Aim : Installation and Configuration of Flutter Environment.

Theory

Flutter, an open-source UI development toolkit by Google, is widely used for building natively compiled applications for mobile, web, and desktop from a single codebase. The installation and configuration of the Flutter environment serve as the initial step for developers to harness this powerful framework for app development.

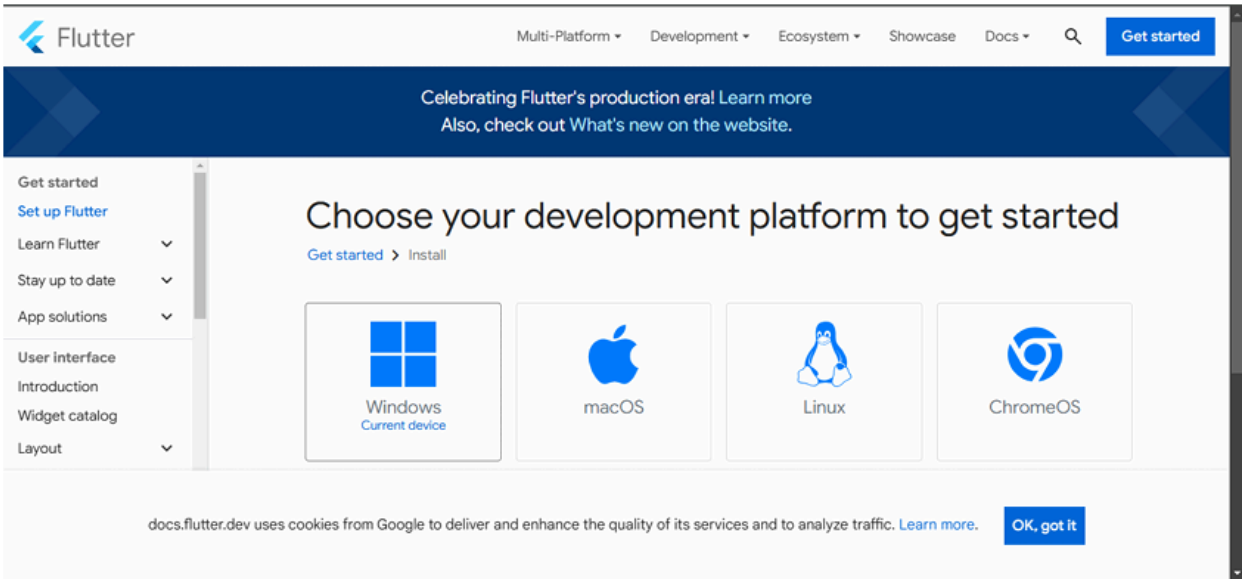
To achieve this setup, the process involves ensuring that all dependencies—such as the Flutter SDK, Android Studio, and required plugins—are appropriately installed. The configuration ensures the seamless functioning of Flutter's development and debugging capabilities. This experiment aims to familiarize developers with the essential tools and establish a robust development environment.

The underlying concepts include:

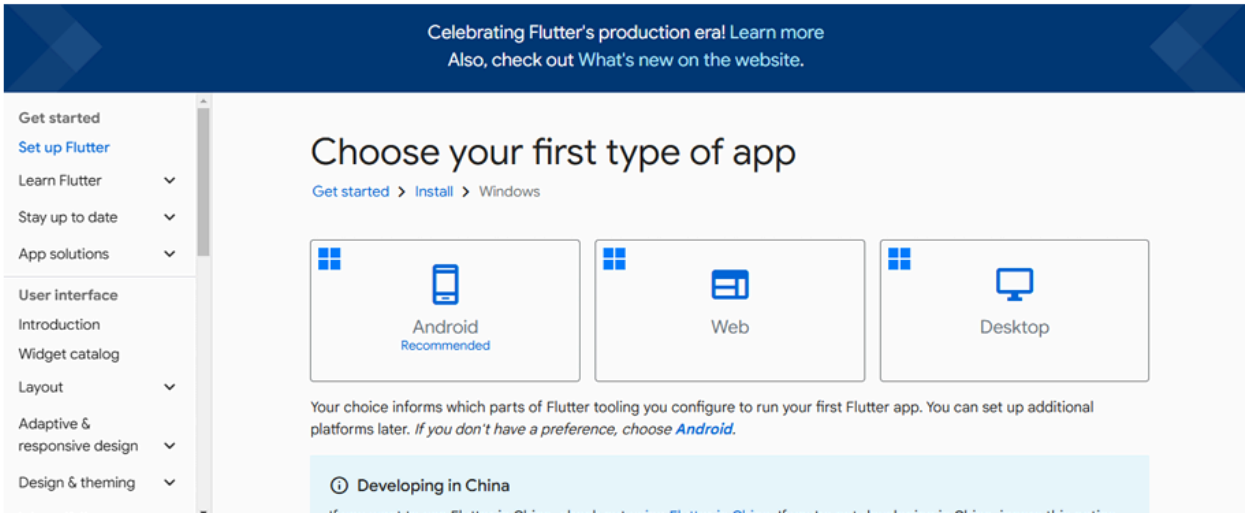
- **Flutter SDK:** A collection of tools, libraries, and documentation necessary for Flutter development.
- **Dart Programming Language:** The language used to write Flutter applications.
- **IDE Integration:** The role of Integrated Development Environments like Android Studio or Visual Studio Code in enhancing productivity.
- **Environment Variables:** Their importance in configuring tools like Flutter and ADB (Android Debug Bridge).

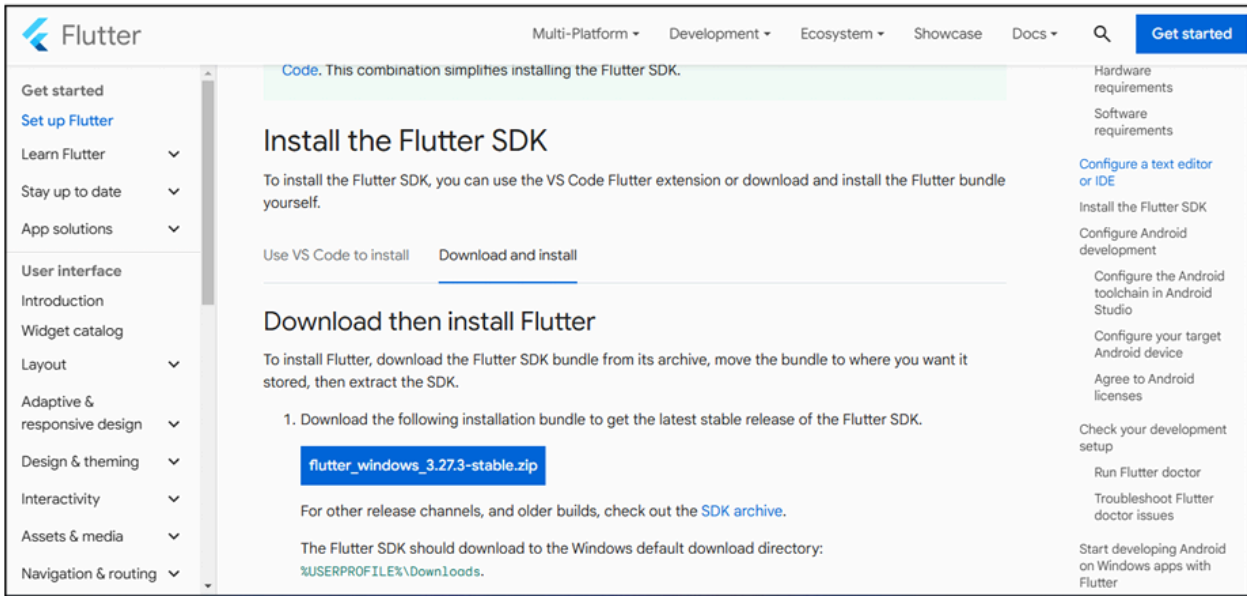
Steps :

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 and then select Android. Here, you will find system requirements and 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 : Now, run the $ flutter command in command prompt.

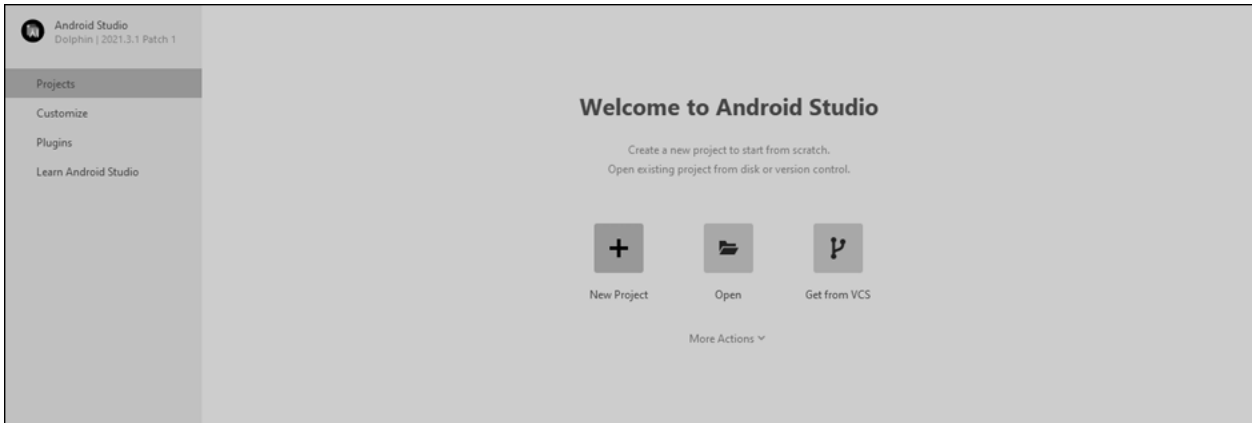
```
Common commands:

  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.

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.
```

Step 5 : 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. Step 8: 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.

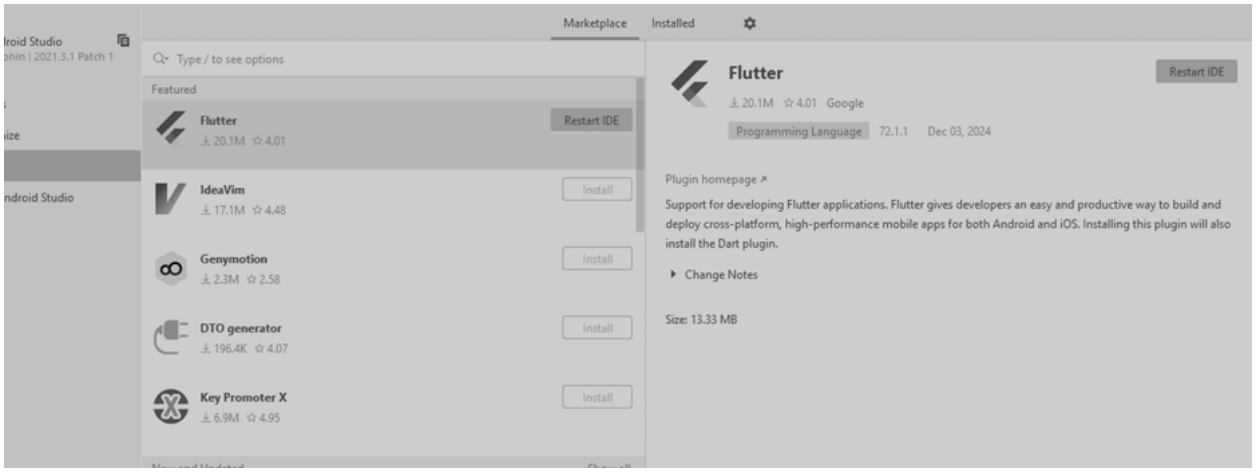
```
Doctor summary (to see all details, run flutter doctor -v):
[√] Flutter (Channel stable, 3.27.3, on Microsoft Windows [Version 10.0.22631.4751], locale en-IN)
[√] Windows Version (Installed version of Windows is version 10 or higher)
[√] Android toolchain - develop for Android devices (Android SDK version 34.0.0)
[√] Chrome - develop for the web
[√] Visual Studio - develop Windows apps (Visual Studio Build Tools 2022 17.12.3)
[!] Android Studio (not installed)
[√] VS Code (version 1.96.4)
[√] Connected device (4 available)
[√] Network resources

! Doctor found issues in 1 category.
```

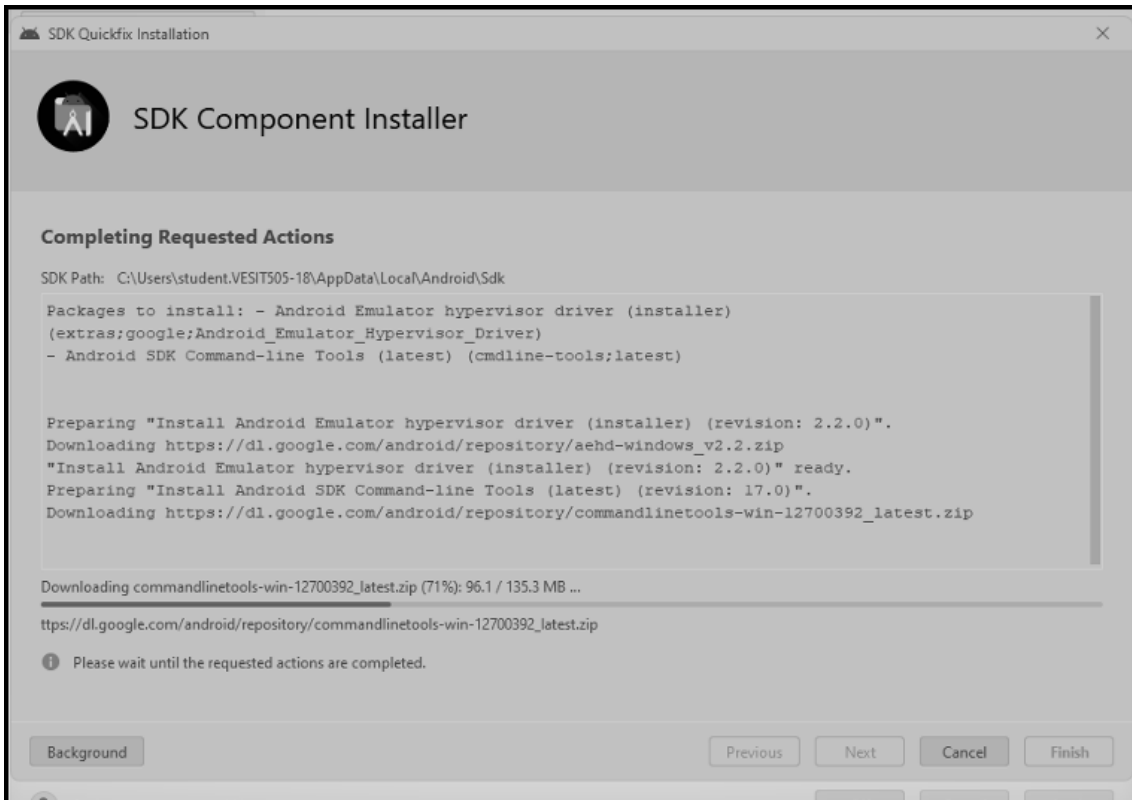
Step 6: Install the Android SDK. If the flutter doctor command does not find the Android SDK tool in your system, then you need first to install the Android Studio IDE. To install Android Studio IDE, do the following steps.

Step 7 : Download the latest Android Studio executable or zip file from the official site by accepting terms and conditions

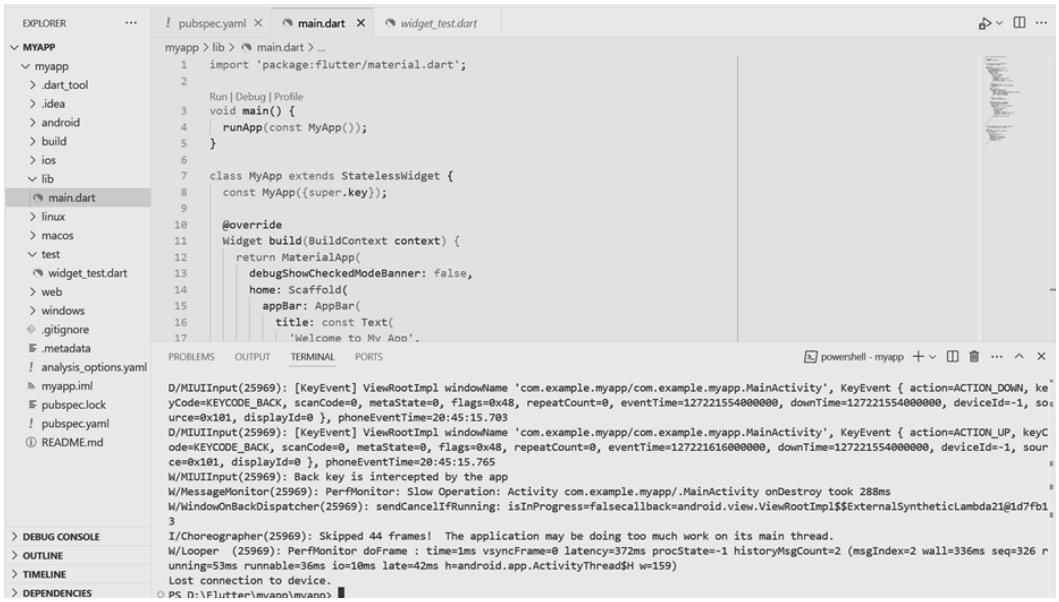


Now open android studio you will see the following window. Click on more actions-> Import an android code Sample -> select Android SDK command-line tools (latest) this will download command-line tools.





Creating project in visual studio code : >flutter create myapp >cd myapp >flutter run



Conclusion: Thus we have successfully installed flutter and ran it to create a sample application.