

IESTP “ANDRES AVELINO CACERES DORREGARAY”



“MANUAL DE INSTALACIÓN DE LENGUAJE DART”

DOCENTE: Ing.Raúl Fernández Bejarano

PRESENTADA POR :

Carhuas Veli , Karen Valeria

Huancayo, 15 de Septiembre

2024

INTRODUCCIÓN

Dentro del universo de los lenguajes de programación, Dart ha emergido como una herramienta poderosa y versátil, especialmente en el desarrollo de aplicaciones modernas. Creado por Google y presentado por primera vez en 2011, hasta la actualidad Dart ha evolucionado significativamente para convertirse en un lenguaje clave en el ecosistema de desarrollo, particularmente en el ámbito de aplicaciones móviles y web.

Dart se distingue por su enfoque en la facilidad de uso y la eficiencia en la ejecución. Su diseño se centra en proporcionar un lenguaje intuitivo y altamente productivo que facilite el desarrollo rápido y escalable de aplicaciones. Esto lo ha convertido en una opción popular para crear aplicaciones tanto para plataformas móviles (Android e iOS) como para la web, gracias a su integración con el marco de trabajo Flutter, que ha revolucionado el desarrollo de interfaces de usuario (UI).

Uno de los principales atributos de Dart es su sintaxis clara y concisa, que combina la familiaridad de lenguajes como JavaScript y Java con características modernas. Su sistema de tipos estático permite detectar errores en tiempo de compilación, lo que contribuye a la robustez del código y a una experiencia de desarrollo más segura.

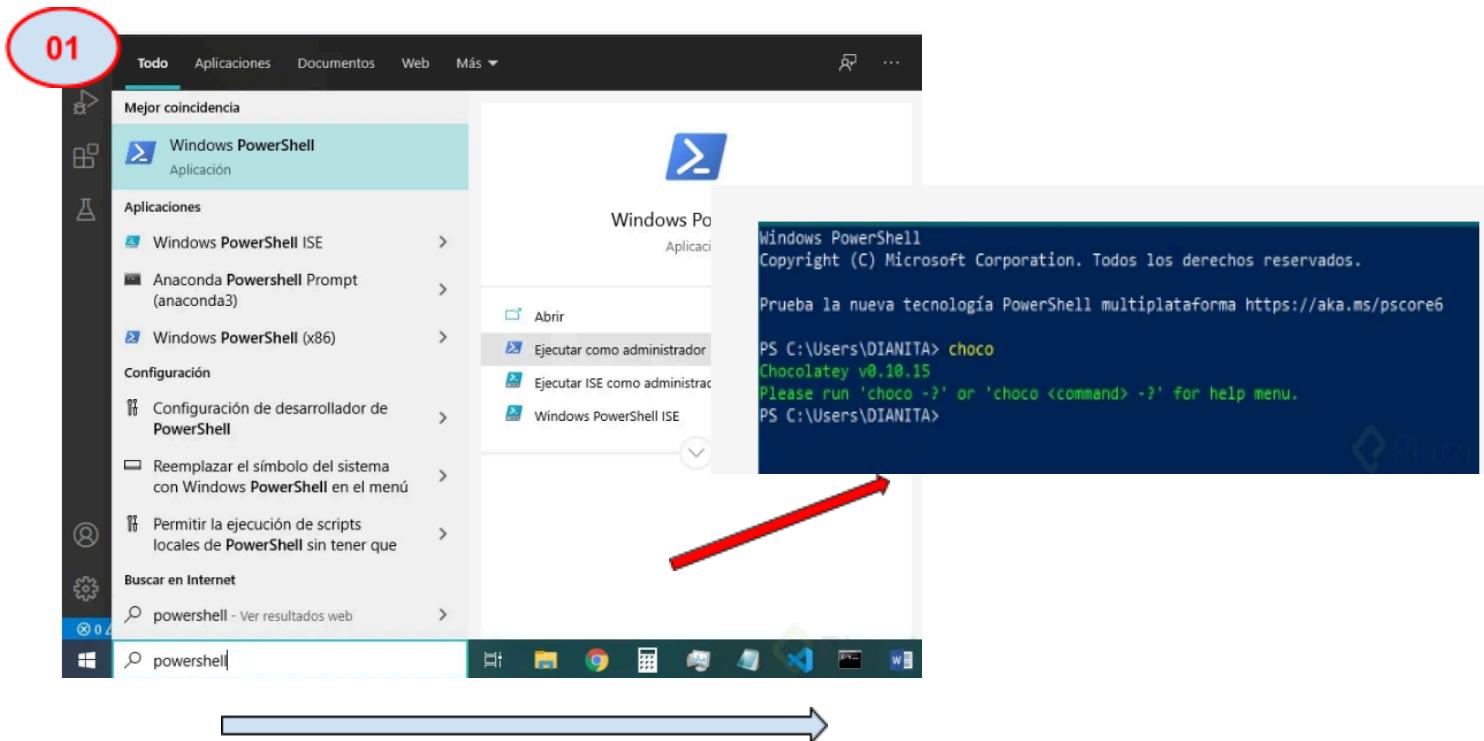
Además, Dart ofrece un rendimiento sobresaliente gracias a su compilador Just-In-Time (JIT) para un desarrollo ágil y un compilador Ahead-Of-Time (AOT) para una ejecución rápida en producción. Esta combinación asegura que las aplicaciones no solo se desarrollen con rapidez, sino que también se ejecuten de manera eficiente.

Por lo tanto podemos decir que, Dart se ha establecido como una herramienta de elección para los desarrolladores que buscan crear aplicaciones de alto rendimiento y mantener una base de código unificada en múltiples plataformas. Su integración con Flutter, junto con su diseño moderno y características innovadoras, lo posiciona como un lenguaje clave en el futuro del desarrollo de software.

I. INSTALAR DART EN VISUAL STUDIO CODE

PASO 1: Para poder instalar primero debemos verificar que tengamos instalado Chocolatey en nuestro equipo (Chocolatey es un gestor de paquetes para la terminal de Windows que nos permite instalar todo tipo de aplicaciones de forma sencilla).

Para verificar si tenemos instalado Chocolatey ejecutamos como Administrador el Windows PowerShell y escribimos la palabra 'choco'.



Si aún no tenemos instalado Chocolatey, debemos ir a la página oficial <https://chocolatey.org/install#individual> y copiamos el siguiente comando en nuestro Windows PowerShell.

```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
```

The screenshot shows a Windows PowerShell window with a red arrow pointing from the left towards the command line. The command being run is:

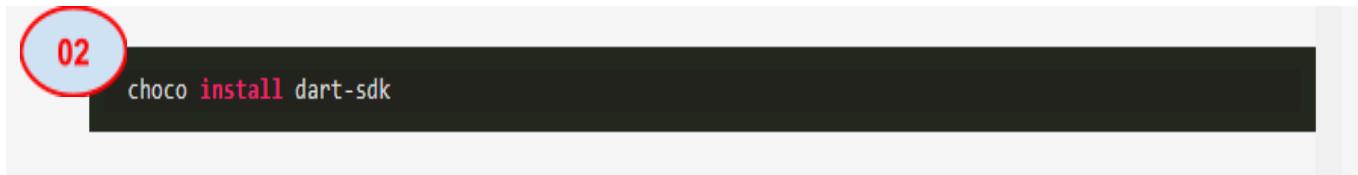
```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
```

The output window shows the execution of the command and the download of Chocolatey. A red arrow points from the right towards the output window.

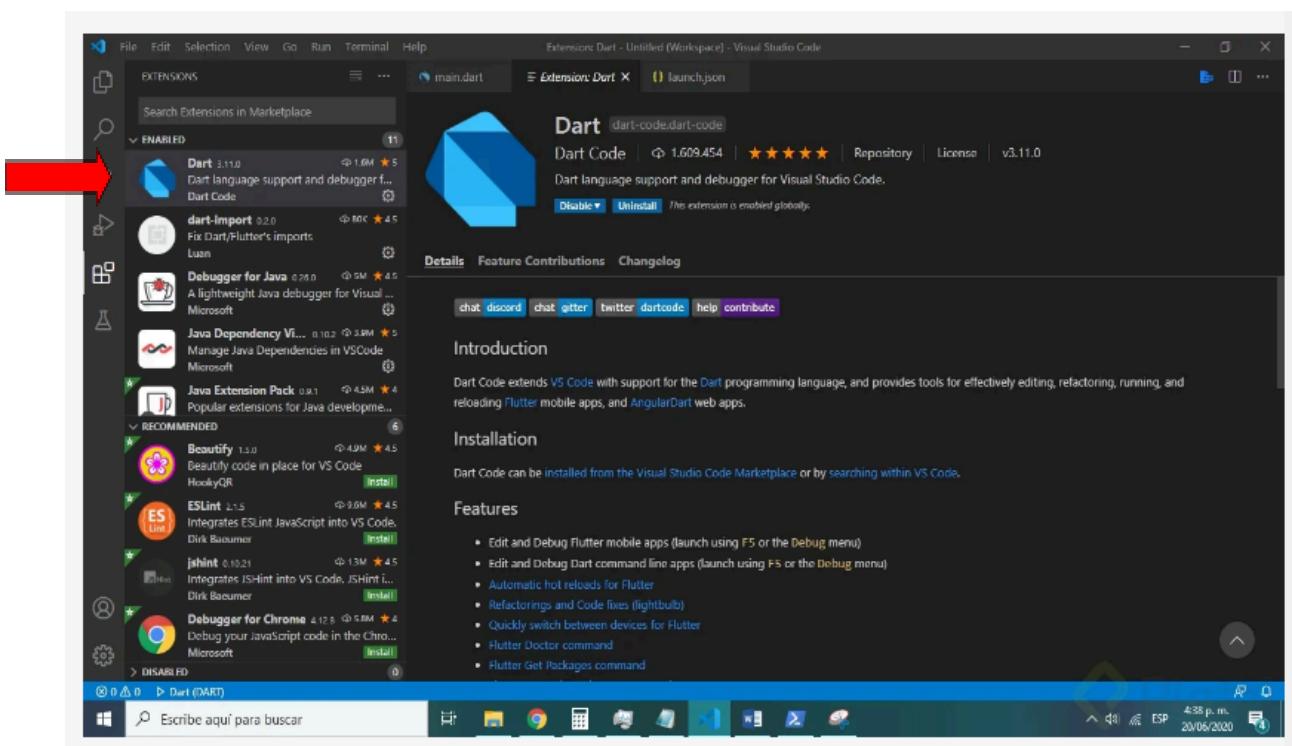
Hola,
y corr
¿me p
tu cor
actual

PASO 2: Una vez instalado Chocolatey cerramos Windows PowerShell y lo abrimos nuevamente como administrador.

Ahora nos dirigimos a la página oficial de Dart <https://dart.dev/get-dart#install> para instalar nuestro SDK de Dart y copiamos el siguiente comando en nuestro Windows PowerShell.

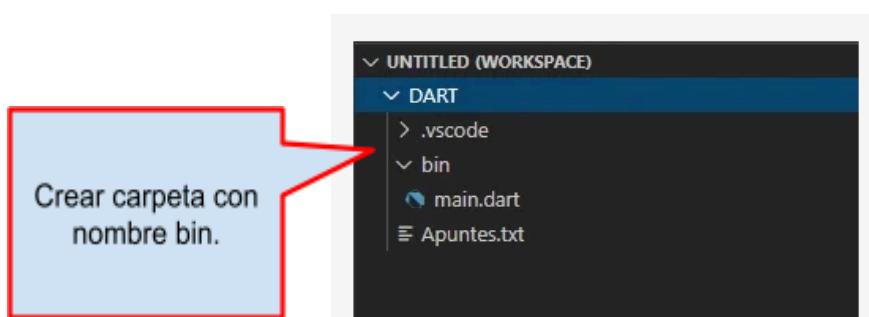


PASO 3: Despues de que hayamos instalado el SDK de Dart vamos a la sección de extensiones de nuestro Visual Studio Code, escribimos la palabra 'Dart' y le damos click a instalar a la extensión de Dart que nos proporciona herramientas para editar, refactorizar, ejecutar y recargar efectivamente aplicaciones móviles Flutter y aplicaciones web AngularDart.

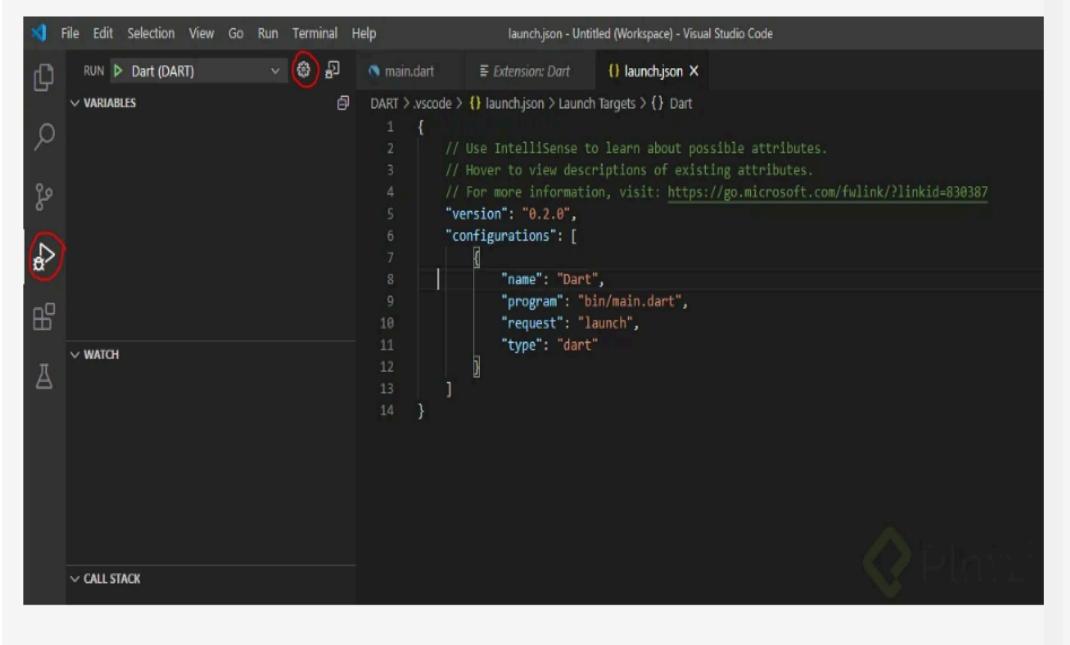


PASO 4: Una vez instalada la extensión de Dart, cerramos nuestro VS Code y lo volvemos a abrir.

Ahora, dentro de la carpeta donde guardaremos nuestros códigos escritos en Dart es importante que creemos una carpeta de nombre 'bin' y nuestro punto de acceso 'main.dart'.

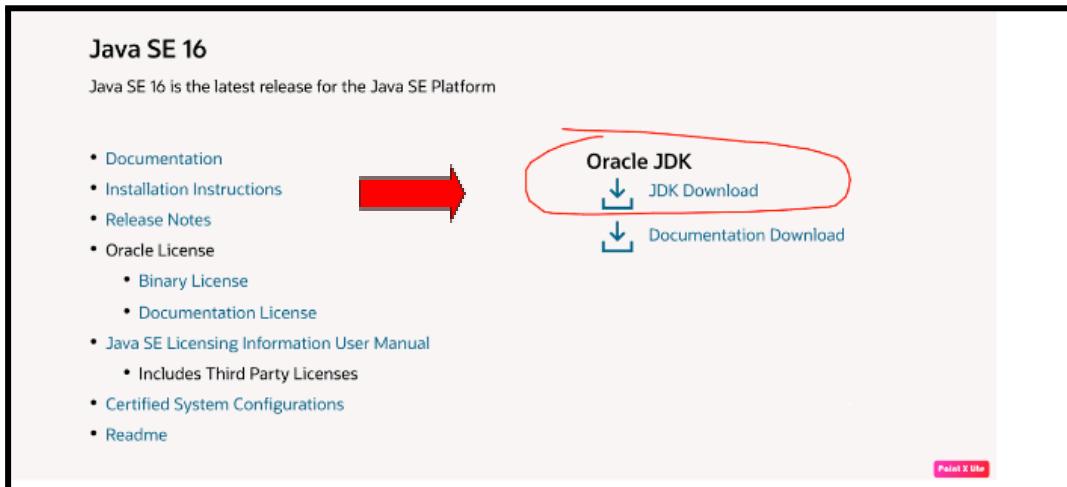


¿Cuál es la razón de hacer esto? La razón es que cuando nosotros vamos a la sección de Debug en configurations, Dart por defecto espera nuestro punto de acceso de nombre ‘main.dart’ dentro de la carpeta ‘bin’.



II. INSTALAR LENGUAJE DART EN ANDROID STUDIO

PASO 1: iremos a Oracle Oficial y hacemos click en descargar Platform (JDK) 16 o la versión más actual en el momento. Damos clic en JDK DOWNLOAD.



Nos aparecerá la siguiente ventana donde encontraremos el JDK para diferentes SO, seleccionamos el que corresponda con nuestro sistema operativo y damos clic sobre este.

Java SE Development Kit 16		
This software is licensed under the Oracle Technology Network License Agreement for Oracle Java SE		
Product / File Description	File Size	Download
Linux ARM 64 RPM Package	144.84 MB	 jdk-16_linux-aarch64_bin.rpm
Linux ARM 64 Compressed Archive	160.69 MB	 jdk-16_linux-aarch64_bin.tar.gz
Linux x64 Debian Package	146.14 MB	 jdk-16_linux-x64_bin.deb
Linux x64 RPM Package	152.96 MB	 jdk-16_linux-x64_bin.rpm
Linux x64 Compressed Archive	170 MB	 jdk-16_linux-x64_bin.tar.gz
macOS Installer	166.56 MB	 jdk-16_osx-x64_bin.dmg
macOS Compressed Archive	16716 MB	 jdk-16_osx-x64_bin.tar.gz
Windows x64 Installer	150.55 MB	 jdk-16_windows-x64_bin.exe

Al dar clic se abrirá esta ventana, en la cual debemos de aceptar los acuerdo de uso de la licencia de Oracle, Aceptamos y descargamos



Una vez descargado lo ejecutamos y lo instalamos, no tiene ninguna complejidad.

Nota importante: Recomiendo dejar la ruta de instalación por defecto, copiamos la ruta y la pegamos en un archivo de texto porque lo usaremos mas adelante.

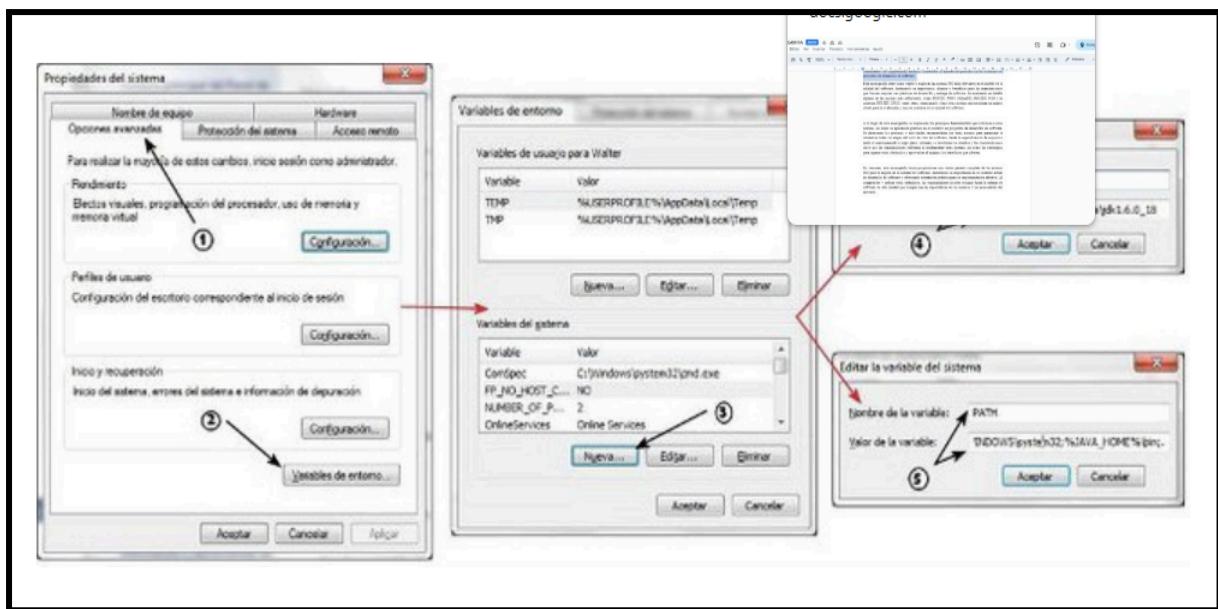
PASO 2:Dentro de este caso vamos a establecer la variable JAVA_HOME,para eso seguimos las siguientes indicaciones:

1.- Vamos al Panel de control > Sistema y Seguridad > Sistema. En la opción Configuración avanzada del sistema, haga clic en Variables del entorno.

2.- Ahora, debe modificar la variable “path” en Variables del sistema para que también contenga la ruta al entorno Java. Seleccione la variable path y haga clic en el botón Editar.

3.- Verá una lista de diferentes rutas, haga clic en el botón Nuevo y luego agregue la ruta donde está instalado java recuerden que les mencione que copiamos la ruta así que la pegamos “C:\Program Files\Java\jdk(version)\bin“. En el caso, has instalado Java en cualquier otra ubicación, luego agrega esa ruta.

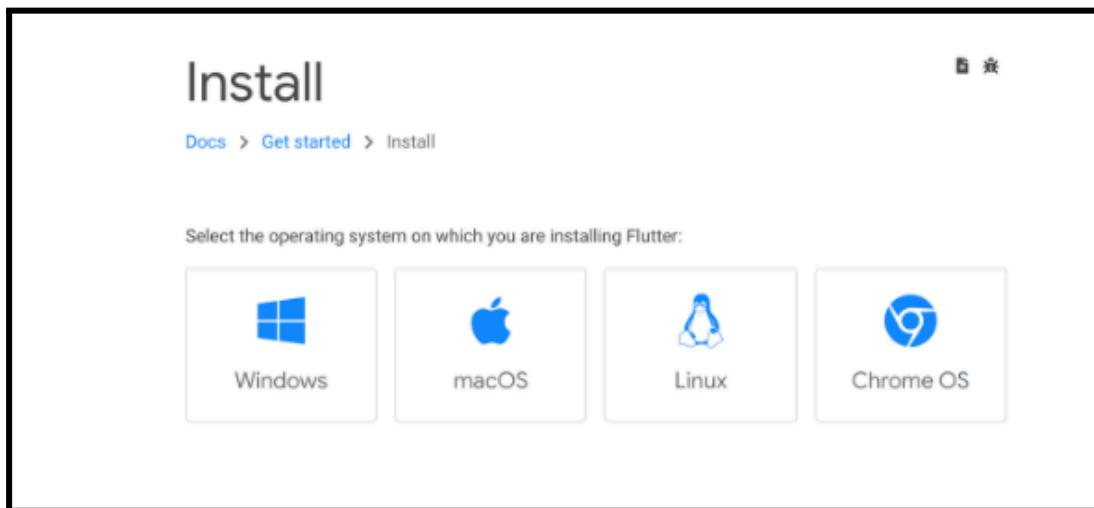
4.- Por último Haga clic en Aceptar, guarde la configuración y listo!!



PASO 3 :Descargar SDK de Flutter

Ahora vamos a descargar Flutter dependiendo de nuestro sistema operativo Descargar Flutter damos clic sobre el icono. Una vez que descargado el archivo lo movemos a una carpeta dentro de nuestros discos y ahí lo descomprimimos

Nota importante: Copiamos la ruta en donde está nuestro SDK Flutter y lo pegamos en un archivo de texto porque lo usaremos más adelante.



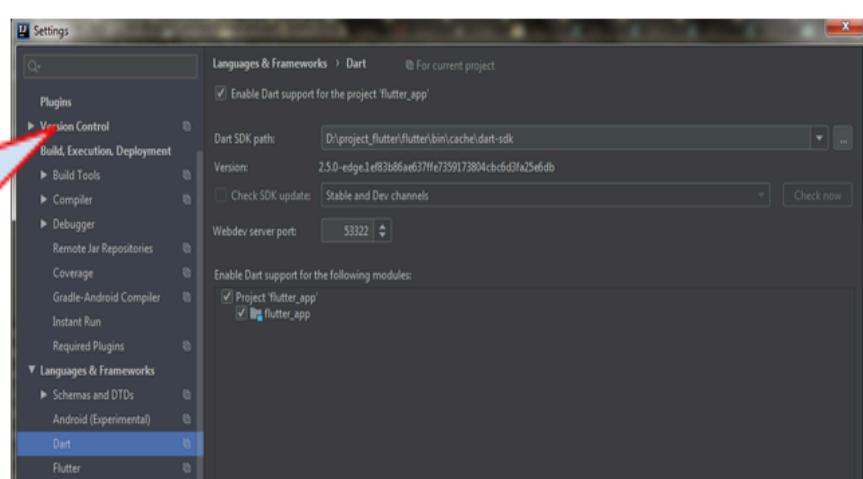
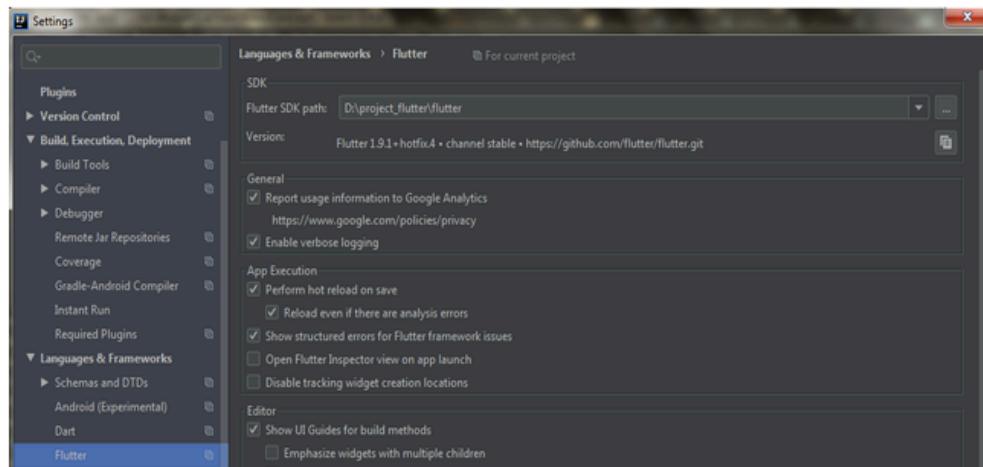
PASO 4: Descargar intelliJ IDEA y configurar Flutter

Para empezar vamos a la página oficial de jetbrains para descargar intelliJ Descargar IntelliJ IDEA existen 2 tipos:

Ultimate => Para desarrollo web y empresarial

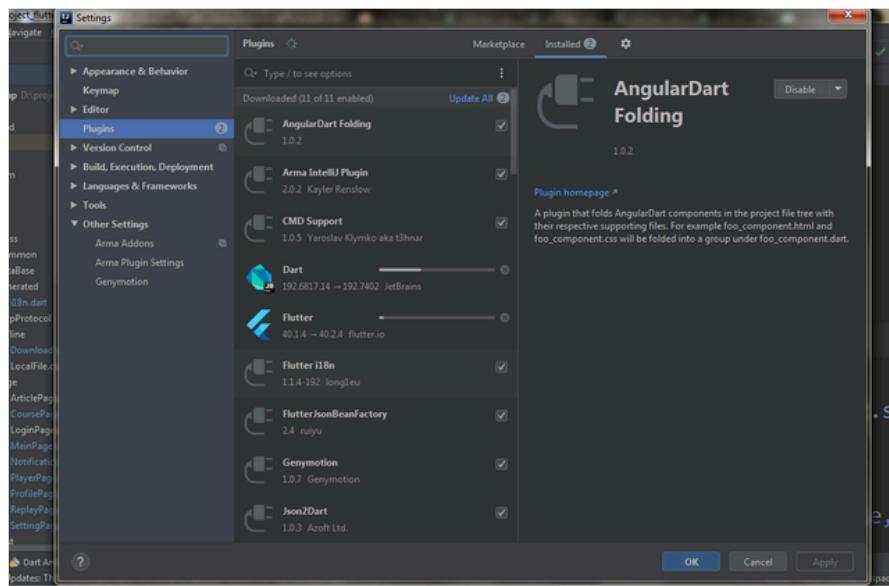
Community => Para desarrollo JVM y Android

Descargamos la versión Community dependiendo de nuestro sistema operativo, la instalación es sencilla, al finalizar abrimos intelliJ Idea y nos vamos a File/Setting en windows para linux y Mac nos vamos a IntelliJ IDEA/Preferences después damos clic en Languages & Frameworks y seleccionamos Flutter y vamos agregar la ruta en Flutter SDK path., recuerden que anteriormente les comente que la pegaramos en un archivo de texto.



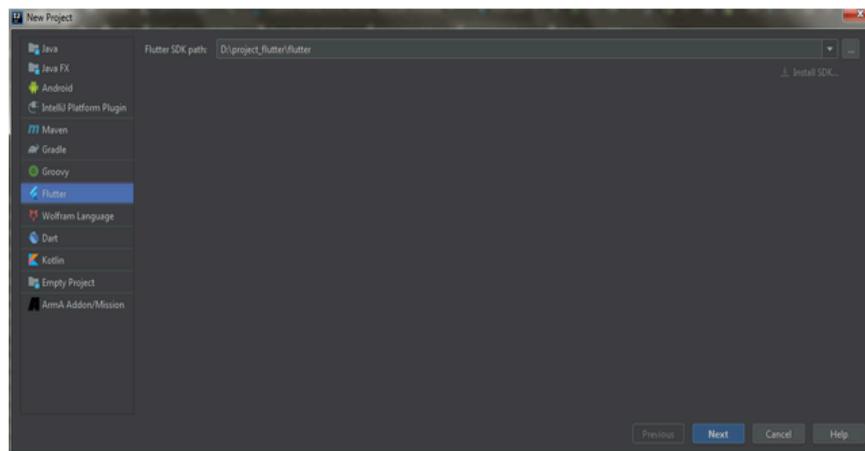
Ahora seleccionamos Dart activamos la casilla Enable Dart support for the project y colocamos la misma ruta con una pequeña modificación seleccionando dart-sdk como se muestra en la imagen:

PASO 5:Vamos a instalar los Plugin de Dart y Flutter.

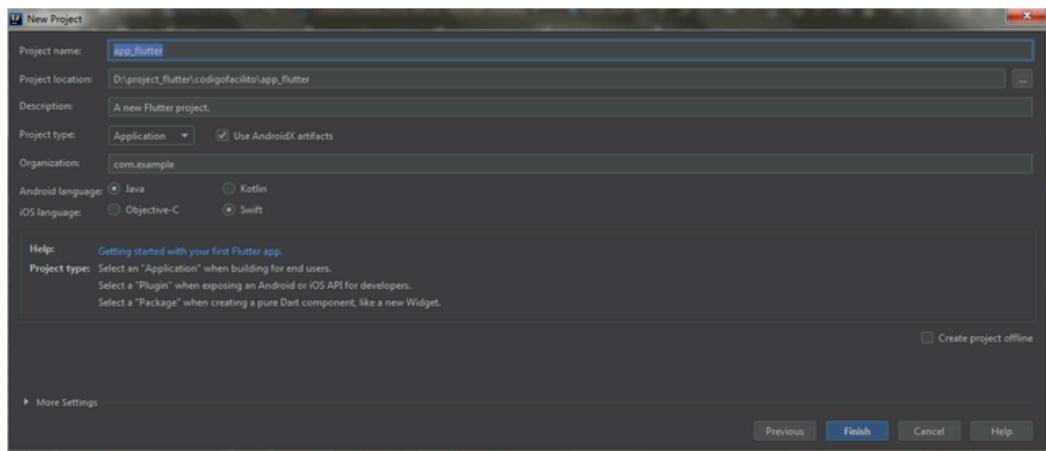


Crear un proyecto

Cuando se abra IntelliJ IDEA creamos un nuevo proyecto nos vamos a “File/Project” a la izquierda seleccionaremos Flutter nos mostrará la ruta del SDK flutter en la parte de abajo damos clic en Next:

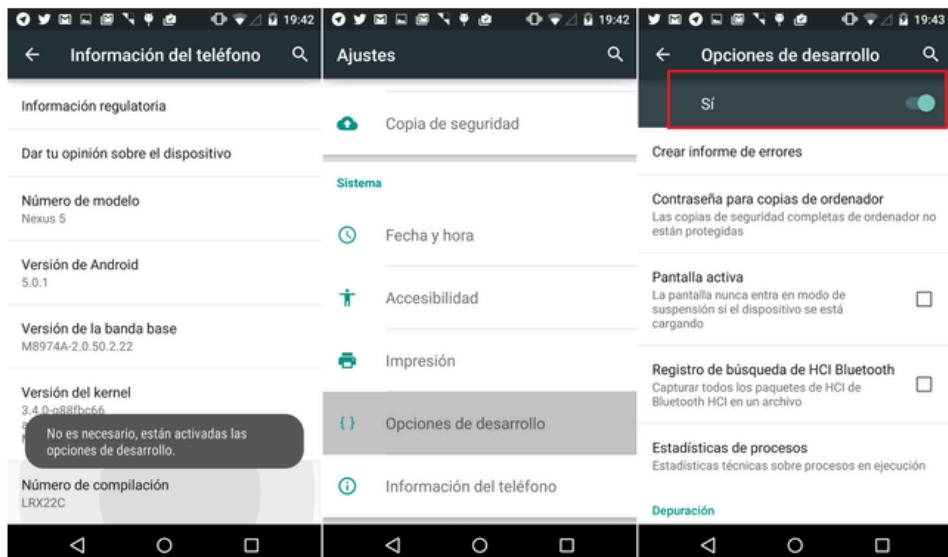


Colocamos el nombre del proyecto ‘flutter_app’ y seleccionamos los lenguajes que deseamos usar en nuestro proyecto android e ios y damos clic en Finish y listo tenemos nuestro proyecto en flutter.

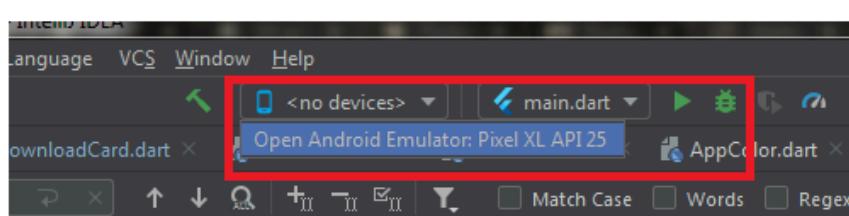


PASO 6: Necesitamos un dispositivo con android lo conectamos y nos vamos a Ajustes > Información del dispositivo y pulsar siete veces sobre el número de compilación.

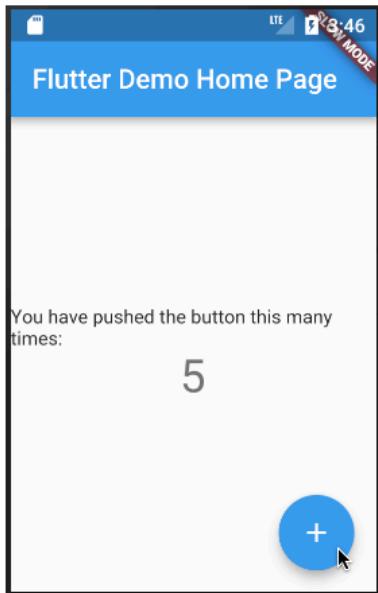
Una vez hecho esto nos saldrá el mensaje ¡Ahora eres un desarrollador! y ya nos mostrará en los Ajustes el apartado de Opciones de desarrollo. Seleccionamos y activamos el modo la Depuración.



Esperamos que reconozca el dispositivo.



Y damos clic en Run cargara y se mostrará nuestra aplicación.



III.COMPARACIÓN

Tanto Visual Studio Code como Android Studio son entornos de desarrollo populares para trabajar con el lenguaje Dart y Flutter, pero cada uno tiene su propio enfoque y características. Al poder instalarlas las diferencias y similitudes en el proceso de instalación y uso del lenguaje Dart en ambos IDEs son las siguientes:

1. Instalación en Visual Studio Code

1.1 Similitudes:

Extensiones: Ambos IDEs utilizan extensiones o plugins para proporcionar soporte para Dart y Flutter. En Visual Studio Code, debes instalar la extensión de Dart y la de Flutter desde el Marketplace.

Configuración: En ambos casos, una vez instaladas las extensiones, necesitas configurar el entorno de desarrollo, como establecer el SDK de Dart y Flutter.

1.2 Diferencias:

Simplicidad: Visual Studio Code es generalmente más ligero y rápido en comparación con Android Studio. La instalación de las extensiones para Dart y Flutter suele ser directa y rápida.

Interfaz: VS Code tiene una interfaz más minimalista y basada en extensiones. Las extensiones para Dart y Flutter se manejan a través de su Marketplace integrado, y la configuración es más sencilla.

Herramientas de desarrollo: VS Code es más flexible y permite una personalización profunda con otros plugins y configuraciones. Sin embargo, puede carecer de algunas herramientas integradas avanzadas que están disponibles en Android Studio.

2. Instalación en Android Studio

2.1 Similitudes:

Plugins: Al igual que VS Code, Android Studio también usa plugins para el soporte de Dart y Flutter. Debes instalar el plugin de Flutter, que a su vez instalará el plugin de Dart si aún no está instalado.

Configuración: Ambos entornos requieren que configures el SDK de Dart y Flutter para que el entorno de desarrollo funcione correctamente.

2.2 Diferencias:

Entorno más integrado: Android Studio ofrece un entorno más completo y con más características integradas, como herramientas avanzadas para depuración, diseño de interfaces y gestión de dependencias. Estas herramientas son parte del IDE y no dependen de plugins adicionales.

Configuración inicial: La instalación de Android Studio y la configuración de los plugins puede ser un poco más compleja debido a la mayor cantidad de opciones y configuraciones, pero a la vez, ofrece una integración más profunda con herramientas de desarrollo.

Rendimiento: Android Studio es más pesado en comparación con VS Code, ya que está basado en IntelliJ IDEA y ofrece muchas más características integradas, lo que puede impactar el rendimiento en sistemas con recursos limitados.

IV.CONCLUSIÓN

Visual Studio Code me parece mucho más ligero, con una configuración más sencilla y rápida a través de extensiones. Es ideal si prefieres un entorno minimalista y flexible.

Por otro lado, Android Studio ofrece un entorno de desarrollo más completo con herramientas avanzadas integradas, ideal si necesitas una integración más profunda y funcionalidades avanzadas, pero su configuración es mucho más tediosa y extensa.

En mi opinión ambos entornos pueden ser utilizados eficazmente para el desarrollo con Dart y Flutter, y la elección entre uno u otro a menudo depende de las preferencias personales y las necesidades específicas del proyecto.