

Link Download Doc Codelab



Flutter

Flutter è un framework open-source di Google per creare User Interface native su un'unica unica code base, unico codice e possiamo compilare applicazioni su più piattaforme(Android, iOS, Windows, macOS, Linux, Web).

Vantaggi:

Prestazioni e compatibilità

- Le applicazioni sviluppate con Flutter girano nativamente sui dispositivi, questo vuol dire che le prestazioni sono elevate(frame rate elevato), il rendering è a basso livello e utilizzando la libreria grafica di Google, Skia.
- Flutter si interfaccia con gli SDK del sistema operativo su cui gira(Android o iOS) quindi è anche possibile creare funzionalità specifiche rispetto al sistema su cui l'applicazione gira.

Debug

- In Flutter, il Debug è molto semplice, durante lo sviluppo è possibile visualizzare ed eseguire l'applicazione in modalità **debug mode**
- **Flutter inspector**, è uno strumento potente per visualizzare il layout della vostra applicazione in sviluppo, oltre che per visualizzare il layout è molto utile per diagnosticare e intercettare problemi di layout dell'app.
- Una funzionalità apprezzata dagli sviluppatori è "l'**hot-reload**" che consente di iniettare e visualizzare istantaneamente le modifiche fatte dall'editor sull'esecuzione dell'app, senza dover ricompilare l'intero codice. Questo ne ottimizza il workflow dello sviluppatore e le varie fasi di debug.

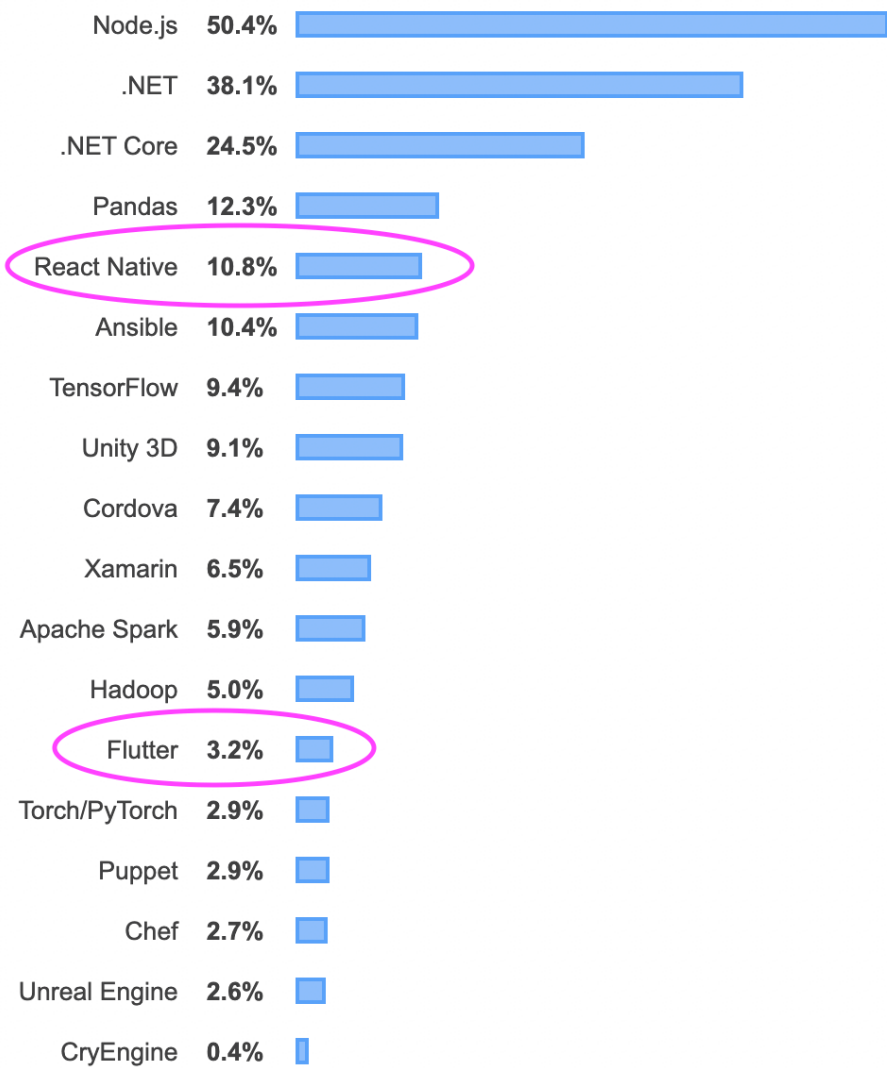
Curva di apprendimento

- Flutter è molto più **facile da imparare**(Dart) rispetto ad altri frameworks in cui si usa javascript come linguaggio per React Native
- Il linguaggio Dart è un linguaggio molto più **intuitivo** e anche **più vicino ai paradigmi di programmazione** ed è molto simile ai linguaggi di programmazione usati nello sviluppo di app mobili native(Java o Kotlin)

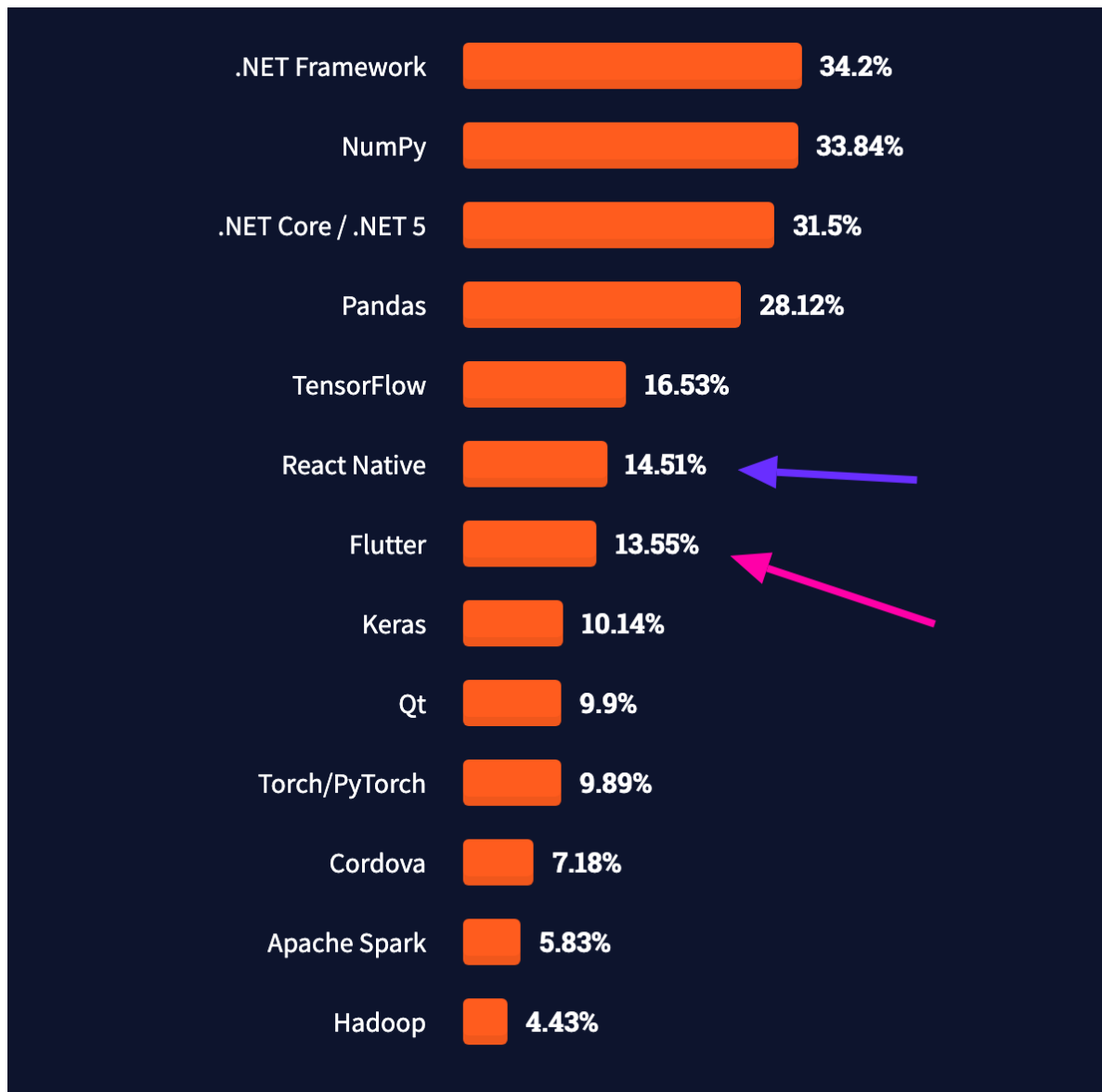
Popolarità e community

- La popolarità di Flutter è in continua **crescita** da quando è disponibile per tutti gli sviluppatori infatti è possibile constatare da sondaggi fatti su Stack Overflow che l'interesse da parte degli sviluppatori è in continua crescita:

Source: Stack Overflow Survey 2019

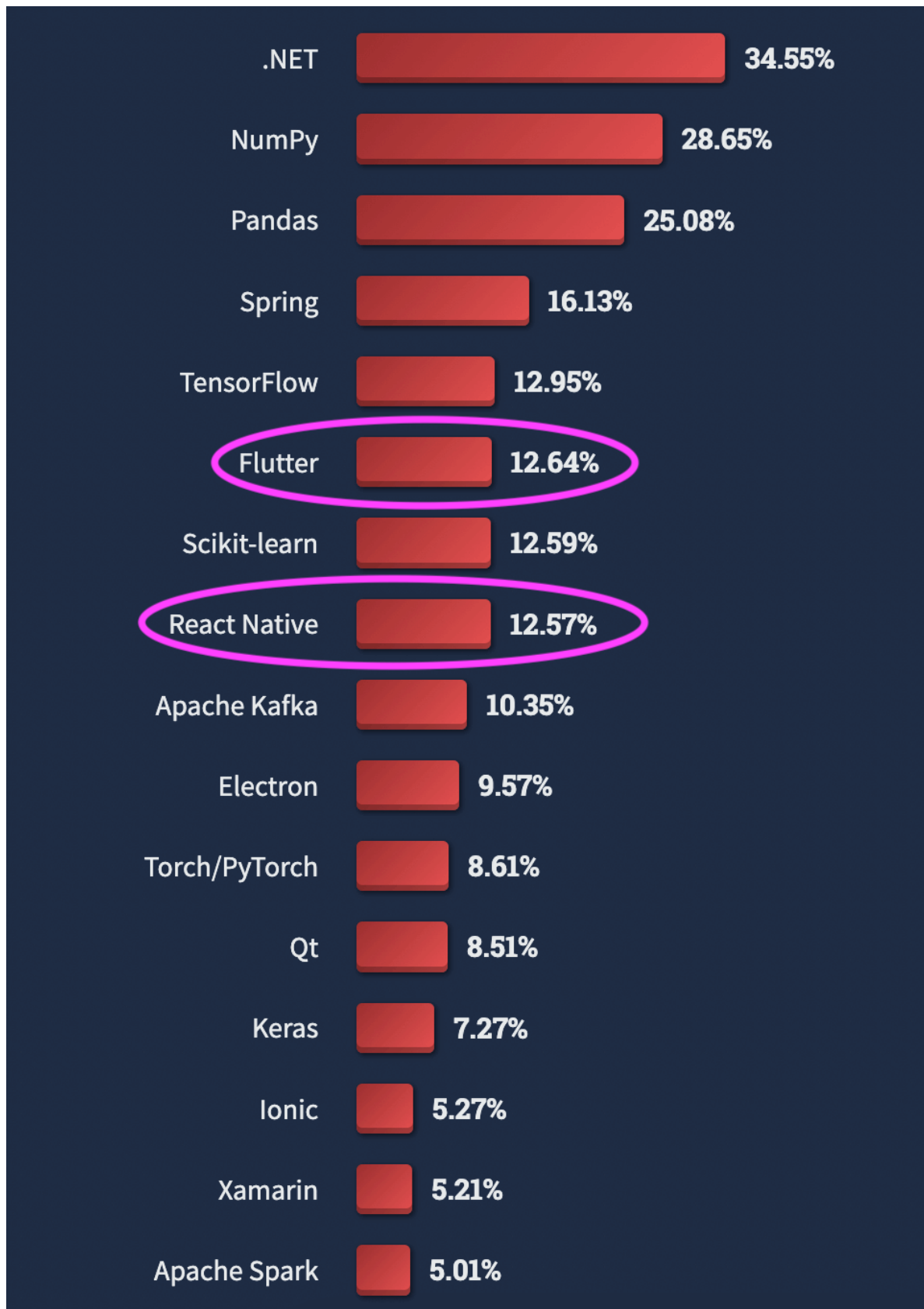


Stack Overflow Survey 2021



Nell'ultimo sondaggio su Stack Overflow notiamo che Flutter e React Native sono in stretta competizione

Stack Overflow Survey 2023



- Essendo anche la Community di Flutter in costante crescita dalla sua uscita, è possibile ritrovare sul web una vasta community di sviluppatori e librerie da integrare per i progetti Flutter

Codelab di oggi

Il codelab di oggi consisterà nel:

- Installare flutter
- Creare la prima app
- Capire le basi del funzionamento di Flutter
- Creare un layout per la nostra prima app
- Collegare le interazioni dell'utente al comportamento dell'app(usando ad esempio un pulsante e visualizzando il risultato a schermo)
- Organizzare il codice Flutter
- Gestire gli stati dell'applicazione(dati)
- Rendere la tua app reattiva per schermi diversi

[Fonte codelab:](#)

Installazione e configurazione

Probabilmente questa sarà la parte più difficile di questo codelab, ovvero installare tutto e assicurarsi che funzioni. È l'unica cosa meno divertente ma è importante per iniziare a sviluppare applicazioni in Flutter.

2) Installazione di Flutter

Per prima cosa scarichiamo l'sdk di flutter da [questo indirizzo](#). Una volta scaricato, estrarre il contenuto dell'archivio in:

C:\SDKs

Creare una cartella SDK per posizionare l'SDK di Flutter. La posizione finale sarà:

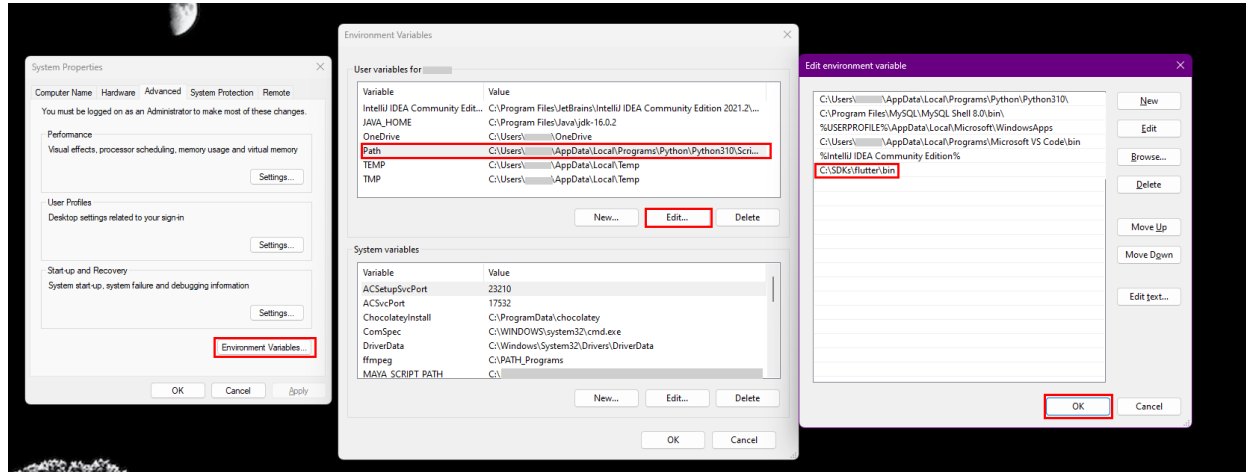
C:\SDKs\flutter

3) Configurare Path di sistema

È estremamente importante aggiornare il PATH nelle variabili di sistema.

Per eseguire i comandi Flutter attraverso la console di Windows sarà necessario aggiungere al PATH la variabile di ambiente. Nella barra di ricerca di Windows inserisci “env” e seleziona “**Modifica variabili di ambiente**” e aggiungi questo percorso:

C:\SDKs\flutter\bin



Per verificare che tutto funzioni, chiudere tutte i terminali Windows eventualmente aperti, aprire un Terminale ed eseguire il comando:

flutter doctor

Se si dovesse verificare un errore con codice “**Error: Unable to find git in your PATH.**” segui la guida chiamata “gitError/GitError.pdf” all’interno della cartella “CodelabFlutterUni-main”.

4) Installazione Tool Sviluppo Windows

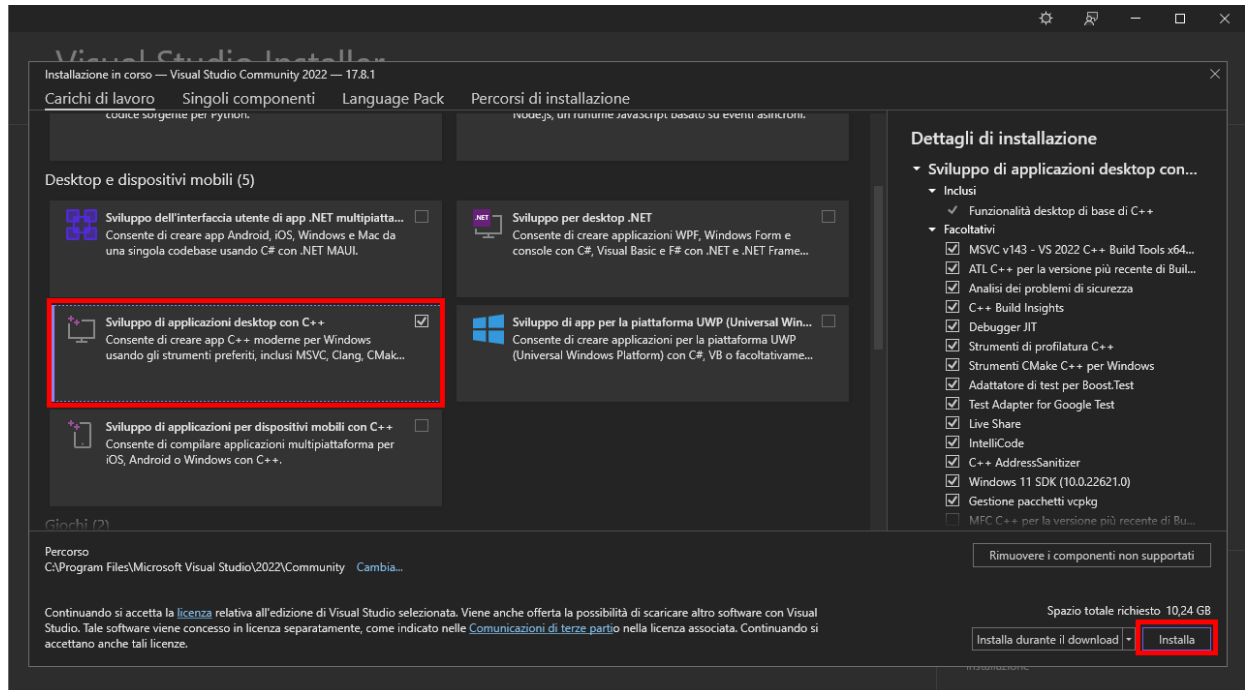
In questo Codelab scriveremo una piccola app per Windows e Flutter per compilare la base di codice avrà bisogno del kit di sviluppo Windows.

Scaricare il setup del kit di sviluppo windows Visual Studio 2022, [Scaricabile da qui.](#)

Attenzione non stiamo parlando dell’editor Visual Studio Code!.

Una volta scaricato il **setup di Visual Studio 2022** bisogna verificare nelle impostazioni di installazione che “Desktop development with C++” sia selezionato, incluse tutte le

componenti di default.



5) Installazione di Visual Studio Code

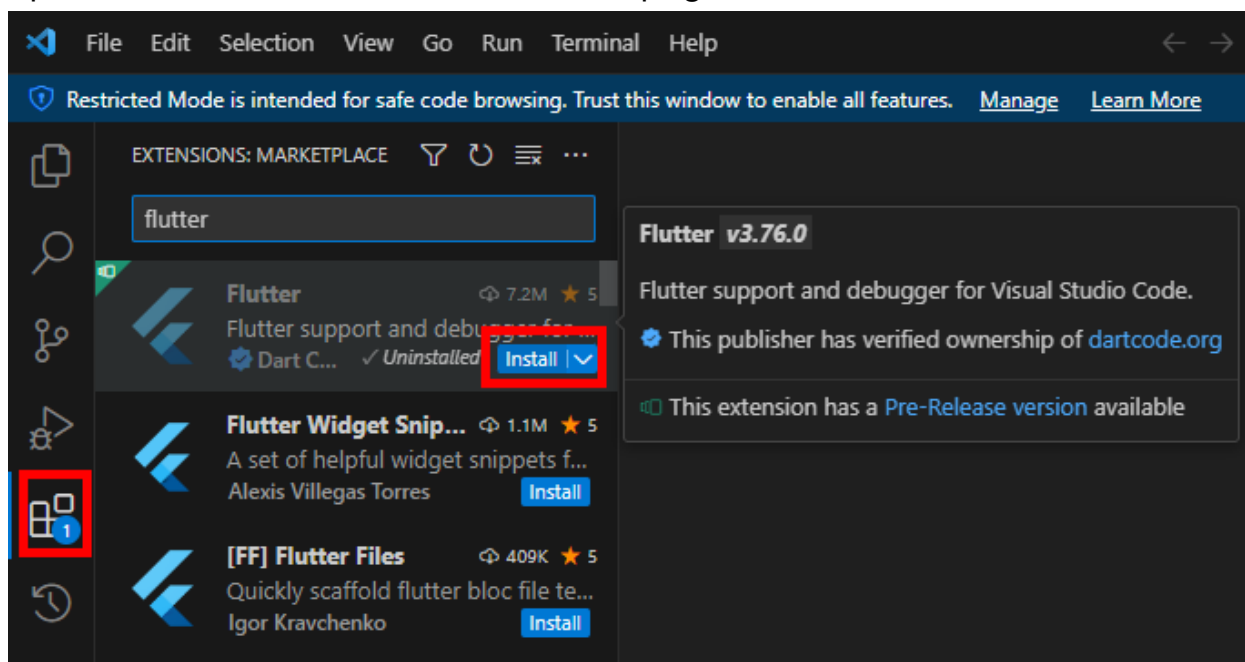
Useremo Visual Studio Code perchè è semplice, affidabile e lo usano molte persone.

Chiaramente è gratis.

[Link installazione](#)

6) Installazione Plugin Flutter

Apri l'editor di Visual Studio Code e installa il plugin flutter



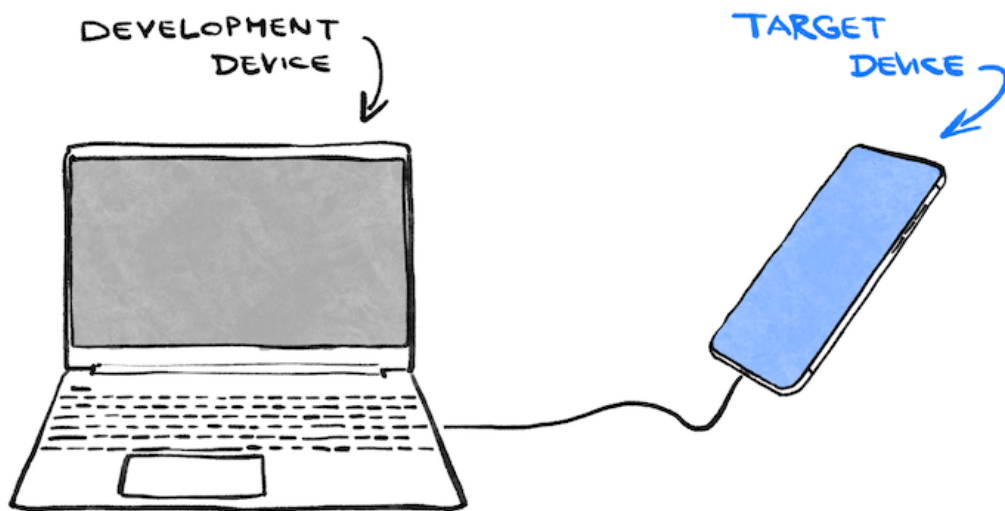
Considerazioni sviluppo con Flutter

Quando svilupperete un'app spesso non la svilupperete per tutte le piattaforme, ma è importante scegliere una piattaforma target iniziale su cui sviluppare e magari in seguito adattare l'app per tutte le altre piattaforme.

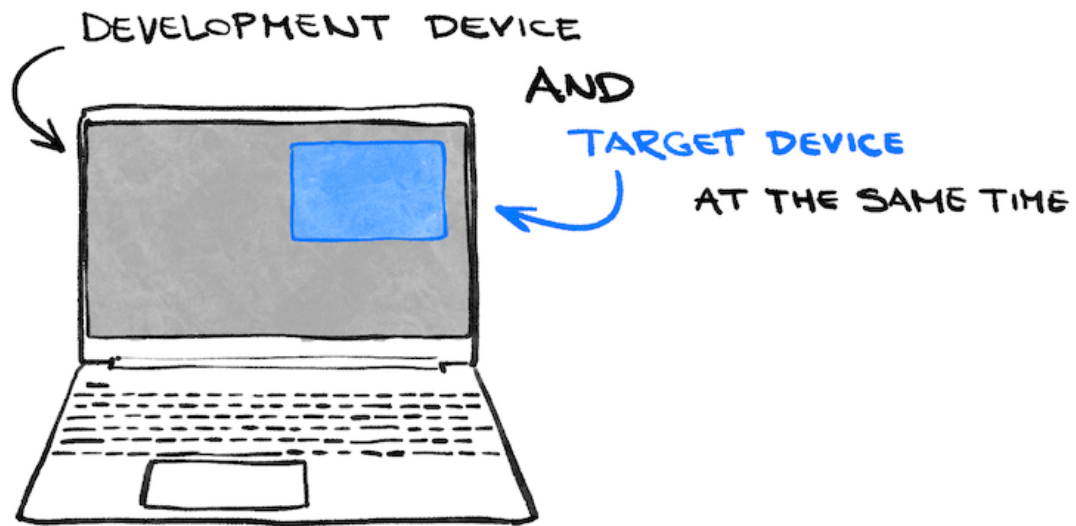
In questo caso sceglieremo Windows come **Target Device** e **Development Device**.

Un altro suggerimento è che se state sviluppando un'app per computer Windows allora usate un sistema operativo Windows per farlo. Questo rende alcune cose più facili e veloci.

Esempio **Development Device** e **Target Device**



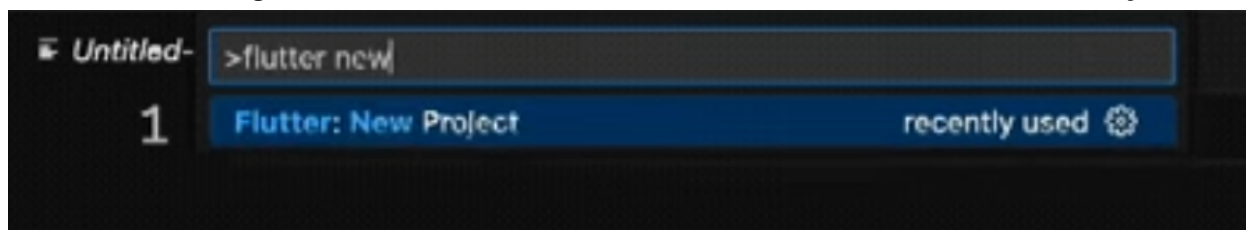
Esempio **Development Device** usato sia per lo sviluppo che come **Target Device**



Nuovo progetto

Creazione del progetto

Avvia Visual Studio Code, aprire la command palette (con F1 o Ctrl+Shift+P o Shift+Cmd+P). Digitare “**flutter new**” e selezionare il comando “**Flutter: New Project**”



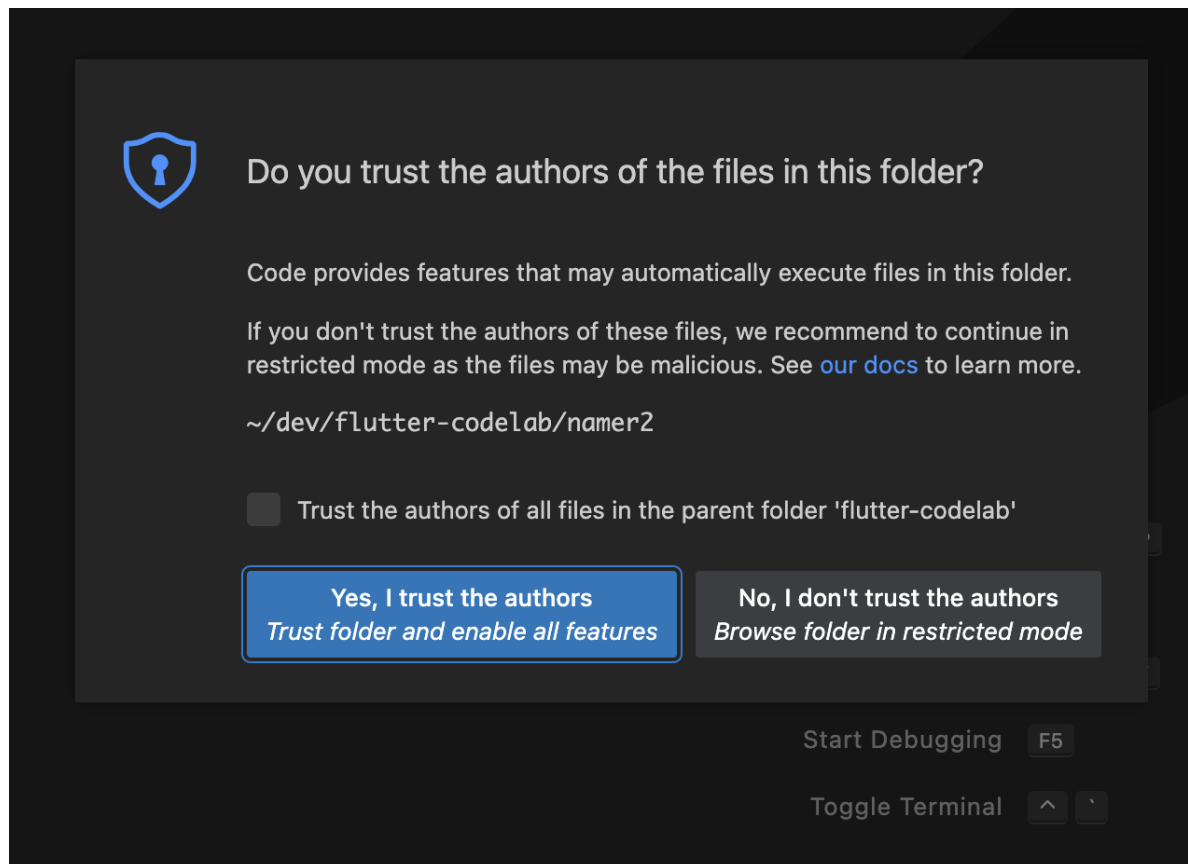
Dopo selezionare “**Application**”, seleziona ora una cartella in cui creare un nuovo progetto. Per comodità nella cartella Documenti Creiamo una cartella chiamata Flutter e la selezioniamo.

Infine dai un nome alla tua app come `flutter_app` o `flutter_application_1`.

A questo punto flutter procederà con la creazione del progetto. Ci potrebbero volere alcuni minuti. Appena il progetto sarà creato VS lo aprirà.

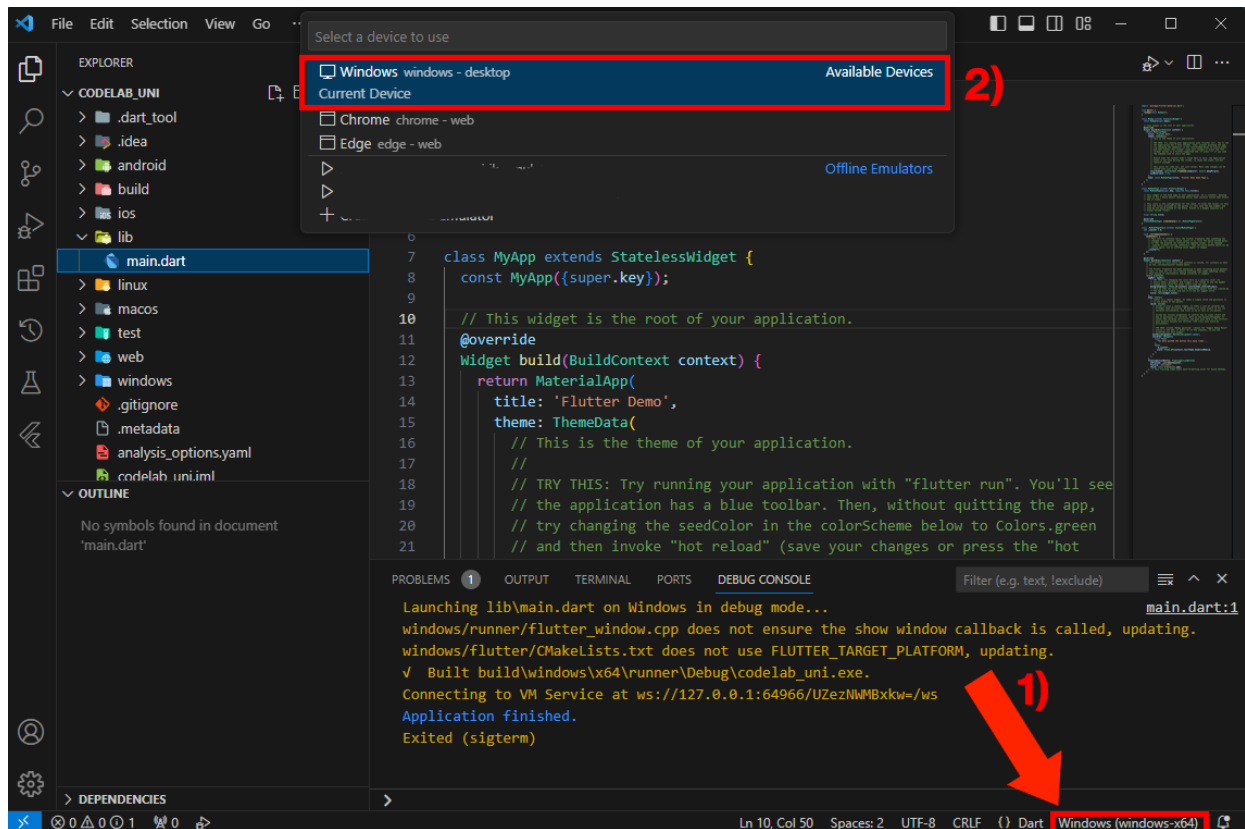
Nota: VS Code mostrerà una finestra chiedendo se può considerare attendibile il contenuto della cartella. Selezionare Sì. Se premi no verrà disabilitata la possibilità di

usare Flutter nella cartella.



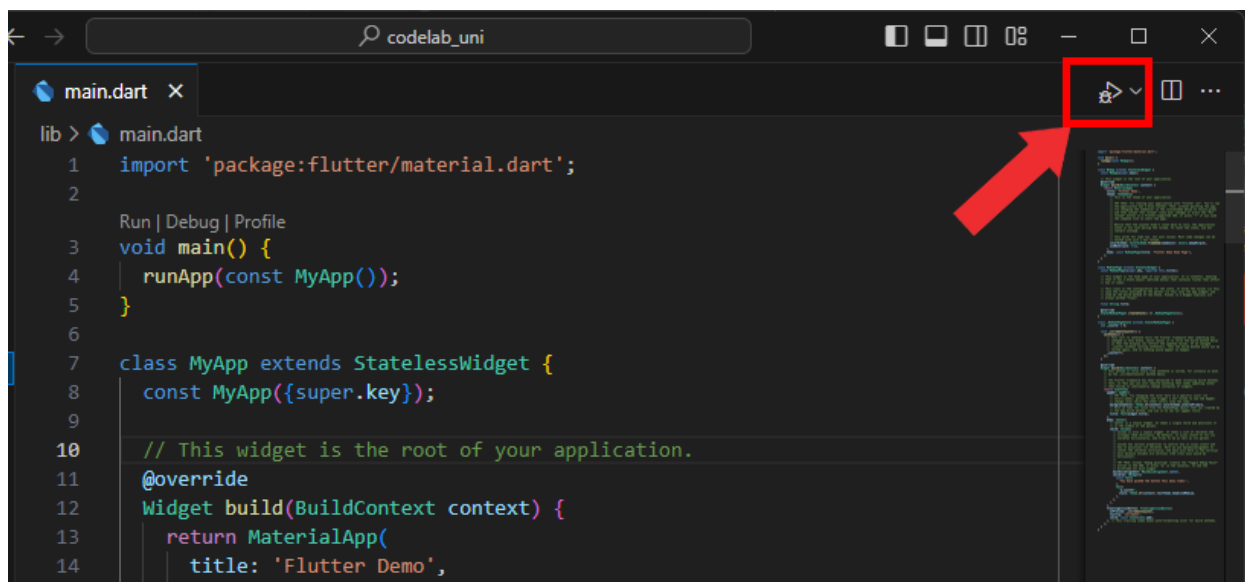
Selezione Target Device

Per selezionare il **target device**, (1) premere nell'angolo in basso a destra di VS Code. (2) Selezionare quindi **Windows - desktop** che dovrebbe comparire tra i device disponibili.



Avvia l'app(Debug)

Per avviare l'app in modalità Debug assicurarsi che sia aperto il file `lib/main.dart` e premere il pulsante in alto a destra della finestra di VS Code



Dopo circa un minuto l'app verrà avviata l'app in modalità debug. Potremo visualizzare

quindi la demo Flutter preimpostata mentre viene eseguita nella finestra.

