**Mémo Flutter**

**Installation environnement**

\* Installer manuellement le SDK flutter

https://docs.flutter.dev/get-started/install/windows

n'installez pas Flutter sur un chemin contenant des caractères spéciaux ou des n'installez pas Flutter dans un répertoire C:\Program Files\qui nécessite des privilèges élevés.

\* Rajouter le chemin flutter/bin dans la variable d’environnement PATH

\* Lancer **flutter doctor** dans un terminal de commande pour finir et controler l’installation.

-->Traiter les anomalies

*C:\Users\stephane.chevrier>flutter doctor*

*Doctor summary (to see all details, run flutter doctor -v):*

*[√] Flutter (Channel stable, 3.7.12, on Microsoft Windows [version 10.0.19044.2846], locale fr-FR)*

*[X] Windows Version (Unable to confirm if installed Windows version is 10 or greater)*

*[√] Android toolchain - develop for Android devices (Android SDK version 34.0.0-rc3)*

*[√] Chrome - develop for the web*

*[X] Visual Studio - develop for Windows*

*X Visual Studio not installed; this is necessary for Windows development.*

*Download at https://visualstudio.microsoft.com/downloads/.*

*Please install the "Desktop development with C++" workload, including all of its default components*

*[√] Android Studio (version 2022.2)*

*[√] IntelliJ IDEA Ultimate Edition (version 2023.1)*

*[√] VS Code, 64-bit edition (version 1.65.2)*

*[√] Connected device (3 available)*

*[√] HTTP Host Availability*

*! Doctor found issues in 2 categories.*

\* Installer AndroïdStudio

https://developer.android.com/studio

\* Configurer l’émulateur Androïd

\* Accepter les licences Androïd

**flutter doctor –android-licenses**

\* Lancer **flutter doctor**

-->Traiter les anomalies

\* **Dans Intelllij, installer les plugins flutter et dart**

Anomalie rencontrée :

Je n’avais pas la bonne version du JDK java

→ installation java jdk 20

→ mettre à jour la variable d’environnement JAVA\_HOME

**Dart**

\* Principe de fonctionnement :

on définit un arbre de widget.

Flutter detecte le changement d’un widget et va reconstruir tout l’empilement de manière optimisé.

On peut mettre une logique dans un widget : si condition alors l’enfant est toto sinon c’est lulu

\* Logique proche de java, mais avec des différences.

\* La valeur null ne peut pas exister pour certains types.

Il faut enrichir le type par ‘ ?’ pour permettre la possibilité d’avoir nu.

Exemple :

String str = null → erreur compilation

String ? str = null → ok

String ? Str → str = null par défaut, seul cas de valeur par defaut.

str ?.length permet de renvoyer null si la chaine str est vide.

\* Aucune valeur par défaut sauf type ? → null

\* 3 sortes de paramètres dans une fonction :

- **positionnel** comme dans java : l’ordre des parametres est obligatoires

f(double x, double y) → f(1,2) → x=1, y=2

- **nommés**

f({double? x, double? y} → f(x:1) → sous entendu y=null

f({ required double x, double? y} → x obligatoire, y optionnel

- **optionnel**

f(double x, [double y]) → parametre y optionnel

\* les constructeurs :

dans dart pour les utiliser, le ‘new’ est optionnel.