



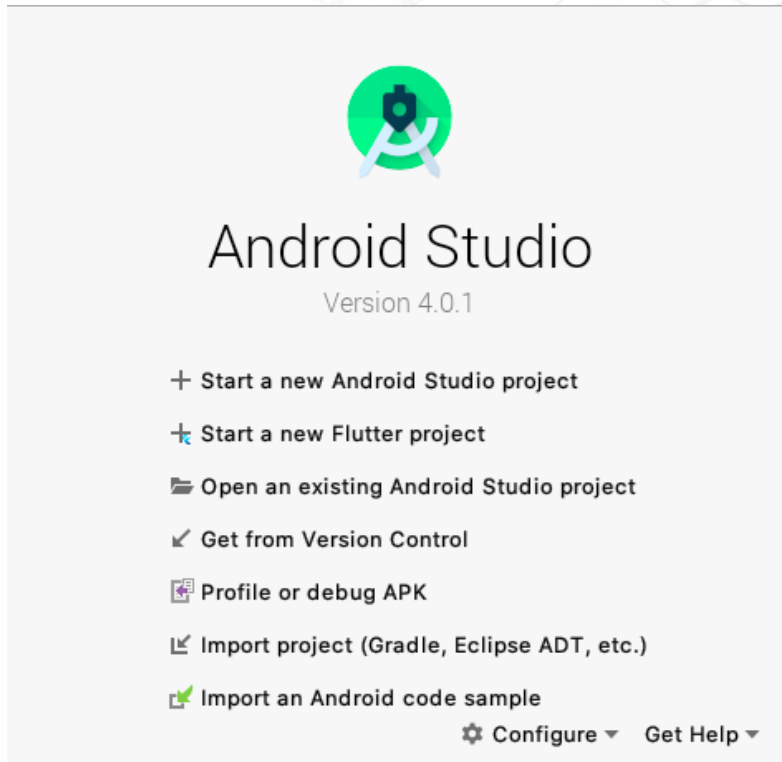
Criando Primeiro Projeto e Interface Android Studio



Prof^a. Ivre Marjorie
(ivre@pucminas.br)

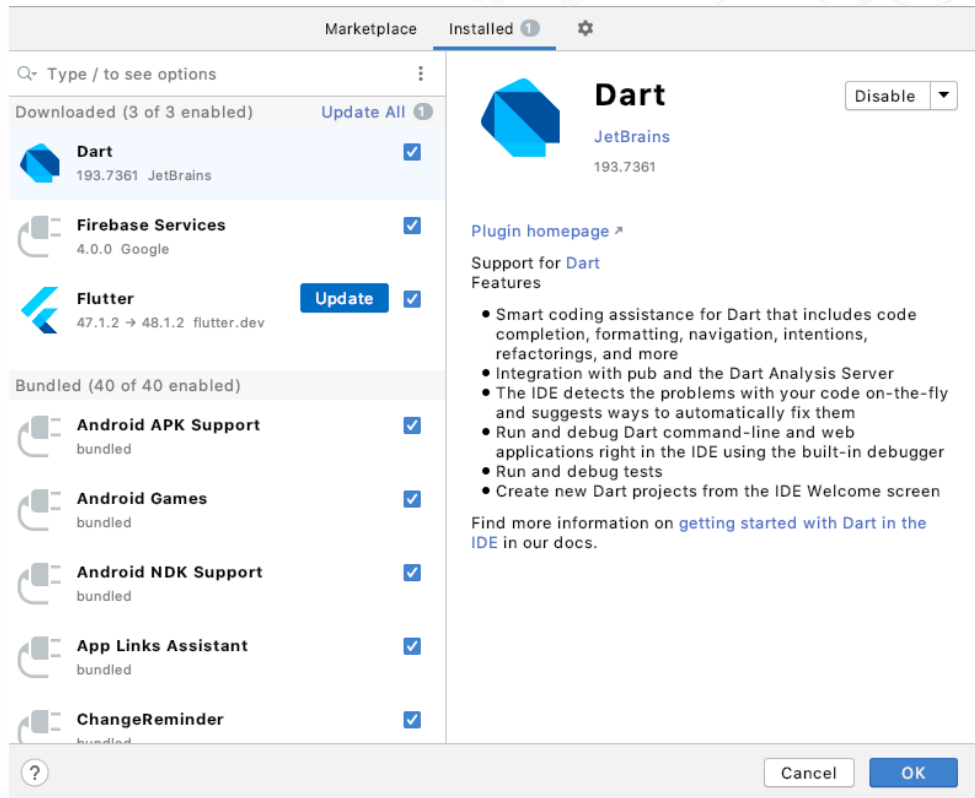
Criando o primeiro projeto

- Ao abrir o Android Studio, vamos escolher a opção Start a new Flutter project
- Caso essa opção não apareça, provavelmente, ainda tenha que instalar o plugging do Flutter, para isso, clique em configure -> plugings



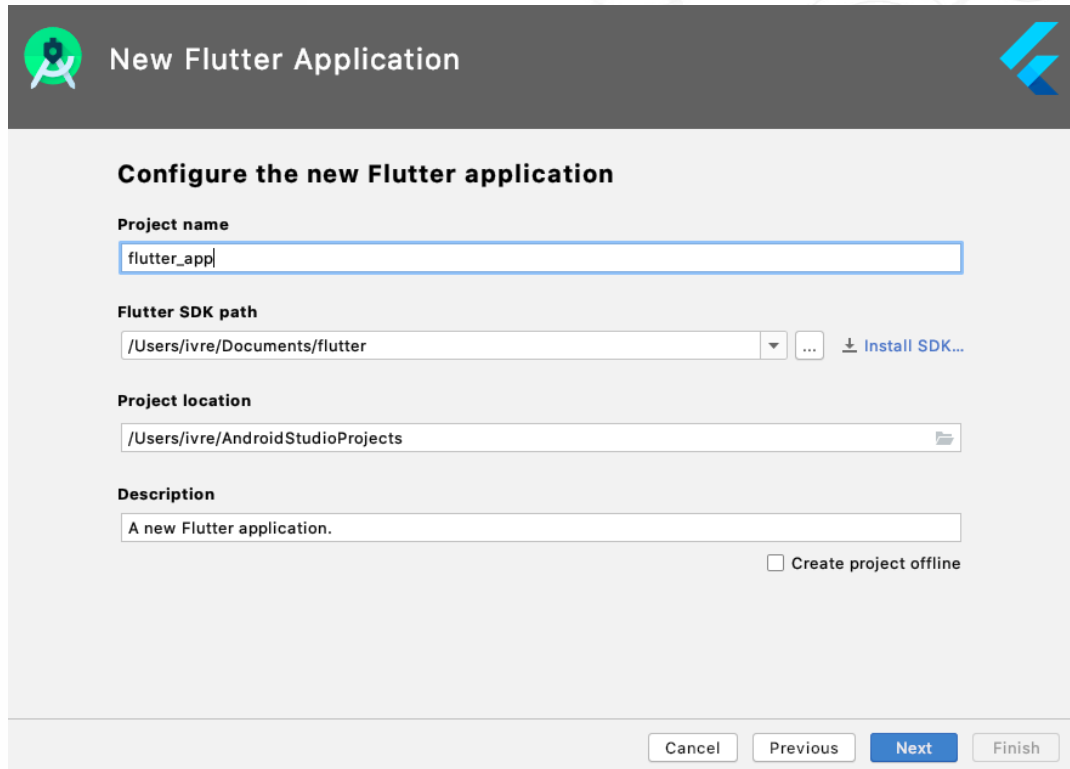
Criando o primeiro projeto

- Ao abrir uma tela como essa, realize a pesquisa por Flutter e Dart e realize a instalação, abra novamente o Android Studio e crie o projeto



Criando o primeiro projeto

- Escolha Flutter Application e clique em next
- E na próxima tela (ao lado), iremos dar um nome para nosso projeto e escolher o local para gravar o projeto



The screenshot shows the 'New Flutter Application' dialog box in Android Studio. The title bar is dark gray with the Flutter logo on the left and the Android Studio logo on the right. The main content area is light gray and titled 'Configure the new Flutter application'. It contains four sections: 'Project name' with a text field containing 'flutter_app'; 'Flutter SDK path' with a text field containing '/Users/ivre/Documents/flutter', a dropdown arrow, an ellipsis button, and a link 'Install SDK...'; 'Project location' with a text field containing '/Users/ivre/AndroidStudioProjects' and a folder icon button; and 'Description' with a text field containing 'A new Flutter application.' and a checkbox 'Create project offline' which is currently unchecked. At the bottom right, there are four buttons: 'Cancel', 'Previous', 'Next' (highlighted in blue), and 'Finish'.

Criando o primeiro projeto

- Para o nosso primeiro exemplo, vamos criar um App para calcular o IMC, nesse caso, vamos chama-lo de `imc_app`, é possível também incluir uma descrição para o APP

Configure the new Flutter application

Project name
imc_app

Flutter SDK path
/Users/ivre/Documents/flutter [Install SDK...](#)

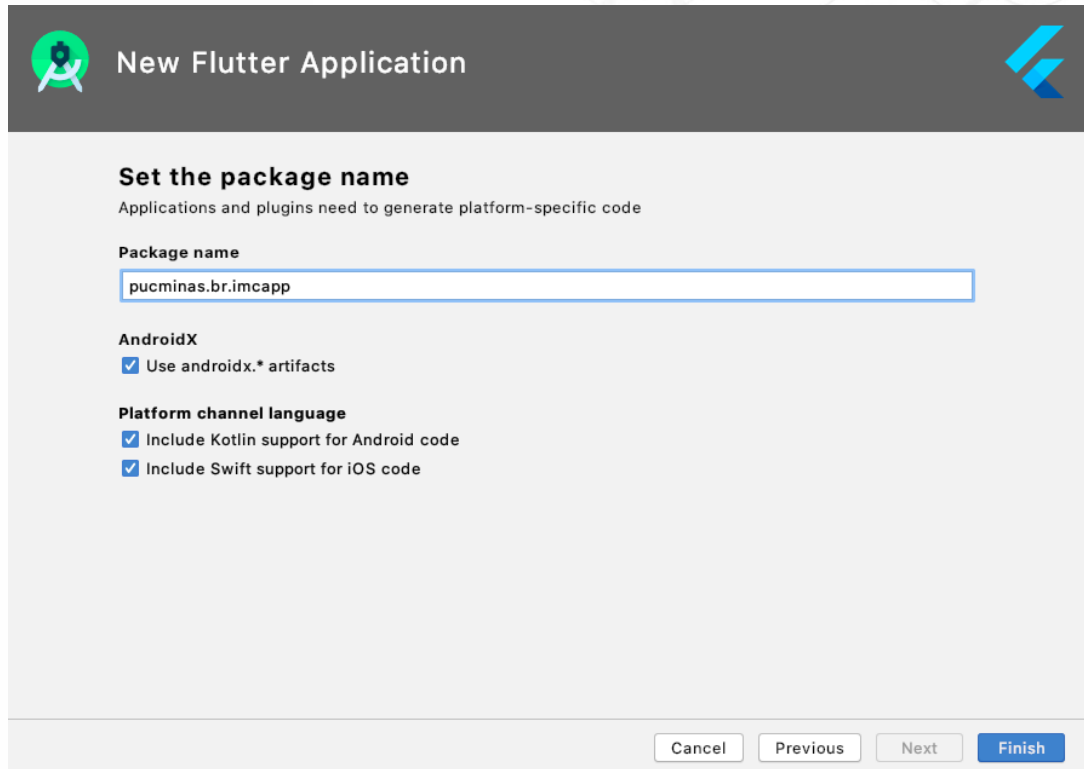
Project location
/Users/ivre/AndroidStudioProjects

Description
A new Flutter application.
☐ Create project offline

Indicar a pasta para o SDK do flutter

Criando o primeiro projeto

- Na próxima tela indicamos o nome do pacote (Package name)
- Ele define um **nome único** para o seu APP, quando for fazer a publicação na loja esse nome será usado



New Flutter Application

Set the package name
Applications and plugins need to generate platform-specific code

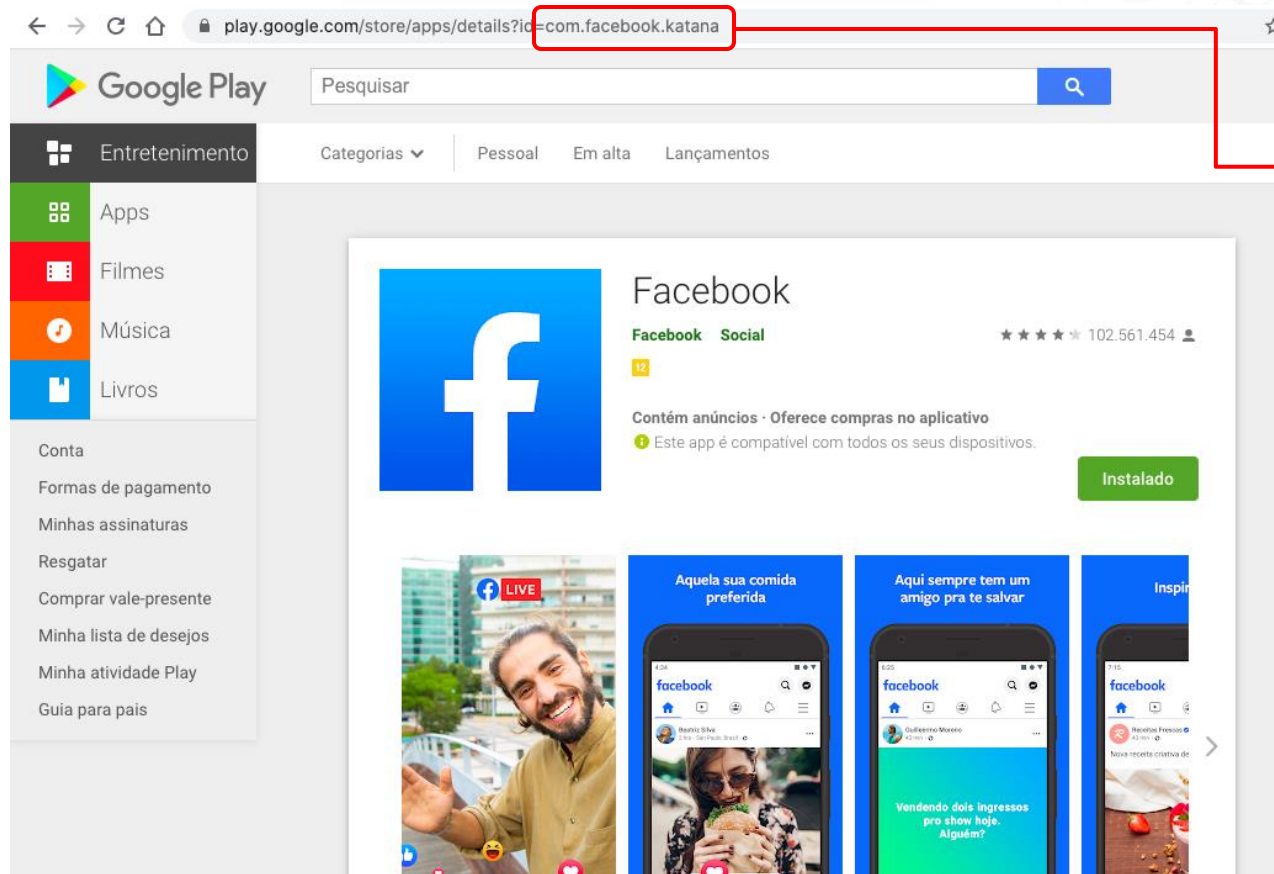
Package name
pucminas.br.imcapp

AndroidX
☒ Use androidx.* artifacts

Platform channel language
☒ Include Kotlin support for Android code
☒ Include Swift support for iOS code

Cancel Previous Next Finish

Package name - Exemplo



Criando o primeiro projeto

- Além disso, nessa tela você pode incluir o suporte ao Kotlin e ao Swift, que nós não vamos utilizar por enquanto, sendo assim, pode desmarcar essas opções e clicar em Finish

New Flutter Application

Set the package name
Applications and plugins need to generate platform-specific code

Package name
pucminas.br.imcapp

AndroidX
☒ Use androidx.* artifacts

Platform channel language
☒ Include Kotlin support for Android code
☒ Include Swift support for iOS code

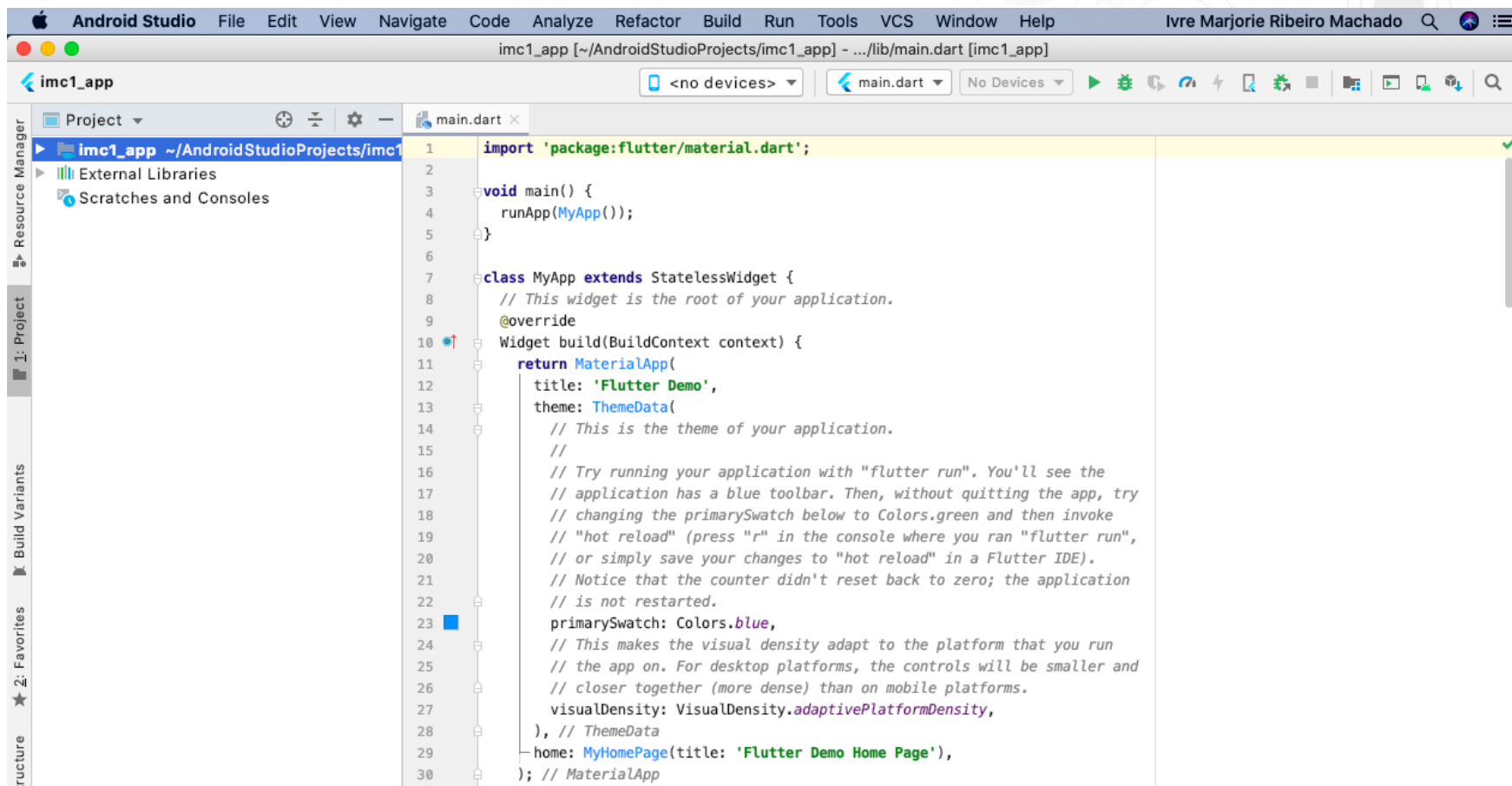
Cancel Previous Next Finish

Criando o primeiro projeto

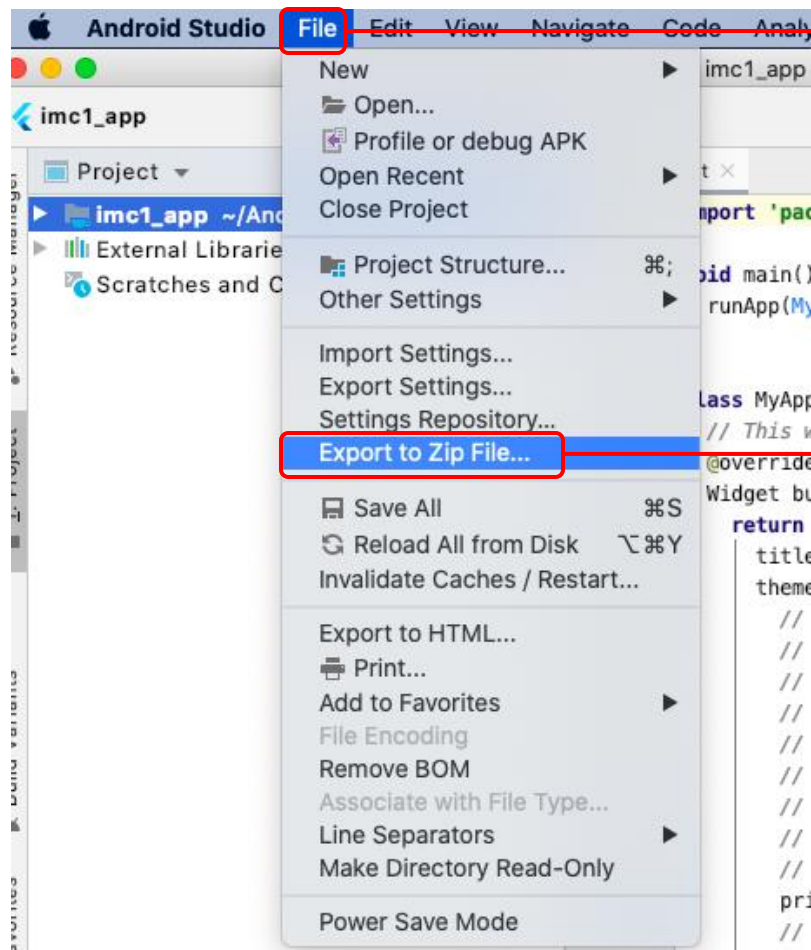
- Ele vai abrir um código de exemplo na main.dart
- Vamos executá-lo e depois alteramos o código para o exemplo de Cálculo de IMC

```
main.dart x
1  import 'package:flutter/material.dart';
2
3  void main() {
4    runApp(MyApp());
5  }
6
7  class MyApp extends StatelessWidget {
8    // This widget is the root of your application.
9    @override
10   Widget build(BuildContext context) {
11     return MaterialApp(
12       title: 'Flutter Demo',
13       theme: ThemeData(
14         // This is the theme of your application.
15         //
16         // Try running your application with "flutter run". You'll see the
17         // application has a blue toolbar. Then, without quitting the app, try
18         // changing the primarySwatch below to Colors.green and then invoke
19         // "hot reload" (press "r" in the console where you ran "flutter run",
20         // or simply save your changes to "hot reload" in a Flutter IDE).
21         // Notice that the counter didn't reset back to zero; the application
22         // is not restarted.
23         primarySwatch: Colors.blue,
24         // This makes the visual density adapt to the platform that you run
25         // the app on. For desktop platforms, the controls will be smaller and
26         // closer together (more dense) than on mobile platforms.
27         visualDensity: VisualDensity.adaptivePlatformDensity,
28       ), // ThemeData
29       home: MyHomePage(title: 'Flutter Demo Home Page'),
30     ); // MaterialApp
```

Tela do projeto no Android Studio

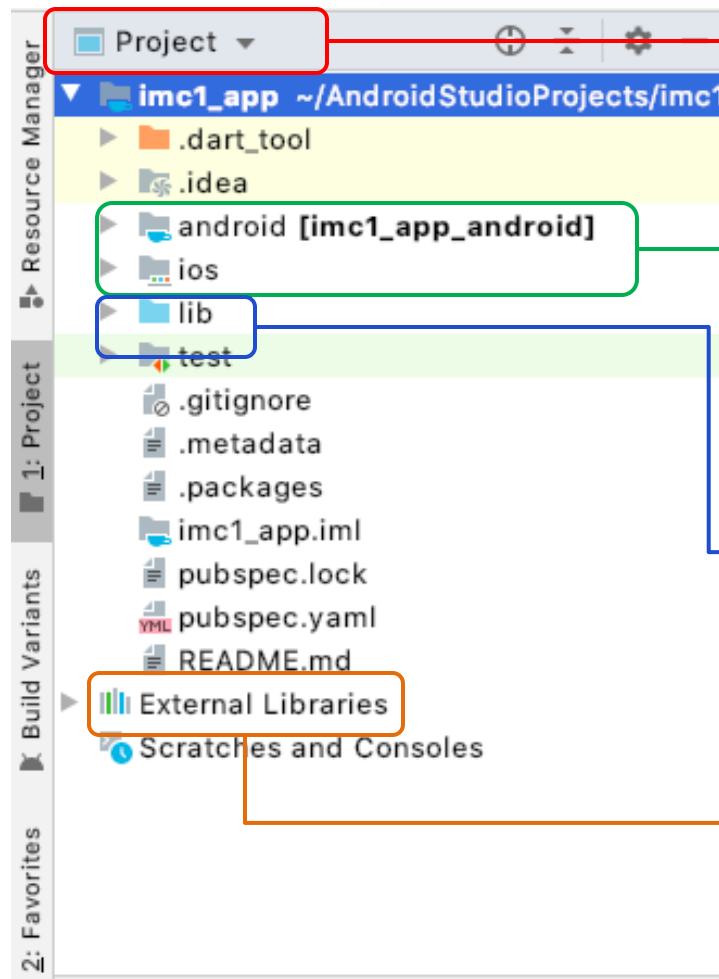


Criando um Zip do Projeto - Android Studio



File > Export to Zip File, vai gerar um arquivo zip do projeto com o que realmente é necessário

Interface do Android Studio



Observe se está com a visão de Projeto (Project)

Estrutura gerada pelo SDK do Flutter para o Android e para o IOS

Pasta onde fica o **main.dart**, arquivo inicial que é gerado automaticamente, mas podemos criar vários arquivos

Bibliotecas externas, consegue definir bibliotecas para usar no projeto

Configurando o Emulador

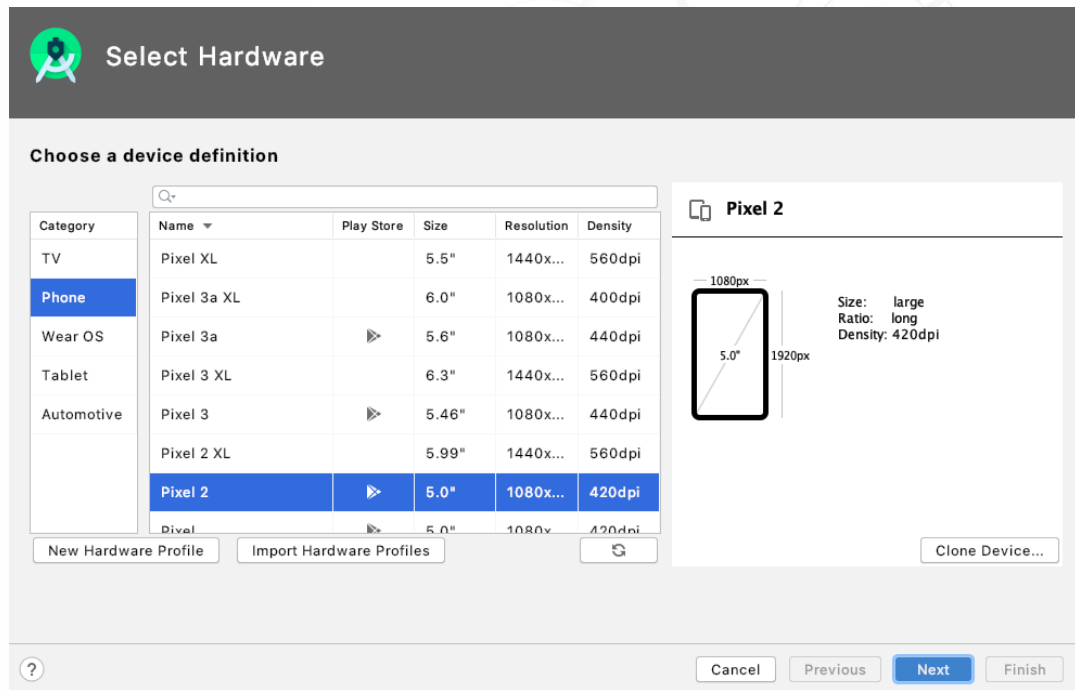
- Vamos configurar um emulador para poder executar o exemplo criado
- Escolha a opção AVD Manager, em seguida, clique em **Create Virtual Device**



AVD
Manager

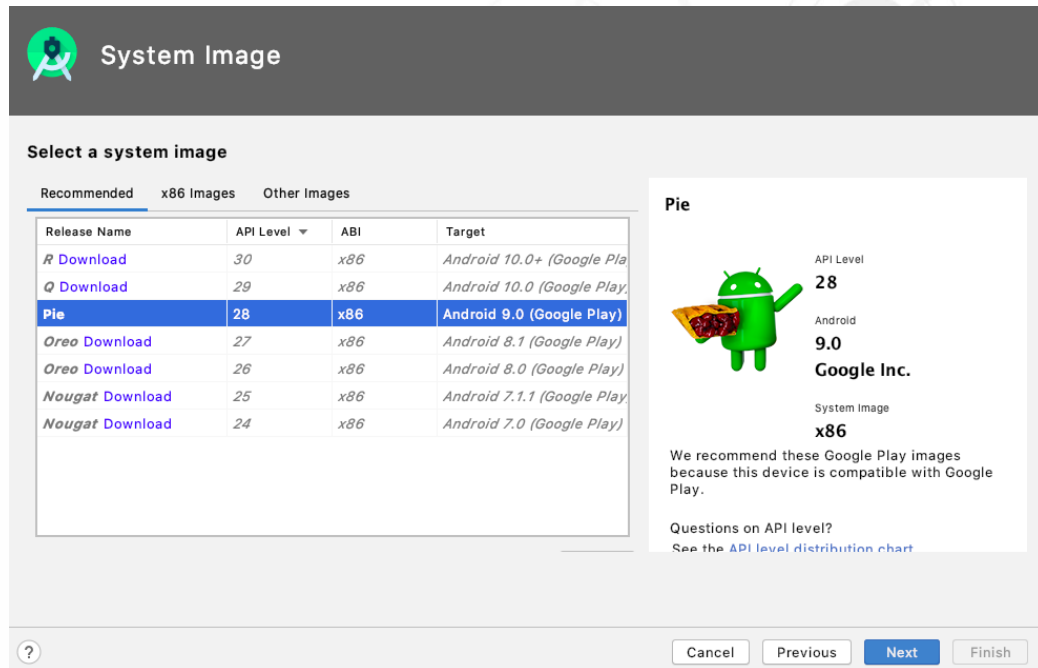
Configurando o Emulador

- Escolha a opção Phone, e vamos escolher uma opção de versão que irá simular um dispositivo para nossos projetos
- Vou usar para meu exemplo o **Nexus 5**, observe que ele tem a Play store



Configurando o Emulador

- Nessa tela vamos instalar uma versão do Android para instalar dentro do simulador
- Em seguida, clicar em Next e aguardar o download



System Image

Select a system image

Recommended x86 Images Other Images

Release Name	API Level	ABI	Target
R Download	30	x86	Android 10.0+ (Google Play)
Q Download	29	x86	Android 10.0 (Google Play)
Pie	28	x86	Android 9.0 (Google Play)
Oreo Download	27	x86	Android 8.1 (Google Play)
Oreo Download	26	x86	Android 8.0 (Google Play)
Nougat Download	25	x86	Android 7.1.1 (Google Play)
Nougat Download	24	x86	Android 7.0 (Google Play)

Pie

API Level
28

Android
9.0

Google Inc.

System image
x86

We recommend these Google Play images because this device is compatible with Google Play.

Questions on API level?
[See the API level distribution chart](#)

Cancel Previous **Next** Finish

Configurando o Emulador

- Nessa tela é possível realizar algumas configurações do emulador
- Pode trocar o nome, a orientação inicial, e pode habilitar ou não os frames e configurações avançadas



The screenshot shows the 'Android Virtual Device (AVD)' configuration window. The title bar includes the Android logo and the text 'Android Virtual Device (AVD)'. Below the title bar, the 'Verify Configuration' section contains several settings:

- AVD Name:** A text field containing 'Nexus 5 API 28 2'.
- Device:** A row with 'Nexus 5' and '4.95 1080x1920 xxhdpi', followed by a 'Change...' button.
- System Image:** A row with the Android robot icon, 'Pie', and 'Android 9.0 x86', followed by a 'Change...' button.
- Startup orientation:** Two buttons, 'Portrait' (selected) and 'Landscape'.
- Emulated Performance:** A section with a 'Graphics' dropdown menu set to 'Automatic'.
- Show Advanced Settings:** A button located below the performance settings.

On the right side of the window, there is a section titled 'AVD Name' with the text 'The name of this AVD.' below it.

At the bottom of the window, there is a navigation bar with four buttons: '?', 'Cancel', 'Previous', and 'Finish'.

Não é necessário realizar nenhuma alteração, apenas clicar em Finish

Emulador criado

Android Virtual Device Manager

Your Virtual Devices
Android Studio

Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Nexus 5 API 28		1080 x 1920: xx...	28	Android 9.0 (Go...	x86	513 MB	

Consequimos visualizar o tamanho ocupado pelo emulador, no entanto, esse espaço pode ir aumentando e chegar até gigabytes

Aqui conseguimos inicializar o emulador, como editar e deletar

? + Create Virtual Device...

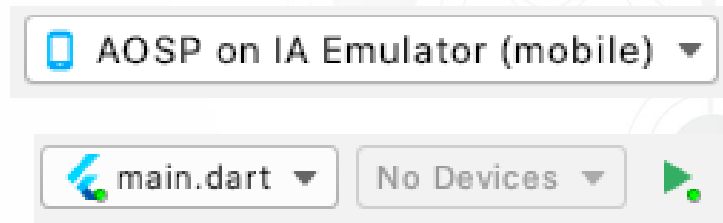
Configurando o Emulador

- Clique no botão verde para inicializar o emulador e aguarde um pouco enquanto o emulador é aberto
- Ele abrirá uma tela semelhante a um celular, e é nele que veremos o App quando executarmos usando o emulador



Executando o Projeto criado

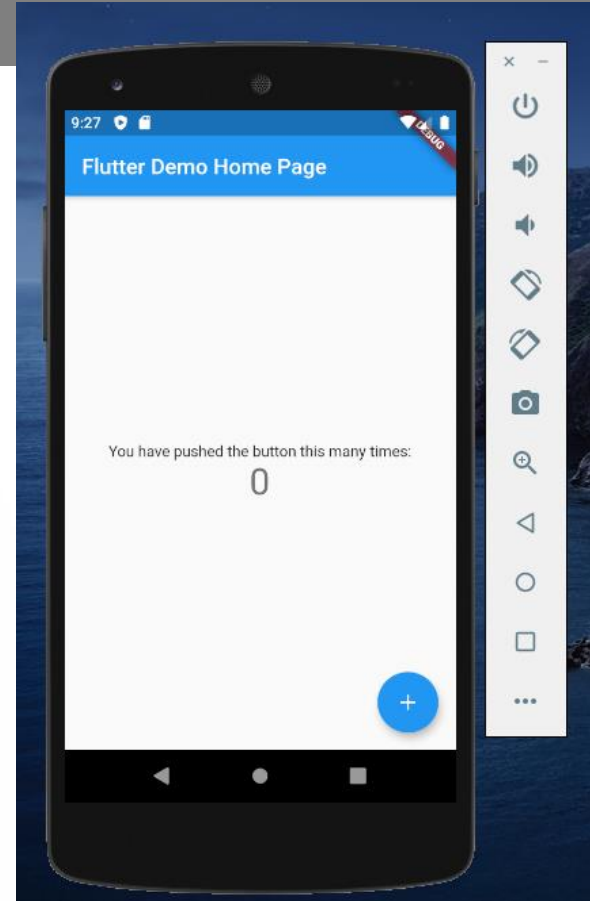
- Agora que o emulador já está em execução, vamos rodar o programa de exemplo, clicando no botão **Run**
- E aguarde, pois a primeira execução é mais demorada



Run main.dart

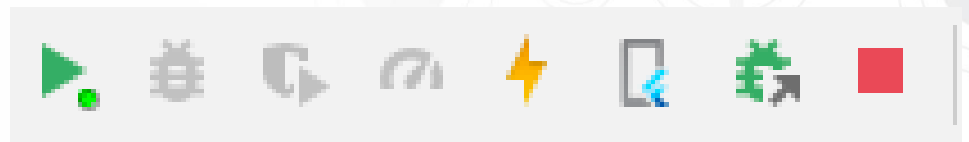
Executando o Projeto criado

- O emulador vai apresentar o APP de exemplo a seguir
- Nesse APP é possível clicar no botão de + e ele vai alterando o valor apresentado na tela
- Além disso, observe que esse APP vem com "Debug" na tela, depois veremos como criar uma versão que será usada para instalar o APP



Executando o Projeto criado

- Caso realize alterações no código e queira ver a alteração de forma mais rápida, usaremos o botão Flutter Hot Reload



Flutter Hot Reload

Executando o Projeto criado

- Para testar uma alteração e execução rápida, vamos trocar o título do código:

home: MyHomePage(title: '**Meu Primeiro APP**'),

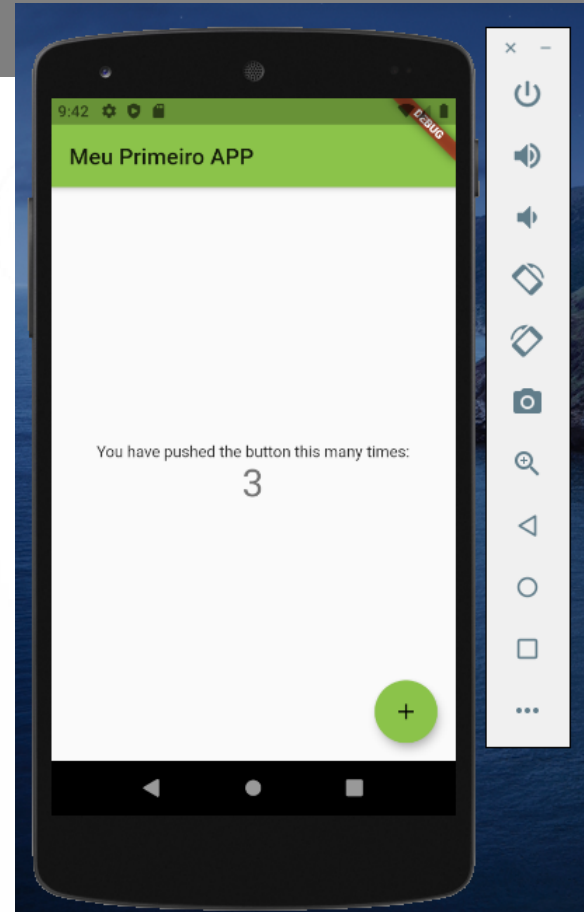
- Vamos trocar também a cor no código:

primarySwatch: Colors.*lightGreen*

- Em seguida, clique no botão Flutter Hot Reload

Executando o Projeto criado

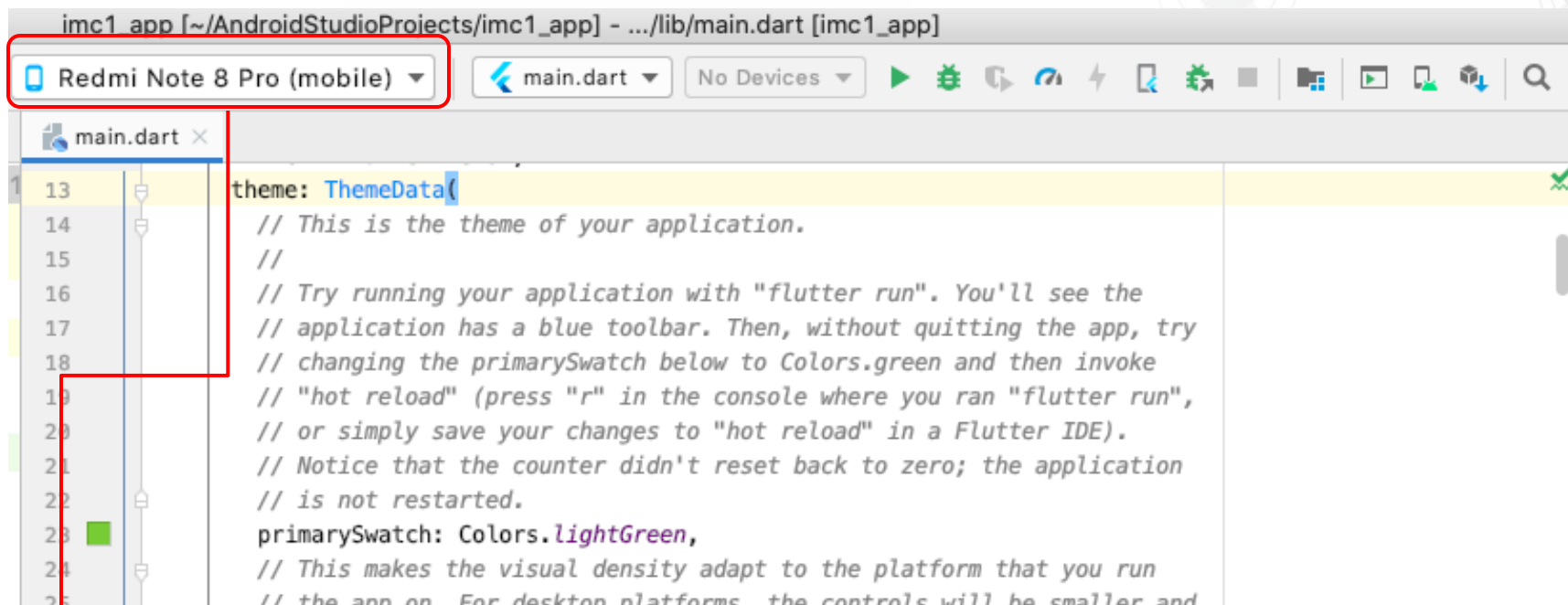
- Observe as alterações realizadas e apresentadas no emulador
- Título da página "Meu Primeiro APP" e cor "Verde claro"



Executando o Projeto criado no celular

- É possível usar um dispositivo real Android para executar os projetos feitos no Android Studio
- Para isso, é necessário realizar dois passos:
 1. Realizar uma configuração no celular para liberar para desenvolvedor (pesquise no Google como fazer no seu celular)
 2. Conectar o celular no computador com um cabo USB

Executando o Projeto criado no celular

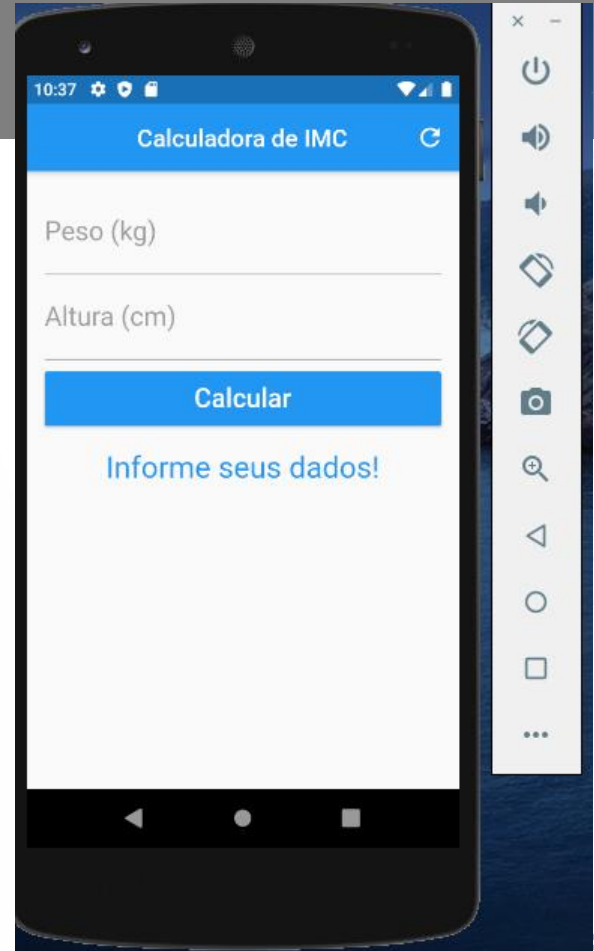


Observe que agora aparece o modelo do celular no lugar do nome do simulador, e ao clicar em Run, o App será executado no celular

nas

Exemplo - Cálculo do IMC

- Execute agora o exemplo do Cálculo do IMC e observe o código
- Ainda teremos aulas a respeito de cada um dos componentes usados
- O APP permite que o usuário digite o valor do peso e da altura e ao clicar no botão apresenta o valor do IMC



Exemplo - Cálculo do IMC

- Vamos acrescentar no APP a classificação da tabela ao lado
- E dessa forma, ao clicar no botão, além de mostrar o valor do IMC, mostrará um texto com a classificação

IMC	Classificação do IMC
< 16	Magreza grave
16 a < 17	Magreza moderada
17 a < 18,5	Magreza leve
18,5 a < 25	Saudável
25 a < 30	Sobrepeso
30 a < 35	Obesidade Grau I
35 a < 40	Obesidade Grau II (severa)
> 40	Obesidade Grau III (mórbida)

Prática 01 - Conversor de Temperatura

- Nesta atividade crie um app para realizar a conversão de temperaturas, o usuário deverá informar a temperatura em Celsius e o aplicativo realiza a conversão para temperatura em Kelvin, Fahrenheit, Reaumur ou Rankine.
- O aplicativo deverá mostrar a conversão na tela de acordo com a **escolha do usuário**.

Prática 01 - Conversor de Temperatura

- O que deve ser entregue: o projeto .ZIP gerado através do Android Studio e um pequeno vídeo apresentando o funcionamento do APP
- O ideal é que o vídeo esteja no YouTube ou OneDrive e seja informado o link para o vídeo em um arquivo .txt
- Entrega: através do Canvas