

Aim -

Installation and Configuration of Flutter Environment.

Theory -

What is Flutter?

Flutter is Google's open-source technology for creating mobile, desktop, and web apps with a single codebase. Unlike other popular solutions, Flutter is not a framework or library; it's a complete SDK – software development kit. Let's briefly clarify it here to make sure we're on the same page.

A library is a reusable piece of code you put in your application to perform a particular function.

A framework is a structure that provides you with a skeleton architecture for building software. It's a set of tools that serves as a foundation for your app, requiring you to fill in the blanks with your code to complete the entire structure and get the desired functionality.

An SDK has a much wider scope as it's a collection of tools, including libraries, documentation, APIs, sometimes frameworks, and more, giving you all you need for software development. And that's the case with Flutter — it already contains everything necessary to build cross-platform applications.

Other technologies such as Xamarin, React Native, Ionic, or NativeScript are also used to develop apps that work across multiple platforms. We have a detailed comparison of these cross-platform tools in a dedicated article. So what does Flutter have to offer?

Flutter SDK

Since Flutter is a full-fledged SDK, it includes a rendering engine, ready-made widgets, testing and integration APIs, etc. Let's talk about the main components and the way it works in general. What is Flutter used for?

Many large businesses have chosen Flutter to design branded mobile apps capable of providing a great customer experience. Among famous clients are Toyota, BMW, eBay, Alibaba Group, Groupon, and Etsy, to name just a few.

The SDK also perfectly fits into the startup ecosystem since it's open-source, feature-rich, and cost-effective. Some startups that have been scaling with Flutter are

Nubank, a Brazilian neobank and the largest digital bank in Latin America, as of today;

Invoice Ninja, an invoice and payment platform for small businesses; and Reflect, an AI-based mental health app.

But besides cross-platform mobile apps, which is Flutter's main target, the SDK has found use in many other areas. The fast development cycle and portability make the technology ideal for building minimum viable products (MVPs) and prototyping: You can quickly test your business idea on different platforms.

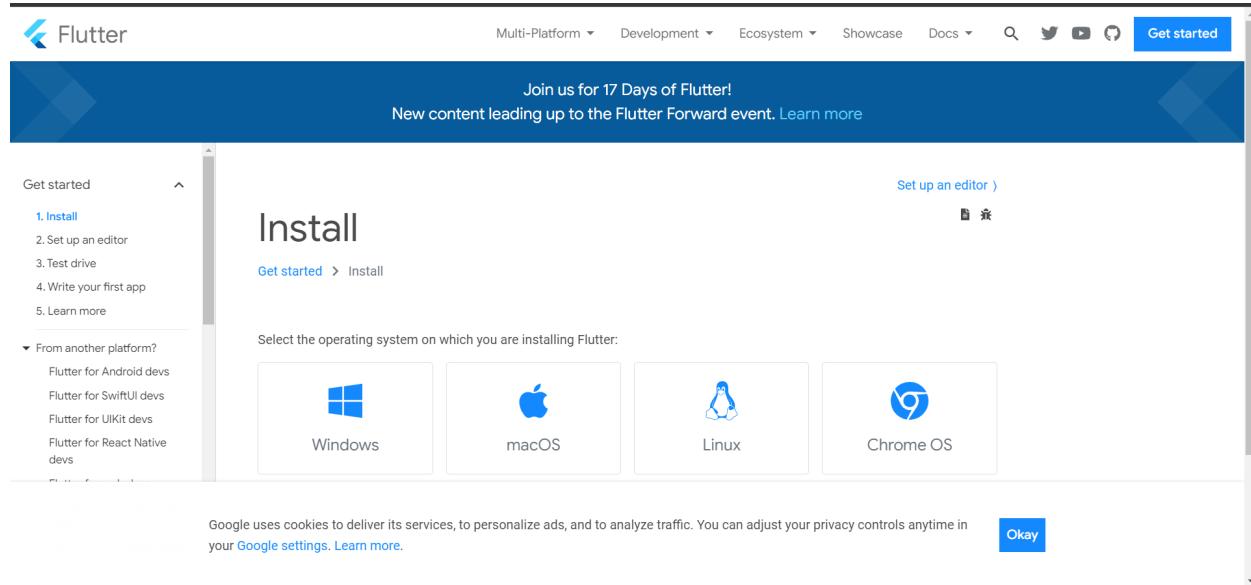
As for web development, Flutter can be a go-to solution for progressive web apps (PWAs) and single-page applications (SPAs). It also enables you to scale your existing mobile project to the web and desktop.

Now that we explained the basics of Flutter technology, let's identify its strong and weak sides.

Output -

Install the Flutter SDK

Step 1: Download the installation bundle of the Flutter Software Development Kit for windows. To download Flutter SDK, Go to its official website <https://docs.flutter.dev/get-started/install>, you will get the following screen.

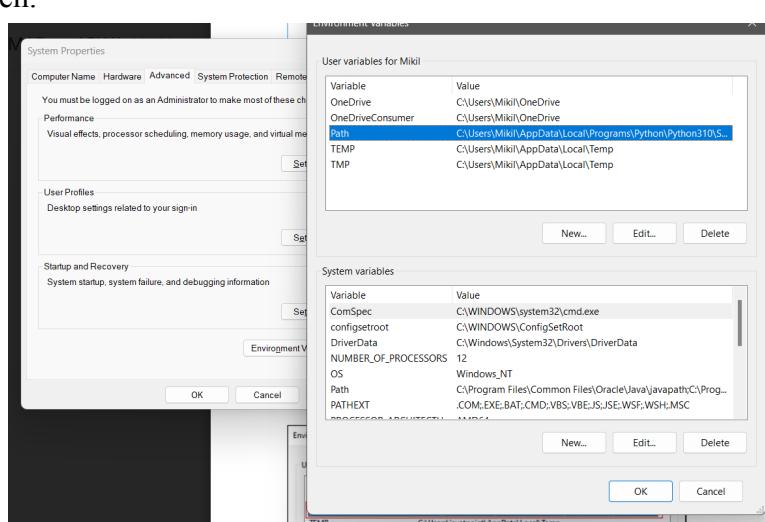


Step 2: Next, to download the latest Flutter SDK, click on the Windows icon. Here, you will find the download link for SDK.

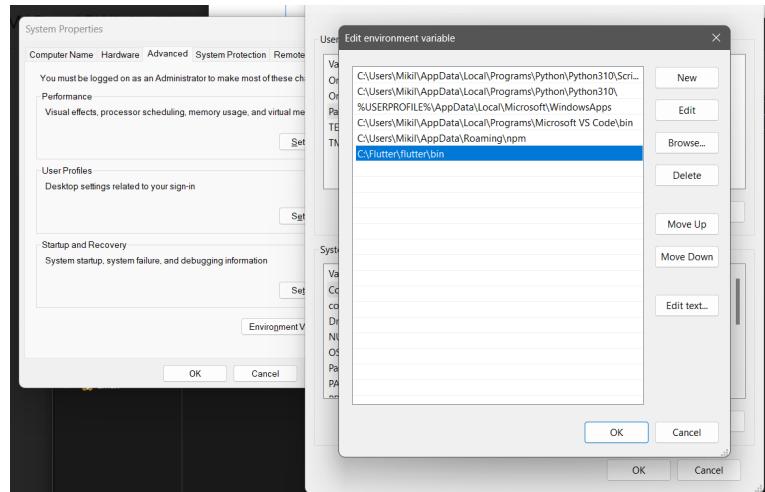
Step 3: When your download is complete, extract the zip file and place it in the desired installation folder or location, for example, C: /Flutter.

Step 4: To run the Flutter command in the regular windows console, you need to update the system path to include the flutter bin directory. The following steps are required to do this:

Step 4.1: Go to MyComputer properties -> advanced tab -> environment variables. You will get the following screen.



Step 4.2: Now, select path -> click on edit. The following screen appears



Step 4.3: In the above window, click on New->write path of Flutter bin folder in variable value -> ok -> ok -> ok.

Step 5: Now, run the **\$ flutter** command in the command prompt.

```
Windows PowerShell
PS C:\Users\Mikil> flutter
Manage your Flutter app development.

Common commands:
  flutter create <output directory>
    Create a new Flutter project in the specified directory.

  flutter run [options]
    Run your Flutter application on an attached device or in an emulator.

Usage: flutter <command> [<arguments>]

Global options:
  -h, --help           Print this usage information.
  -v, --verbose        Noisy logging, including all shell commands executed.
                      If used with "--help", shows hidden options. If used with "flutter doctor", shows additional
                      diagnostic information. (Use "-vv" to force verbose logging in those cases.)
  -d, --device-id     Target device id or name (prefixes allowed).
  --version           Reports the version of this tool.
  --suppress-analytics Suppress analytics reporting when this command runs.

Available commands:

Flutter SDK
  bash-completion   Output command line shell completion setup scripts.
  channel           List or switch Flutter channels.
  config            Configure Flutter settings.
  doctor             Show information about the installed tooling.
  downgrade         Downgrade Flutter to the last active version for the current channel.
  precache           Populate the Flutter tool's cache of binary artifacts.
  upgrade            Upgrade your copy of Flutter.
```

Now, run the **\$ flutter doctor** command. This command checks for all the requirements of Flutter app development and displays a report of the status of your Flutter installation.

```
Windows PowerShell
Moreover, Flutter includes the Dart SDK, which may send usage metrics and
crash reports to Google.

Read about data we send with crash reports:
https://flutter.dev/docs/reference/crash-reporting

See Google's privacy policy:
https://policies.google.com/privacy

PS C:\Users\Mikil> flutter doctor
Running "flutter pub get" in flutter_tools...                                11.6s
Doctor summary (to see all details, run flutter doctor -v):
[!] Flutter (Channel stable, 3.3.10, on Microsoft Windows [Version 10.0.22621.963], locale en-IN)
[!] Android toolchain - develop for Android devices
    X Unable to locate Android SDK.
      Install Android Studio from: https://developer.android.com/studio/index.html
      On first launch it will assist you in installing the Android SDK components.
      (or visit https://flutter.dev/docs/get-started/install/windows#android-setup for detailed instructions).
      If the Android SDK has been installed to a custom location, please use
      'flutter config --android-sdk' to update to that location.

[!] Chrome - develop for the web
[!] Visual Studio - develop for Windows
    X Visual Studio not installed; this is necessary for Windows development.
      Download at https://visualstudio.microsoft.com/downloads/.
      Please install the "Desktop development with C++" workload, including all of its default components
[!] Android Studio (not installed)
[!] VS Code (version 1.74.1)
[!] Connected device (3 available)
[!] HTTP Host Availability

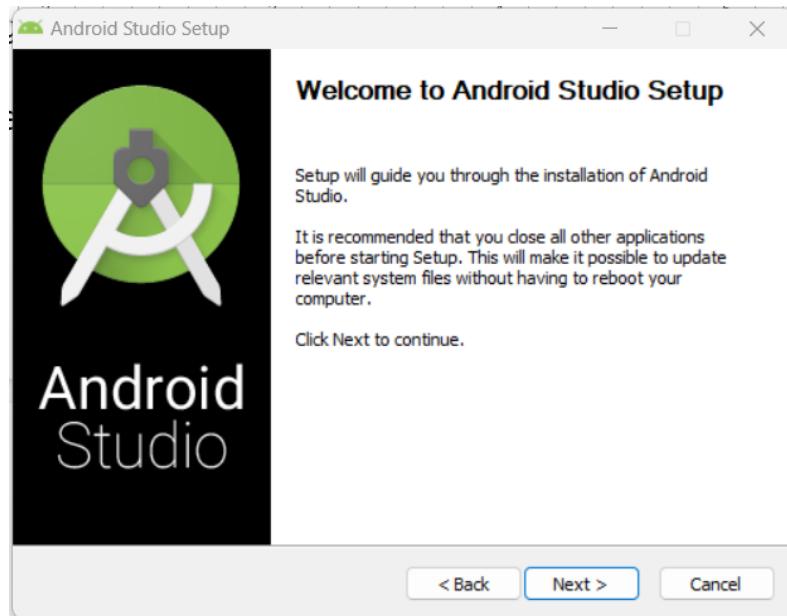
! Doctor found issues in 3 categories.
PS C:\Users\Mikil>
```

Step 6: When you run the above command, it will analyze the system and show its report, as shown in the below image. Here, you will find the details of all missing tools, required to run Flutter as well as the development tools that are available but not connected to the device.

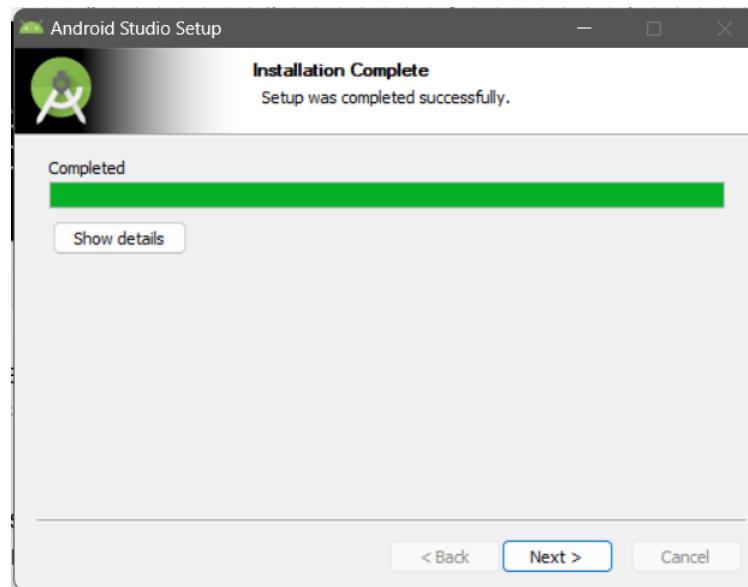
Step 7: Install the Android SDK. If the flutter doctor command does not find the Android SDK tool in your system, then you need first to install the Android Studio IDE. To install Android Studio IDE, do the following steps.

Step 7.1: Download the latest Android Studio executable or zip file from the official site.

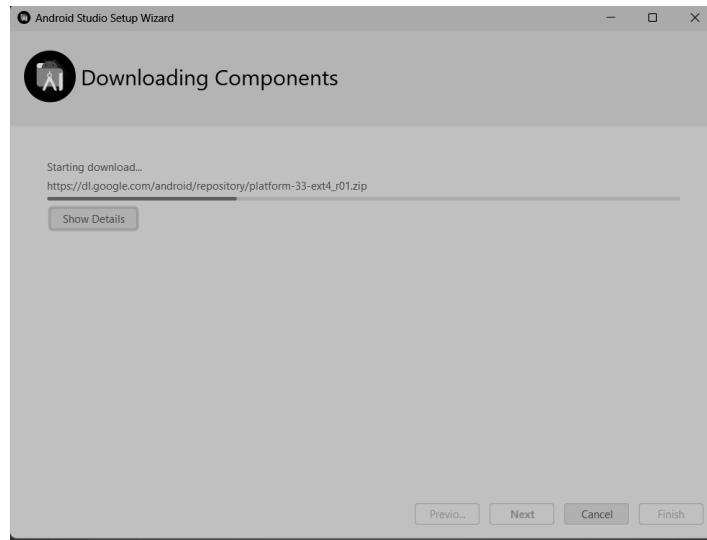
Step 7.2: When the download is complete, open the .exe file and run it. You will get the following dialog box.



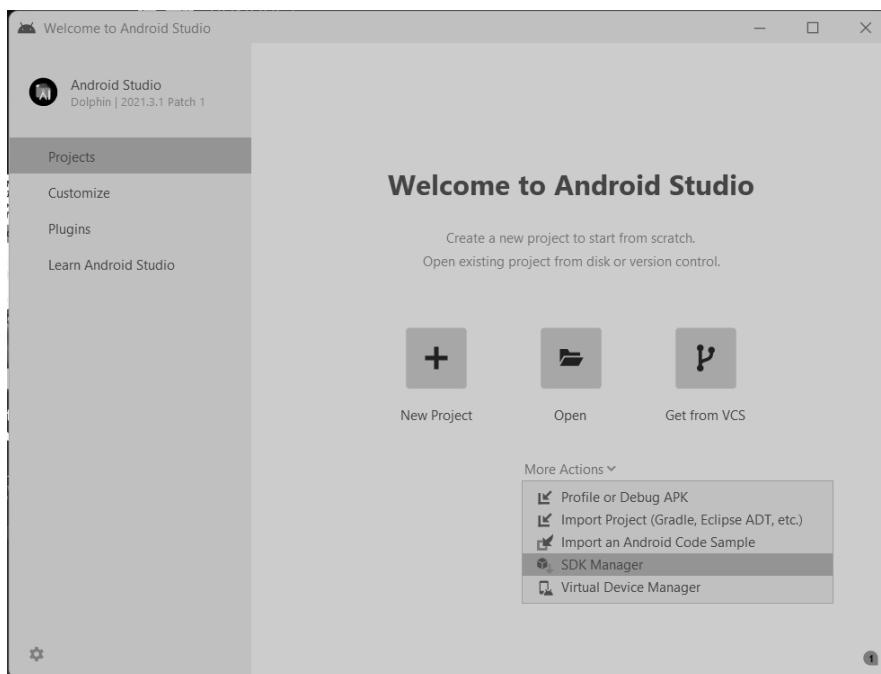
Step 7.3: Follow the steps of the installation wizard. Once the installation wizard completes, you will get the following screen.

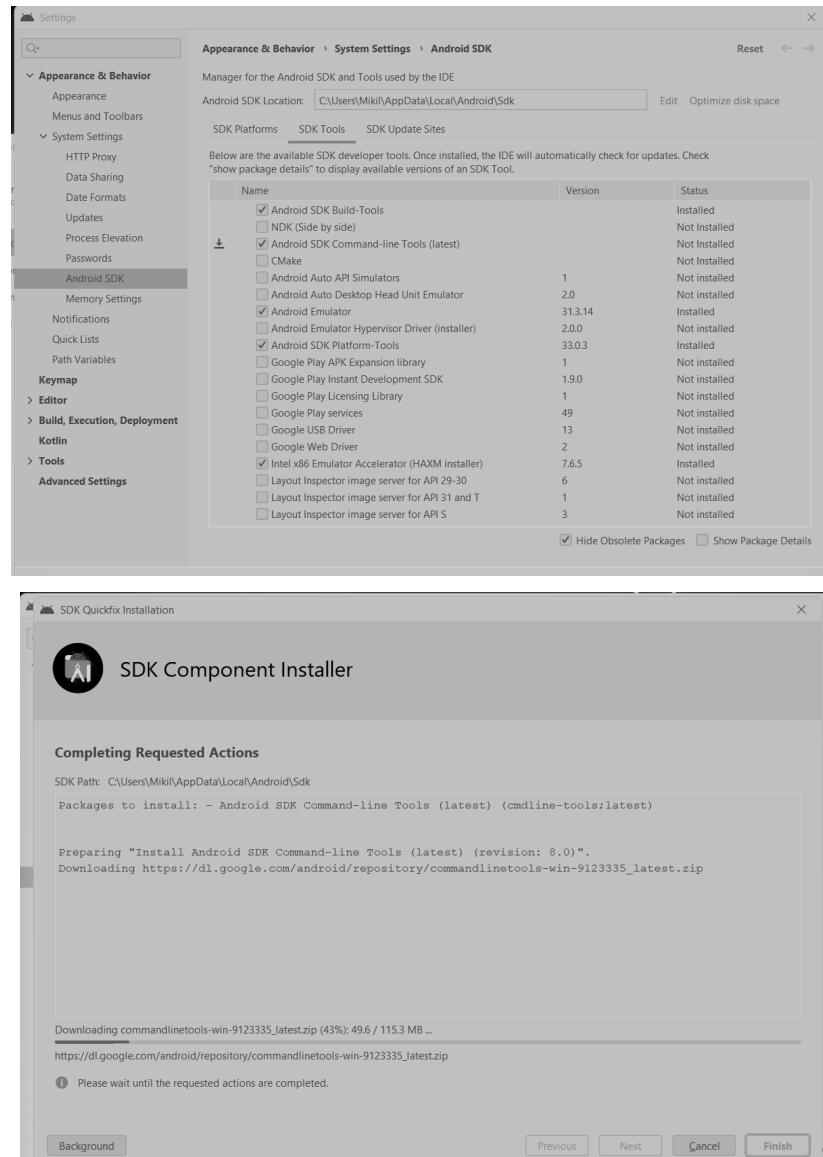


Step 7.4: In the above screen, click Next-> Finish. Once the Finish button is clicked, you must choose the 'Don't import Setting option' and click OK. It will start the Android Studio.



Step 7.5: Goto SDK manager in more actions and then SDK Tools and install Android SDK Tools and Android SDK Command-line Tools.





Step 7.6: run the **\$ flutter doctor** command and Run the **\$ flutter doctor --android-licenses** command.

```

PS C:\Users\Mikil> flutter doctor --android-licenses
[=====] 100% Computing updates...
5 of 7 SDK package licenses not accepted.
Review licenses that have not been accepted (y/N)? y

1/5: License android-googletv-license:
-----
Terms and Conditions

This is the Google TV Add-on for the Android Software Development Kit License Agreement.

1. Introduction

1.1 The Google TV Add-on for the Android Software Development Kit (referred to in this License Agreement as the "Google TV Add-on" and specifically including the Android system files, packaged APIs, and Google APIs add-ons) is licensed to you subject to the terms of this License Agreement. This License Agreement forms a legally binding contract between you and Google in relation to your use of the Google TV Add-on.

1.2 "Google" means Google Inc., a Delaware corporation with principal place of business at 1600 Amphitheatre Parkway, Mountain View, CA 94043, United States.

2. Accepting this License Agreement

2.1 In order to use the Google TV Add-on, you must first agree to this License Agreement. You may not use the Google TV Add-on if you do not accept this License Agreement.

2.2 You can accept this License Agreement by:

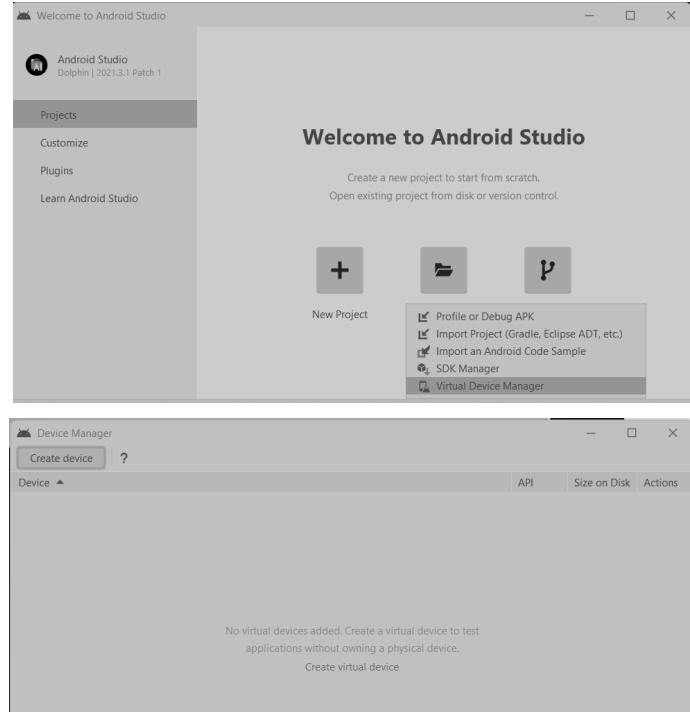
(A) clicking to accept or agree to this License Agreement, where this option is made available to you; or
(B) by actually using the Google TV Add-on. In this case, you agree that use of the Google TV Add-on constitutes acceptance of the License Agreement from that point onwards.

2.3 You may not use the Google TV Add-on and may not accept the Licensing Agreement if you are a person barred from rece

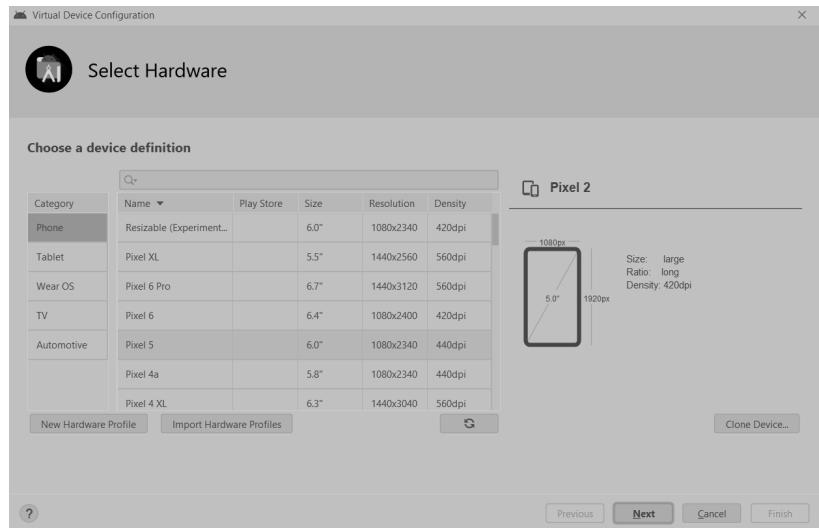
```

Step 8: Next, you need to set up an Android emulator. It is responsible for running and testing the Flutter application.

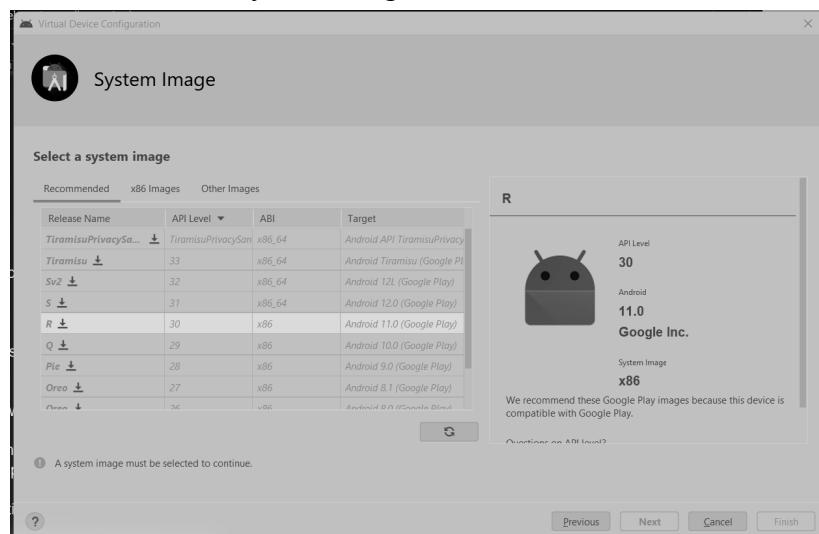
Step 8.1: To set an Android emulator, go to **Android Studio > More Actions > Virtual Device Manager >** and select **Create Virtual Device**.

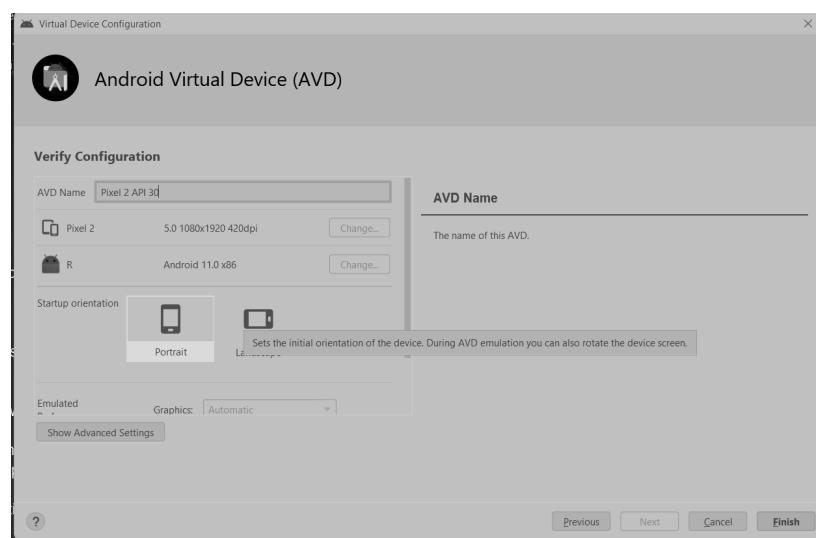
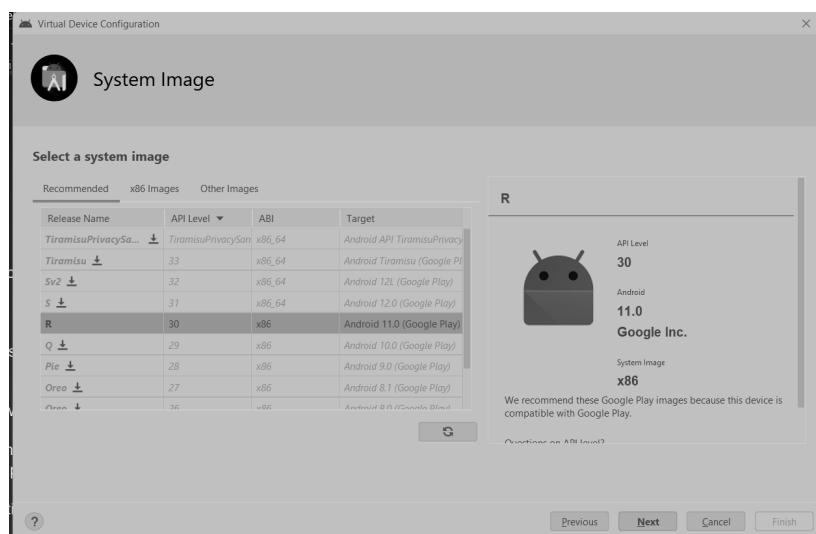
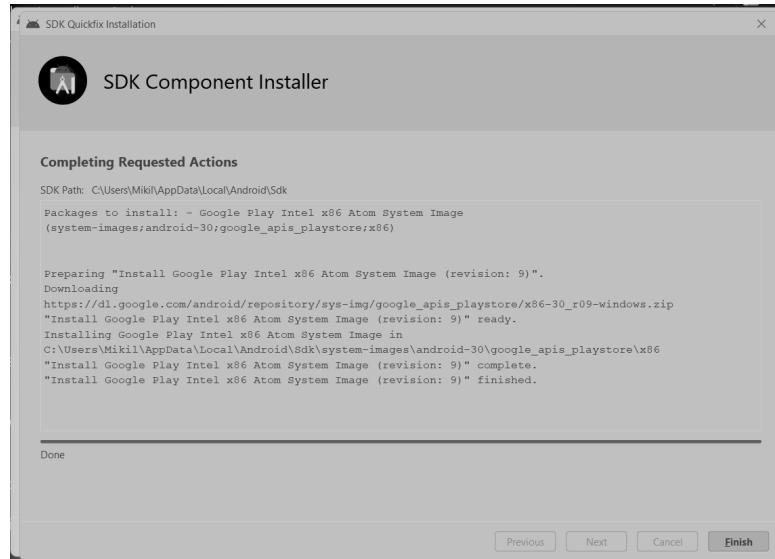


Step 8.2: Choose your device definition and click on Next.

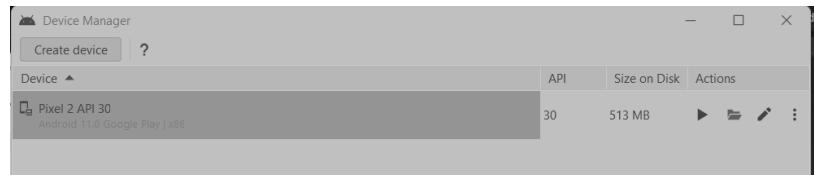


Step 8.3: Download and select the system image for the latest Android version and click on Next.





Step 8.4: Now, verify all AVD configurations. If it is correct, click on Finish. The following screen appears.



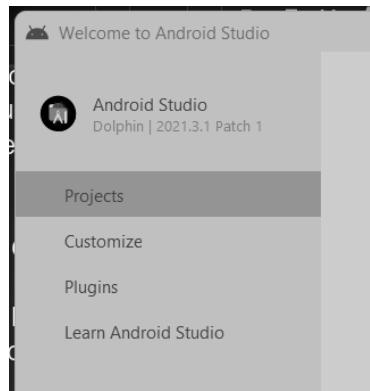
Step 8.5: Last, click on the icon pointed to the red color rectangle. The Android emulator displayed below the screen.



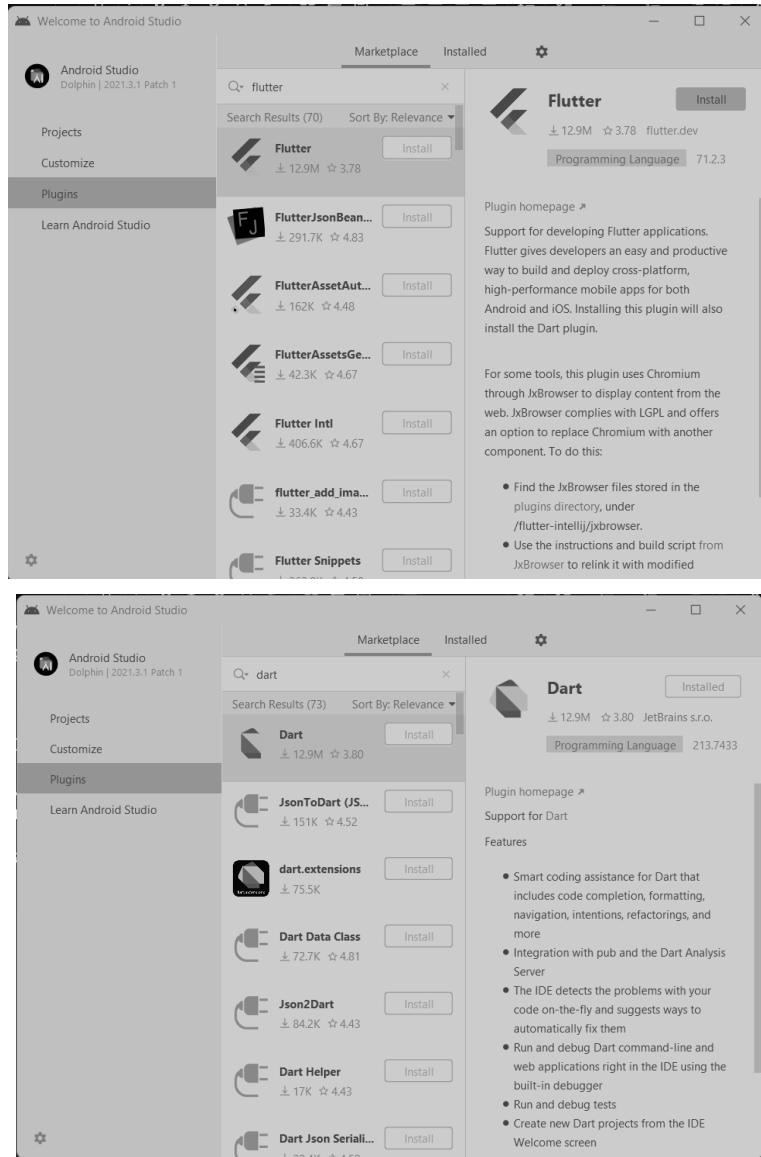
Step 9: Now, install Flutter and Dart plugins for building the Flutter application in Android Studio.

These plugins provide a template to create a Flutter application and give the option to run and debug the Flutter application in the Android Studio itself. Do the following steps to install these plugins.

Step 9.1: Open the Android Studio and go to Plugins.



Step 9.2: Now, search the Flutter plugin. If found, select the Flutter plugin and click install.
When
you click on install, it will ask you to install Dart plugin as below screen. Click yes to proceed.



Step 9.3: Restart the Android Studio.

Conclusion -

Thus we have successfully installed Flutter and executed our first program.