Download Flutter SDK

**Flutter**is a **Mobile UI Framework**, an open source created by **Google**. Its purpose is to create applications that can run on a different variety of platforms such as **Android, iOS**and **Web**.

**Google** offers three channels for you to download **Flutter** depending on your purpose.

**Stable Channel**

**Stable Channel** allows you to download the stable version of **Flutter** at the product level. You can use it without worrying about bugs in the code.

**Beta Channel**

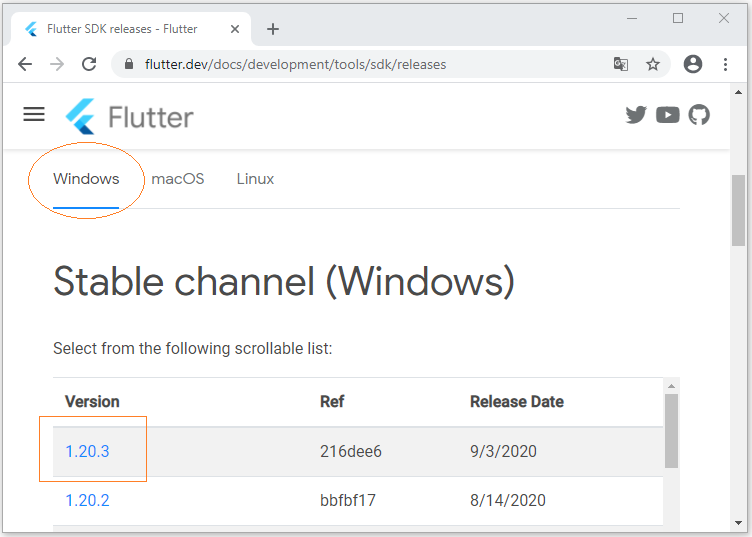
**Beta Channel** allows you to download the version of **Flutter** with the latest features. However, it has not been officially released yet because it takes more time to test for its stability. You can download this version if you want to learn about the new features in advance.

**Dev Channel**

**Dev Channel** allows you to download the version of **Flutter** with all the features being developed by the **Flutter**design team. Nonetheless, this version is constantly changing and unstable.

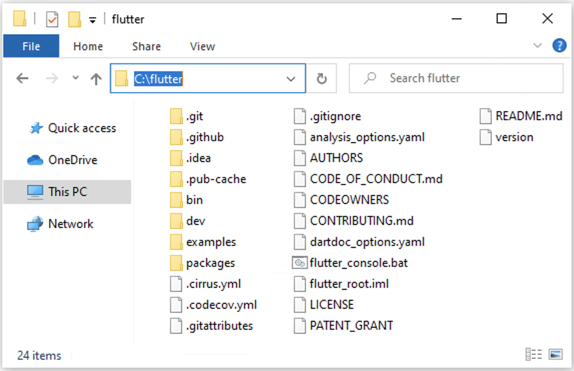
To download **Flutter**,visit the link below.

* <https://flutter.dev/docs/development/tools/sdk/releases>

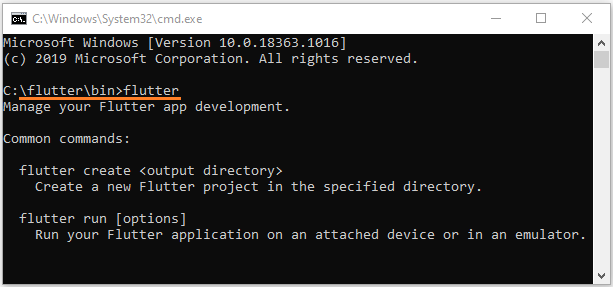
2. Installing Flutter SDK

Firstly, extract the file that you have downloaded in the above step to a certain folder, for example:

* **C:/fluter**



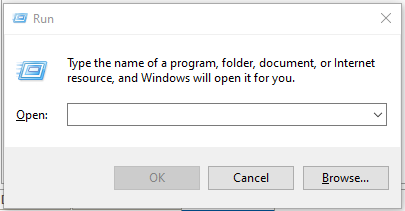
Next, open **CMD** window and go to **C:/flutter/bin**. Enter the **"flutter"** command and press **Enter**.



3. Add an environment variable

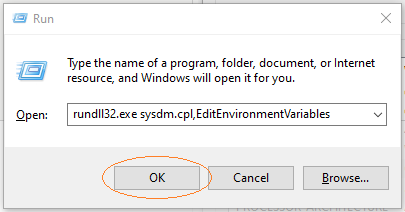
You also need to add the **Flutter**path to the Environment Variable, which is necessary for you to use **Flutter SDK** from anywhere on your computer.

Then press **WIN + R** to open the **RUN**window.

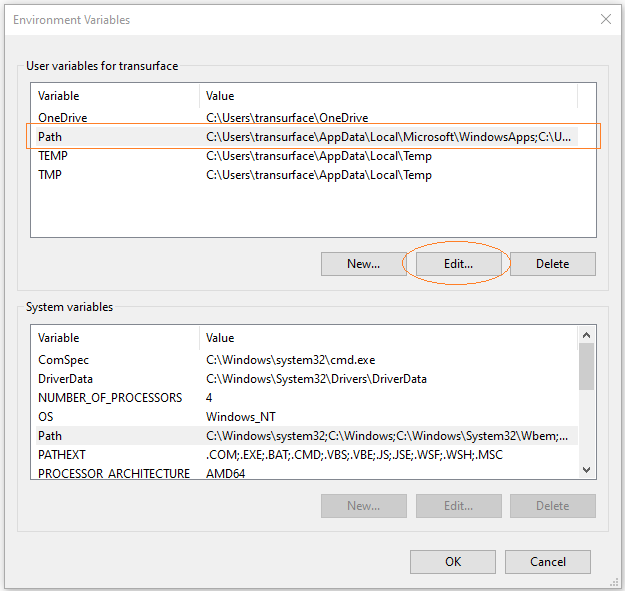


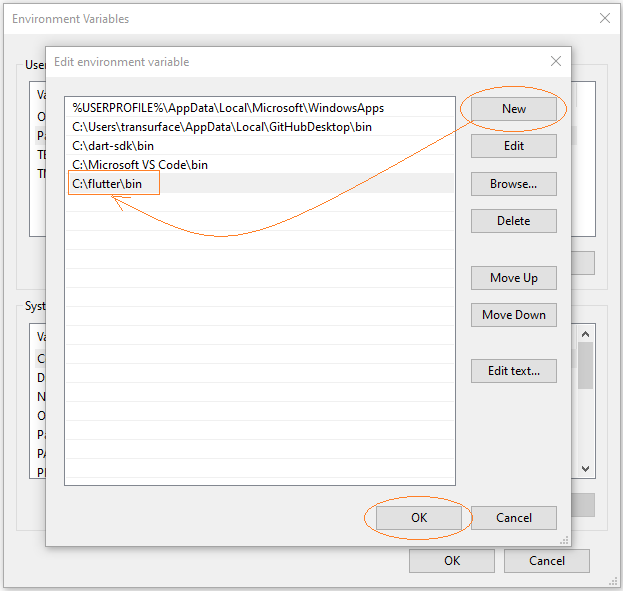
After that, enter the following command to open the**Environment Variables**window:

rundll32.exe sysdm.cpl,EditEnvironmentVariables



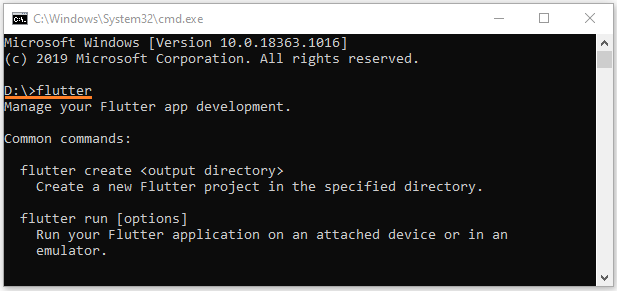
Later, select the **"Path"** variable and press **"Edit"**:





Now you can use the "**flutter**" command from any folder on your computer.

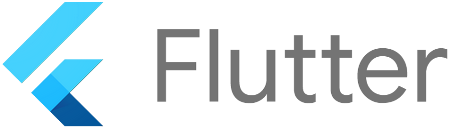
Note: You may need to restart your computer for the changes to take effect.



## Flutter Plugin

As you all know**, Android Studio** is an **IDE** used to develop **Android**apps and created by **Google**. You can also use **Android Studio** in order to learn **Dart** language or develop **Flutter** apps, which requires you to install some more related plugins. All thee, **Android Studio**, **Dart** and **Flutter** are created by **Google**; therefore, **Android Studio** is definitely the most appropriate **IDE** to develop the **Flutter** apps due to its greatest support.

**Flutter** is a **Mobile UI Framework**. It uses **Dart** language to write apps, so make sure you have already installed **Dart SDK**:



Download Dart

**.**

* *Introduction to Dart*

**Google** offers three channels for you to download **Dart** depending on your goals.

**Stable Channel**

**Stable Channel** allows you to download a stable version of **Dart** at a product level. You can use it without worrying about bugs in the code.

**Beta Channel**

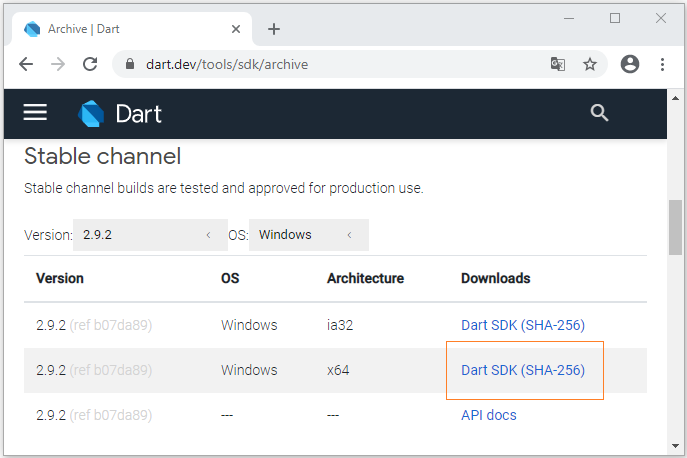
**Beta Channel** allows you to download the version of **Dart**with the newest features which have not officially released yet because of the needs of more time for stability test. You can download this version if you desire to learn about the new features in advance.

**Dev Channel**

**Dev Channel**allows you to download the **Dart**version with all the features being developed by the **Dart**design team. This version is constantly changing and not stable.

To download **Dart**, access the link below.

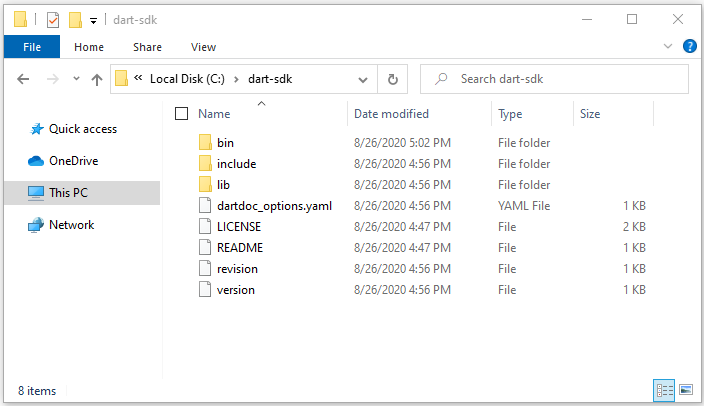
* <https://dart.dev/tools/sdk/archive>



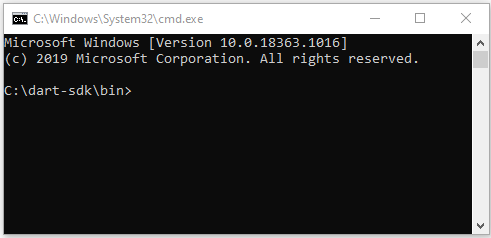
After downloading it successfully, you will get a **zip**file.

2. Dart setting

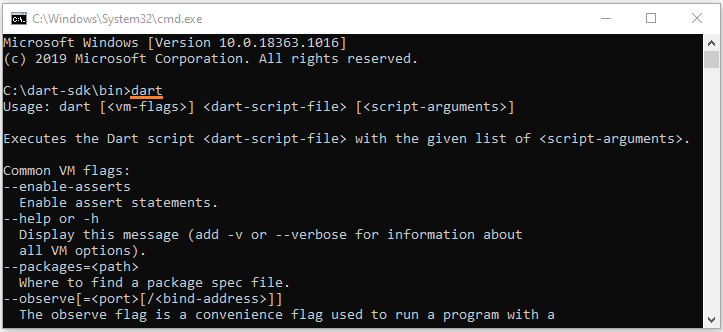
Firstly, extract the **zip** file that you have already downloaded. For example, extract the **C:/dart-sdk**folder:



Next, open the **bin**folder of the **dart-sdk** in **Windows Explorer**. Enter the keyword **"CMD"**in the address bar and press **Enter**.A **CMD** window will open thereafter.



Then enter **"dart"** in the **CMD**window and press **Enter**. You will get a list of arguments of the**"dart"**command and their usage.



C:\dart-sdk\bin>dart

Usage: dart [<vm-flags>] <dart-script-file> [<script-arguments>]

Executes the Dart script <dart-script-file> with the given list of <script-arguments>.

Common VM flags:

--enable-asserts

Enable assert statements.

--help or -h

Display this message (add -v or --verbose for information about

all VM options).

--packages=<path>

Where to find a package spec file.

--observe[=<port>[/<bind-address>]]

The observe flag is a convenience flag used to run a program with a

set of options which are often useful for debugging under Observatory.

These options are currently:

--enable-vm-service[=<port>[/<bind-address>]]

--pause-isolates-on-exit

--pause-isolates-on-unhandled-exceptions

--warn-on-pause-with-no-debugger

This set is subject to change.

Please see these options (--help --verbose) for further documentation.

--write-service-info=<file\_uri>

Outputs information necessary to connect to the VM service to the

specified file in JSON format. Useful for clients which are unable to

listen to stdout for the Observatory listening message.

--snapshot-kind=<snapshot\_kind>

--snapshot=<file\_name>

These snapshot options are used to generate a snapshot of the loaded

Dart script:

<snapshot-kind> controls the kind of snapshot, it could be

kernel(default) or app-jit

<file\_name> specifies the file into which the snapshot is written

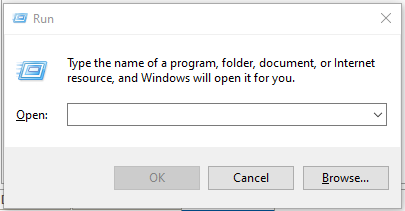
--version

Print the SDK version.

3. Adding environment variable

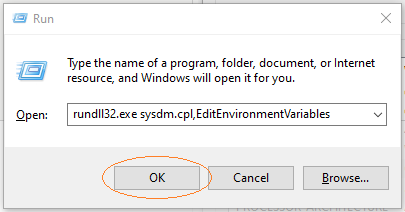
You also need to add the **Dart** path to the Environment Variable, which is necessary for you to use the **Dart SDK**from anywhere on your computer.

Press **WIN + R** to bring up the **RUN**window.

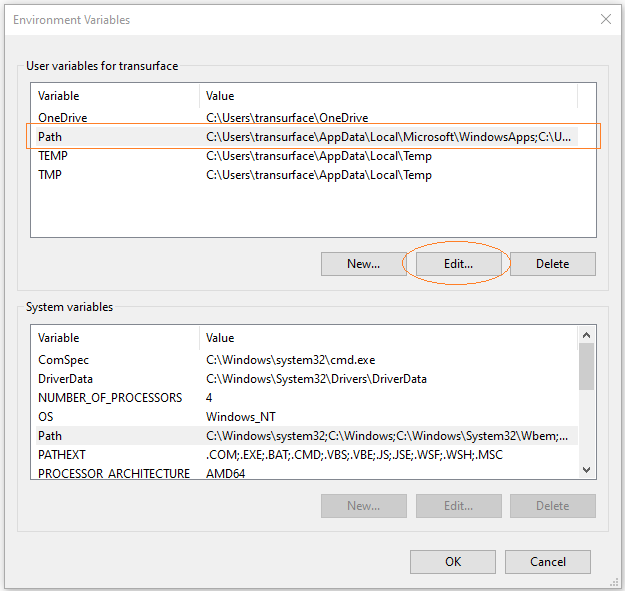


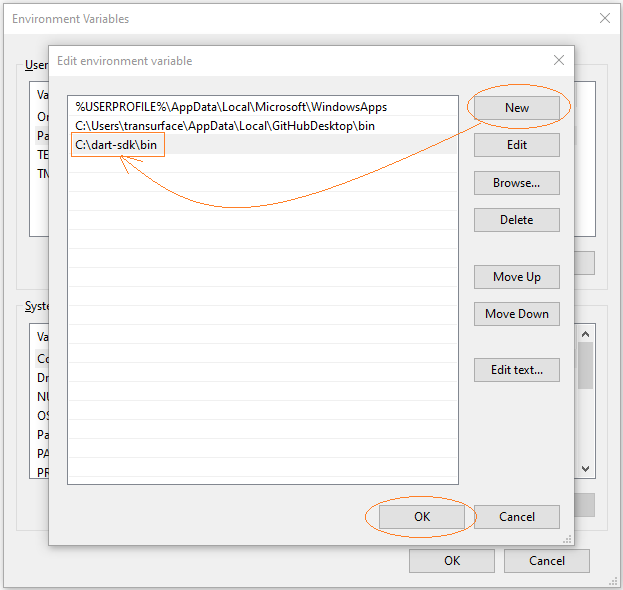
Enter the command below to open the **Environment Variables**window:

rundll32.exe sysdm.cpl,EditEnvironmentVariables



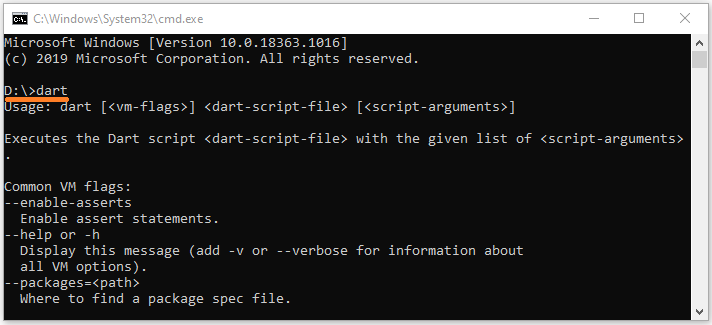
Select **"Path"** variable and press **"Edit"**:





Now you can use the **"dart"** command from any folders on your computer.

Note: You may need to restart your computer for the changes to take effect.



4. Write your first Dart example

And **Dart Plugin** in **Android Studio**:

# **Install Dart Plugin for Android Studio**

## Dart Plugin

As we all know,**Android Studio** is an **IDE** created by **Google**. Its initial purpose is to develop some **Android** apps, but currently, **Android Studio** permits you to install plugins for programming other languages such as **Dart**.

**Android Studio** is a good choice to learn the **Dart** language and build **Flutter**apps. All of these three products are created by **Google,**therefore, you will receive the best support.

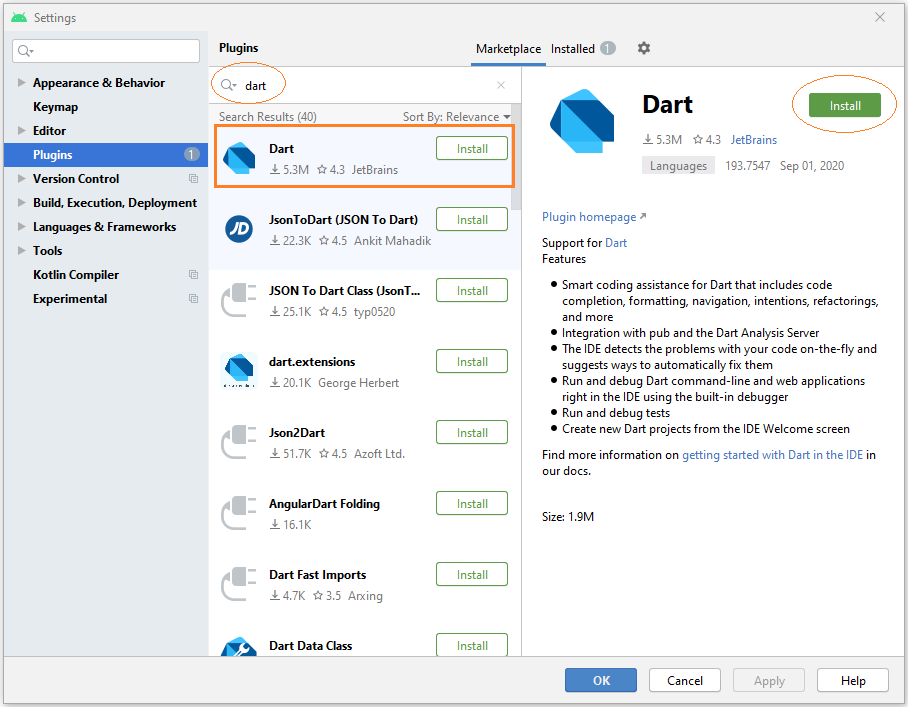
* [*Install Android Studio on Windows*](https://o7planning.org/10405/install-android-studio-on-windows)

In this article I'm going to guide you to install the **Dart Plugin** in **Android Studio**.

Firstly, open **Android Studio** and select:

* **File > Settings... > Plugins**

Find the keyword **"Dart"**:



After the installation is complete, you need to restart**Android Studio** for the changes to take effect

Next, Install **Flutter SDK**:

# **Install Flutter SDK on Windows**

1. Download Flutter SDK
2. Installing Flutter SDK
3. Add an environment variable

## 1. Download Flutter SDK

**Flutter**is a **Mobile UI Framework**, an open source created by **Google**. Its purpose is to create applications that can run on a different variety of platforms such as **Android, iOS**and **Web**.

**Google** offers three channels for you to download **Flutter** depending on your purpose.

**Stable Channel**

**Stable Channel** allows you to download the stable version of **Flutter** at the product level. You can use it without worrying about bugs in the code.

**Beta Channel**

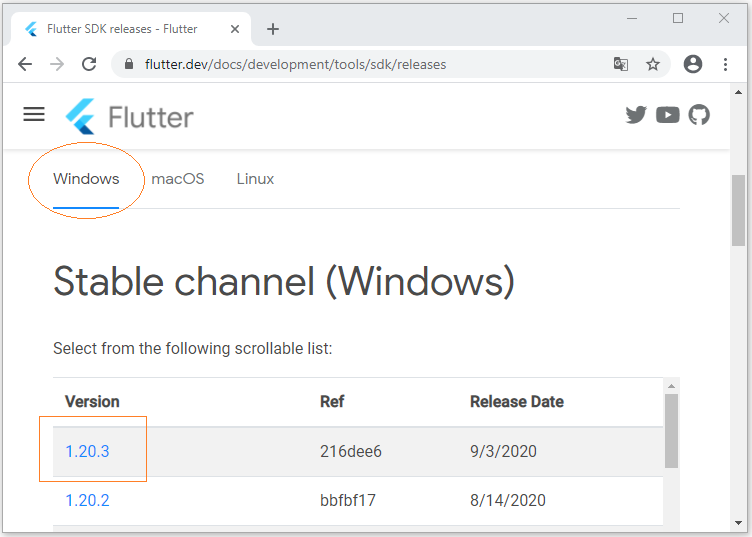
**Beta Channel** allows you to download the version of **Flutter** with the latest features. However, it has not been officially released yet because it takes more time to test for its stability. You can download this version if you want to learn about the new features in advance.

**Dev Channel**

**Dev Channel** allows you to download the version of **Flutter** with all the features being developed by the **Flutter**design team. Nonetheless, this version is constantly changing and unstable.

To download **Flutter**,visit the link below.

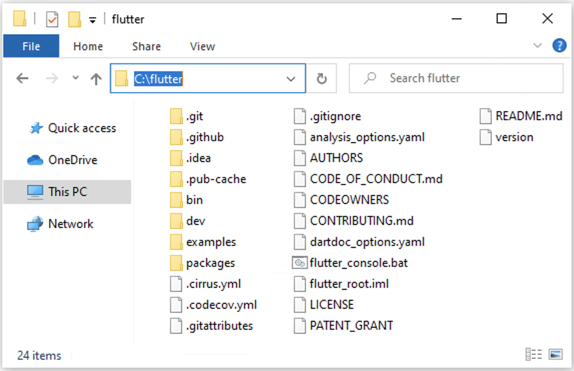
* <https://flutter.dev/docs/development/tools/sdk/releases>



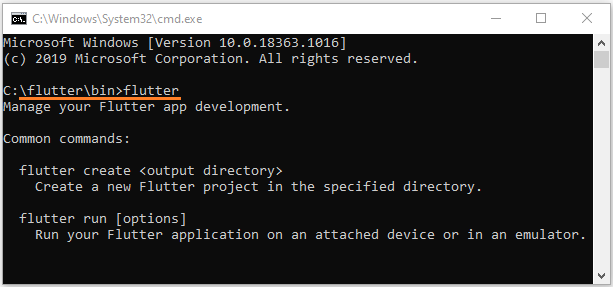
## 2. Installing Flutter SDK

Firstly, extract the file that you have downloaded in the above step to a certain folder, for example:

* **C:/fluter**



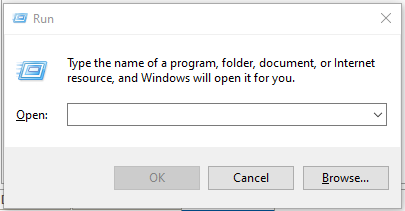
Next, open **CMD** window and go to **C:/flutter/bin**. Enter the **"flutter"** command and press **Enter**



## 3. Add an environment variable

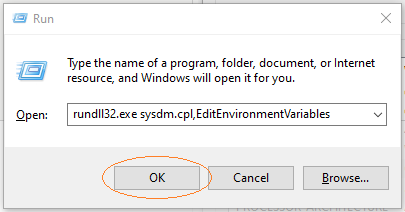
You also need to add the **Flutter**path to the Environment Variable, which is necessary for you to use **Flutter SDK** from anywhere on your computer.

Then press **WIN + R** to open the **RUN**window.

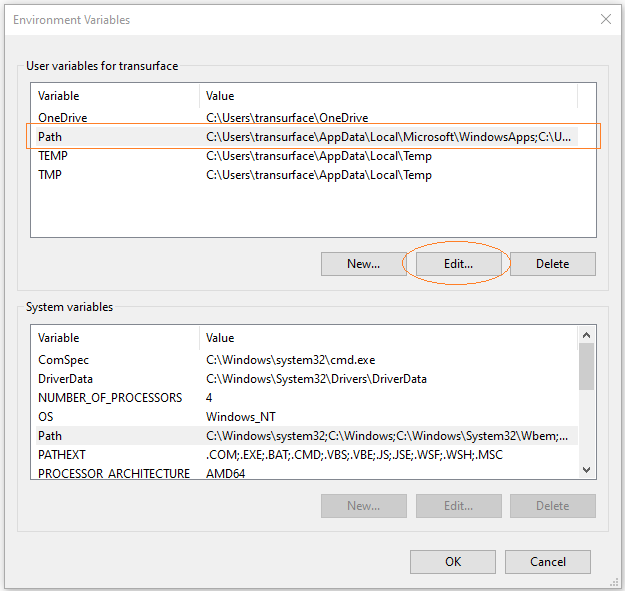


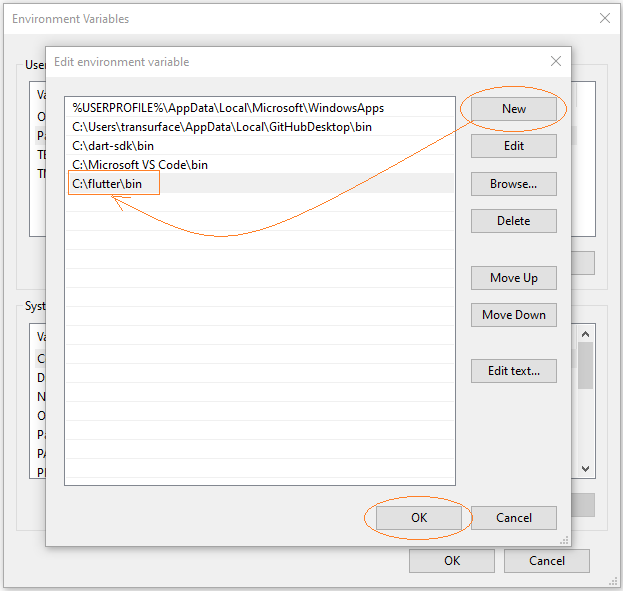
After that, enter the following command to open the**Environment Variables**window:

rundll32.exe sysdm.cpl,EditEnvironmentVariables



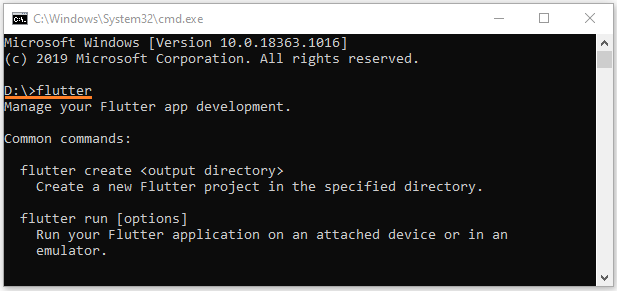
Later, select the **"Path"** variable and press **"Edit"**





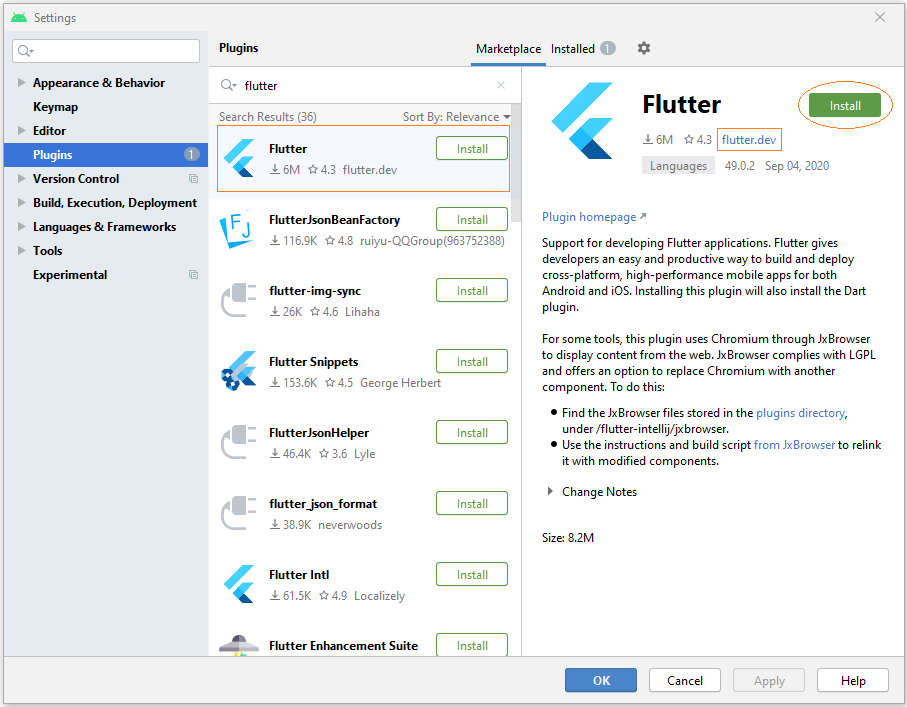
Now you can use the "**flutter**" command from any folder on your computer.

Note: You may need to restart your computer for the changes to take effect.



In order to install **Flutter Plugin**, first of all, open **Android Studio** and select:

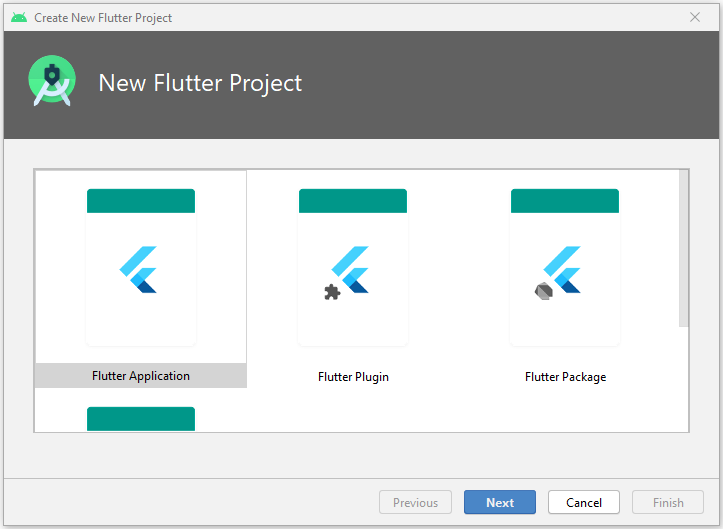
* **File > Settings... > Plugins** (Windows)
* **Preferences > Plugins** (Mac OS)



After installing **Flutter Plugin** successfully, you need to restart **Android Studio** for changes to take effect.

Finally, And check to make sure the functionality to create a new **Flutter** project is available on **Android Studio**.

* **File > New > New Flutter Project...**



## 2. Creating your first Flutter app

# **Create your first Flutter app - Hello Flutter**

1. Article objectives
2. Creating Flutter project
3. Project structure explanation
4. Writing code for application
5. Run the application

## 1. Article objectives

In this article, I'm going to guide you to create your first **Flutter** application on **Android Studio** and run the app with **Android Emulator**successfully.

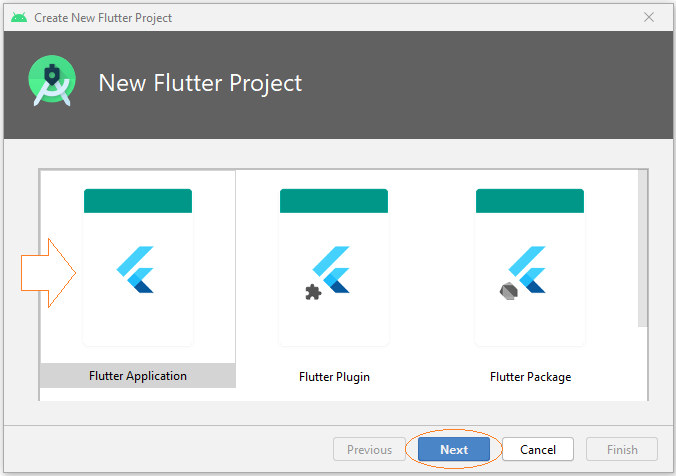
First of all, make sure that you have successfully installed the following necessary tools:

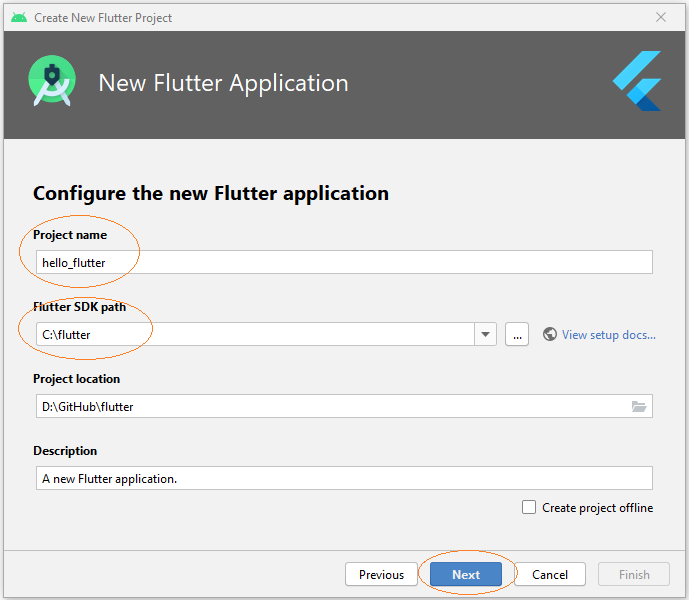
* [Install Android Studio on Windows](https://o7planning.org/10405/install-android-studio-on-windows)
* [Install Flutter Plugin for Android Studio](https://o7planning.org/12829/install-flutter-plugin-for-android-studio)

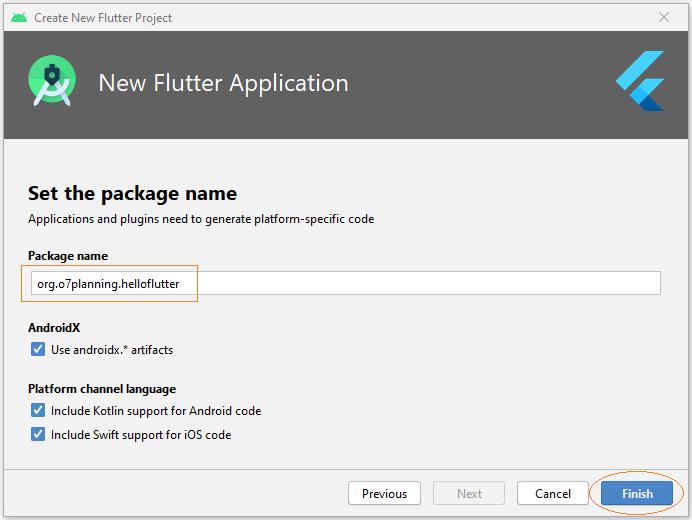
## 2. Creating Flutter project

On **Android Studio**, create a **Flutter** project.

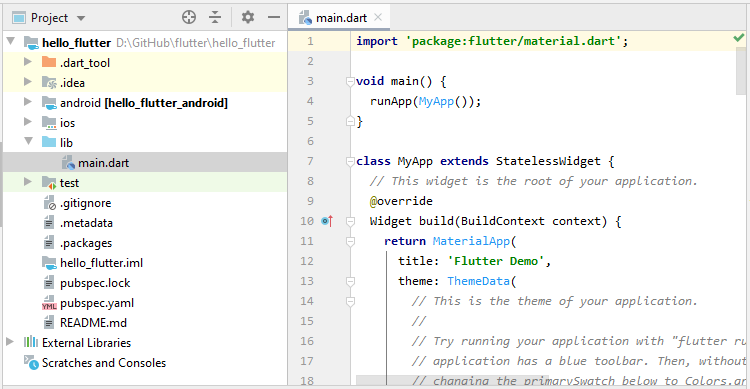
* **File > New > New Flutter Project...**







The project has been successfully created. Here is its structure:



## 3. Project structure explanation

**android**

The folder automatically generates code for the **Android** app.

**ios**

The folder automatically generates code for the **iOS**app.

**lib**

The home folder contains the **Dart** code of the app.

**lib/main.dart**

The file will be called to start the app.

**test**

The folder contains **Dart** codes used to test the app.

**test/widget\_test.dart**

Sample code

**.gitignore**

**Git version control file** - This file contains the configuration of the **GIT** project.

**.metadata**

The folder is automatically generated by the **Flutter** tool.

**.packages**

The file is automatically generated which contains a list of dependencies being used by the project.

**.iml**

The project file of **Android Studio**.

**pubspec.yaml**

The file is used to declare resources related to the project such as images, fonts, etc.

**pubspec.lock**

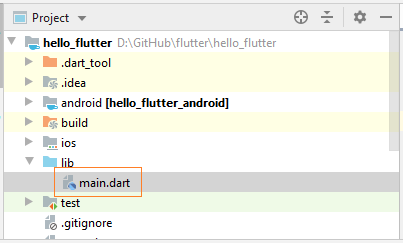
This file should be added to **GIT Control** to ensure that your development team members use the same library versions.

**README.md**

The file describes the project, which is written according to the **Markdown** structure.

## 4. Writing code for application

Delete all the contents of the **main.dart** file and replace them with new ones.



lib/main.dart

**import** 'package:flutter/material.dart';

**void** main() => runApp(MyApp());

**class** **MyApp** **extends** **StatelessWidget** {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

**return** MaterialApp(

title: 'Hello World Demo Application',

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: MyHomePage(title: 'Home page'),

);

}

}

**class** **MyHomePage** **extends** **StatelessWidget** {

MyHomePage({Key key, **this**.title}) : **super**(key: key);

**final** String title;

@override

Widget build(BuildContext context) {

**return** Scaffold(

appBar: AppBar(

title: Text(**this**.title),

),

body: Center(

child:

Text(

'Hello World',

)

),

);

}

}

5. Run the application

The **Flutter** app needs to be deployed to an **Android**or **iOS** device to run. Therefore, during the app development process, you have one of the following options:

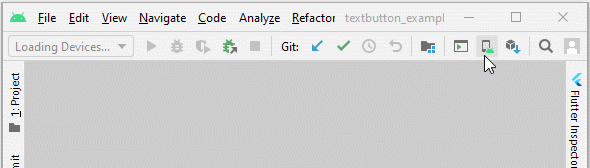
* Connect your physical **Android**device to your computer, and turn on Developer mode.
* Connect your physical **iOS** device (such as an **iPhone**) to your computer, and turn on Developer mode.
* Run an **Android Emulator**.

We are programming the **Flutter** application on **Android Studio**, sothe best approach is to run an **Android Emulator**.

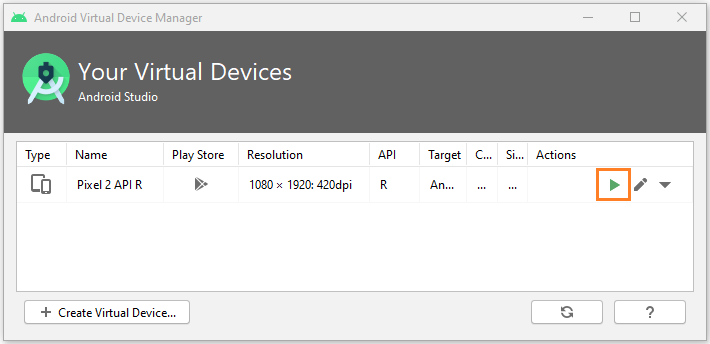
Now on **Android Studio**, select:

* **Tools > AVD Manager**

Or click on the **"AVD Manager"** icon on the toolbar:



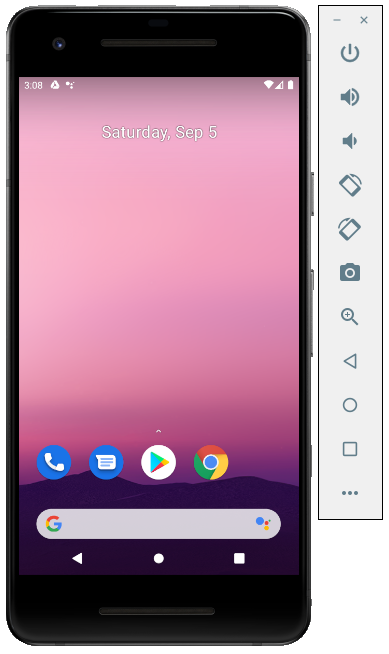
Then launch a virtual device on the list:



*If you do not see any virtual device on the list, create one by following the instruction below:*

* [*Configure Android Emulator in Android Studio*](https://o7planning.org/10413/configure-android-emulator-in-android-studio)

**Android Emulator** is launched and get ready for deploying the **Flutter**app.



On the **Android Studio**toolbar, run your **Flutter** app as what you see in the illustration below:



Here is the result you get:

