DAA – Lab 2

**Auteurs** :  
Thibault Seem  
Pascal Perrenoud

**Date** : 16.11.2022



[1. Développement de la vue 2](#_Toc119503916)

[Mise en œuvre 2](#_Toc119503917)

[2. Contrôleur 2](#_Toc119503918)

[3. Questions complémentaires 2](#_Toc119503919)

[3.1. Pour le champ remark, quelle configuration faut-il faire dans le fichier XML pour un comportement correct? 2](#_Toc119503920)

[3.2. Comment gérer le format des dates au mieux? 2](#_Toc119503921)

[3.3. A quoi est due l'exception android.view.WindowLeaked lors d'une rotation de l'écran avec le dialogue ouvert? 2](#_Toc119503922)

[3.4. 3](#_Toc119503923)

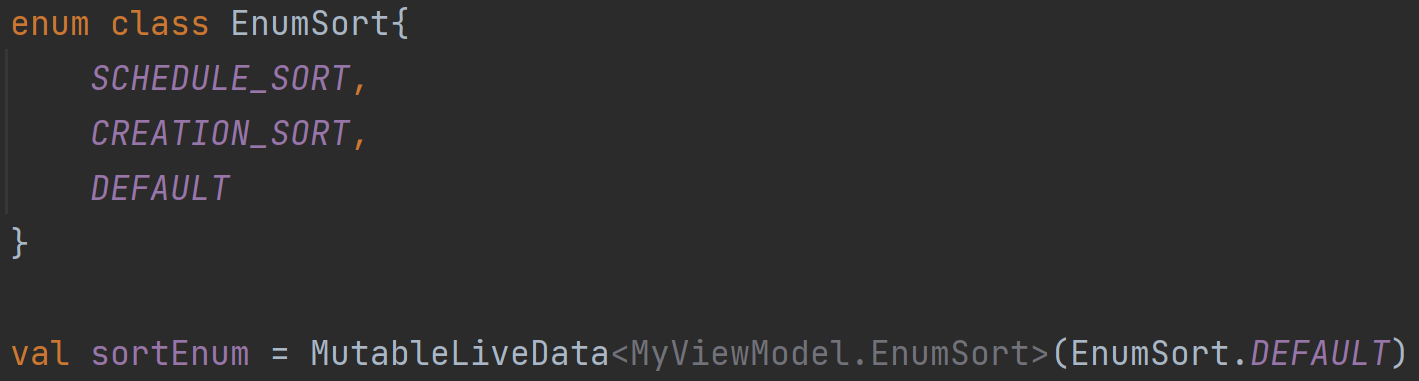
[3.4.1. Est-ce possible de spécifier son propre ordre de remplissage? 3](#_Toc119503924)

[3.4.2. Arrivé sur le dernier champ, est-il possible de faire en sorte que ce bouton soit lié au bouton de validation du questionnaire? 3](#_Toc119503925)

[3.5. Pour les deux spinners, comment peut-on faire en sorte que le premier choix corresponde au choix null, et comment faire pour ne pas la confondre avec une réponse? 3](#_Toc119503926)

# Quelle est la meilleure approche pour sauver le choix de l'option de tri de la liste de notes?

Il n'y a pas vraiment de meilleur choix, mais dans notre cas, nous avons optés pour une LiveData d'Enumération stockée dans la ViewModel qui indiquera quel tri nous souhaitons. Lors d'un choix de tri, nous allons changer cette variable pour, qui est observée par le Fragment contenant la RecyclerView, afin d'indiquer quel tri effectuer.



# L'accès à la liste des notes issues de la base de données Room se fait avec une LiveData. Est-ce que cette solution présente des limites?

# Les notes affichées dans la RecyclerView ne sont pas sélectionnable ni cliquables. Comment procéder pour proposer une interface permettant de sélectionner une note pour l'éditer ?

Il est possible de d'ajouter une détection de click sur un item d'un RecyclerView. Pour ce faire, il faut surcharger la méthode setOnClickListener de la view passée en paramètre de l'inner class ViewHolder, et ceci dans son init. Dans cette surcharge, la définition de la méthode que l'on souhaite utiliser.

Ensuite, il faut, depuis l'activité/fragment gérant la RecyclerView, surcharger la méthode définie dans la surcharge de setOnClickListener afin de faire ce que l'on souhait lors d'un click. Nous nous somme basé sur cette vidéo pour la réponse:

<https://www.youtube.com/watch?v=dB9JOsVx-yY>