120A3051

Shreya Idate

Batch: E3

Experiment No. : 1

Aim: To install and configure Flutter Environment in Windows.

Theory:

Cross Platform App/Dev

Cross-platform application development is about building a single application that can run on various operating systems, instead of developing different app versionsfor each platform. The driving force for cross-platform application development is to produce software that works well in more than one specific digital environment, with the main purpose of selling it to a wider customer base. When you work on cross-platform app development, you can launch your software quickly on various platforms. The source code is written once for all platforms. This means you don't need to hire a separate software development team for each platform, as it's possible to launch and update the software by using a variety of cross-platform development tools.

It requires the deployment of a single script instead of writing separate scripts for each platform. This significantly speeds up development time and cuts time to market**,** which benefits everyone, from the dev team all the way to marketing.

Since cross-platform applications are Internet-based, updates are nice and easy. Users don't have to download separate updates, which would require the maintenance and support of multiple app versions. The app is updated automatically for all customers to ensure they always have the most current version of the app, which positively impacts itsperformance.

What is Flutter?

Flutter was introduced by Google as an open-source technology for coding and creating native apps for Android and iOS. Flutter is relatively new as it was officially presented in December 2018 as the first stable version 1.0 at the Flutter Live event.

Flutter combines ease of development with performance similar to native performance while maintaining visual consistency between platforms. Flutter's programming language, Dart, was initially intended as a replacement for JavaScript. Most importantly, Flutter is open-source and completely free. As a cross-platform framework, Flutter mostclosely resembles React Native. Both allow for a reactive and declarative style of programming. Unlike React Native, however, Flutter doesn't need to use a JavaScript bridge, which improves app startup times and overall performance. Dart achieves this byusing Ahead-Of-Time (AOT) compilation.

What is Dart?

Dart is an open-source, general-purpose, object-oriented programming language with C- style syntax developed by Google in 2011. The purpose of Dart programming is to createa frontend user interfaces for the web and mobile apps. It is under active development, compiled to native machine code for building mobile

apps, inspired by other programming languages such as Java, JavaScript, C#, and is Strongly Typed. Since Dart is a compiled language so you cannot execute your code directly; instead, the compiler parses it and transfer it into machine code.

It supports most of the common concepts of programming languages like classes, interfaces, functions, unlike other programming languages. Dart language does not support arrays directly. It supports collection, which is used to replicate the data structuresuch as arrays, generics, and optional typing.

Example:

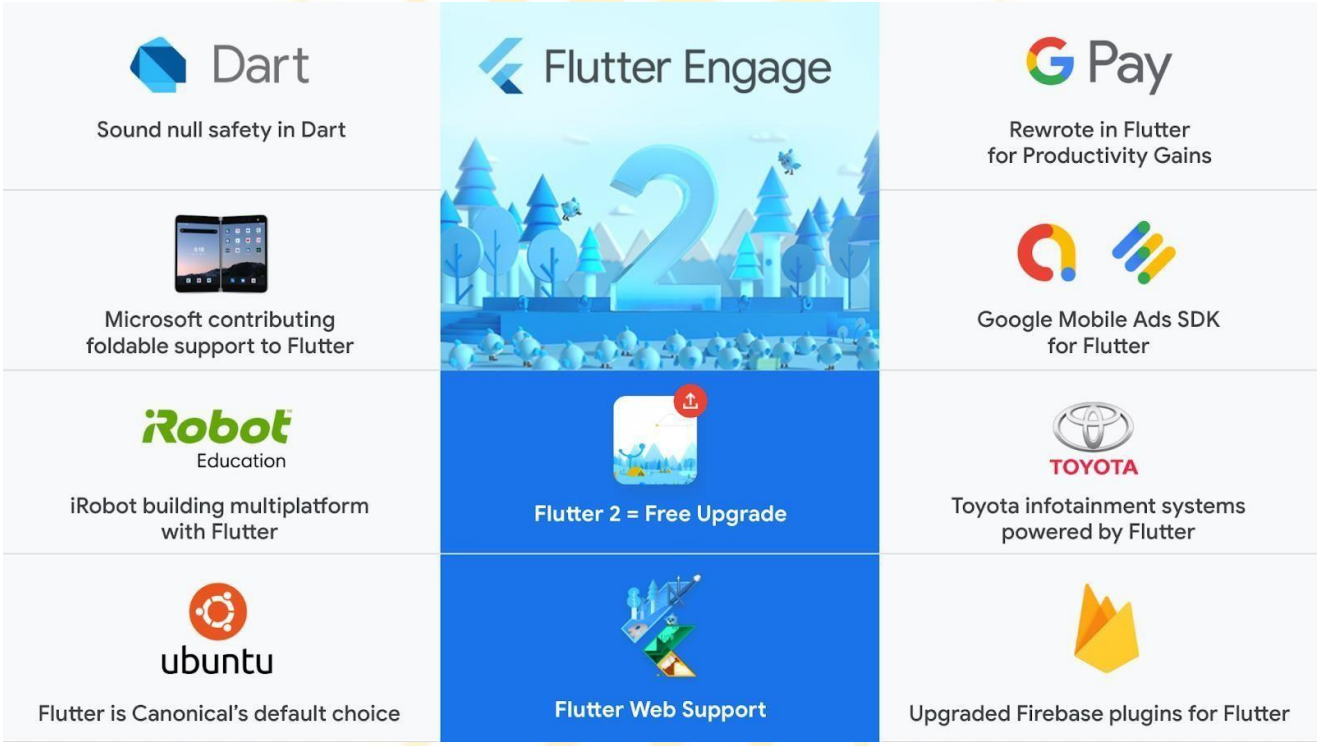
```
void main() {
    for (int i = 0; i < 5;
    i++) {print('hello ${i +
    1}');
}
}
```

Features of Flutter-

- **High productivity:** Since Flutter is cross-platform, you can use the same code base for your iOS and Android app. This can definitely save you both time and resources.
- **Great performance:** Dart compiles into native code and there is no need to access OEM widgets as Flutter has its own. This means less mediated communication between the app and the platform.
- **Fast and simple development:** One of the most lauded features of Flutter is *hot reload* which allows you to instantly view the changes made in the code on emulators, simulators and hardware. In less than a second, the changed code is reloaded while the app is running with no need for a restart. This is great not just for building UIs or adding features but also for bug fixing. As far as simplicity is concerned, Flutter claims in its docs that programming with Flutter is so easy that no prior programming

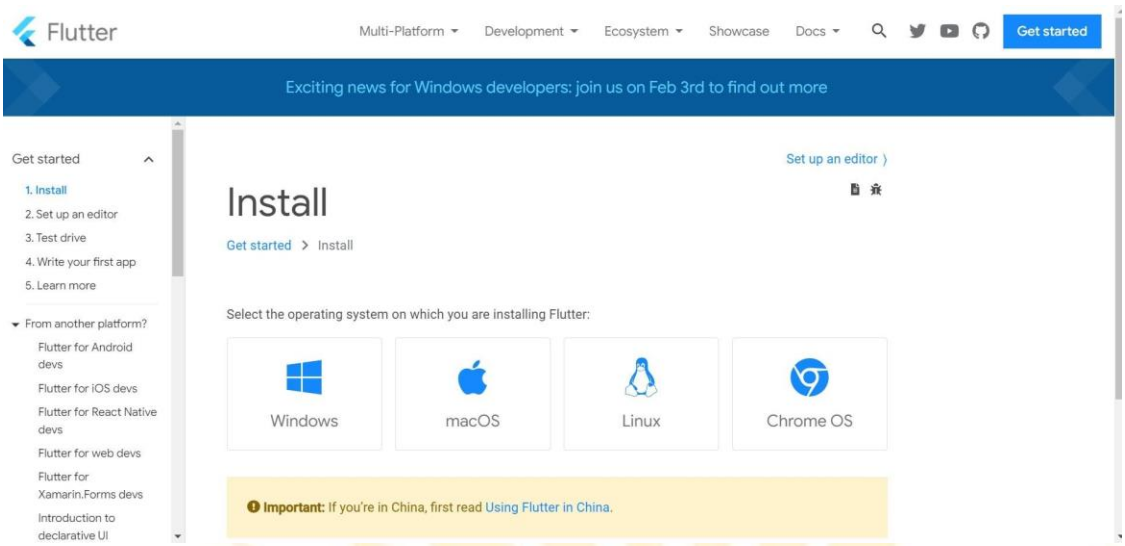
knowledge is required.

- **Compatibility:** Since widgets are part of the app and not the platform, you'll likely experience less or no compatibility issues on different OS versions. This in turnmeans less time spent on testing.
- **Open-source:** Both Flutter and Dart are open-source and free to use, and provideextensive documentation and community support to help out with any issues you may encounter.



Steps For Installation: VSCode

Step 1: Download the installation bundle of the Flutter Software Development Kit forwindows. To download Flutter SDK, go to its official website, click on Get started button, you will get the following screen.



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

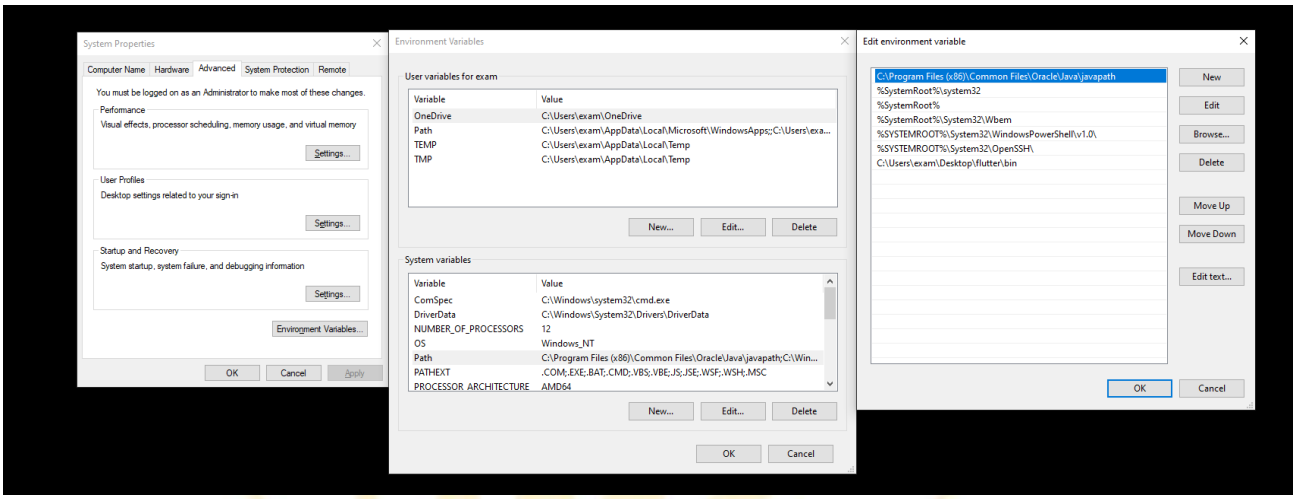
Step 3: When your download is complete, extract the zip file and place it in the desiredinstallation folder or location, for example, D: /Flutter.required.

Step 4: To run the Flutter command in regular windows console, you need to update thesystempath 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. Youwill getthe 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.

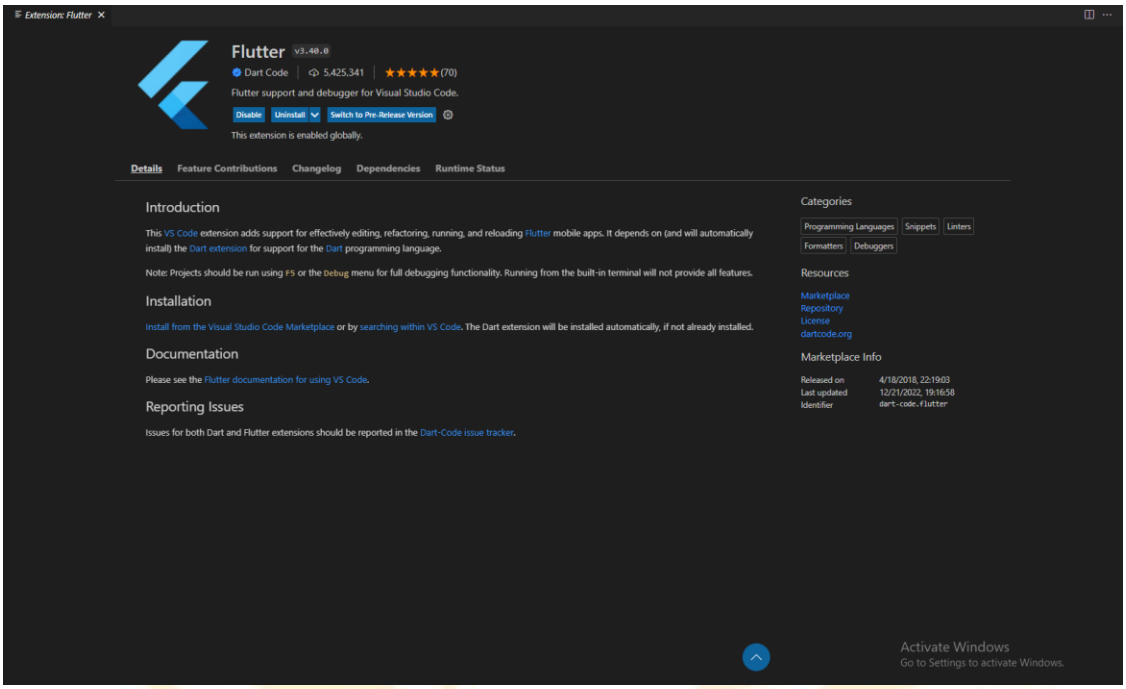


Step 5: Now, run the $ flutter doctor command. This command checks for all the requirements of Flutter app development and displays a report of the status of your Flutter installation.

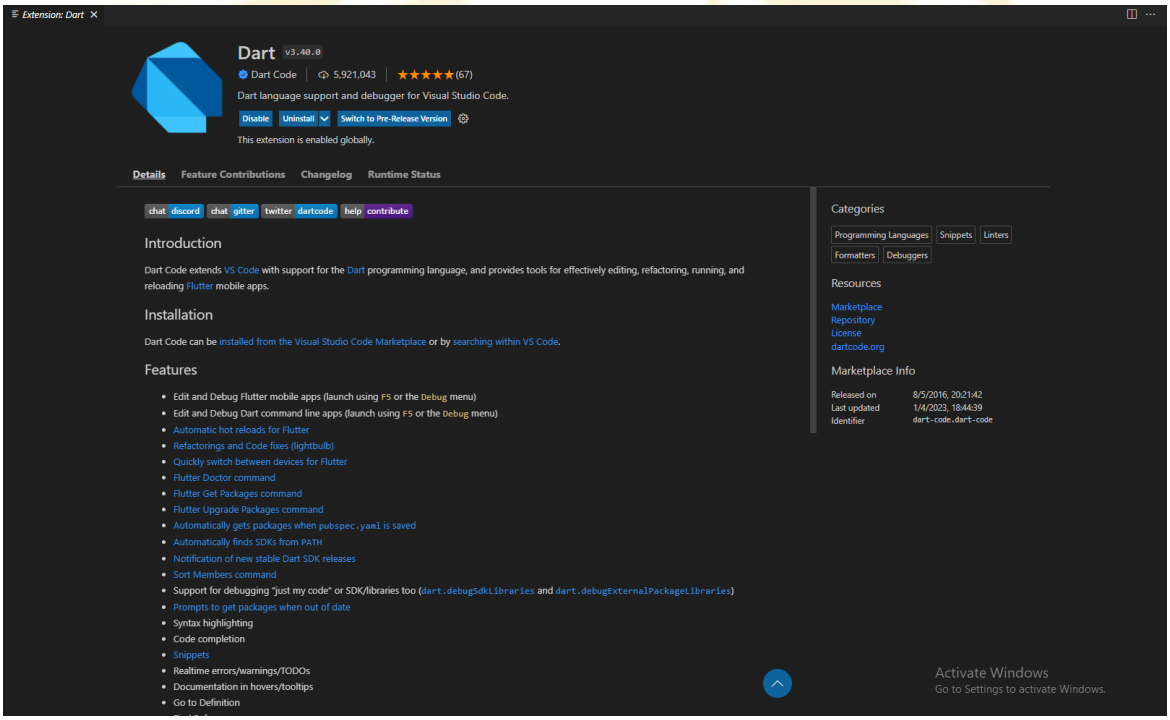
$ flutter doctor

```
C:\Users\exam>flutter doctor
Running "flutter pub get" in flutter_tools...                    11.9s
Doctor summary (to see all details, run flutter doctor -v):
[√] Flutter (Channel stable, 3.3.10, on Microsoft Windows [Version 10.0.18363.592], locale en-US)
[X] 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 (Visual Studio Community 2022 17.4.3)
    X The current Visual Studio installation is incomplete. Please reinstall Visual Studio.
[!] Android Studio (not installed)
[√] VS Code (version 1.66.2)
[√] Connected device (2 available)
[√] HTTP Host Availability

! Doctor found issues in 3 categories.

C:\Users\exam>
```

Step 6: Installing Flutter in Visual Studio Code

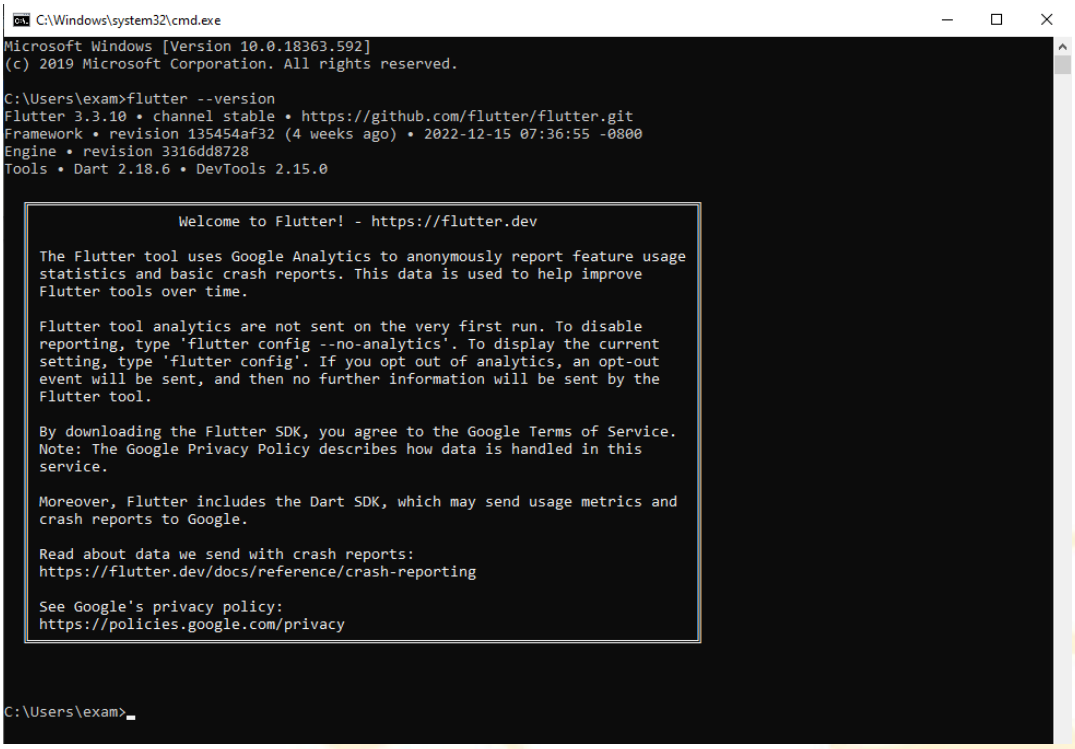


Step 7: Now install Dart in Visual Studio Code

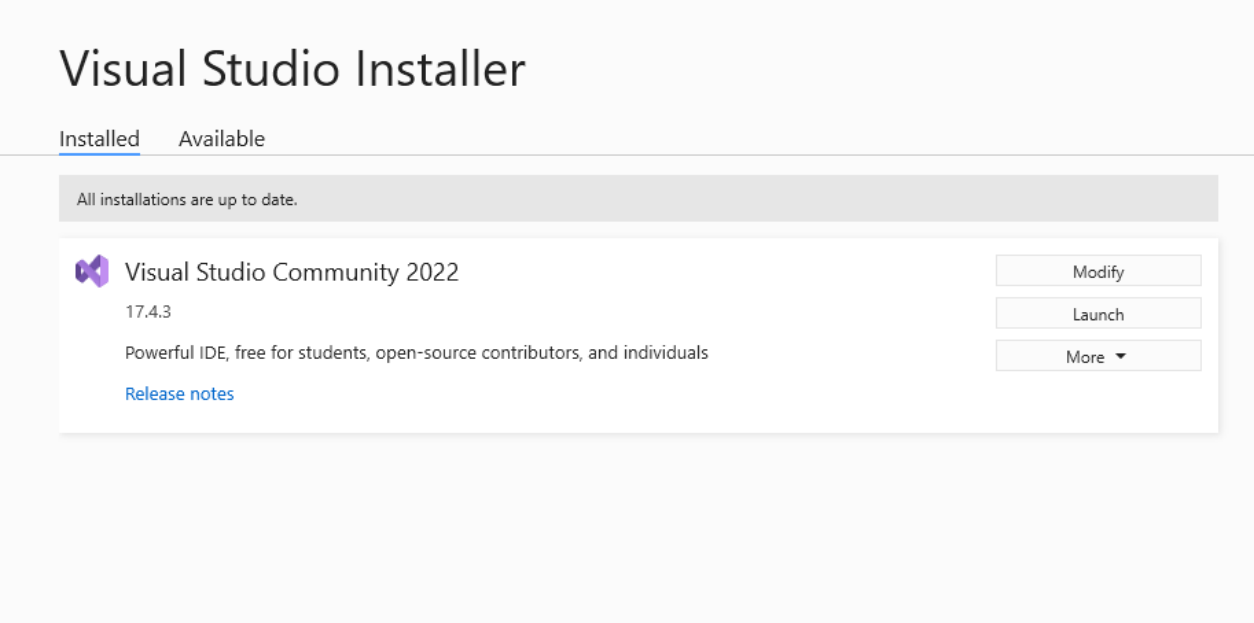


Step 8: Now we have successfully added Flutter and Dart to the Visual studio code, now let's check if flutter is installed or not. For this we will open a new terminal in Visual Studio Code and type the following "flutter –version", if everything is fine then it will

normally show the version of the installed flutter

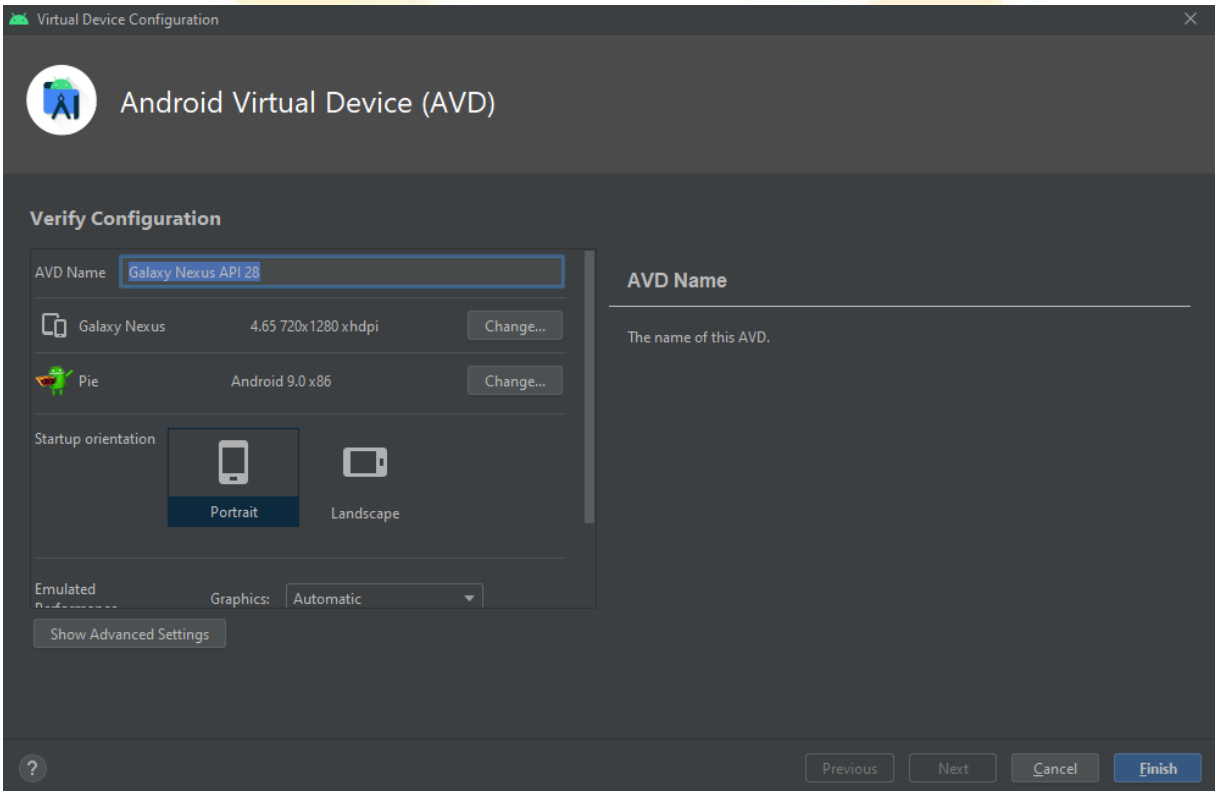


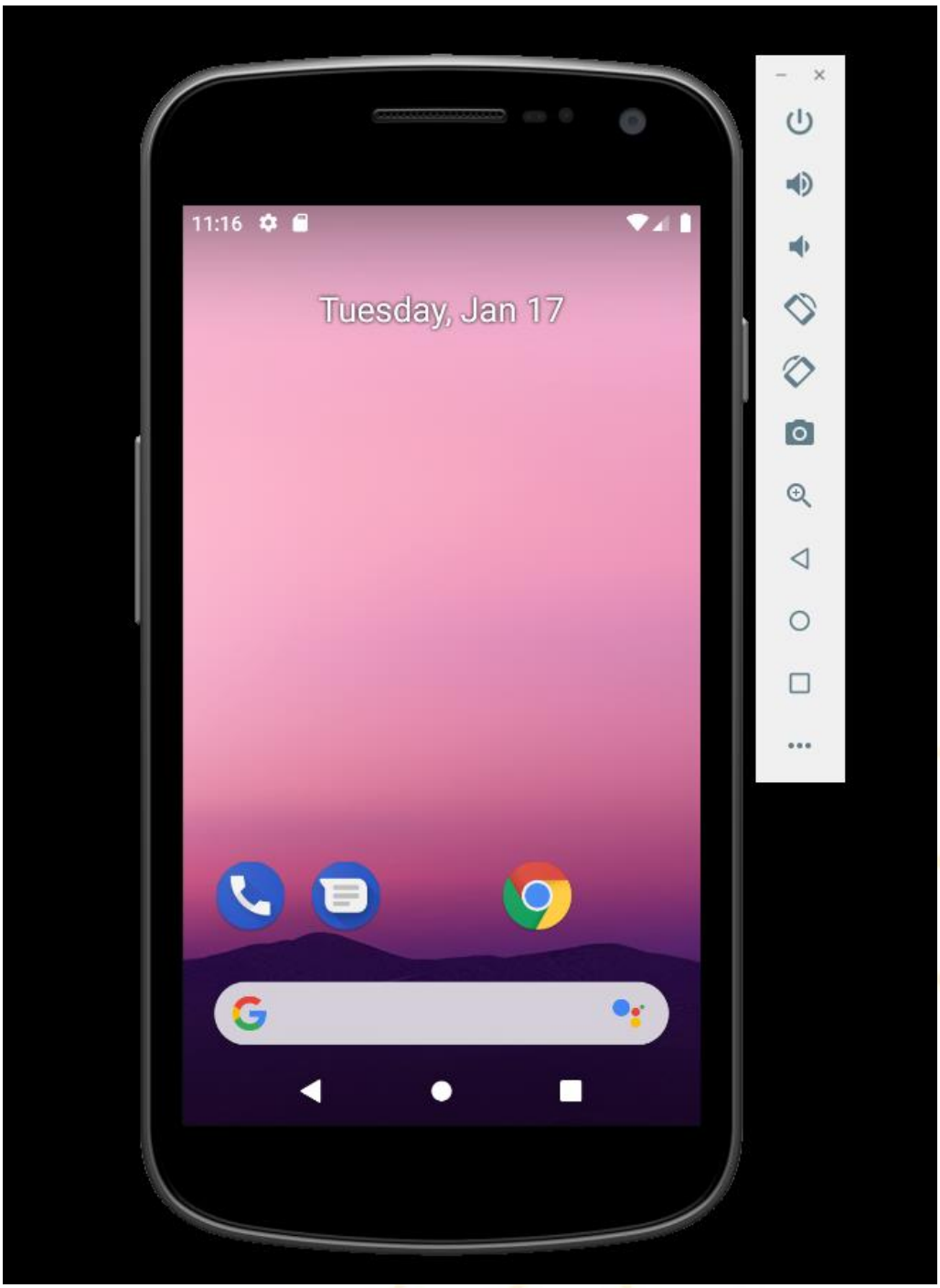
Step 9: Install Visual Studio



Step 10: Install and set up Android Studio

Step 11: Create a Virtual Device using Android Studio



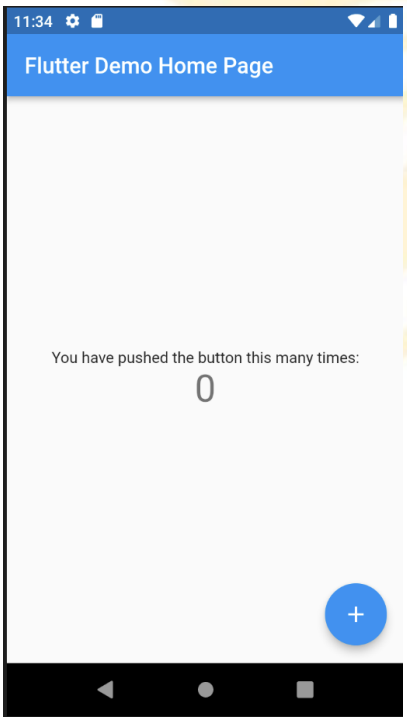


Step 11: Now for creating a new flutter project write the following in the visual studio Code terminal, "**flutter create sample**". After that project will be created inside the test project directory.

```
PS C:\Users\exam\Desktop\120A3051> cd basicWidgets
PS C:\Users\exam\Desktop\120A3051\basicWidgets> flutter create myapp
Wrote 127 files.

All done!
In order to run your application, type:

  $ cd myapp
  $ flutter run

Your application code is in myapp\lib\main.dart.

PS C:\Users\exam\Desktop\120A3051\basicWidgets> cd myapp
PS C:\Users\exam\Desktop\120A3051\basicWidgets\myapp> flutter run
Using hardware rendering with device AOSP on IA Emulator. If you notice graphics artifacts, consider enabling software rendering with "--enable-software-rendering".
Launching lib\main.dart on AOSP on IA Emulator in debug mode...
Running Gradle task 'assembleDebug'...                             32.1s
√  Built build\app\outputs\flutter-apk\app-debug.apk.
Installing build\app\outputs\flutter-apk\app.apk...              1,573ms
Syncing files to device AOSP on IA Emulator...                      83ms

Flutter run key commands.
r Hot reload.
R Hot restart.
h List all available interactive commands.
d Detach (terminate "flutter run" but leave application running).
c Clear the screen
q Quit (terminate the application on the device).

 Running with sound null safety

An Observatory debugger and profiler on AOSP on IA Emulator is available at: http://127.0.0.1:60297/cykR2PdRhgM=/
D/eglCodecCommon( 6699): setVertexArrayObject: set vao to 0 (0) 1 0
The Flutter DevTools debugger and profiler on AOSP on IA Emulator is available at: http://127.0.0.1:9101?uri=http://127.0.0.1:60297/cykR2PdRhgM=/
D/EGL_emulation( 6699): eglMakeCurrent: 0xdf105720: ver 2 0 (tinfo 0xdf103c90)


Taking screenshot for AOSP on IA Emulator...                       366ms
Screenshot written to flutter_01.png (41kB).

Performing hot reload...
Reloaded 0 libraries in 107ms (compile: 8 ms, reload: 0 ms, reassemble: 27 ms).
D/EGL_emulation( 6699): eglMakeCurrent: 0xdf105720: ver 2 0 (tinfo 0xdf103c90)
```

Output:

Now we are ready to see the output, by default it has a program in which there is a buttonbyclicking on the that a counter will be displayed in the centre.



We have also completed the steps for flutter installation in VSCode.