# Annexe 6B – Débogueur avec Android Studio

* Il s’agit d’un outil permettant de découvrir les bogues d’un programme en l’exécutant pas-à-pas ou en l’exécutant jusqu’à certains points d’arrêts.

Vous pouvez utiliser l’exercice de l’annexe 4 ( Transfert intérac amélioré ) pour réaliser l’exercice.

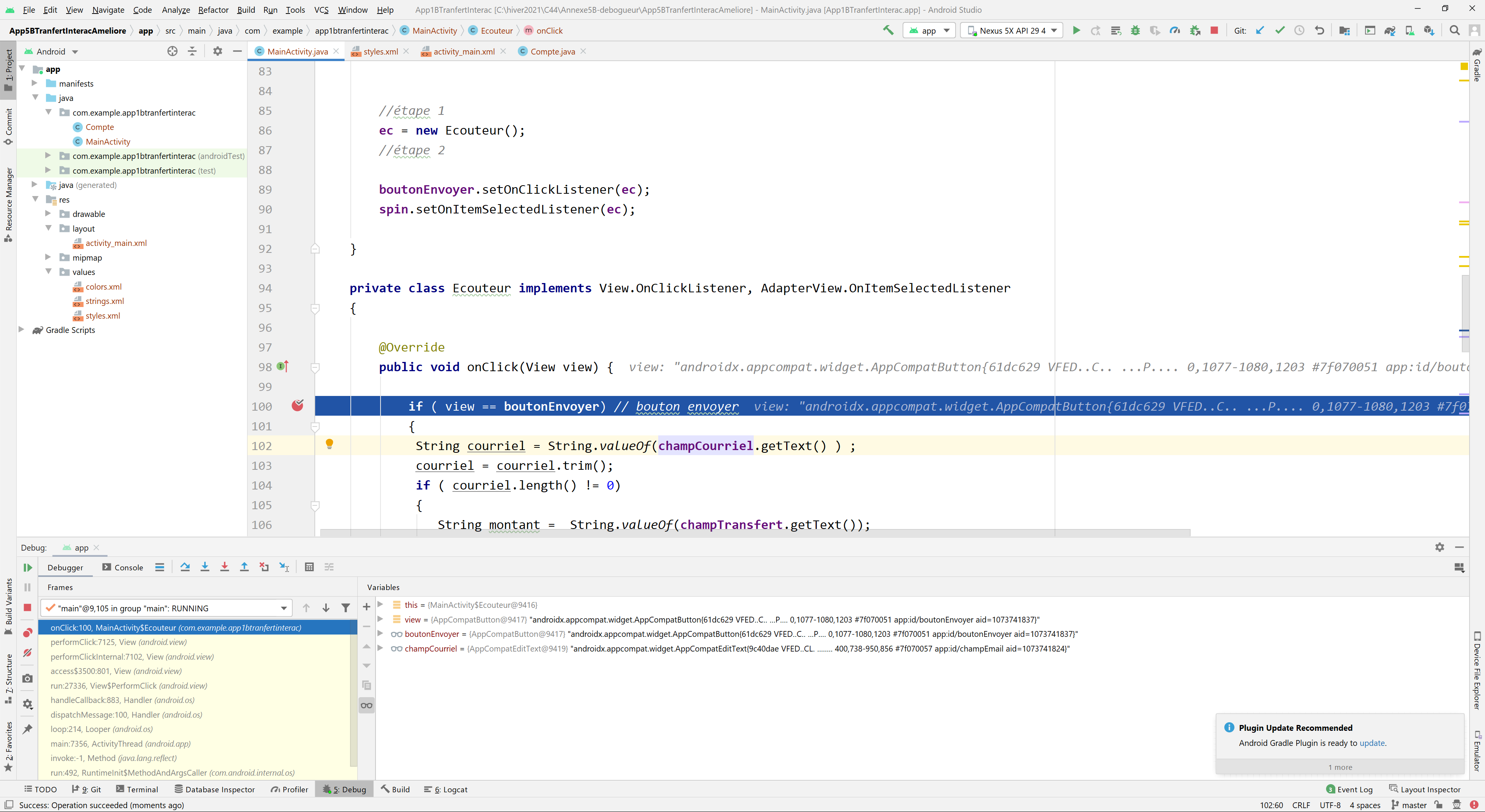
## 1.Point d’arrêt

Supposons qu’on veut s’assurer que le programme entre bien dans la méthode onClick de notre classe-écouteur.

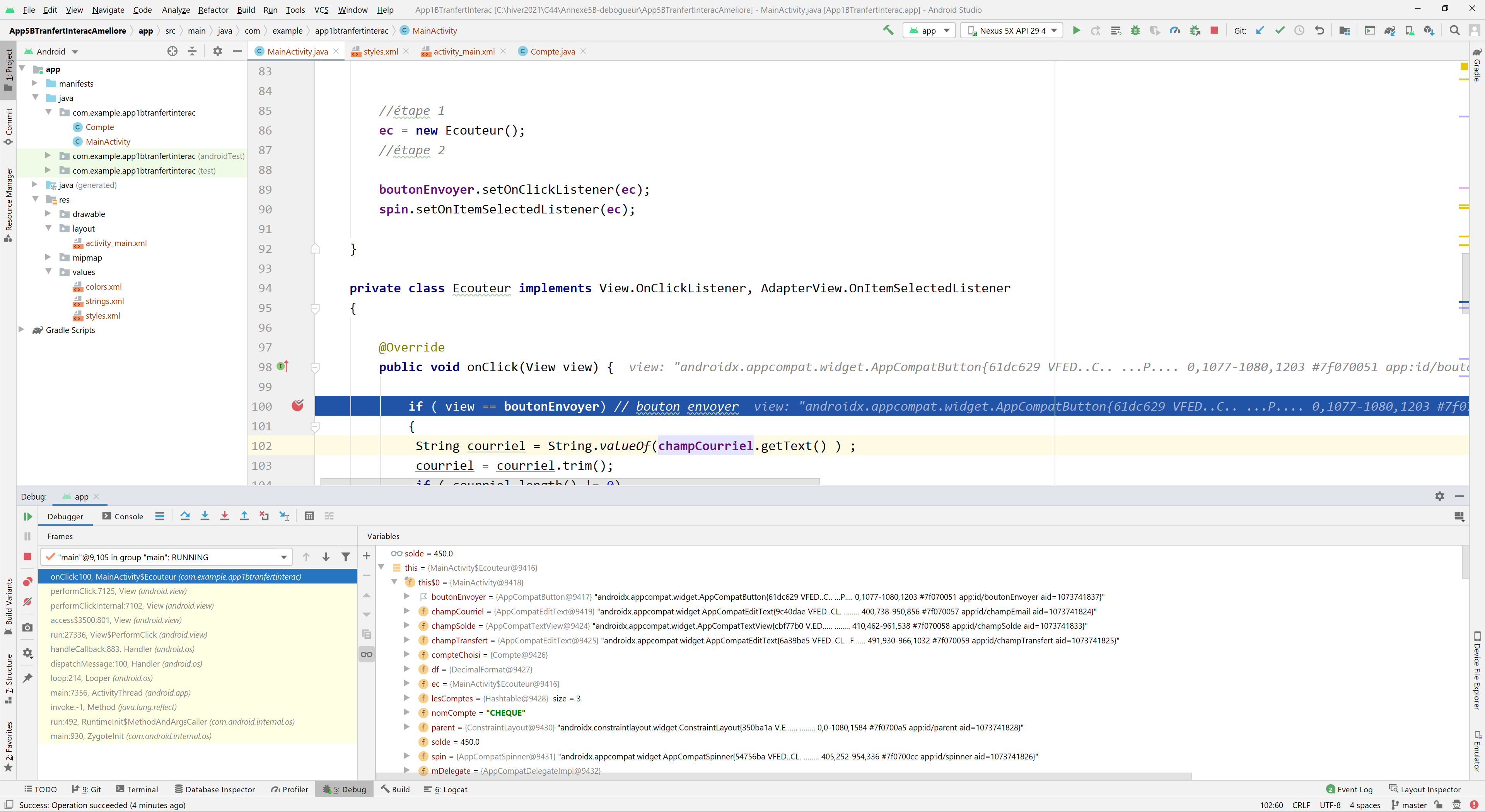
Cliquez dans la marge d’une ligne de code au début de la méthode onClick, un cercle rouge apparaît; il s’agit d’un point d’arrêt

Démarrez le programme avec l’icône Debug plutôt que Run. ( une coccinelle verte )

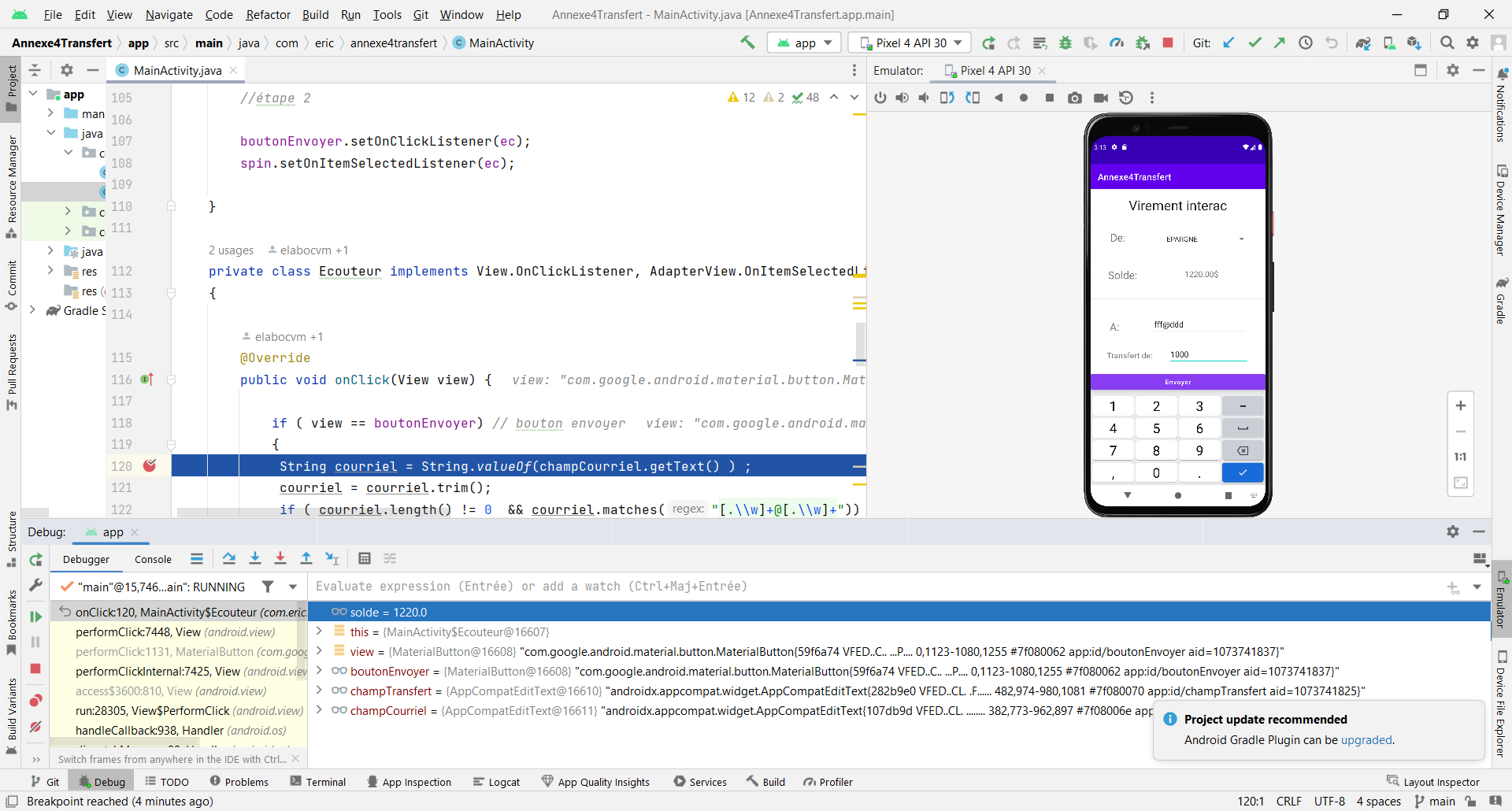
Lorsqu’on appuie sur le boutonEnvoyer, le programme arrête et on passe en mode « debug » :



La fenêtre Variables devrait être apparue en bas de l’éditeur de code. Développez-y le bloc this et this$0 afin de trouver la valeur de la variable solde ( pour moi c’est 450 )



Ca peut devenir embrouillant, une autre technique si on cherche la valeur de la variable solde est de faire bouton de droite 🡪 New Watch et écrire le nom de la variable solde pour toujours avoir sa valeur en temps réel



## 2. Pas-à-pas

On peut ainsi faire le pas-à-pas du programme à partir du point d’arrêt avec les touches F7 et F8 :

F8 : permet d’avancer dans la classe du point d’arrêt. Essayez de le faire à quelques reprises jusqu’à la ligne compteChoisi.getSolde(). Les valeurs des variables peuvent aussi être écrites en italique dans le code.

F7 : permet un déboguage en profondeur, c’est-à-dire qu’on peut aller dans les autres classes appelées dans la classe où se trouve le point d’arrêt.

Vous pouvez donc utiliser F7 lors de la ligne compteChoisi.getSolde de manière à aller dans la méthode getSolde de la classe Compte

## 3. Revenir à l’utilisation du programme

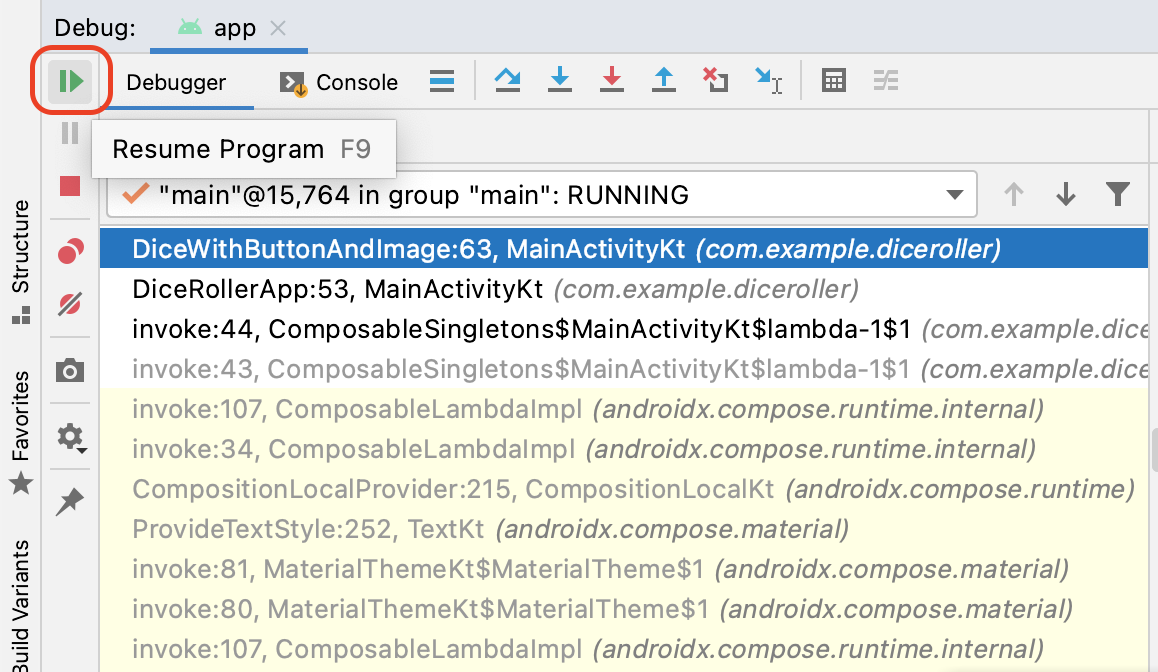
De retour dans l’activité suite à notre détour dans la classe Compte, continuez à faire F8 jusqu’à ce que l’exécution soit terminée. Une ligne jaune pâle devrait apparaître ou vous tombez dans des classes que vous n’avez pas codées comme ceci :



*Je suis passé tout droit ! Je fais F9 pour revenir à l’émulateur*

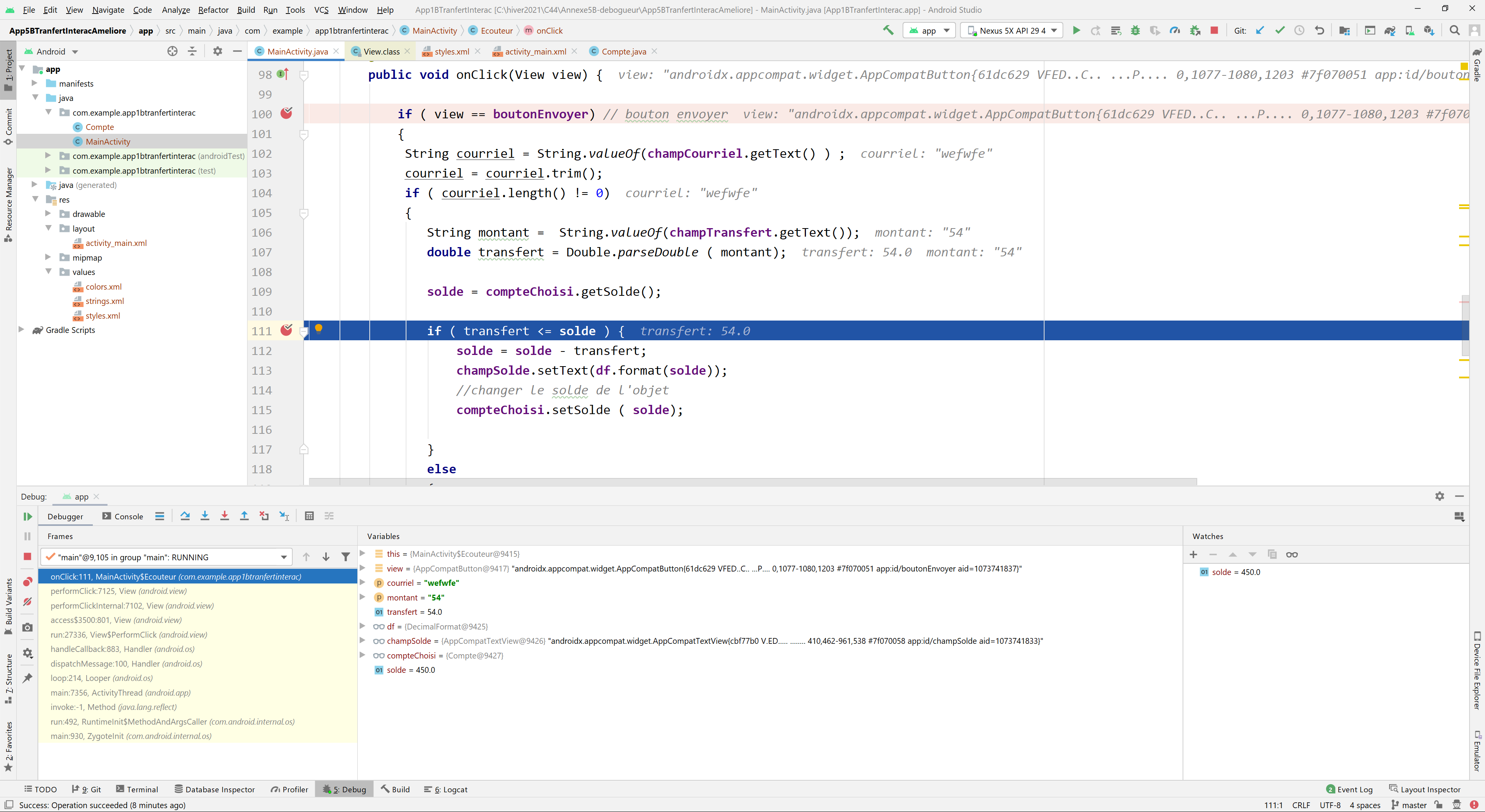
F9 : permet de revenir au programme principal, pour pouvoir continuer à rouler l’interface graphique.

## 4. Passer d’un point d’arrêt à un autre

F9 peut également permettre de passer d’un point d’arrêt à un autre.

Fermez le débogueur avec le bouton Arret en haut ( carré rouge )

Placez deux ou plusieurs points d’arrêts dans votre code et essayez de faire F9 plutôt que F7 ou F8 :

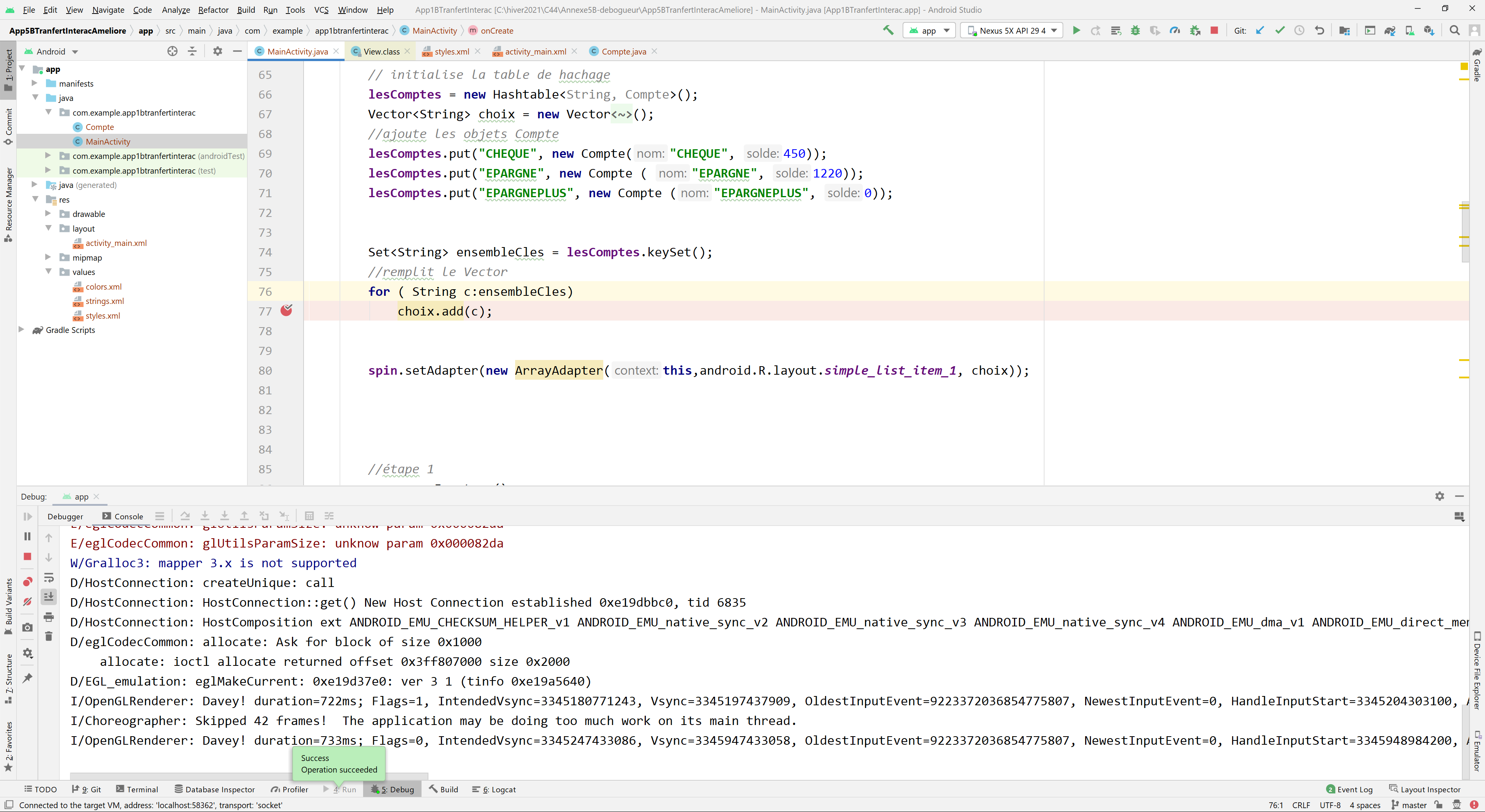


*F9 permet de passer d’un point d’arrêt à un autre plutôt que faire du pas-à-pas*

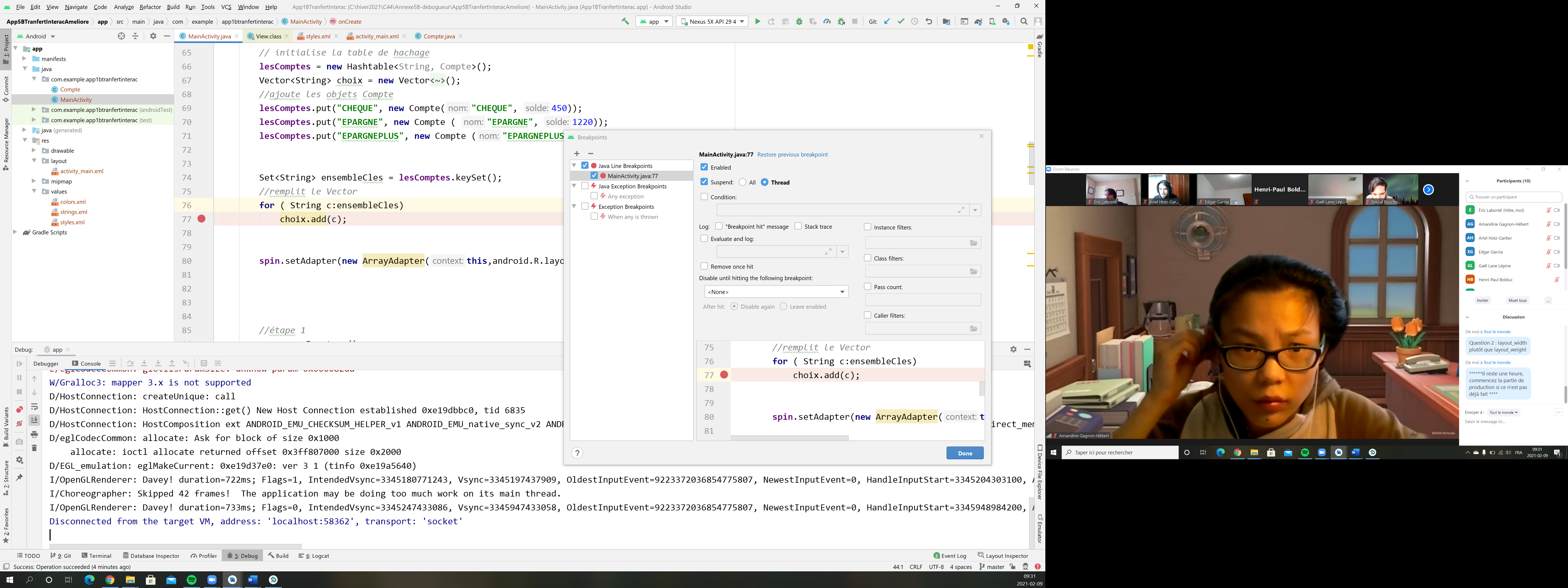
## 5. Points d’arrêt conditionnels

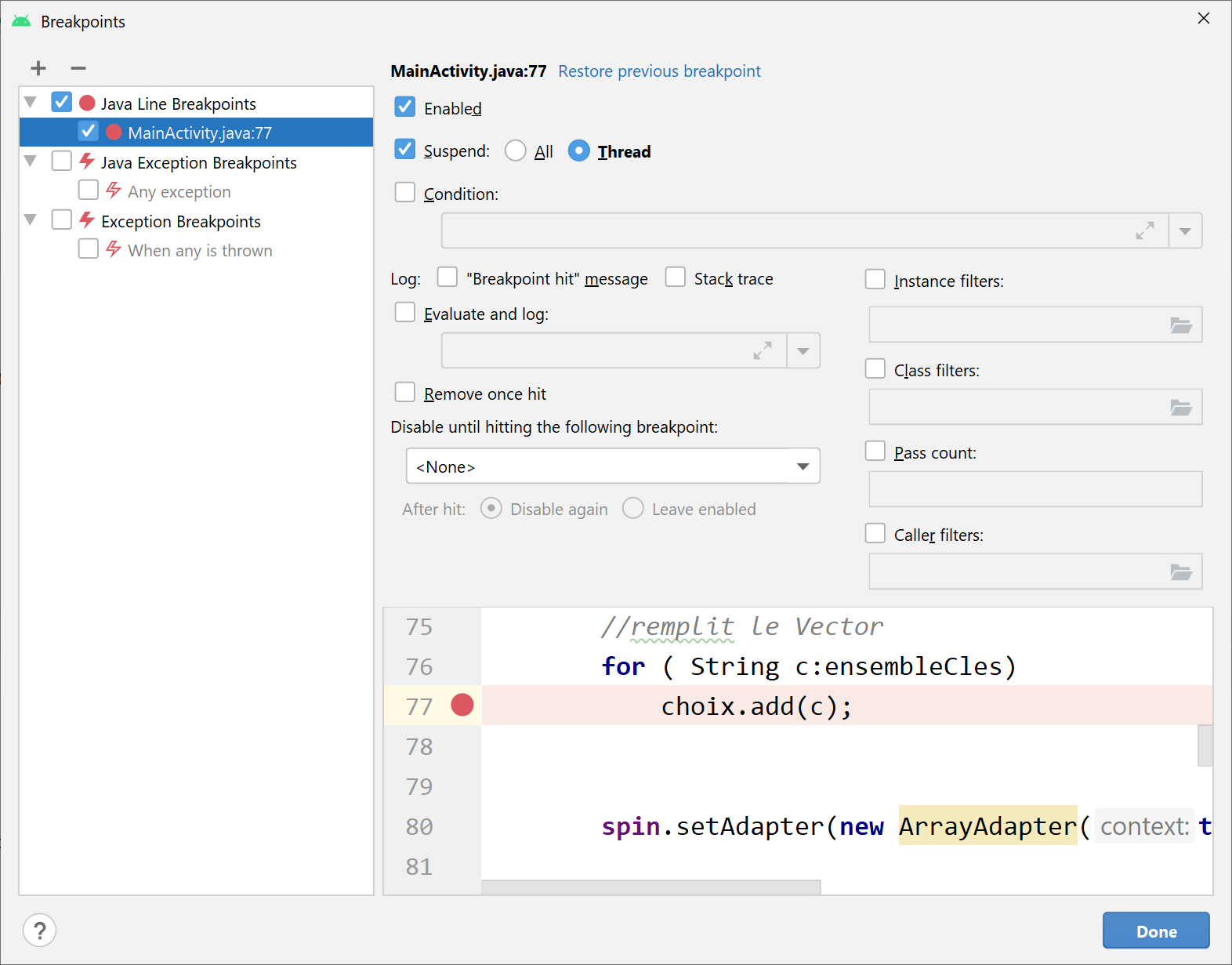
On peut faire en sorte que le point d’arrêt n’arrête le programme qu’en certaines circonstances

Exemple : Retirez les points d’arrêts ( en cliquant sur les indicateurs rouges dans les marges ) et ajoutez-en un dans la ligne contenue par la boucle for dans la méthode onCreate :



Accédez ensuite à la fenêtre Breakpoints :





Sélectionnez PassCount et écrivez 4.

Faites rouler le programme, le point d’arrêt n’interrompt pas le programme car on ne fait que 3 tours de cette boucle…ce n’est pas vraiment utile dans ce cas mais ce l’est si on veut pas avoir à être stoppé à chaque itération de boucle si elle en compte 100 !