

#2 Installing Flutter and Setting up Environment for Coding using VS Code

Introduction

This guide will walk you through the process of installing Flutter and setting up your environment for coding using Visual Studio Code. So, let's get started! 🚀

Installing Flutter

Downloading Flutter SDK

The first step is to download the Flutter SDK from the official Flutter website. Once downloaded, extract the contents of the zip file to a desired location on your machine.

Setting up environment variables

Next, you need to add the Flutter tool to your path. To do this, you can add the following line to your shell configuration file (for example, `~/.bashrc` or `~/.zshrc`):

```
export PATH="$PATH:[PATH_TO_FLUTTER_GIT_DIRECTORY]/flutter/bin"
```

Testing the installation

To verify that Flutter is installed correctly, run the following command in your terminal:

```
flutter doctor
```

This command will check for any dependencies you may need to install, and provide you with a summary of the current state of your development environment.

Here is an example result from running the "flutter doctor" command:

```
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 2.2.3, on macOS 11.5.1 20G80 darwin-x64, locale en-US)
[✓] Android toolchain - develop for Android devices (Android SDK version 31.0.0)
[✓] Xcode - develop for iOS and macOS
[✓] Chrome - develop for the web
[✓] Android Studio (version 4.1)
[✓] VS Code (version 1.60.2)
[✓] Connected device (2 available)

• No issues found!
```

This output shows that Flutter is installed correctly, along with the Android toolchain, Xcode, Chrome, Android Studio, and VS Code. It also indicates that there are two connected devices available for running the app. The message "No issues found!" at the end of the output confirms that everything is set up properly and you can start coding with Flutter.

Setting up Visual Studio Code

Installing Dart and Flutter extensions

To start coding with Flutter, you will need to install the Dart and Flutter extensions for Visual Studio Code. To do this, open the extensions view in VS Code by clicking on the extension icon in the activity bar on the side of the window or by pressing `Ctrl+Shift+X` (Windows, Linux) or `Cmd+Shift+X` (macOS).

In the search bar, type "Dart" and click on the "Install" button next to the "Dart" extension by Dart Code. Repeat the same process for the "Flutter" extension by Flutter Team.

Creating a new Flutter project

To create a new Flutter project, open the command palette in VS Code by pressing `Ctrl+Shift+P` (Windows, Linux) or `Cmd+Shift+P` (macOS). In the search bar, type "Flutter" and select "Flutter: New Project". Follow the prompts to give your project a name and select a directory to save it in.

<https://www.loom.com/embed/d45c3a3ffcb64a4aa12a876877ca3a0b>

Running the project on an emulator or device

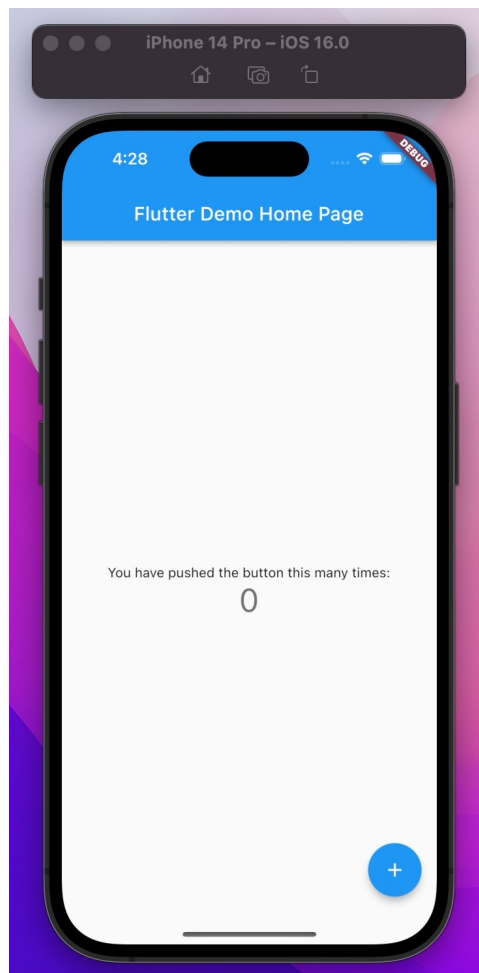
Once your project is created, you can run it on an emulator or device. To do this, open the command palette again and type "Flutter" and select "Flutter: Run". Follow the prompts to select the device or emulator you want to run the app on.

To run or debug your Flutter app in Visual Studio Code, you can use the following keyboard shortcuts:

- **Windows:**
 - `F5` to start debugging
 - `Ctrl+F5` to start without debugging
 - `Shift+F5` to stop debugging
- **macOS:**
 - `F5` to start debugging
 - `Control+F5` to start without debugging
 - `Shift+F5` to stop debugging
- **Linux:**
 - `F5` to start debugging
 - `Ctrl+F5` to start without debugging
 - `Shift+F5` to stop debugging

As a suggestion, you can also use the VS Code Run and Debug panel to start, stop and manage your Flutter app. To access this panel, click on the Run and Debug icon in the left sidebar of the VS Code window. From the panel, you can select your Flutter app and

choose to run, debug or stop it. You can also add breakpoints, inspect variables, and more.



Conclusion

Congratulations! You have successfully installed Flutter and set up your environment for coding using Visual Studio Code. Now you can start building beautiful and performant mobile apps with Flutter. Happy coding! ⚡