



# **Programação Funcional com Elixir**

The background features a series of overlapping, semi-transparent gray shapes that resemble stylized leaves or petals, creating a layered, organic effect. These shapes are primarily concentrated on the left side of the frame, with some extending towards the center.

# **Conhecendo o Mix**

# Conhecendo o Mix

- Mix é uma CLI (command-line interface) que fornece “tasks” para você criar, compilar e testar projetos Elixir, além de gerenciar suas dependências.

`https://hexdocs.pm/mix/Mix.html`

# Conhecendo o Mix

- Vamos criar um CRUD para uma aplicação pequena de agenda/contatos.
- Vamos usar o Mix para criar a estrutura padrão

```
mix new friends_app
```

# Conhecendo o Mix

- Após rodar o comando anterior, teremos as seguintes pastas e arquivos...

```
config/  
lib/  
test/  
.formatter.exs  
.gitignore  
mix.exs  
README.md
```

# Conhecendo o Mix

- O arquivo **README.md** serve para descrever o projeto...
- O arquivo **mix.exs** gerencia todas as dependências do projeto.

```
config/  
lib/  
test/  
.formatter.exs  
.gitignore  
mix.exs  
README.md
```

# Conhecendo o Mix

- O **.gitignore** é do versionador de código, git.
- O arquivo **.formatter.exs** é onde fica a configuração para a formatação do estilo de forma automática pelo próprio Mix.

`https://elixir-lang.org/getting-started/mix-otp/introduction-to-mix.html#automatic-code-formatting`

# Conhecendo o Mix

- A pasta **test** é onde ficarão os arquivos de teste.
- A pasta **lib** é onde ficarão todos os arquivos que vamos criar para executar o projeto.

```
config/  
lib/  
test/  
.formatter.exs  
.gitignore  
mix.exs  
README.md
```



# Conhecendo o Mix

- A pasta **config** é onde ficam os arquivos de configuração do projeto.

```
config/  
lib/  
test/  
.formatter.exs  
.gitignore  
mix.exs  
README.md
```