

# Flutter Windows installation

- **Get the Flutter SDK**
  1. Download the following installation bundle to get the latest stable release of the Flutter SDK: [flutter\\_windows\\_2.8.1-stable.zip](#)
  2. Extract the zip file and place the contained flutter in the desired installation location for the Flutter SDK (for example, C:\Users\\Documents).
  3. **Warning:** Do not install Flutter in a directory like C:\Program Files\ that requires elevated privileges.

# Flutter Windows installation

- **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.

# Flutter Windows installation

- **Run flutter doctor**
- From a console window that has the Flutter directory in the path, 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 X  
**Android SDK is missing command line tools; download from <https://goo.gl/XxQghQ>** • Try re-installing or updating your Android SDK, visit <https://docs.flutter.dev/setup/#android-setup> for detailed instructions.

# IDEs for Flutter app development

- There are many IDEs available for cross-platform mobile application development such as Android Studio, Visual Studio Code, Xcode, and IntelliJ. On which **Android Studio and Visual Studio Code are the most common IDEs for Flutter app development.**

# 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 confirm 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.
- Install plugins for your code editor:
  - [Flutter](#) and [Dart](#) plugins installed for Android Studio.
  - [Flutter](#) extension installed for Visual Studio Code.
-

# 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**.

# Create First Flutter app

- Open the IDE and select **Create New Flutter Project**.
- Select **Flutter Application** as the project type. Then click **Next**.
- Verify the Flutter SDK path specifies the SDK's location (select **Install SDK...** if the text field is blank).
- Enter a project name (for example, myapp). Then click **Next**.
- Click **Finish**.
- Wait for Android Studio to install the SDK and create the project.