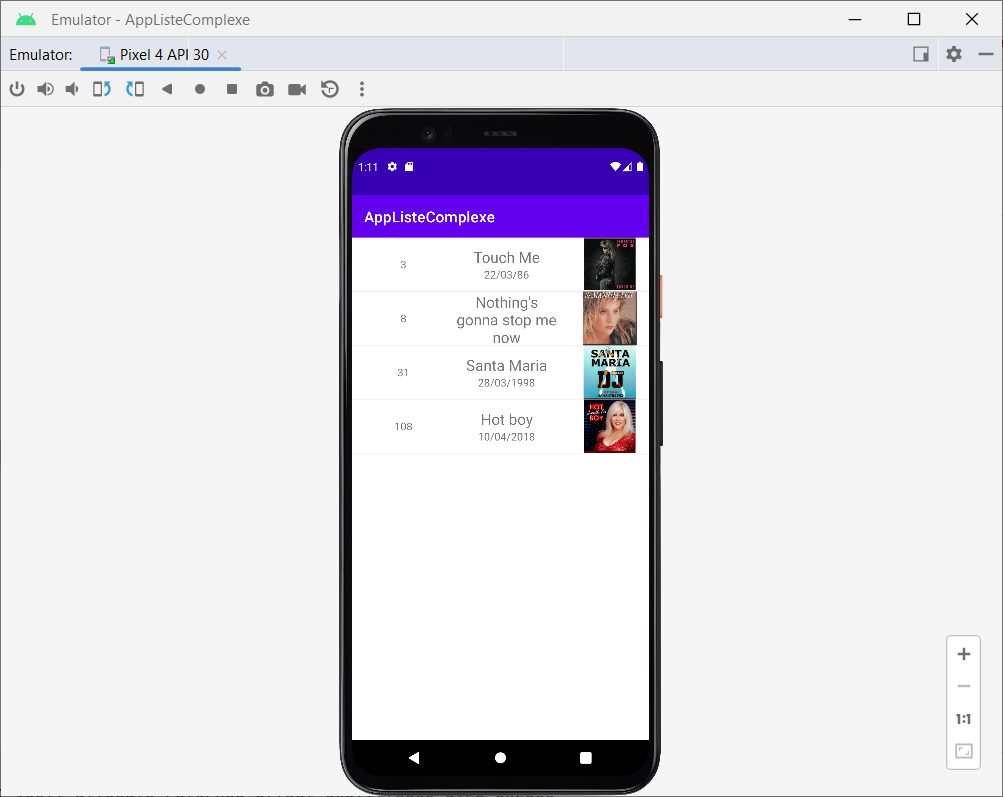
# Annexe 5 – Les listes complexes ( avec SimpleAdapter )

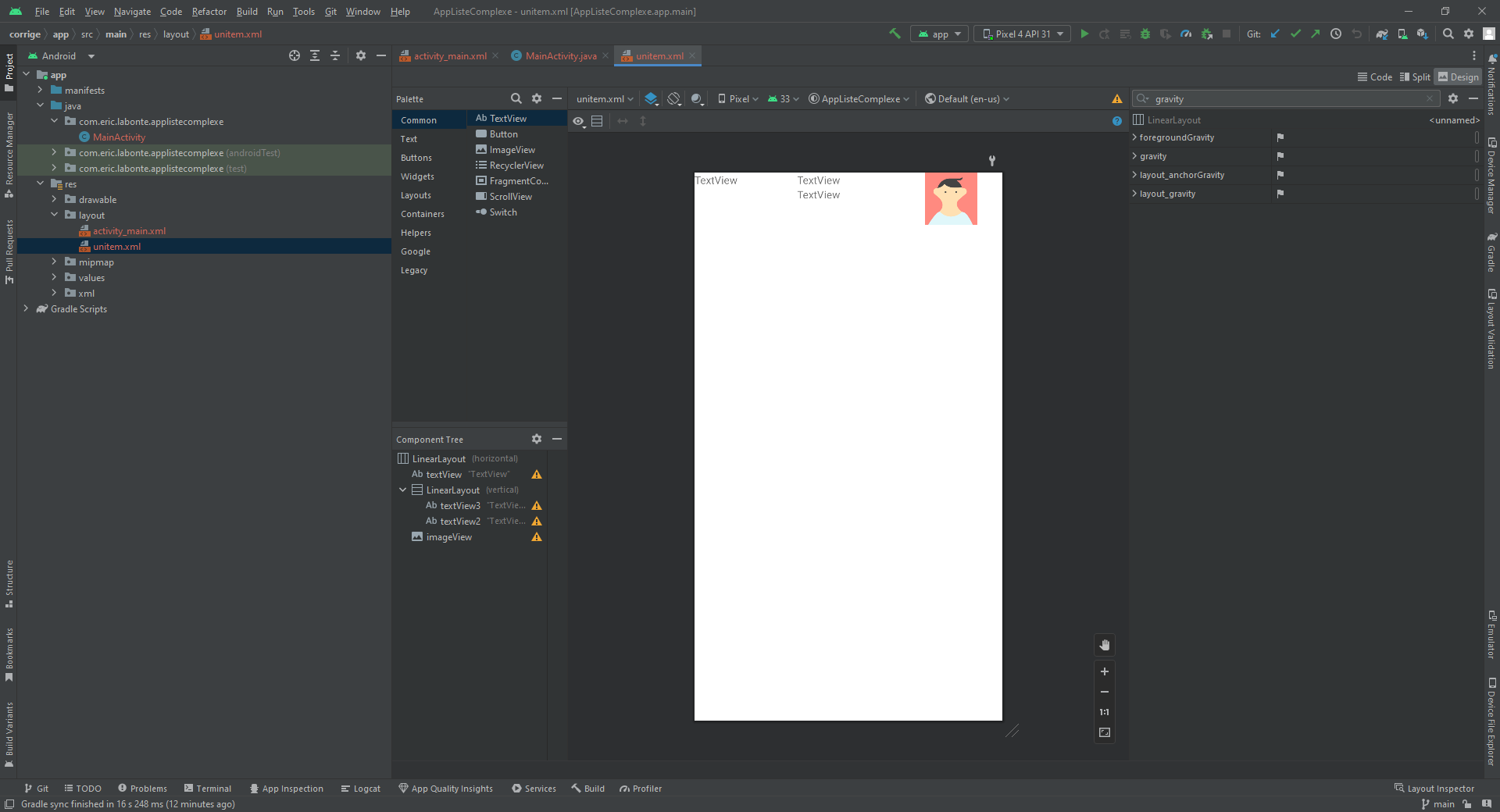
Une liste complexe est une liste où chaque item de la liste est composé de plusieurs items au lieu d’un seul TextView comme avec les listes simples :



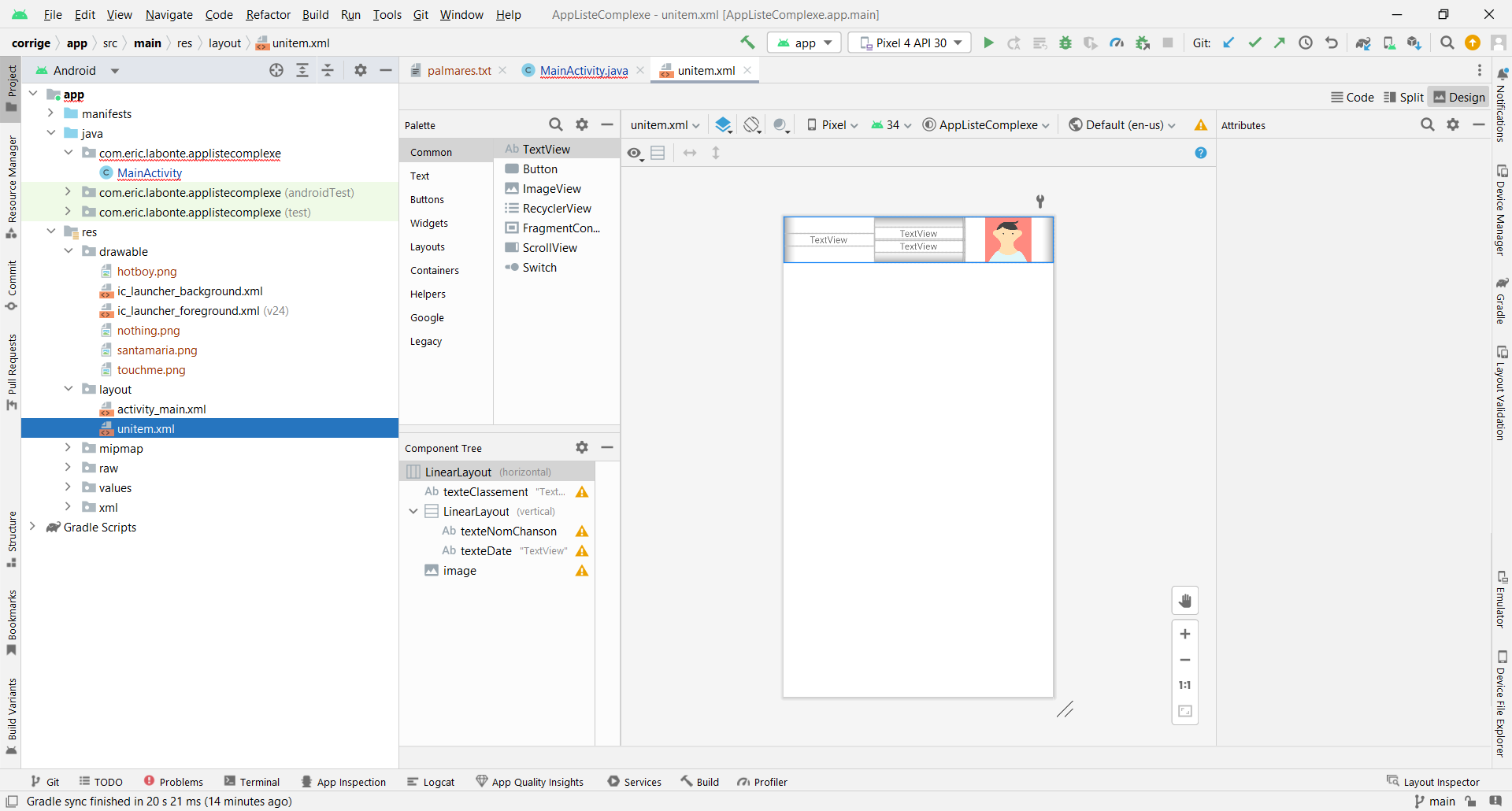
Exercice :

1. Présentes dans LEA, placez les pochettes représentant les chansons dans le dossier drawable
2. Créez un nouveau projet où l’activité ne contiendra qu’un ListView
3. On doit faire un autre fichier de positionnement .xml pour représenter **un item du ListView**

* Faire New sur layout dans le navigateur 🡪 Layout Resource File . Ne prenez pas le fichier .xml déjà présent pour votre activité car celui nous servira à représenter l'activité au complet.
* Le parent principal de ce second fichier de positionnement sera un LinearLayout. Utilisez des poids et d’autres LinearLayout pour avoir un résultat similaire à la capture d’écran de la page 1 :



*// au départ, je place les widgets dans le LinearLayout du .xml représentant un item de la liste*



*// un peu plus loin, après avoir utilisé gravity pour centrer les éléments*

1. Habituellement, on utilisait un ArrayAdapter pour remplir un ListView; pour remplir un ListView composé d’items plus complexes comme ici, on doit le remplir à l’aide d’un SimpleAdapter.

À l’aide de l’API Android, décrivez les 5 paramètres nécessaires à la création d’un SimpleAdapter :

*public SimpleAdapter (contexte contextuel, List<? étend les données Map<String, ?>>, ressource int, String[] from, int[] to)*

***context :****le contexte dans lequel la vue associée à ce SimpleAdapter s'exécute*

***data :****une liste de map. Chaque entrée de la liste correspond à une ligne de la liste. Les cartes contiennent les données de chaque ligne et doivent inclure toutes les entrées spécifiées dans « de » un vector de hashmap/table*

***ressource :****identifiant de ressource d'une disposition de vue qui définit les vues pour cet élément de liste. Le fichier de mise en page doit inclure au moins les vues nommées définies dans « à »*

***from :****Une liste de noms de colonnes qui seront ajoutés à la carte associée à chaque élément.*

***to :****Les vues qui doivent afficher la colonne dans le paramètre « from ». Ceux-ci devraient tous être des TextViews. Les N premières vues de cette liste reçoivent les valeurs des N premières colonnes du paramètre from.*

1. Le 2e paramètre « data » doit contenir les données à afficher dans la liste, sous forme de List de Maps

Quel est la collection qu’on connait qui met en œuvre l’interface List ? Cliquez ou appuyez ici pour entrer du texte.

Quel est la collection qu’on connaît qui implémente l’interface Map ? Cliquez ou appuyez ici pour entrer du texte.

On doit donc déposer nos données dans une collection de collections ce qui fait du sens car la collection parent représente les items tandis que la collection enfant représente les informations d’une chanson.