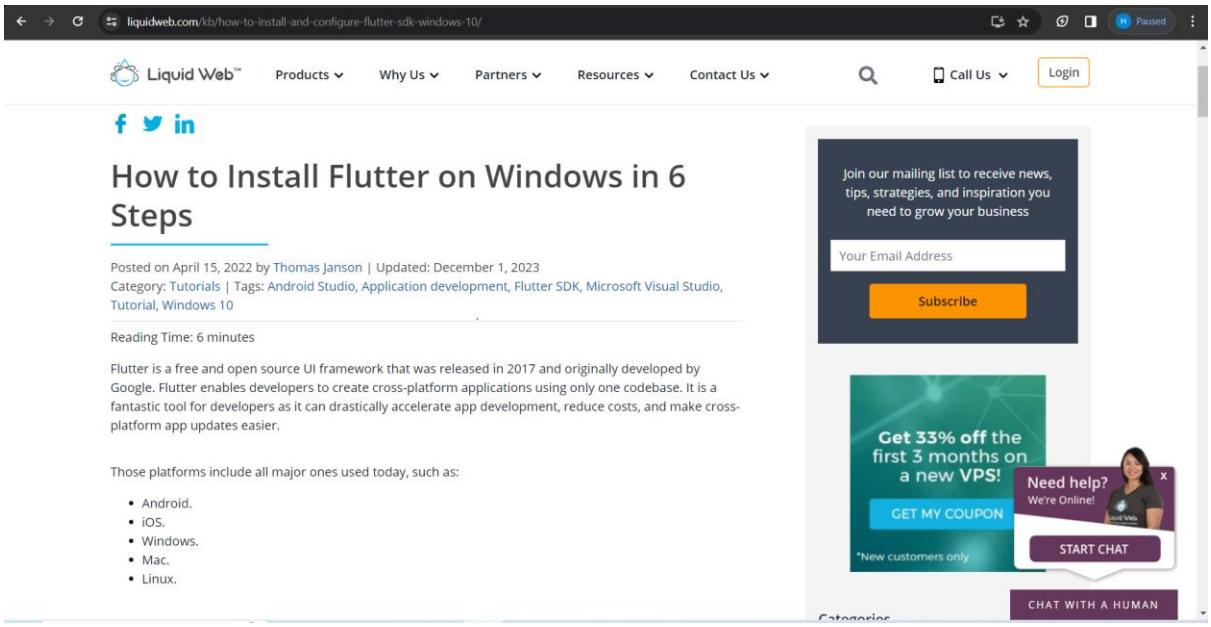
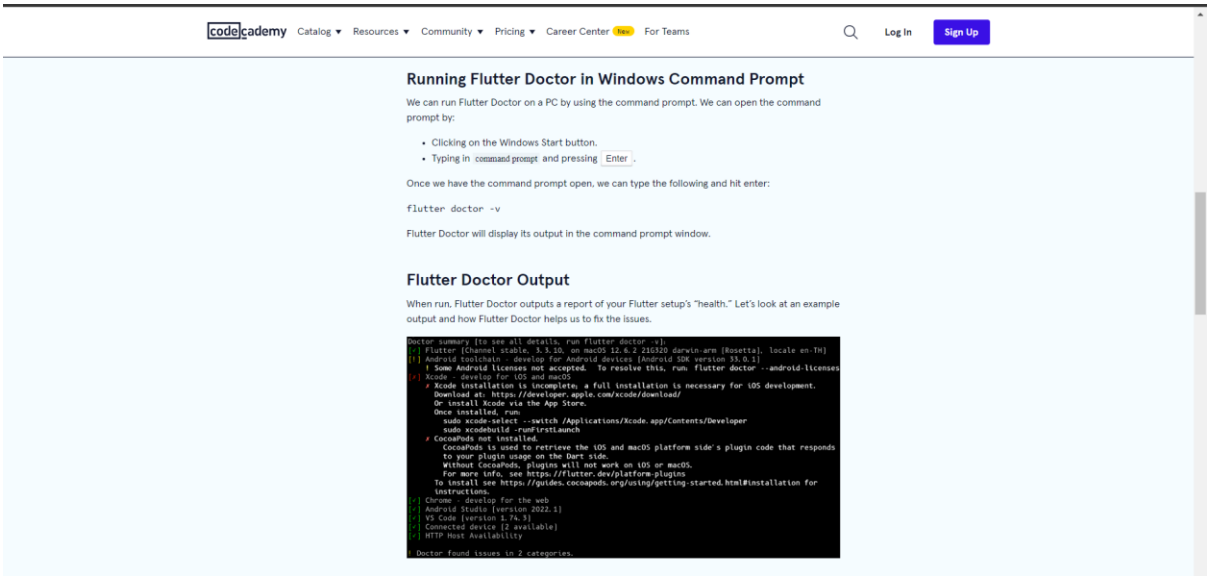
Website for Installation

1. LiquidWeb



2. CodeCademy



FAQ

What is Flutter?

Flutter is Google's portable UI toolkit for crafting beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.

Who is Flutter for?

For users, Flutter makes beautiful apps come to life.

For developers, Flutter lowers the bar to entry for building apps. It speeds app development and reduces the cost and complexity of app production across platforms.

For designers, Flutter provides a canvas for high-end user experiences. Fast Company described Flutter as one of the top design ideas of the decade for its ability to turn concepts into production code without the compromises imposed by typical frameworks. It also acts as a productive prototyping tool with drag-and-drop tools like FlutterFlow and web-based IDEs like Zapp!.

For engineering managers and businesses, Flutter allows the unification of app developers into a single *mobile, web, and desktop app team*, building branded apps for multiple platforms out of a single codebase. Flutter speeds feature development and synchronizes release schedules across the entire customer base.

Who uses Flutter?

Developers inside and outside of Google use Flutter to build beautiful natively-compiled apps for iOS and Android. To learn about some of these apps, visit the showcase.

What makes Flutter unique?

Flutter is different than most other options for building mobile apps because it doesn't rely on web browser technology nor the set of widgets that ship with each device. Instead, Flutter uses its own high-performance rendering engine to draw widgets.

In addition, Flutter is different because it only has a thin layer of C/C++ code. Flutter implements most of its system (compositing, gestures, animation, framework, widgets, etc) in Dart (a modern, concise, object-oriented language) that developers can easily approach read, change, replace, or remove. This gives developers tremendous control over the system, as well as significantly lowers the bar to approachability for the majority of the system.