Getting Started with Flutter:

❖ Installation for windows Platform:

➢ System requirements:
To install and run Flutter, your development environment must meet these minimum requirements:

- Operating Systems: Windows 7 SP1 or later (64-bit), x86-64 based.
- Disk Space: 1.64 GB (does not include disk space for IDE/tools)
- Tools: Flutter depends on these tools being available in your environment.
  - Windows PowerShell 5.0 or newer (this is pre-installed with Windows 10)
  - Git for windows 2.x, with the **Use Git from the Windows Command Prompt** option

    if Git for windows is already installed, make sure you can run git commands from the command prompt or PowerShell.

❖ Get the Flutter SDK:

1. Download the following installation bundle to get the latest stable release of the Flutter SDK:

   https://flutter.dev/docs/development/tools/sdk/releases

   For the release channels, and older builds, see the SDK releases page.

2. Extract the zip file and place the contained flutter in the desired installation location for the Flutter SDK (for example C:\Users\<your-username>\Documents).

   Note – Do not install Flutter in a directory like C:\Program Files\ that requires elevated privileges.

3. If you don't want to install a fixed version of the installation bundle, you can skip steps 1 and 2. Instead, get the source code from the

Flutter repo on GitHub and change branches or tags as needed. For example:

https://github.com/flutter/flutter

You are now ready to run Flutter commands in the Flutter Console.

❖ Update your path:

➢ If you wish to run Flutter commands in the regular Windows console, take these steps to add Flutter to the PATH environment variable:

- From the Start search bar, enter 'env' and select **Edit environment variables for your account**.

- Under **User variables** check if there is an entry called **Path**:
  - If the entry exists, append the full path to flutter\bin using ; as a separator from existing values.
  - If the entry doesn't exist, create a new user variable named Path with the full path to flutter\bin as its value.

You have to close and reopen any existing console windows for these changes to take effect.

❖ Run flutter doctor:

From a console window that has the Flutter directory in the path (see above), run the following command to see if there are any platform dependencies you need to complete the setup:

```
C:\src\flutter>flutter doctor
```

This command checks your environment and displays a report of the status of your Flutter installation. Check the output carefully for other software you might need to install or further tasks to perform (shown in **bold** text).

For example:

```
[-] Android toolchain - develop for Android devices
    • Android SDK at D:\Android\sdk
    ✗ Android SDK is missing command line tools; download from https://goo.gl/XxQghQ
    • Try re-installing or updating your Android SDK,
      visit https://flutter.dev/setup/#android-setup for detailed instructions.
```

Note: If flutter doctor returns that either the Flutter plugin or Dart plugin of Android Studio are not installed, move on to **Set up an editor** to resolve this issue.

Android setup:

Flutter relies on a full installation of Android Studio to supply its Android platform dependencies. However, you can write your Flutter apps in several editors; a later step discusses that.

- **Install Android Studio:**
  - Download and install Android Studio.
  - Start Android Studio and go through the 'Android Studio Setup Wizard'. This installs the latest Android SDK, Android SDK Command-line Tools, and Android SDK Build-Tools, which are required by Flutter when developing for Android.
  - Run flutter doctor to conform that Flutter has located your installation of Android Studio. If Flutter cannot locate it, run flutter config –android-studio-dir <directory> to set the directory that Android Studio is installed to.

❖ Set up Your Android device:

- To prepare to run and test your Flutter app on an Android device, you need an Android device running Android 4.1 (API level 16) or higher.

  - Enable **Developer options** and **USB debugging** on your device. Detailed instructions are available in the Android documentation.

  - Windows-only: Install the Google USB Driver.

  - Using a USB cable, plug your phone into your computer. If prompted on your device, authorize your computer to access your device.

  - In the terminal, run the flutter devices command to verify that Flutter recognizes your connected Android device. By default, Flutter uses the version of the android SDK where your adb tool is based. If you want Flutter to use a different installation of the Android SDK, you must set the ANDROID_SDK_ROOT environment variable to that installation directory.

❖ Set up the Android emulator:

- To prepare to run and test your Flutter app on the Android emulator, follow these steps:
  - Enable VM acceleration on your machine.
  - Launch **Android Studio**, click the **AVD Manager** icon, and select **Create Virtual Device…**
    - In older versions of Android Studio, you should instead launch **Android Studio > Tools > Android > AVD Manager** and select **Create Virtual Device….** (The **Android** submenu is only present when inside an Android project.)
    - If you do not have a project open, you can choose **Configure > AVD Manager** and select **Create Virtual Device…**
  - Choose a device definition and select **Next**.
  - Select one or more system images for the Android versions you want to emulate and select **Next**. An x86 or x86_64 image is recommended.
  - Under Emulated Performance, select **Hardware – GLES 2.0** to enable hardware acceleration.
  - Verify the AVD configuration is correct and select **Finish**.

- In Android Virtual Device Manager, click **Run** in the toolbar. The emulator starts up and displays the default canvas for your selected OS version and device.

❖ Agree to Android Licenses:

➢ Before you can use Flutter, you must agree to the licenses of the Android SDK platform. This step should be done after you have installed the tools listed above.

- Make sure that you have a version of java 8 installed and that your JAVA_HOME environment variable is set to the JDK's folder.

Android Studio versions 2.2 and higher come with a JDK, so this should already be done.

- Open an elevated console window and run the following command to begin signing licenses.

```
$ flutter doctor --android-licenses
```

- Review the terms of each license carefully before agreeing to them.

- Once you are done agreeing with licenses, run flutter doctor again to conform that you are ready to use Flutter.