

Инструкция к буткемпу Flutter

Инструкция по установке Flutter и запуску на эмуляторе/устройстве

Эта инструкция основана на официальной документации Flutter. Она описывает два варианта установки Flutter, настройку для тестирования на эмуляторах или устройствах, а также рекомендуемые расширения для VS Code. Для каждого шага приведены ссылки на документацию с кратким описанием ключевых инструкций.

Варианты установки Flutter

Существует 2 варианта установки Flutter и его компонент.

Установка Flutter через VS Code (рекомендуется, самый простой вариант)

<https://docs.flutter.dev/install/quick>

С помощью этого туториала из официальной документации вы:

- Установите необходимое предварительное ПО, а именно Git и VS Code, если их нет.
- Установите расширения Flutter и Dart в VS Code.
- Создадите новый проект Flutter через Command Palette, скачаете SDK и добавите его в PATH.

Примечание

Для SDK выбирайте папку, в пути к которой нет специальных символов или пробелов и которая не требует повышенных привилегий (например, `%USERPROFILE%\develop` на Windows или `~/develop/` на macOS/Linux).

- Создадите пример приложения на основе шаблона.
- Протестируете функции, такие как горячая перезагрузка (hot reload) с сохранением состояния, запуск на Chrome и использование Flutter Inspector для просмотра дерева виджетов.

Установка Flutter напрямую через архив

<https://docs.flutter.dev/install/manual>

- Скачайте архив с последней стабильной версией Flutter SDK для вашей ОС.

- Создайте папку для SDK (избегайте путей с пробелами или специальными символами).
- Извлеките архив в выбранную папку с помощью команд (например, `unzip` на macOS или `tar -xf` на Linux).
- Добавьте папку `bin` Flutter SDK в переменную PATH (с инструкциями для Windows, macOS и Linux).
- Проверьте установку, выполнив `flutter --version` и `dart --version` в терминале.

Настройка для тестирования на эмуляторе или устройстве

Для тестирования результатов разработки мобильного приложения потребуется установить эмулятор или использовать физическое устройство. Инструкции ниже:

Для Android

<https://docs.flutter.dev/platform-integration/android/setup>

- Установите Android Studio и необходимые компоненты через SDK Manager (Android SDK Build-Tools, Command-line Tools, Emulator, Platform-Tools, CMake, NDK).
- Примите лицензии Android SDK с помощью `flutter doctor --android-licenses`.
- Настройте эмулятор: создайте виртуальное устройство в Device Manager, выберите образ системы (x86 или ARM) и включите аппаратное ускорение.
- Подключите физическое устройство: включите USB-отладку, установите драйверы (на Windows) и проверьте подключение с `flutter devices`.
- Проверьте настройку с `flutter doctor` и `flutter emulators`.

Для iOS (доступно только на macOS)

<https://docs.flutter.dev/platform-integration/ios/setup>

- Установите Xcode и настройте command-line tools с помощью `xcode-select` и `xcodebuild -runFirstLaunch`.
- Примите лицензии Xcode и скачайте поддержку iOS платформы.
- Установите Rosetta 2 на Apple Silicon (если нужно) и CocoaPods.
- Запустите симулятор iOS с `open -a Simulator` или скачайте дополнительные версии.
- Для физического устройства: подключите по USB, доверьте компьютер, включите Developer Mode, создайте сертификат разработчика (бесплатно для тестирования) и настройте в VPN & Device Management.

Рекомендуемые расширения для VS Code

Рекомендую установить следующие расширения для удобства разработки в Flutter:

- [Awesome Flutter Snippets - Visual Studio Marketplace](#) — предоставляет набор сокращений для быстрого создания громоздких конструкций, таких как StatelessWidget, StatefulWidget, StreamBuilder и другие. Примеры: введите `statelessW` для StatelessWidget, `statefulW` для StatefulWidget, `streamBldr` для StreamBuilder, `futureBldr` для FutureBuilder.
 - [Android iOS Emulator - Visual Studio Marketplace](#) — позволяет запускать эмуляторы Android и симуляторы iOS прямо из VS Code без перехода в Android Studio или Xcode. Доступно через Command Palette (тип "Emulator") или иконку в интерфейсе; поддерживает кастомные пути и cold boot для Android.
 - [Flutter Tree - Visual Studio Marketplace](#) — позволяет по сокращенному синтаксу строить большие конструкции из вложенных друг в друга виджетов, ускоряя создание сложных UI-структур в Flutter.
-