

Configurando a IDE

Transcrição

O Flutter é dependente tanto do **Android Studio** quanto do **Apple Xcode**, então é necessário que baixemos essas duas ferramentas para podermos utilizar o Flutter dentro do nosso computador. Se você não tem algumas destas ferramentas instaladas, baixe-as e instale-as. O Android Studio fica dentro do site do **Android Studio** e Xcode dentro da **Apple Store**.

Após instalá-los, dentro do nosso terminal, onde extraímos o SDK e fizemos a configuração da ferramenta, vamos colocar o comando `flutter.doctor` e damos `Enter`. Essa é uma ferramenta do Flutter que faz uma comparação entre as ferramentas que ele precisa e as ferramentas que já possuímos. Ele colocará algumas informações sobre a análise feita.

Há três tipos de informação que podemos ter. A primeira será o símbolo `]` em **verde** e significa que está tudo certo com essa configuração. Em segundo, temos o `!` em **amarelo**, que significa que você não tem o correspondente instalado, mas isso não será um impeditivo. O último caso é um `x` **vermelho** que significa que você não tem aquela configuração e sem ela não podemos trabalhar. No caso de `x`, ele não fala apenas o que não está faltando, mas também dá um passo a passo de como corrigir isso.

Por exemplo, no meu caso, ele me dá uma aviso de que não tenho algumas ferramentas como o `libimobiledevice` e o `ideviceinstaller`

```
iOS tools - develop for iOS devices
x libimobiledevice and ideviceinstaller are not installed. To install with Brew, run:
  brew update
  brew install --HEAD usbmuxd
  brew link usbmuxd
  brew install --HEAD libimobiledevice
  brew install ideviceinstaller
```

Então ele diz que se rodar esses comando do Brew - que faz a gestão de pacotes dentro do mac - conseguimos resolver esse problema. Ele também aponta que o Android Studio não tem o plugin do Flutter e do Dart, o mesmo acontece com o IntelliJ e o VS Code.

```
] Android Studio (version 3.4)
  x Flutter plugin not installed; this adds Flutter specific functionality.
  Dart plugin not installed this adds Dart specific functionality.
] IntelliJ IDEA Community Edition (version 2019.2)
  x Flutter plugin not installed; this adds Flutter specific functionality.
  Dart plugin not installed this adds Dart specific functionality.
] VS Code (version 1.34.0)
  x Flutter extension not installed; install from
  https://marketplace.visualstudio.com/items?itemName=Dart-Code.
```

Nenhum desses são graves e nos impede de desenvolver, mas vamos corrigir o erro do Android Studio, que vamos utilizar neste curso. É possível também desenvolver dentro do **IntelliJ** e do **VS Code**, mas é uma outra forma de configurar o ambiente.

Ele está apontando que não existe o plugin do Flutter e do Dart, então vamos instalar ambos. Para isso, vamos entrar no Android Studio. Logo na primeira página, na parte inferior, encontramos a opção `Configure`, clicamos nela, em seguida em `Plugins`. Ele abrirá o **Marketplace** e os plugins que já estão instalados.

Vamos escrever "flutter" na barra de buscas e ele mostrará alguns plugins. Escolheremos a primeira opção, `Flutter`, que é oficial, a seguir selecionamos `Install` que estará no canto superior direito em verde. Ao fazer isso, o programa abrirá uma janela avisando que esse plugin tem como dependência o plugin do Dart. Isso porque o Flutter é uma *framework* da linguagem Dart. Então, nessa janela selecionamos a opção `Yes`.

Em seguida, onde estava `Install` aparecerá `Restart IDE` devemos selecionar essa opção, em seguida `Restart` na janela que se abrirá. Assim, o Android Studio reiniciará já com essas novas ferramentas que instalamos.

Antes de instalarmos os plugins aparecia apenas a opção `+ Start a new Android Studio project` para começar um novo projeto do Android Studio, agora temos também a opção `+ Start a new Flutter project` para começarmos um novo projeto Flutter. Ou seja, agora podemos começar nosso projeto Flutter dentro dessa IDE.