

# RecyclerView

Une RecyclerView est une nouvelle façon d'afficher une liste ou une grille de vues.

Elle peut s'apparenter à une ListView mais permet beaucoup plus de personnalisation.

Cette nouvelle vue est disponible dans la librairie de support v7 d'android, il vous faut donc commencer par importer dans votre fichier gradle :

```
dependencies {  
    implementation 'com.android.support:recyclerview-v7:28.0.0'  
}
```

Comme toutes vues, il faut la déclarer dans nos layout xml :

```
<?xml version="1.0" encoding="utf-8"?>  

```

```

override fun getItemCount(): Int {
    // definir la taille
    return myData.size
}

override fun onBindViewHolder(holder: MyViewHolder, position: Int) {
    // Mettre à jour un vue déjà crée

    var e=myData.get(position)
    // récupérer l'élément depuis la liste des données myData

    holder.tvnom.text=e.nom
    holder.tvnum.text=e.num.toString()
    // modifier le contenu d'un vue par les valeurs de cet élément
}

inner class MyViewHolder(parent: View) : RecyclerView.ViewHolder(parent)
{
    //recuperation des sous composant du layout depuis son Context : parent
    var tvnom=parent.findViewById(R.id.tv_nom) as TextView
    var tvnum=parent.findViewById(R.id.tv_num) as TextView
}
}

```

Utilisons le maintenant dans notre Activity

Une étape importante est d'affecter un LayoutManager à notre activity, sinon une erreur sera levée : **RecyclerView: No layout manager attached; skipping layout**

```

var viewManager = LinearLayoutManager(getApplicationContext())
var viewAdapter = MyAdapter(this,data)

// in content do not change the layout size of the RecyclerView
lv.setHasFixedSize(true)
//définit l'agencement des cellules, ici de façon verticale, comme une ListView
lv.setLayoutManager(LinearLayoutManager(getApplicationContext()));
//pour adapter en grille comme une RecyclerView, avec 2 cellules par ligne
//recyclerView.setLayoutManager(new GridLayoutManager(this,2));

// specify an viewAdapter (see also next example)
lv.adapter = viewAdapter

```