EXP 1: Installation and Configuration of Flutter Environment.

Aim: To install and configure the Flutter development environment on your system to start building applications with Flutter.

Theory:Flutter is an open-source framework developed by Google to create cross-platform applications from a single codebase. It enables the development of high-performance apps for Android, iOS, Web, and Desktop. Flutter uses Dart, a programming language developed by Google, which offers fast compilation and powerful async features.

Key Components for Setting Up the Flutter Environment:

1. Flutter SDK:
    - The core development kit for building Flutter applications. It includes the Flutter framework, which provides widgets and tools for creating UIs, and the engine, which handles rendering using the Skia graphics library.
    - It also includes Flutter CLI tools like flutter doctor, flutter run, and flutter build for managing projects and debugging.
2. Dart SDK:
    - Dart is the programming language used by Flutter. It is object-oriented and supports asynchronous programming.
    - Dart uses Just-In-Time (JIT) compilation during development for fast build times and Hot Reload, and Ahead-Of-Time (AOT) compilation for production builds for better performance.
3. IDE (Integrated Development Environment):
    - Android Studio: A full-featured IDE that includes Flutter and Dart plugins, device emulators, and tools for debugging and profiling.

- Visual Studio Code (VS Code): A lightweight editor with excellent Flutter and Dart plugin support for development and debugging.

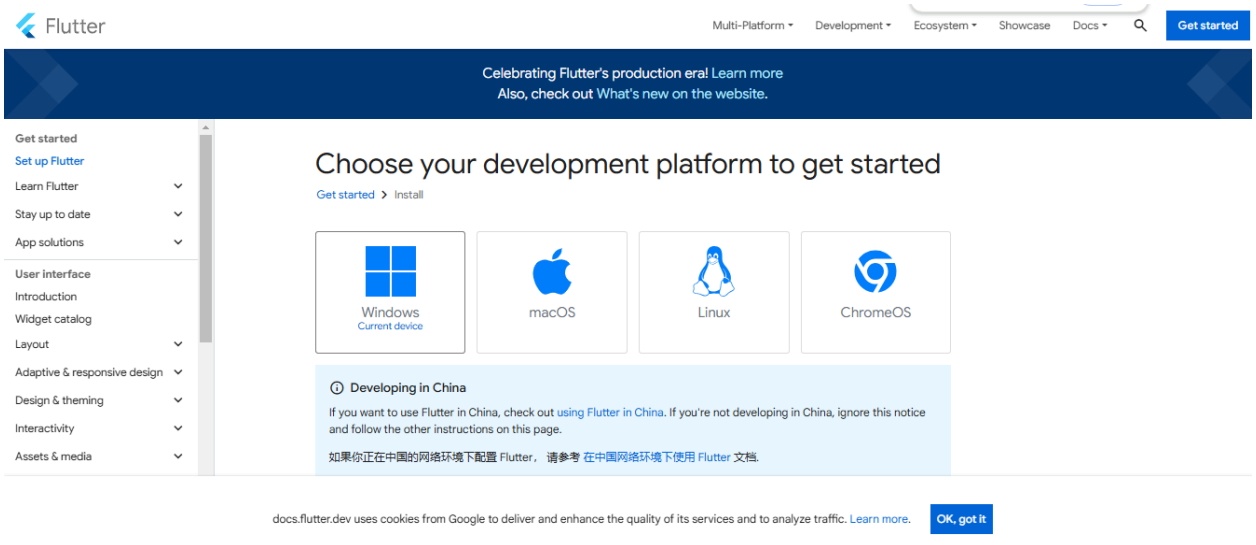
4. Platform-Specific Tools:

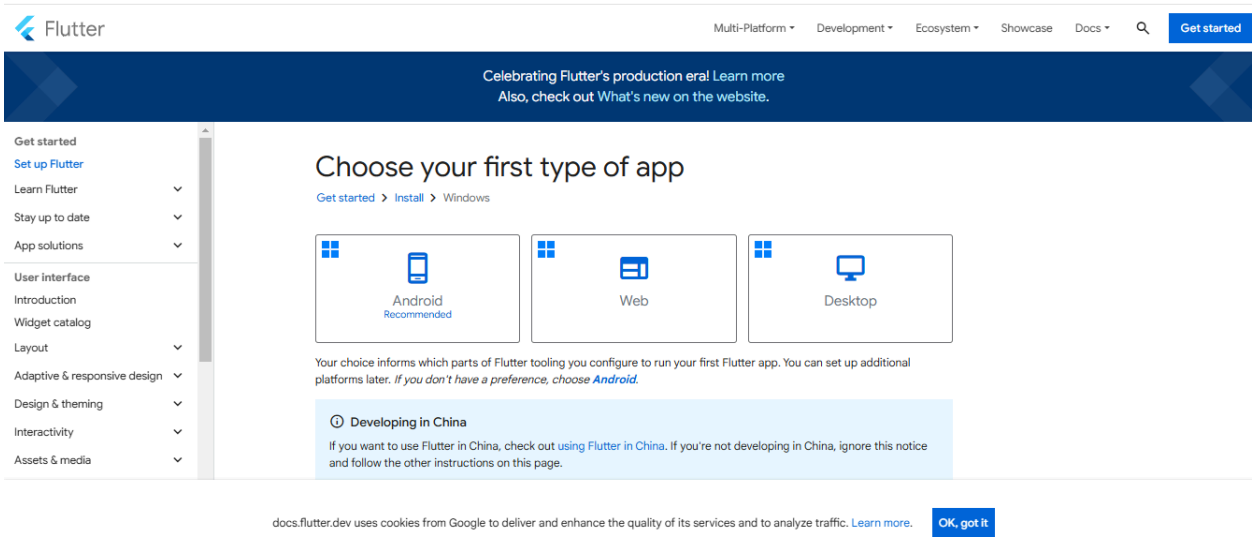
   - Android Studio: Required for Android development with Flutter, including the Android SDK and emulator.

   - Xcode: Needed for iOS development on macOS, with tools like the iOS Simulator

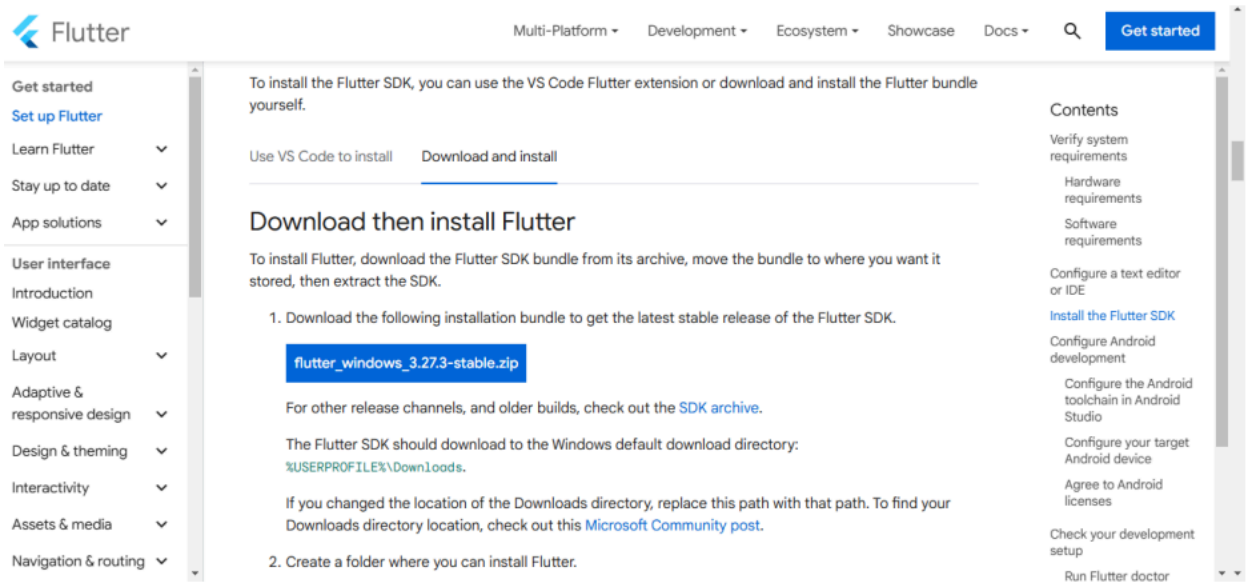
Steps To Install Flutter:

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.





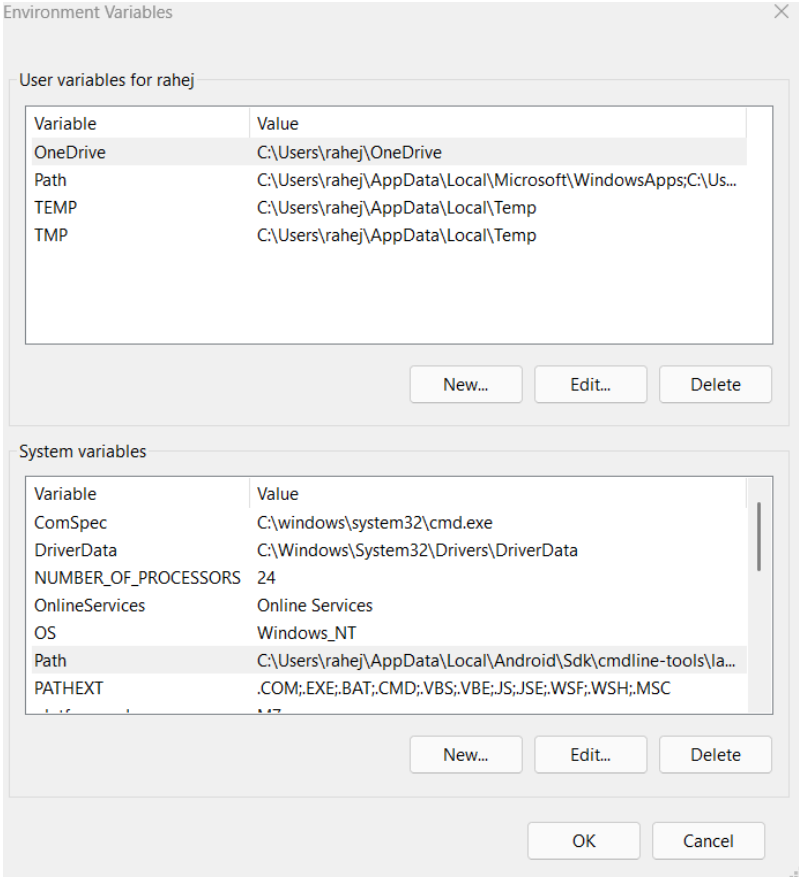
Step 2: Next, to download the latest Flutter SDK, click on the Windows icon. Here, you will
find the download link for SDK.



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.

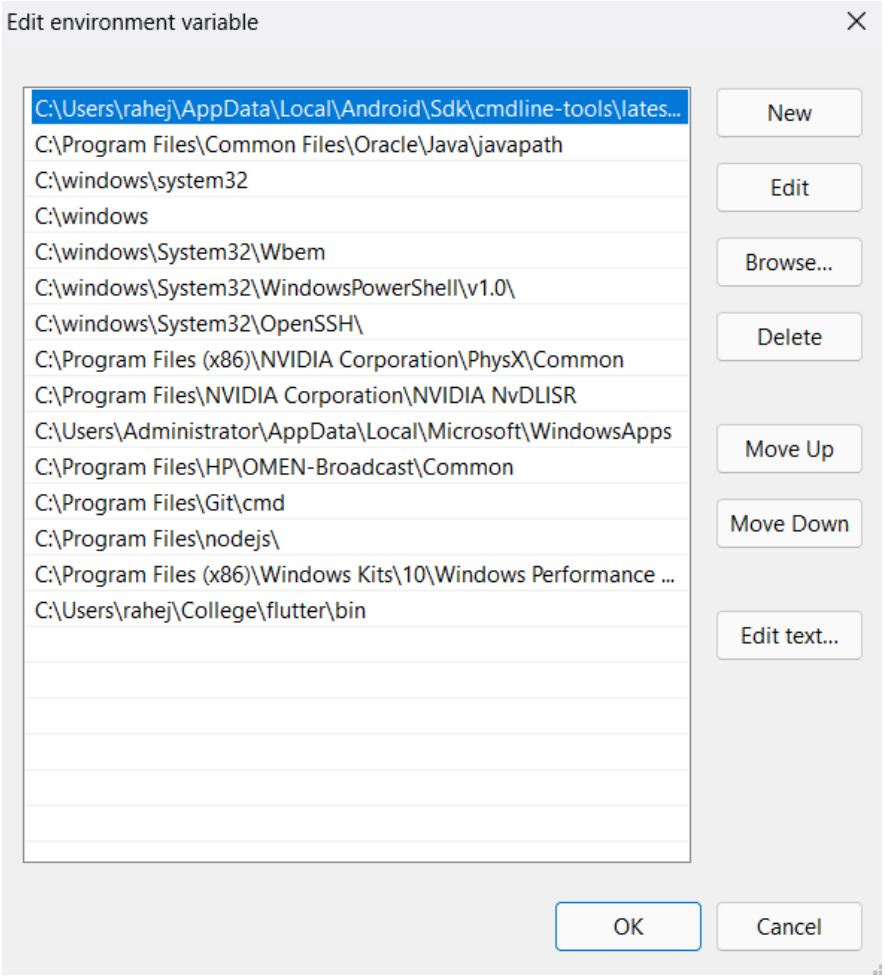
Step 4: To run the Flutter command in regular windows console, you need to update the system
path to include the flutter bin directory. The following steps are required to do this:

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 command prompt.

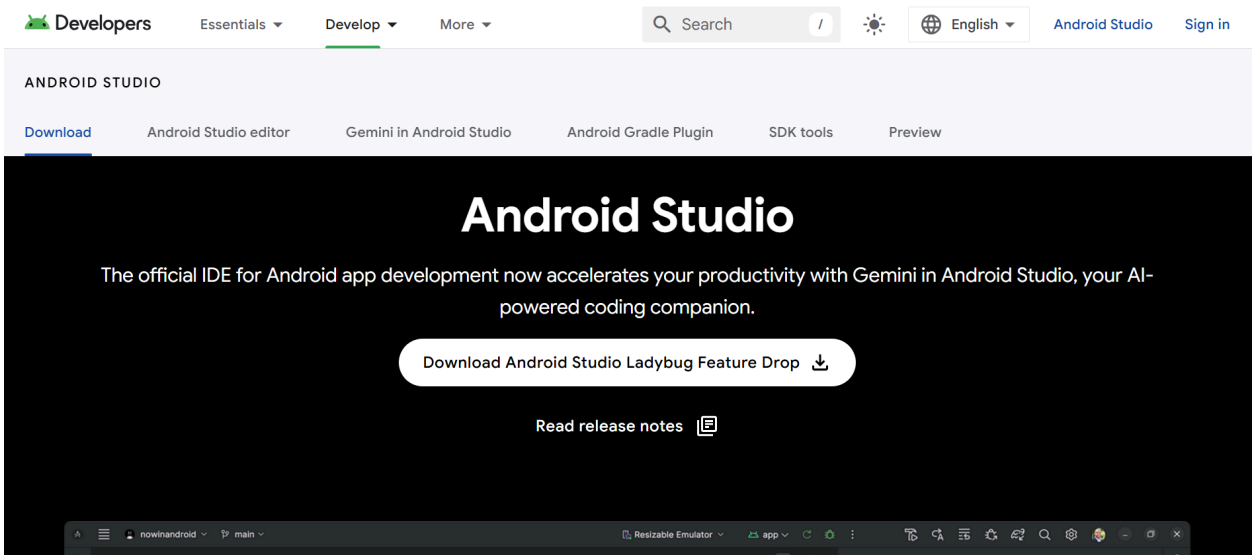
Now, run the $ flutter doctor command. This command checks for all the requirements of Flutter app development and displays a report of the status of your Flutter installation.

```
C:\Users\rahej>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.
    --enable-analytics      Enable telemetry reporting each time a flutter or dart command runs.
    --disable-analytics     Disable telemetry reporting each time a flutter or dart command runs, until it is
                            re-enabled.
    --suppress-analytics    Suppress analytics reporting for the current CLI invocation.
```

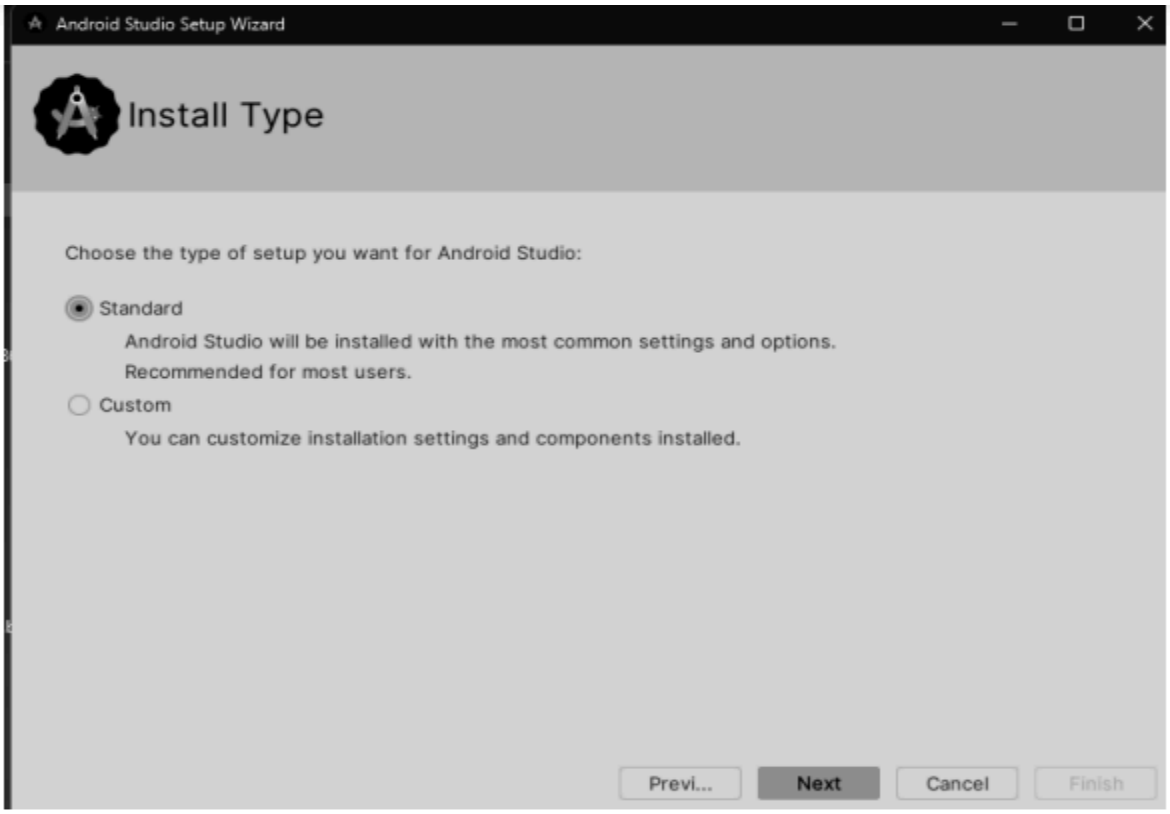
Step 6: When you run the above command, it will analyze the system and show its report, as shown in the below image. Here, you will find the details of all missing tools, which required to run Flutter as well as the development tools that are available but not connected with the device.

```
C:\Users\rahej>flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 3.29.2, on Microsoft Windows [Version 10.0.26100.3476], locale en-IN)
[✓] Windows Version (Windows 11 or higher, 24H2, 2009)
[✓] Android toolchain - develop for Android devices (Android SDK version 35.0.1)
[✓] Chrome - develop for the web
[✓] Visual Studio - develop Windows apps (Visual Studio Community 2022 17.13.2)
[✓] Android Studio (version 2024.3)
[✓] VS Code (version 1.98.0)
[✓] Connected device (3 available)
[✓] Network resources

• No issues found!
```

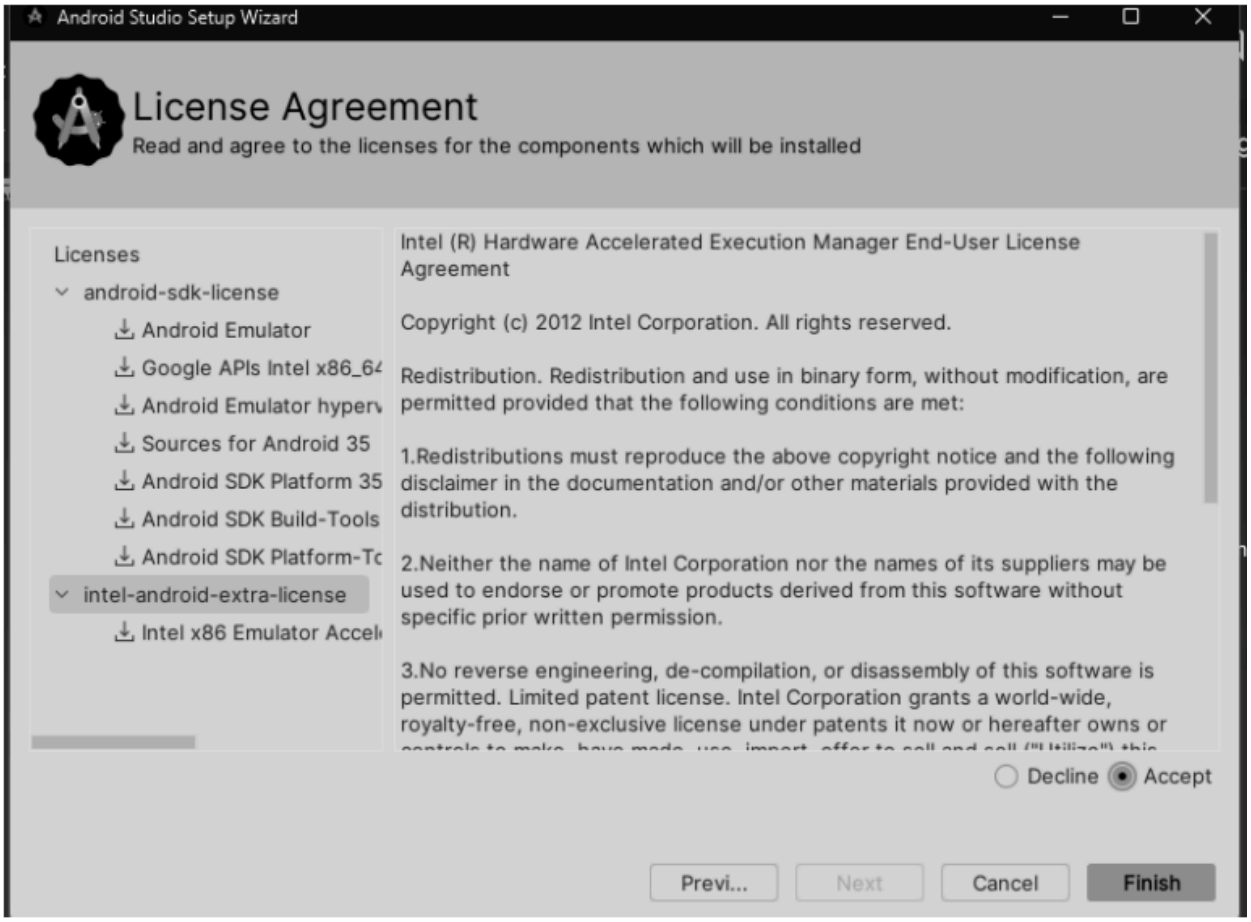
Step 7: Install the Android SDK. If the flutter doctor command does not find the Android SDK tool in your system, then you need first to install the Android Studio IDE. To install Android Studio IDE, do the following steps.



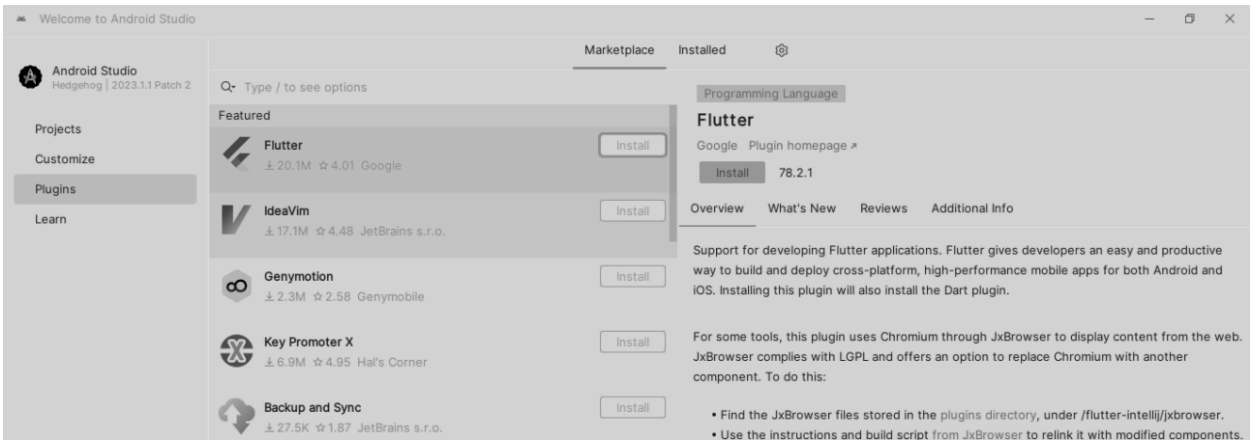
Step 8: After opening the installer you will see the following. Select standard and click next.



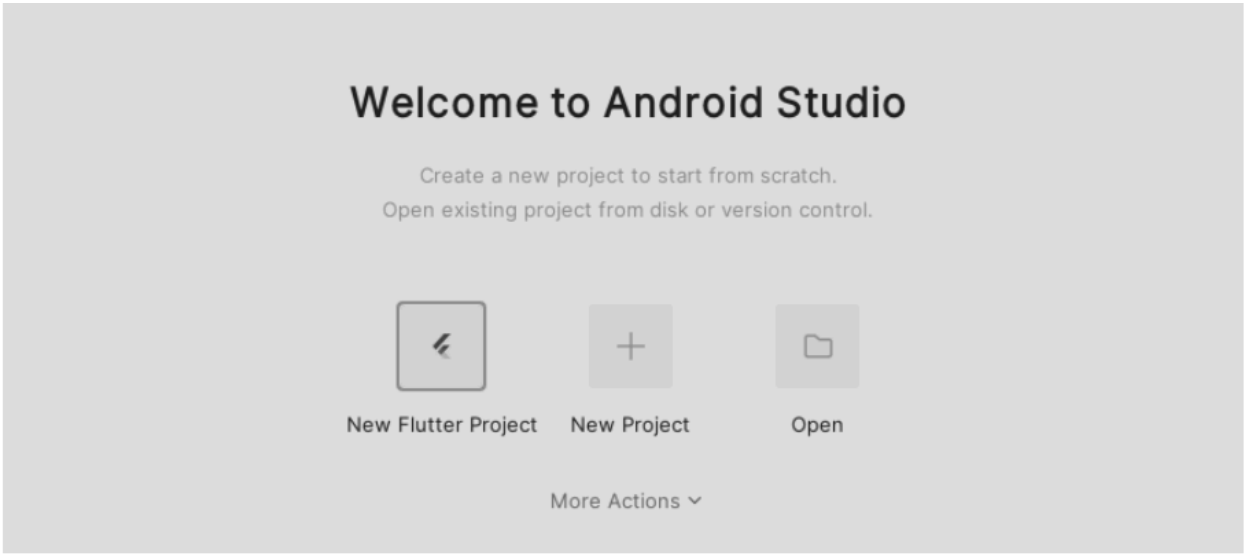
Step 9: Accept all and finish the installation.



Step 10: Install Flutter and Dart plugins from Marketplace.

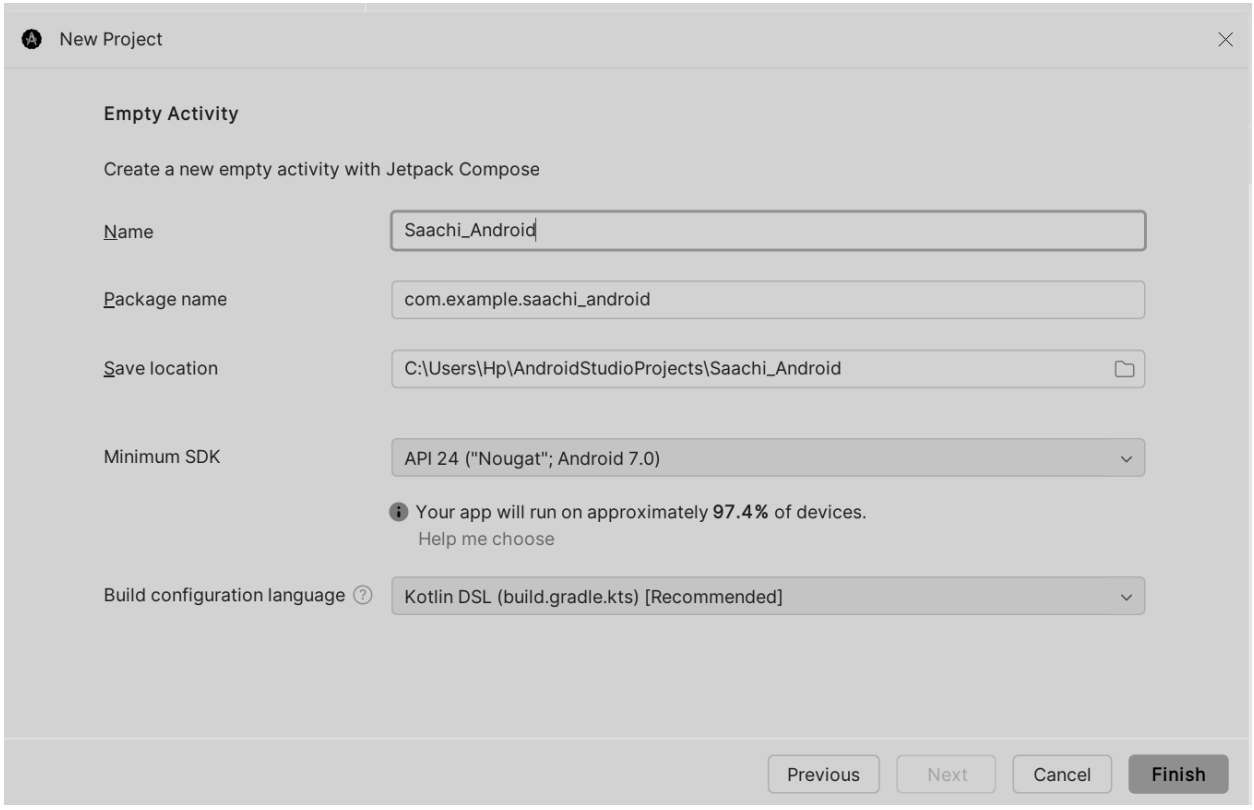


Step 11: Click on New Flutter Project.



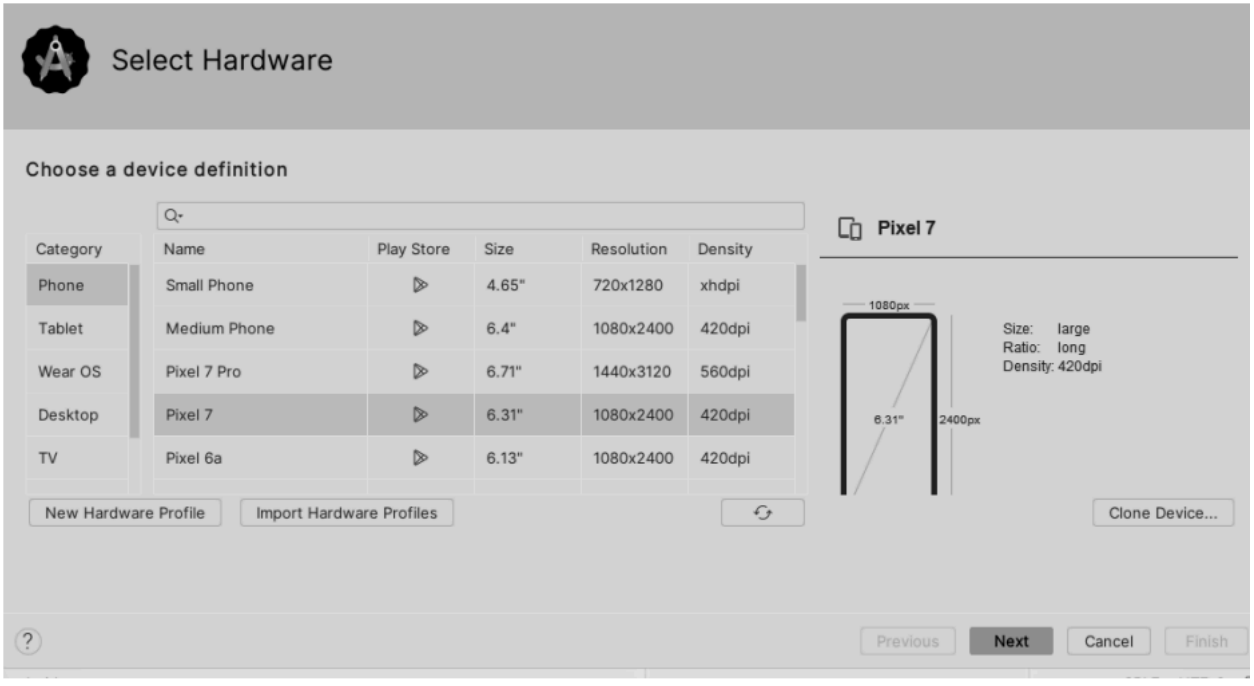
Step 12: Set path C:\flutter\flutter

Step 13: Enter a name for project and click on create.

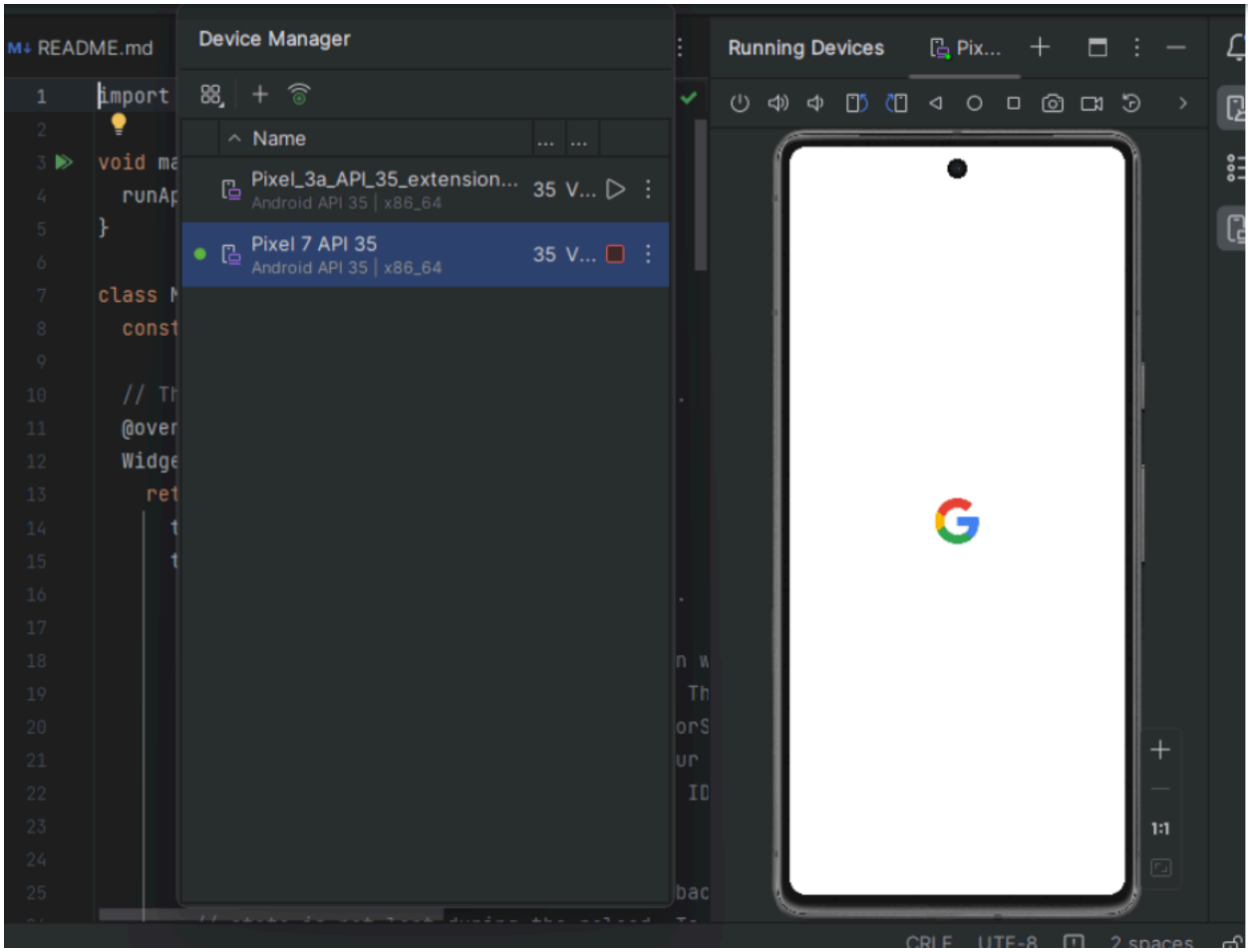


Step 14: Go to Menu > Tools > Device Manager

Step 15: Choose your device on which you want to run your project and click Next.



Step 16 : Click on play button in the toolbar above and you will see the emulator starting. It will take time to load for the first time.



Conclusion:Flutter allows developers to build cross-platform applications with a **single codebase**. The **Flutter SDK**, **Dart programming language**, and the right **IDE** (Android Studio or Visual Studio Code) are essential for setting up the development environment. With tools for fast compilation, real-time changes (Hot Reload), and native-like performance, Flutter makes it easy to create powerful, multi-platform apps.