("clear")
```

```
  Shell.info("Escolha uma opção:")
```

```
  menu_items = MenuItem.all()
```

```
  find_menu_item_by_index = &Enum.at(menu_items, &1, :error)
```

```
  ...
```

Refatorando algumas funções

- Altere a função **parse_answer**

```
defp parse_answer(answer) do
  case Integer.parse(answer) do
    :error -> invalid_option()
    {option, _} -> option - 1
  end
end
```

Refatorando algumas funções

- Crie a função **invalid_option**

```
def invalid_option do
  Shell.cmd("clear")
  Shell.error("Opção inválida!")
  Shell.prompt("Pressione ENTER para tentar novamente.")
  start()
end
```

Refatorando algumas funções

- Ajuste a função **confirm_menu_item**

```
defp confirm_menu_item(chosen_menu_item) do
  case chosen_menu_item do
    :error -> invalid_option()
    _ -> confirm_message(chosen_menu_item)
  end
end
```

Refatorando algumas funções

- Crie a função **confirm_message**

```
defp confirm_message(chosen_menu_item) do
  Shell.cmd("clear")
  Shell.info("Você escolheu... [#{chosen_menu_item.label}]")

  case Shell.yes?("Confirma?") do
    true -> Shell.info("... #{chosen_menu_item.label} ...")
    false -> start()
  end
end
```

Refatorando algumas funções

- Faça um novo teste....

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

```
1 - Cadastrar um amigo
2 - Pesquisar um amigo
3 - Atualizar um amigo
4 - Excluir um amigo
```

```
Qual das opções acima você escolhe? [1,2,3,4]
```

xyz

Pressione ENTER para tentar novamente.
Opção inválida!