Run “flutter doctor“ before you upgrade

Open a new VS Code. In case it automatically opens a project, close it with menu “File/Close Folder“, open a new Terminal with menu “Terminal/New Terminal“ and enter “**flutter doctor**“:

```
PS C:\Users\GS> flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[√] Flutter (Channel stable, 3.24.3, on Microsoft Windows [Version 10.0.19045.5737], locale de-DE)
[√] Windows Version (Installed version of Windows is version 10 or higher)
[√] Android toolchain - develop for Android devices (Android SDK version 34.0.0)
[√] Chrome - develop for the web
[√] Visual Studio - develop Windows apps (Visual Studio Community 2019 16.11.2)
[√] Android Studio (version 2024.2)
[√] VS Code (version 1.102.0)
[√] Connected device (3 available)
[√] Network resources

• No issues found!
```

Don’t worry in case you have some issues here. After the “flutter upgrade“, “flutter doctor“ is automatically executed again, and then you should not have more issues compared to this call before the Upgrade.

Run “flutter upgrade”

Now run “**flutter upgrade**” in the Terminal. On my PC I saw the following output:

```
PS C:\Users\GS> flutter upgrade
Upgrading Flutter to 3.32.6 from 3.24.3 in C:\flutter\sdk...
Checking Dart SDK version...
Downloading Dart SDK from Flutter engine 72f2b18bb094f92f62a3113a8075240ebb59affa...
Expanding downloaded archive with PowerShell...
Building flutter tool...
Running pub upgrade...
Resolving dependencies...
Downloading packages...
Got dependencies.

Upgrading engine...
Downloading android-arm-profile/windows-x64 tools...                581ms
Downloading android-arm-release/windows-x64 tools...                402ms
Downloading android-arm64-profile/windows-x64 tools...              469ms
Downloading android-arm64-release/windows-x64 tools...              428ms
Downloading android-x64-profile/windows-x64 tools...                465ms
Downloading android-x64-release/windows-x64 tools...                422ms
Downloading android-x86 tools...                                    14,7s
Downloading android-x64 tools...                                    15,3s
```

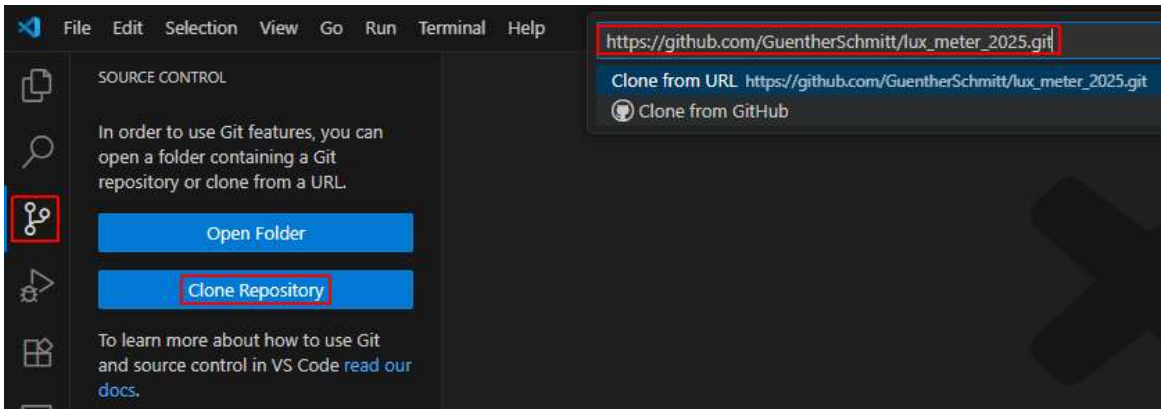
…

```
Downloading windows-x64-release/windows-x64-flutter tools...         9,9s
Downloading windows-x64/font-subset tools...                        264ms

Flutter 3.32.6 • channel stable • https://github.com/flutter/flutter.git
Framework • revision 077b4a4ce1 (4 days ago) • 2025-07-08 13:31:08 -0700
Engine • revision 72f2b18bb0 (4 days ago) • 2025-07-08 10:33:53 -0700
Tools • Dart 3.8.1 • DevTools 2.45.1

Running flutter doctor...
Doctor summary (to see all details, run flutter doctor -v):
[√] Flutter (Channel stable, 3.32.6, on Microsoft Windows [Version 10.0.19045.5737], locale de-DE)
[√] Windows Version (10 Pro 64-bit, 22H2, 2009)
[√] Android toolchain - develop for Android devices (Android SDK version 34.0.0)
[√] Chrome - develop for the web
[√] Visual Studio - develop Windows apps (Visual Studio Community 2019 16.11.2)
[√] Android Studio (version 2024.2)
[√] VS Code (version 1.102.0)
[√] Connected device (3 available)
[√] Network resources

• No issues found!
```

and this took about 4 minutes.

Clone and debug a prepared project implementing a basic lux meter

The project is stored in the following repository: https://github.com/GuentherSchmitt/lux_meter_2025
Clone it from there (use as URL for cloning “https://github.com/GuentherSchmitt/lux_meter_2025.git”):



and select the folder where this project should be stored on your PC.

When the project is opened, confirm to run “flutter pub get” in the upcoming message on bottom-right:

Now select either Android emulator or your (via USB connected) Android smartphone as device in VS Code.
In case you have issues to use your smartphone as device, see our old Powerpoint https://github.com/GuentherSchmitt/fdg_2024/blob/main/docs/02%20Test%20Flutter%20apps%20on%20an%20Android%20device.pdf from last year for details.

I used the emulator as device and pressed F5 to start debugging:

```
Launching lib\main.dart on sdk gphone64 x86 64 in debug mode...
F/bluetooth(16869): system/gd/hci/hci_layer.cc:478 on_hardware_error: Hardware Error Event with code 0x42
Checking the license for package NDK (Side by side) 27.0.12077973 in C:\Users\GS\AppData\Local\Android\sdk\licenses
License for package NDK (Side by side) 27.0.12077973 accepted.
Preparing "Install NDK (Side by side) 27.0.12077973 v.27.0.12077973".
"Install NDK (Side by side) 27.0.12077973 v.27.0.12077973" ready.
Installing NDK (Side by side) 27.0.12077973 in C:\Users\GS\AppData\Local\Android\sdk\ndk\27.0.12077973
"Install NDK (Side by side) 27.0.12077973 v.27.0.12077973" complete.
"Install NDK (Side by side) 27.0.12077973 v.27.0.12077973" finished.
Checking the license for package Android SDK Platform 35 in C:\Users\GS\AppData\Local\Android\sdk\licenses
License for package Android SDK Platform 35 accepted.
Preparing "Install Android SDK Platform 35 (revision 2)".
"Install Android SDK Platform 35 (revision 2)" ready.
Installing Android SDK Platform 35 in C:\Users\GS\AppData\Local\Android\sdk\platforms\android-35
"Install Android SDK Platform 35 (revision 2)" complete.
"Install Android SDK Platform 35 (revision 2)" finished.
Checking the license for package CMake 3.22.1 in C:\Users\GS\AppData\Local\Android\sdk\licenses
License for package CMake 3.22.1 accepted.
Preparing "Install CMake 3.22.1 v.3.22.1".
"Install CMake 3.22.1 v.3.22.1" ready.
Installing CMake 3.22.1 in C:\Users\GS\AppData\Local\Android\sdk\cmake\3.22.1
"Install CMake 3.22.1 v.3.22.1" complete.
"Install CMake 3.22.1 v.3.22.1" finished.
√ Built build\app\outputs\flutter-apk\app-debug.apk
I/flutter (19453): [IMPORTANT:flutter/shell/platform/android/android_context_gl_impeller.cc(94)] Using the Impeller rendering backend (OpenGLES).
Connecting to VM Service at ws://127.0.0.1:53323/0EV90iyqd4I=/ws
Connected to the VM Service.
I/Choreographer(19453): Skipped 148 frames!  The application may be doing too much work on its main thread.
D/ProfileInstaller(19453): Installing profile for com.example.lux_meter_2025
```

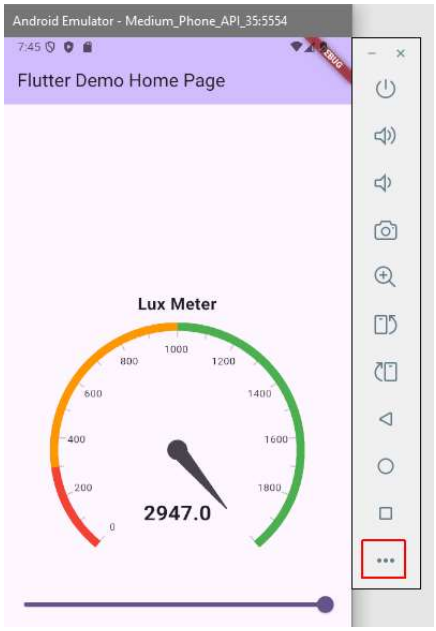
On my PC this took about 10 minutes. On our “family laptop”, where I had tested this before, it took significantly longer, but I did not stop the exact time there.

Because these installations of SDK (Software Development Kit), NDK (Native Development Kit) and other tools take a lot of time during the first build, I have asked you to perform this before our next flutter training.

Congratulation if this build process succeeded on your PC and the app started on your device.
In case you had issues, please contact Günther by mail and ask him for a meeting to analyze and hopefully fix these issues.

When the app runs on your smartphone, it should measure and display the lux value in your environment.

In case you used the emulator, open its menu:



and select Tab “Virtual sensors” to simulate different lux values.

Good luck !