

12 лекция.

Тема: Flutter: Полноценное мобильное приложение в Android Studio: создаем простой мессенджер

План:

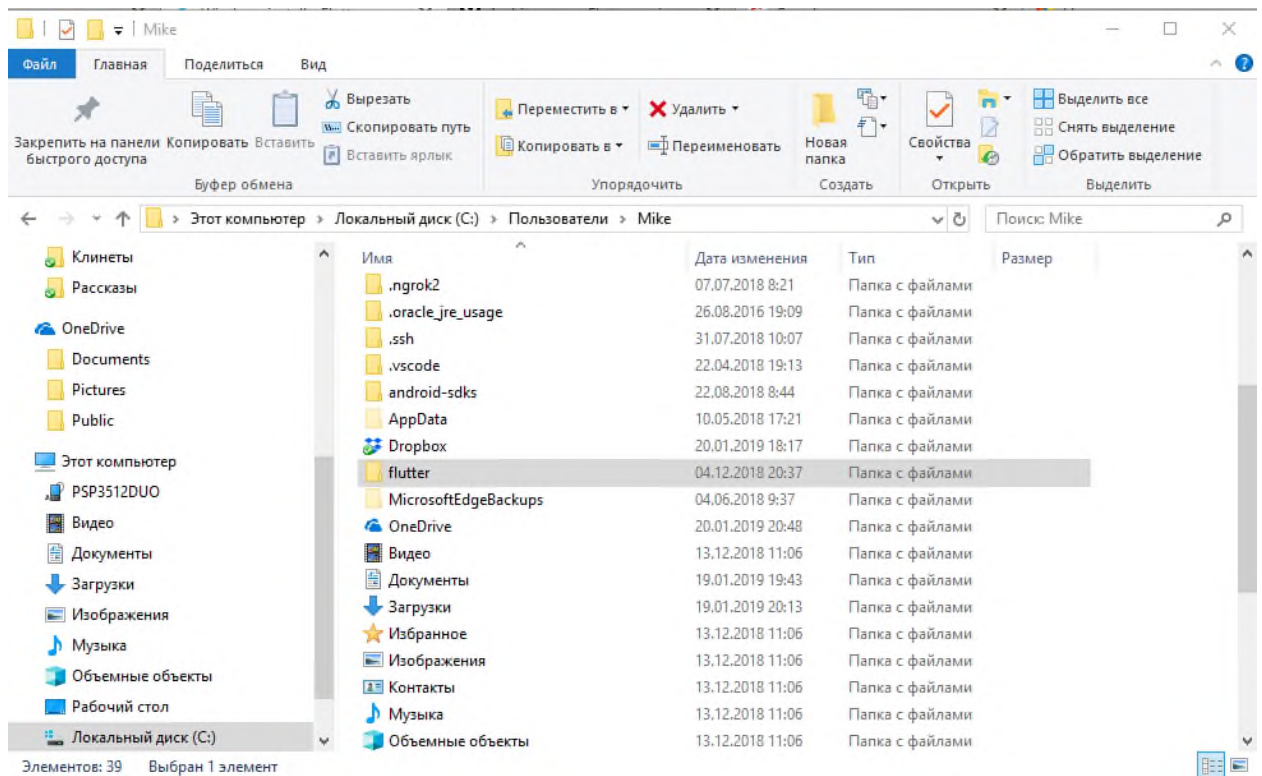
1. Создание простого мессенджера

Поэтому, если вы изучите **Dart** и **Flutter**, вы сможете писать нативные приложения для двух самых популярных мобильных операционных систем, т.е. быть одновременно Android и iOS разработчиком. И это, вероятно, ещё не предел, поскольку было объявлено, что Google ведёт работу по расширению Flutter на предмет возможности создания с его помощью приложений для Windows, Mac и Web (проект Flutter для Web носит название Hummingbird — Колибри). В итоге может получиться так, что зная Dart и Flutter вы сможете писать всё очень многое. Поэтому многие IT эксперты называли 2018 год — годом Flutter.

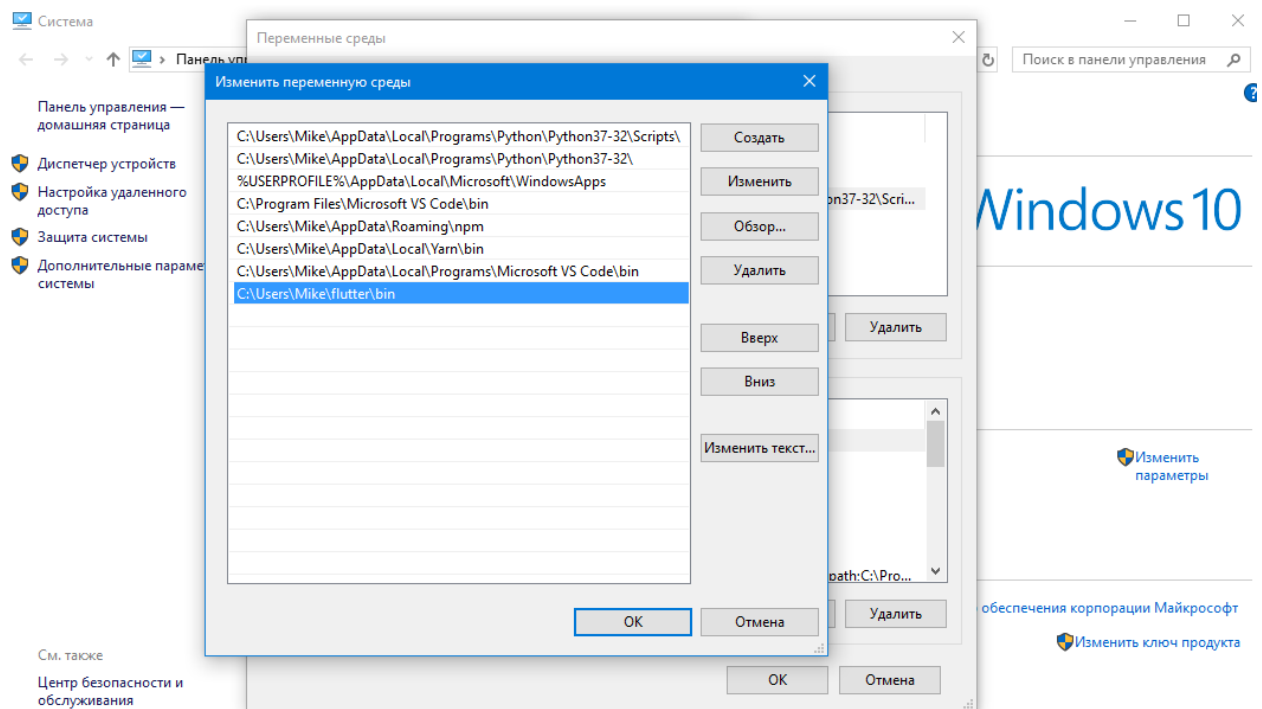
Теперь за дело. Сейчас мы сделаем следующее

1. Установим Flutter SDK, и создадим проект из командной строки Windows.
2. Установим, JDK, Android Studio, плагины для Flutter и Dart, и создадим проект в Android Studio.
3. Установим VS Code, расширения для Flutter и Dart, и создадим проект в VS Code.

Устанавливаем Flutter и создаём проект из командной строки
Переходим на [страницу установки Flutter](#), выбираем свою операционную систему — Windows, Mac или Linux (здесь будет описано для Windows 10, как наиболее популярной ОС), и скачиваем zip файл, содержащий Flutter SDK. Затем распаковываем zip, например, в папку текущего пользователя, как показано на скриншоте:

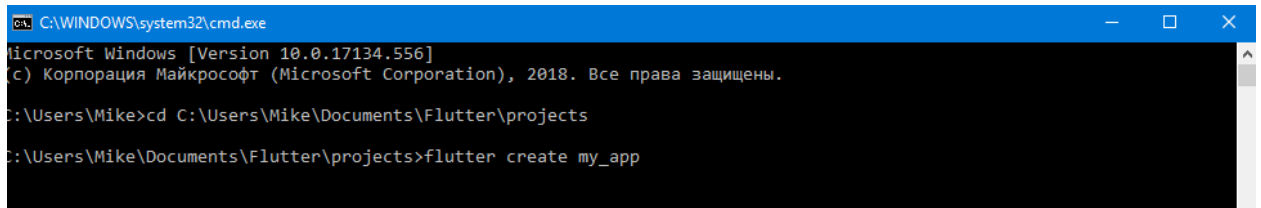


Сейчас пропишем путь к `flutter\bin` в переменную `Path` среды пользователя Windows (Этот компьютер -> Свойства -> Дополнительные параметры среды):



Можно создавать проект из командной строки Windows:

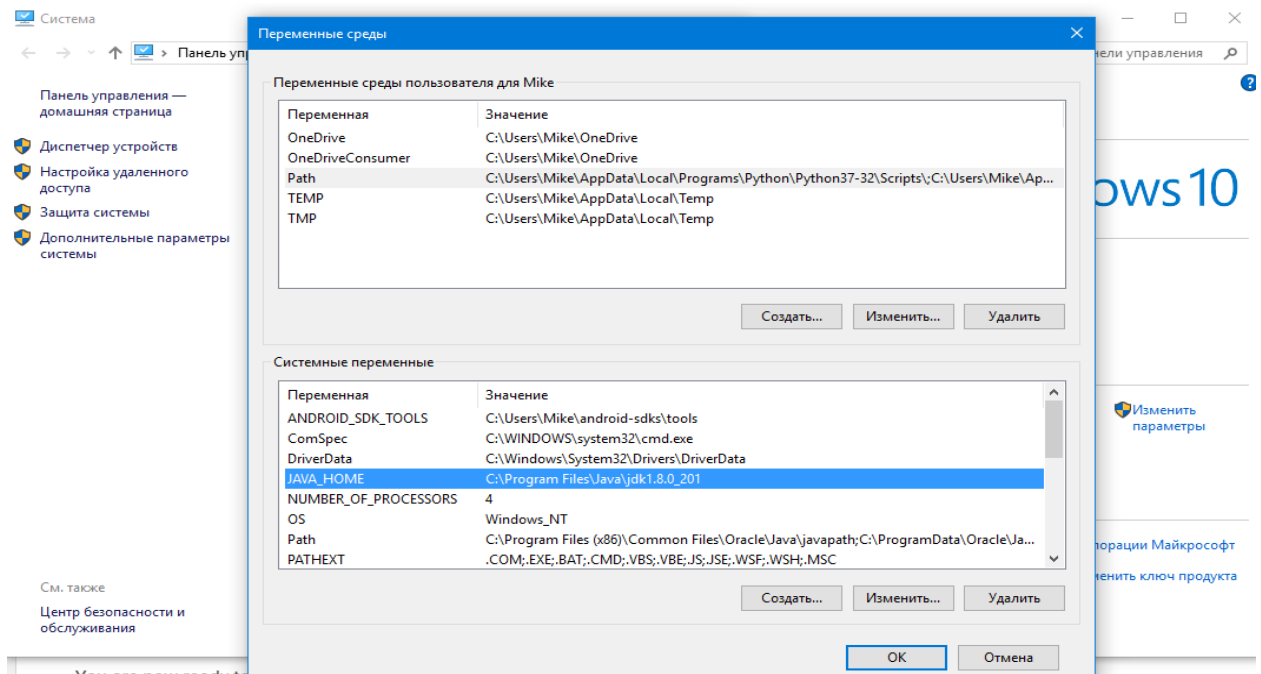
```
flutter create my_app
```



Готово! Файлы проекта можно редактировать любым текстовым редактором, хоть в блокноте. Но это хорошо разве что для мелких правок. Поэтому мы...

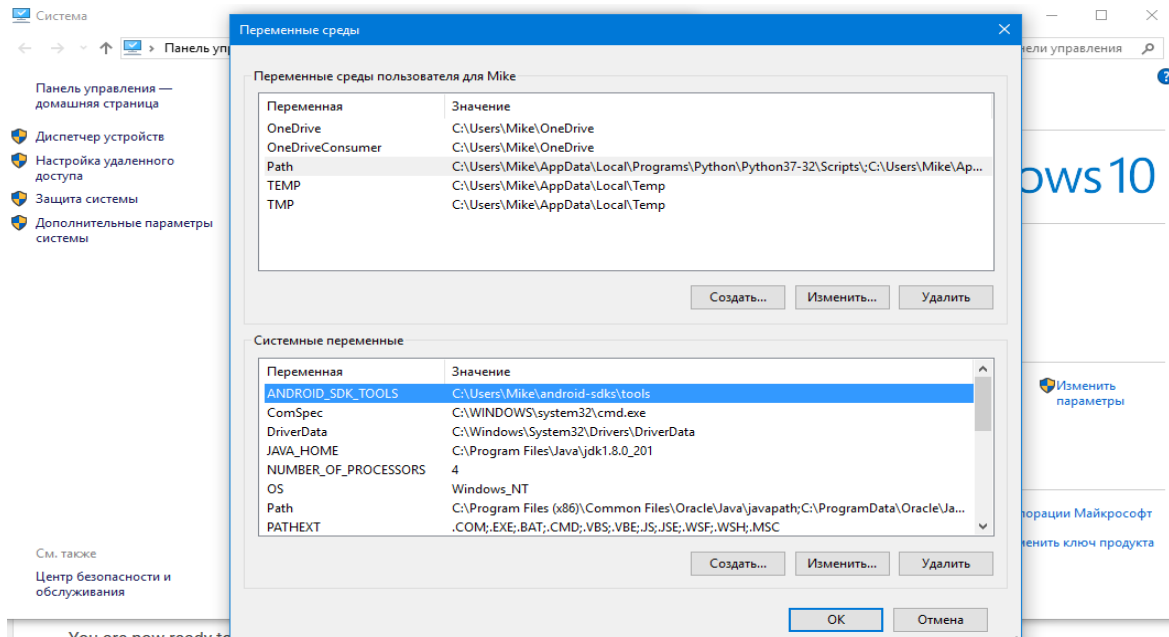
Устанавливаем JDK, Android Studio (вместе с Android SDK) и необходимые плагины

Скачиваем последнюю версию [Java SE Development Kit 8](#) для своей операционной системы (потребуется для Android SDK), устанавливаем на свой компьютер, следуя за мастером установки, и создаём системную переменную среды `JAVA_HOME` с указанием пути к JDK, например: `C:\Program Files\Java\jdk1.8.0_201`.

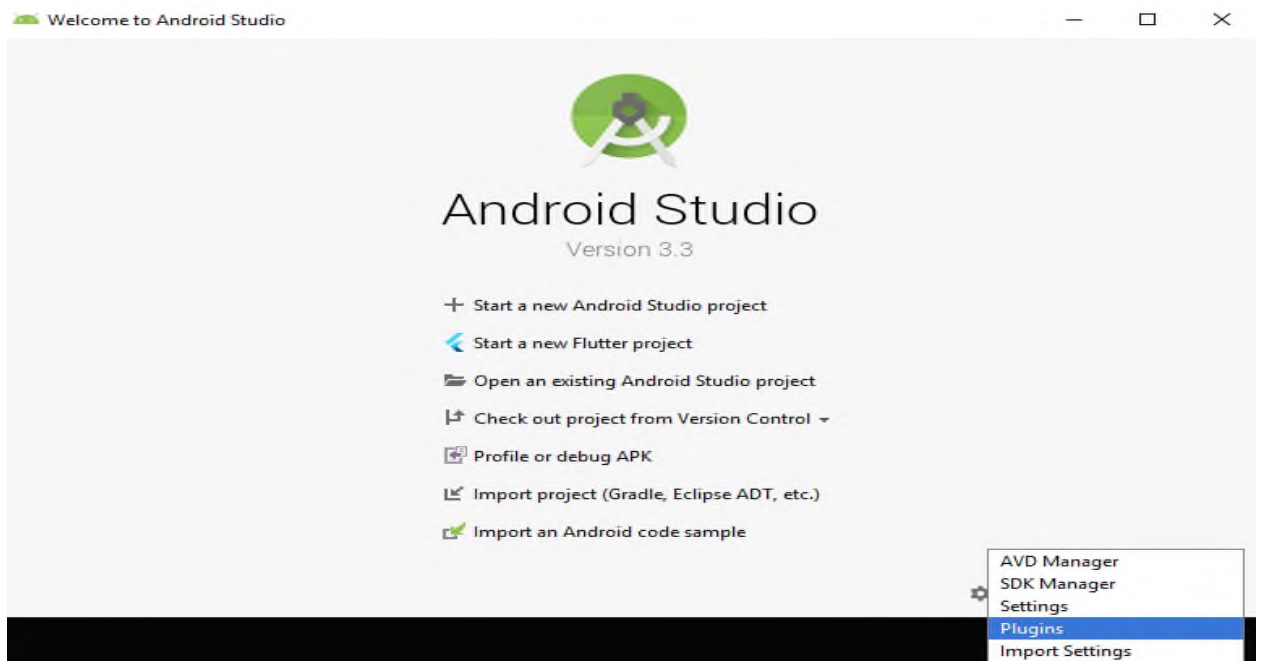


Теперь скачиваем [Android Studio](#). Запускаем процесс установки, следуя за мастером установки, и обращаем внимание на путь, куда будет установлен Android SDK. Создаём системную переменную

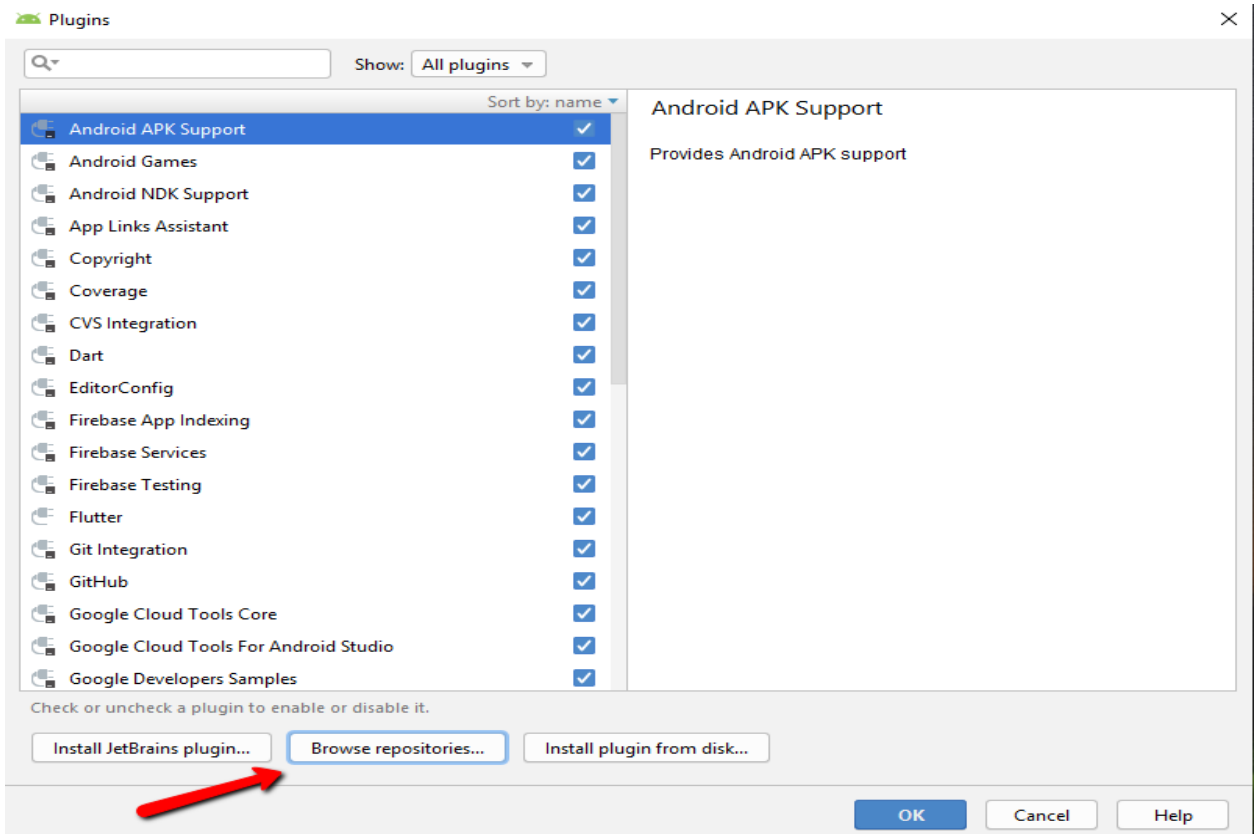
среды *ANDROID_SDK_TOOLS* с указанием пути к папке *\tools* в Android SDK, примерно так:



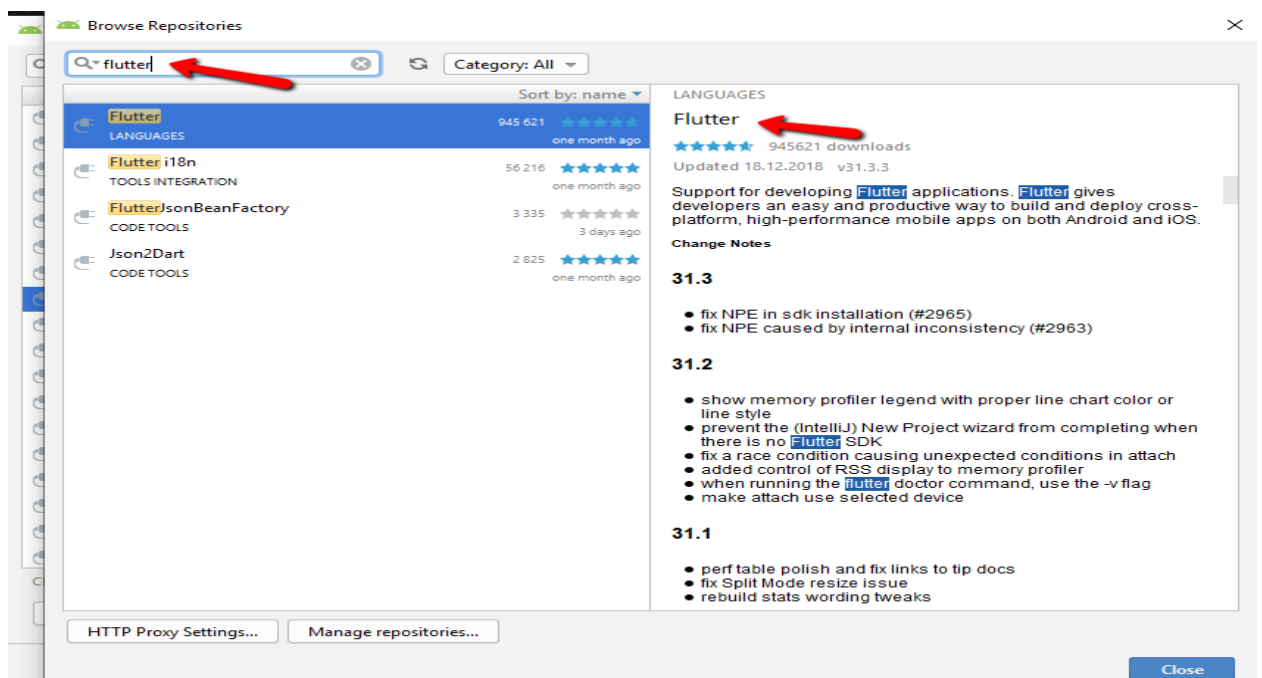
Когда всё готово — запускаем Android Studio, и устанавливаем плагины Flutter и Dart. Для этого в начальном экране Android Studio справа внизу жмём на значок шестерёнки и выбираем Plugins:



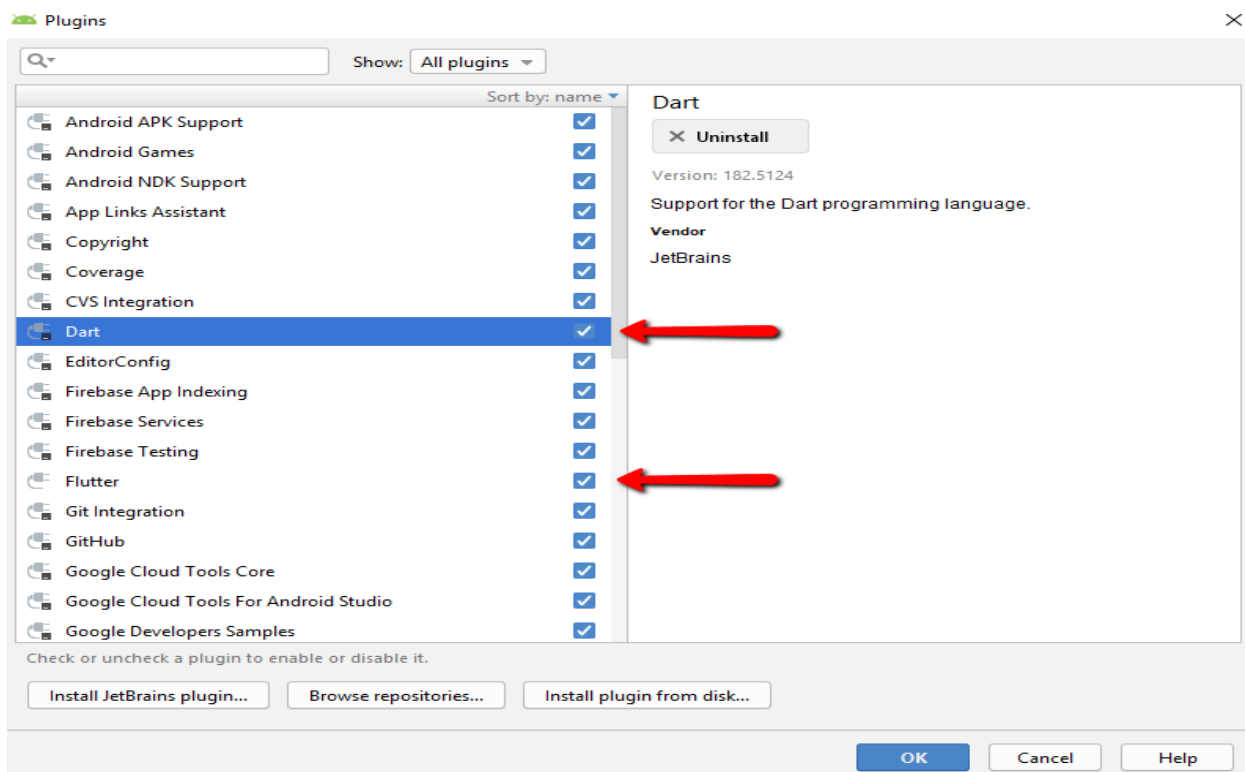
В открывшемся окне внизу нажимаем кнопку *Browse repositories...*



В поисковую строку вводим *flutter*, выбираем и устанавливаем (у меня уже установлен, поэтому не видно соответствующей кнопки):



Android Studio предложит также установить плагин Dart от которого зависит работа плагина Flutter. Соглашаемся. В итоге у вас должно быть установлено как минимум два плагина:



Перезапускаем Android Studio, и теперь давайте убедимся, что всё идёт хорошо. Для этого в командной строке выполним команду:

```
flutter doctor
```

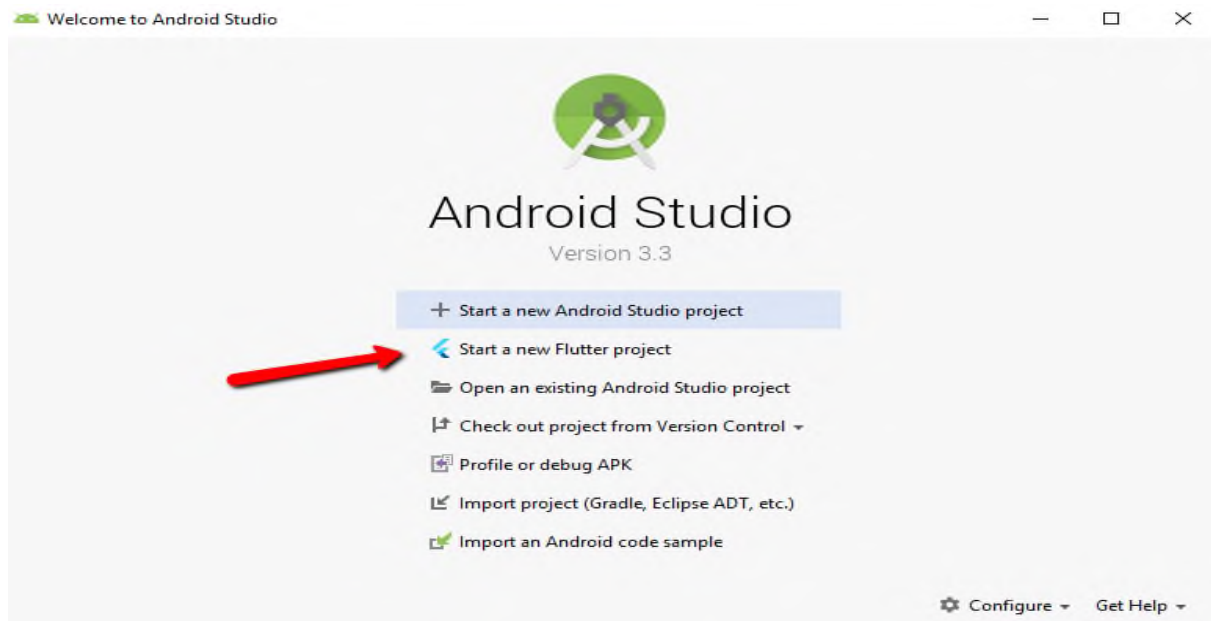
Сканирование займёт десяток секунд, и затем вы можете увидеть примерно такой результат:

```
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, v1.0.0, on Microsoft Windows [Version 10.0.17763.253], locale ru-RU)
[✓] Android toolchain - develop for Android devices (Android SDK 28.0.3)
[✓] Android Studio (version 3.3)
```

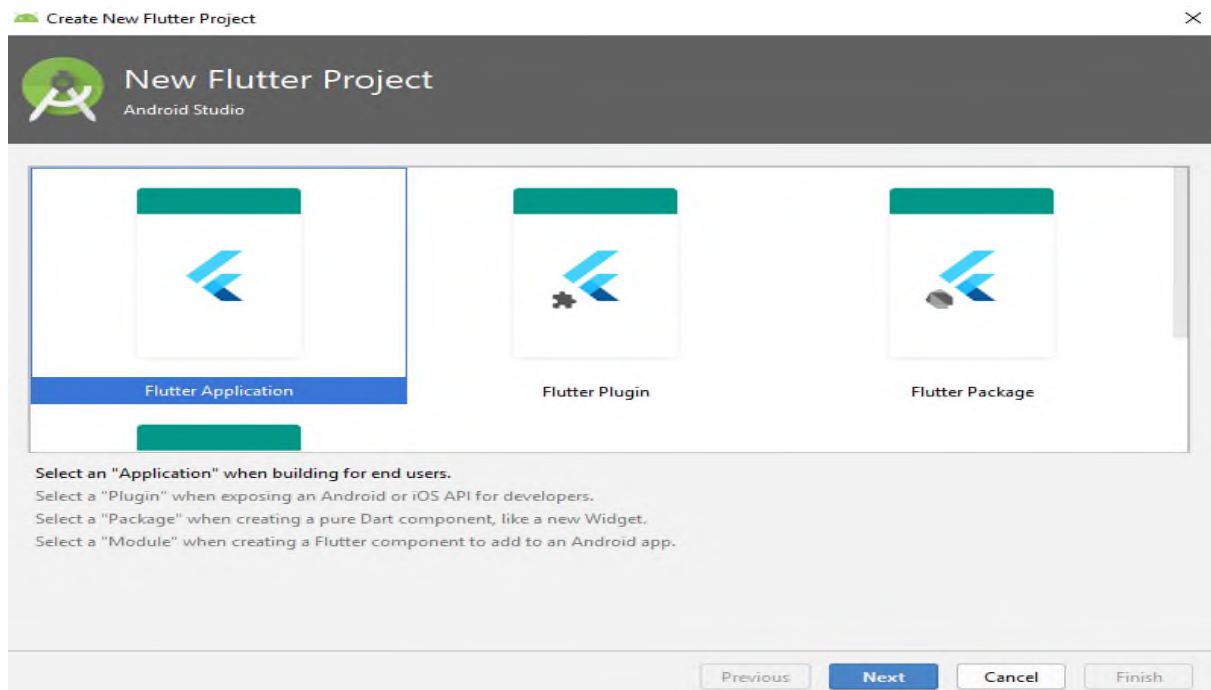
А возможно будет пункт, отмеченный красным крестиком, с пояснением (на английском), что вами ещё не приняты какие-то лицензии (licences), касающиеся Android SDK, и предложение их принять (Y/n). Примите их, напечатав в командной строке Y. Возможно это придётся сделать несколько раз (если имеется несколько лицензий).

Вот мы и готовы создать Flutter проект в Android Studio. После установки плагинов Flutter и Dart в начальном экране Android Studio должна появиться

опция *Start a new Flutter project*. Выбираем её:





Далее соглашаемся с выбранной по умолчанию опцией *Flutter Application* и нажимаем кнопку *Next*:



Указываем название проекта, путь к папке Flutter SDK, путь к папке проекта, даём краткое описание проекта (опционально), и вновь нажимаем кнопку *Next*:

Create New Flutter Project



 New Flutter Application
Android Studio




Configure the new Flutter application

Project name

Flutter SDK path

  [Install SDK...](#)

Project location





Description

☐ Create project offline

Наконец, указываем доменное имя (которое в реверсивном порядке будет использовано как ID Android приложения), а также опционально — поддержку языков Kotlin и Swift (если не указать — по умолчанию будут поддерживаться только Java и Objective-C). Нажимаем кнопку *Finish*.

Create New Flutter Project

 New Flutter Application
Android Studio




Set the package name

Applications and plugins need to generate platform-specific code

Company domain

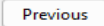
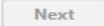
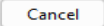

Package name

com.mzaharov.mycoolapp 

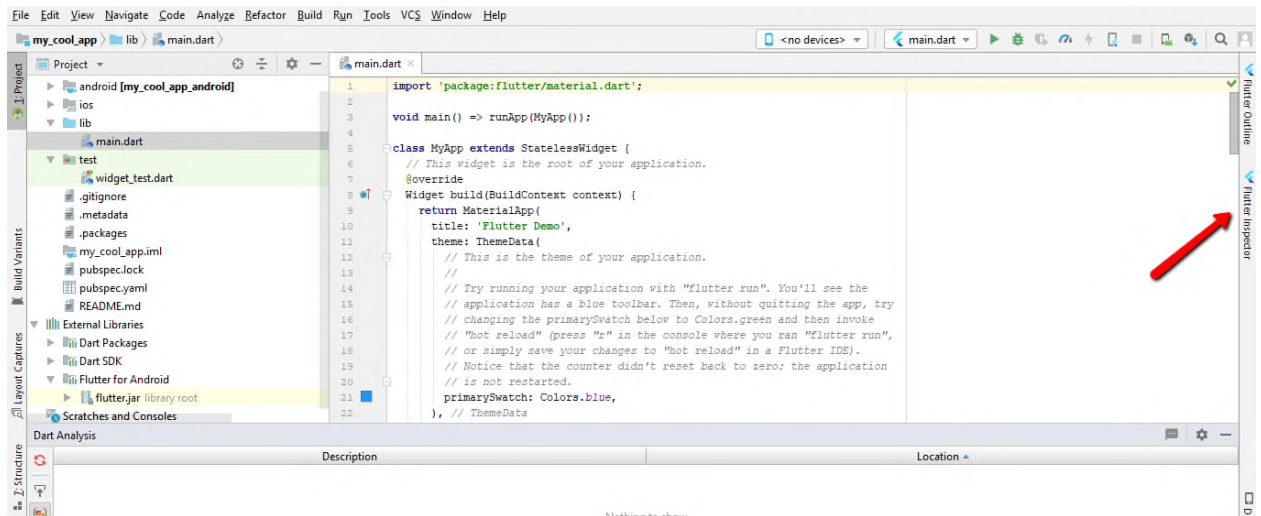
Platform channel language

☐ Include Kotlin support for Android code

☐ Include Swift support for iOS code

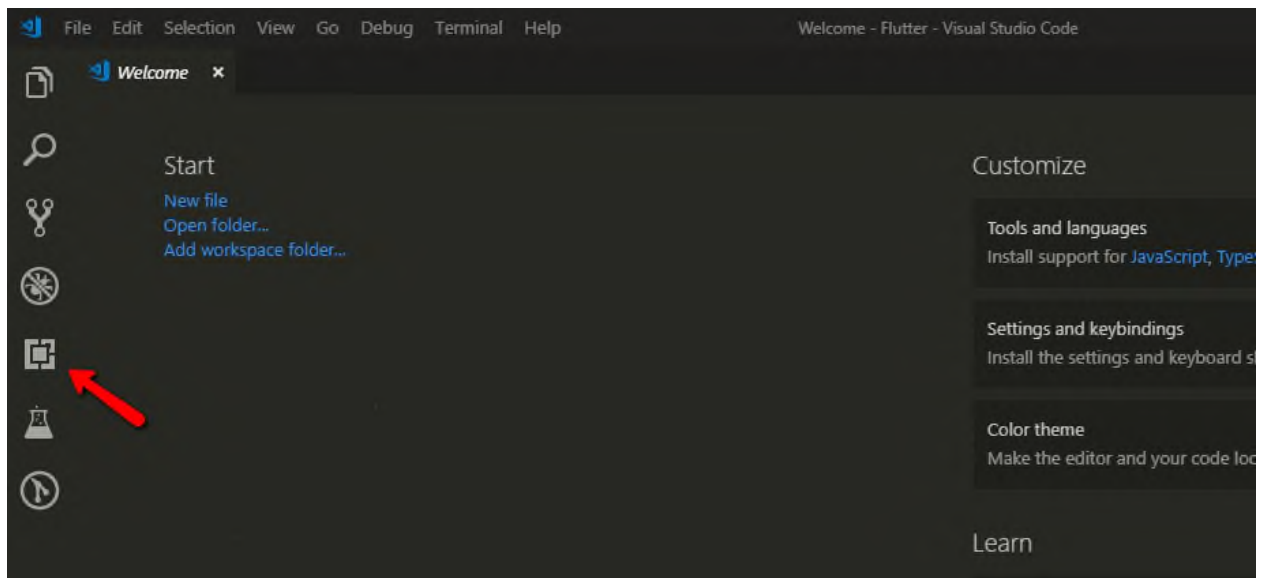
В зависимости от производительности компьютера, ждём несколько минут пока проект будет создан... Готово! Он должен выглядеть примерно так:



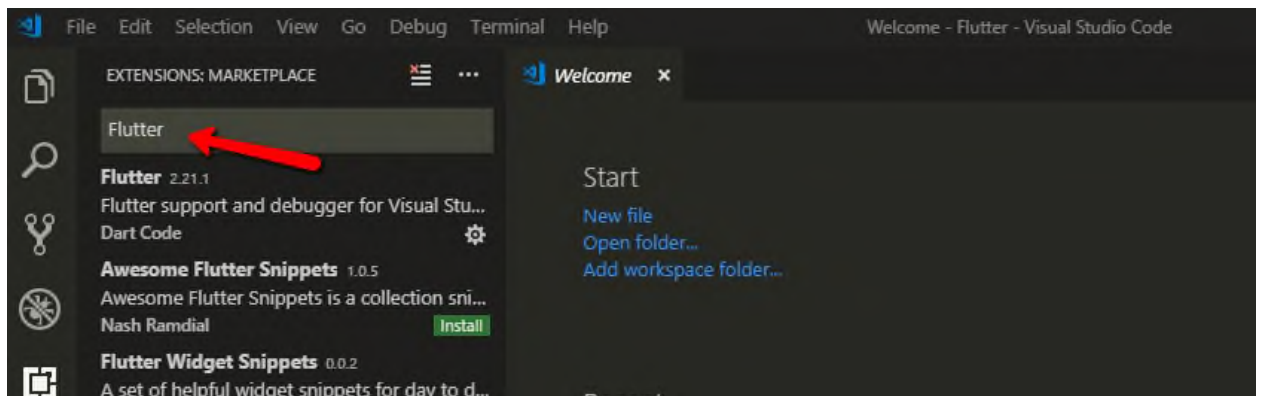
Обратите внимание на стрелку, указывающую на вкладку *Flutter Inspector*. В этом инспекторе имеется функционал, позволяющий делать ряд очень полезных во время разработки вещей, в т.ч. просмотр приложения на девайсе Android в режиме представления на iOS!

И наконец, устанавливаем VS Code, расширения, и создаём третий Flutter проект

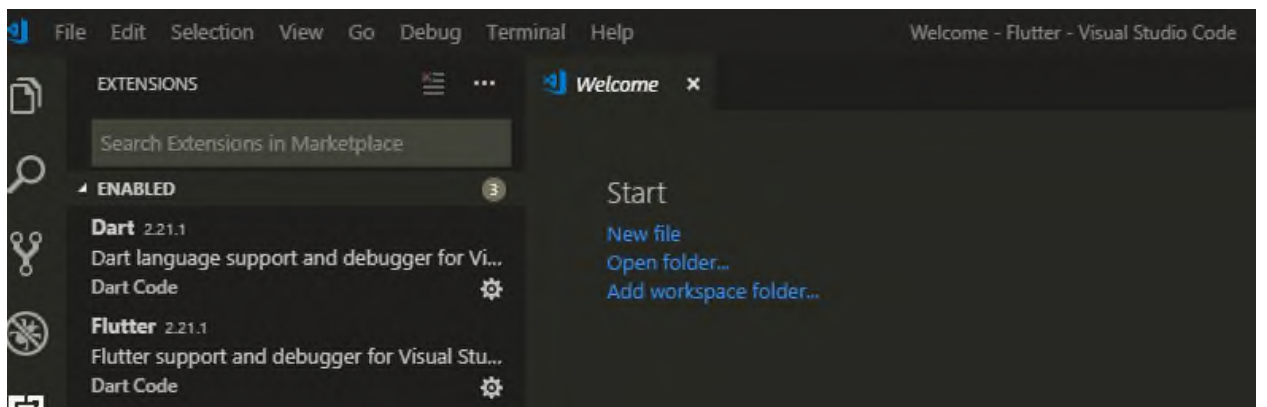
Скачиваем последнюю версию Visual Studio Code для своей операционной системы, устанавливаем на свой компьютер, следуя за мастером установки, и запускаем VS Code. Затем на боковой панели нажимаем на кнопку *Extensions* (показана стрелкой) или на клавиатуре — *Ctrl+Shift+X*:



С помощью поиска ищем расширение Flutter.

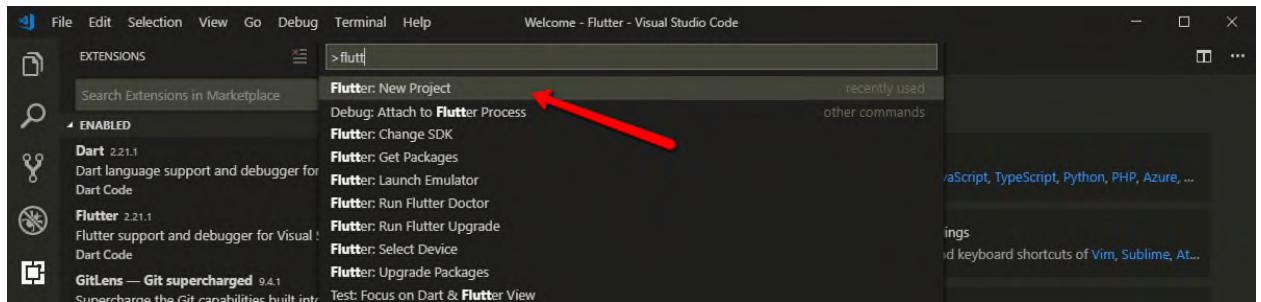


VS Code, как и в случае с Android Studio, предложит установить необходимое дополнительное расширение Dart. Устанавливаем и его. В итоге должны иметь два (или более) активированных расширения:

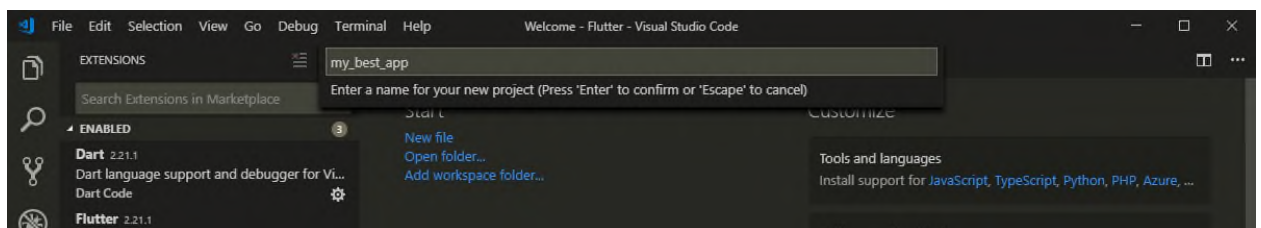


А теперь создаём Flutter проект. Нажимаем на значок шестерёнки в левом нижнем углу, и выбираем *Command Palette...* (или на клавиатуре

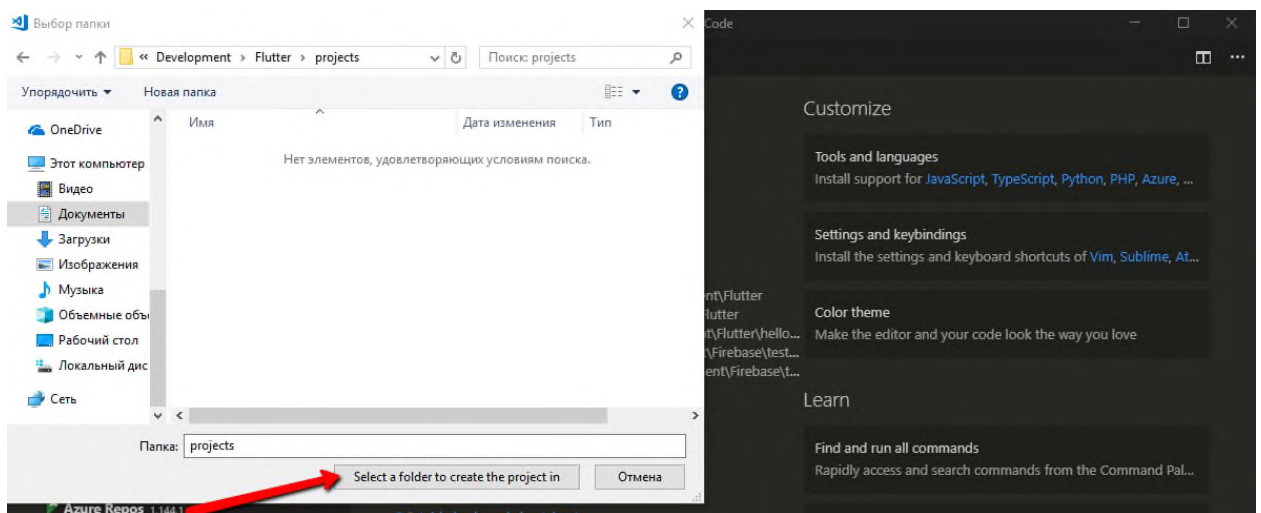
— *Ctrl+Shift+P*). В командной строке Command Palette начинаем печатать *flutter*, и из появившегося списка выбираем *Flutter: New Project*:



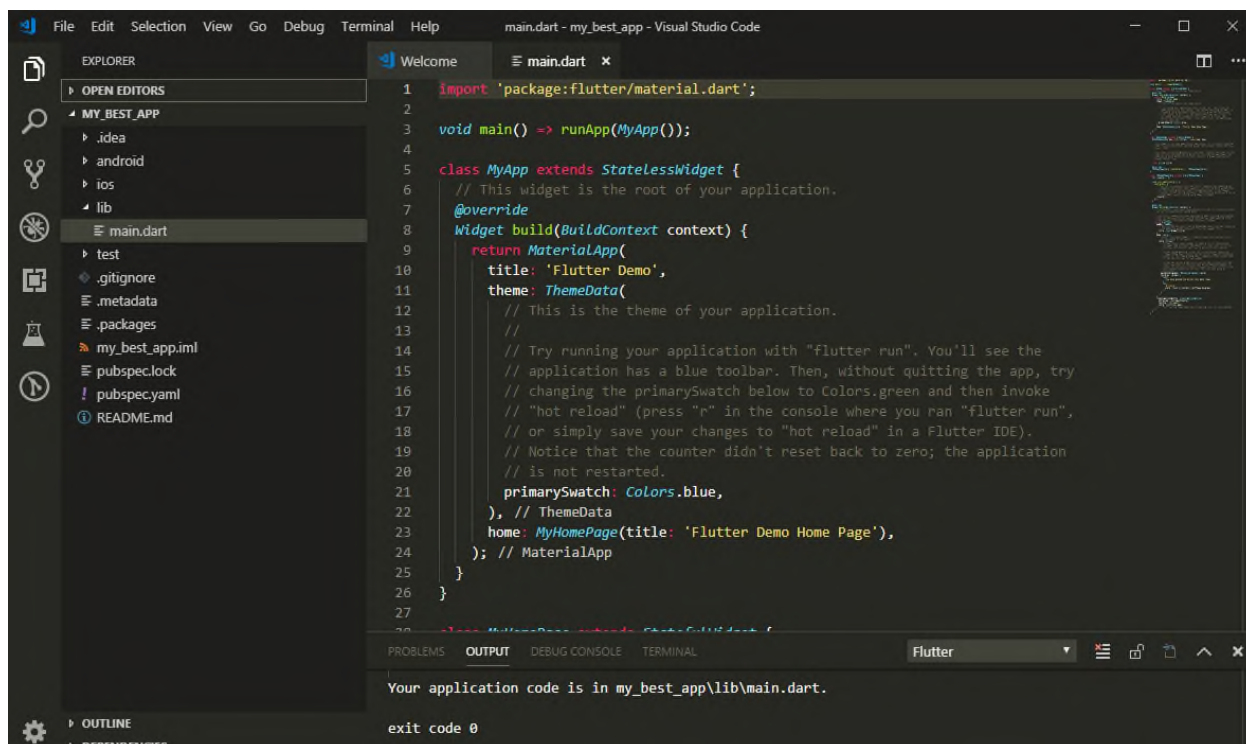
Даём проекту название и нажимаем клавишу *Enter*:



Появится диалоговое окно, предлагающее выбрать папку, в которой необходимо создать Flutter проект. Выбираем и нажимаем кнопку с длинным названием *Select a folder to create the project in*:



Минута ожидания... И, готово!



Мы установили необходимое программное обеспечение (всё бесплатное!), и создали проект Flutter тремя различными способами: из командной строки, с помощью Android Studio, с помощью VS Code.

Каким способом создавать проект и в какой среде разработки (IDE) лучше с ним работать — каждый решает сам. Например, я предпочитаю быстро создать проект из командной строки, затем открыть его в VS Code и большую часть времени над проектом работать именно в нём, т.к. VS Code довольно быстрый.

Но когда надо хорошо протестировать внешний вид и производительность приложения — открываю проект в Android Studio, чтобы использовать имеющийся пока только там *Flutter Inspector*, обладающий, как я уже говорил, рядом очень полезных опций.