

## VIII. Les RecyclerView

Le RecyclerView permet plus de flexibilité qu'une simple ListView. Il permet par exemple de faire du scrolling horizontal ou vertical et plus de customisation.

lisez la doc officielle android [exemple](#) et [doc](#)

### Import des dépendances

Si vous travaillez comme moi avec le sdk 29, ajoutez au fichier `build.gradle(:app)` nos dépendances recycler et card (oui on va faire des cards):

```
dependencies {  
    ...  
    implementation 'androidx.recyclerview:recyclerview:1.1.0'  
    implementation 'androidx.cardview:cardview:1.0.0'  
}
```

et synchronisez votre projet s'il ne se synchronise pas automatiquement.

### Le layout principale

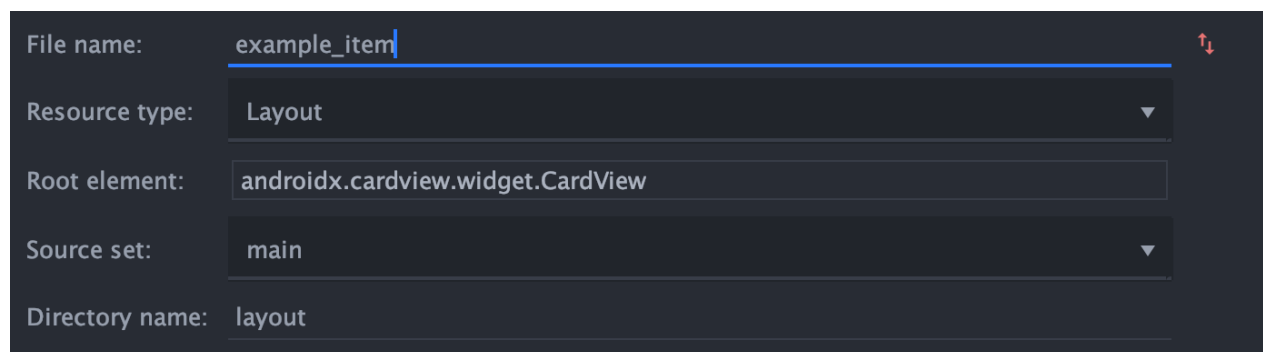
Dans le `activity_main.xml` :

- modifiez le layout pour avoir un `RelativeLayout`
- ajoutez un `RecyclerView` et nommez le **recyclerView**
- donnez lui les attributs suivants `scrollbars="vertical"` et `background="@android:color/darker_gray"`
- enfin il doit prendre la place dans son parent (`match_parent`)

### La Vue à recycler

#### Le layout

Créez un layout `example_item.xml` de type `cardview`



il comprendra:

- un RelativeLayout avec un margin de **4dp**
- une ImageView nommée "imageView" de **50dp** par **50dp** et un padding de **2dp**
- 2 TextView "textView" et "textView2" l'un sur l'autre, à côté de l'image. (Utilisez les propriétés de positionnement ou avec l'interface WYSWYG)

### Le contrôleur

Créez un contrôleur ExampleItem.java ayant pour propriétés :

```
private int mImageResource;
// int parce que les ressources on des indices numériques
private String mText1;
private String mText2;
```

ainsi que le constructeur associé et les assesseurs.

### Quelques images

Ajoutez 3 Vector Asset dans le dossier **drawable**

### Une liste dans le MainActivity

```
ArrayList<ExampleItem> exampleList = new ArrayList<>();
exampleList.add(new ExampleItem(R.drawable.ic_android,"Ligne 1","Ligne 2"));
exampleList.add(new ExampleItem(R.drawable.ic_radio,"Ligne 3","Ligne 4"));
exampleList.add(new ExampleItem(R.drawable.ic_rowing,"Ligne 5","Ligne 6"));
```

### L'Adapter

Pour que nos éléments s'affichent, il va nous manquer 3 choses:

- D'abord déclarer la RecyclerView dans le MainActivity
- l'Adapter qui va inclure nos items dans le recycleur
- Le layoutManager qui va s'occuper des layout.

```
private RecyclerView mRecyclerView;
private RecyclerView.Adapter mAdapter;
private RecyclerView.LayoutManager mLayoutManager;
```