Flutter Installation Guide

Kahkashan Anjum
01-134222-072
6B-BSCS

March 8, 2025

Contents

1 Main Steps to Install Flutter 3

1.1 Download Flutter SDK 3

1.2 Set Up Environment Variables 3

1.3 Install Dependencies 3

1.4 Run Flutter Doctor 3

1.5 Set Up an Emulator or Physical Device 4

1.6 Install a Code Editor (Recommended) 4

2 Connecting a Physical Device Instead of a Virtual Device 4

2.1 Enable Developer Options 4

2.2 Enable USB Debugging 4

2.3 Connect Your Phone to Your Computer 4

2.4 Check Device Connection 5

3 Extensions to Install in VS Code 5

4 Creating a Flutter Project 5

4.1 Using VS Code 5

4.2 Using Terminal 5

5 Advantages of Setting the PATH in Environment Variables 6

1 Main Steps to Install Flutter

To install Flutter, follow these steps:

1.1 Download Flutter SDK

- Visit the official Flutter website: `https://flutter.dev/docs/get-started/install`
- Download the latest stable Flutter SDK for your OS (Windows, macOS, Linux)
- Extract the downloaded file to a suitable location

1.2 Set Up Environment Variables

- Add the Flutter bin directory to your system's PATH variable:
  - Windows: Edit system environment variables and add `<flutter_directory>\bin` to Path.
  - macOS/Linux: Add `export PATH="$PATH:<flutter_directory>/bin"` to your shell profile (.bashrc, .zshrc, .bash_profile)

1.3 Install Dependencies

Windows: Install Git, PowerShell, and necessary Windows tools. **macOS:** Install Xcode, Homebrew, and dependencies:

```
brew install cocoapods
```

Linux: Install necessary libraries:

```
sudo apt-get install clang cmake ninja-build pkg-config
   libgtk-3-dev
```

1.4 Run Flutter Doctor

Open a terminal or command prompt and run:

```
flutter doctor
```

Install any missing dependencies suggested in the output.

1.5 Set Up an Emulator or Physical Device

For Android:

- Install Android Studio
- Set up the Android SDK and virtual device
- Accept Android licenses using:

```
    flutter doctor --android-licenses

```

For iOS (macOS only): Install Xcode and set up an iOS simulator.

1.6 Install a Code Editor (Recommended)

VS Code: Install the Flutter and Dart plugins. **Android Studio:** Install the Flutter and Dart plugins from the plugins marketplace.

2 Connecting a Physical Device Instead of a Virtual Device

To use your mobile phone instead of an emulator:

2.1 Enable Developer Options

1. Open **Settings** on your phone.
2. Go to **About Phone**.
3. Find **Build Number** and tap it **7 times** until you see ”You are now a developer!”

2.2 Enable USB Debugging

- Go back to **Settings** →Developer Options.
- Turn on **USB Debugging**.

2.3 Connect Your Phone to Your Computer

- Use a USB cable to connect your phone.
- If prompted, **Allow USB Debugging**.

2.4 Check Device Connection

Open a terminal and run:

```
flutter devices
```

If your phone is listed, it is connected successfully.

3 Extensions to Install in VS Code

- **Flutter Extension**: Provides Flutter support in VS Code.
- **Dart Extension**: Provides syntax highlighting, code completion, and debugging tools for Dart.

4 Creating a Flutter Project

4.1 Using VS Code

1. Open VS Code.
2. Press **Ctrl + Shift + P** to open the command palette.
3. Type **"Flutter: New Project"** and select it.
4. Choose **"Flutter Application"** as the project type.
5. Select a folder and enter a project name.
6. Wait for the setup to complete.

4.2 Using Terminal

```
cd path/to/your/folder
flutter create my_app
cd my_app
flutter run
```

5 Advantages of Setting the PATH in Environment Variables

- **Run Flutter Commands from Anywhere:** Use commands like `flutter run` without navigating to the Flutter directory.
- **Saves Time and Effort:** No need to type long file paths.
- **Avoids Errors:** Prevents ”Command not found” issues.
- **Makes Development Easier:** Helps integrate Flutter with VS Code and Android Studio.
- **Works with Other Tools:** Allows seamless use of Dart SDK, `adb`, and `flutter upgrade`.