

Workshop GridView/ListView

Exemple :

Catalogue de Livres.

Objectif Principal:

Comment afficher un contenu en Grille (Grid).

Objectif Bonus :

Comment Basculer entre deux modes d'affichage Grille et Liste selon l'orientation du téléphone Portrait et Landscape (2 méthodes)

Contexte :

Comme son nom l'indique, une **GridView** permet d'afficher des « éléments » sous forme d'une grille. L'avantage du GridView est sa bonne gestion de l'espace dans une interface graphique. L'implémentation de cet élément graphique est plutôt simple, la GridView peut être utilisée dans plusieurs modèles d'interface pour représenter par exemple :

Un menu principal d'une application.

Une interface d'application divisée en grille par exemple Calculatrice, pavée numérique, instrument de musique...

Une interface de jeu par exemple sudoku, dame, échecs, mot croisée, jeu de mots, nombres..

Lors du développement d'une application, il faut tenir en compte l'adaptation de l'interface aux différentes tailles d'écrans sachant que la GridView se compose de Lignes et Colonnes, aussi le remplissage de la GridView doit se faire en utilisant un Adapter correspondant.

Enoncé :

Utiliser une GridView pour la réalisation d'une Interface de Catalogue de Livres d'une application Librairie en mode Landscape et qui change en ListView en mode Portrait.

Données :

- Projet d'une application avec bout de code d'une ListView à compléter.
 1. L'interface principale contient : Un Une ListView
 2. Un fichier XML qui correspond à un élément (item) de la liste composée d'une ImageView et un TextView.
- Le dossier **Drawable** contient les ressources nécessaires :
 1. 20 images miniaturisées correspondent aux livres.
 2. Une image de fond (à utiliser comme papier peint).

Etapes à suivre :

On implémente la méthode `setOnClickListener` de la `ListView` qui affiche en `Toast` le nom du livre et on ajoute l'Adapter dans le même fichier. L'Adapter doit remplir la liste de la façon la plus simple et la plus optimisée.

On crée un Objet de Type `GridView` et on ajoute l'Adapter correspondant, on modifie le même fichier XML de l'interface Principale pour qu'il fonctionne bien pour la `ListView` et la `GridView`.

1. Méthode pour la détection de l'orientation (Java) :

On ajoute le nécessaire dans la Classe Principale pour détecter l'orientation du smartphone et puis afficher la vue correspondante : `PORTRAIT-> LISTVIEW`, `LANDSCAPE->GRIDVIEW`

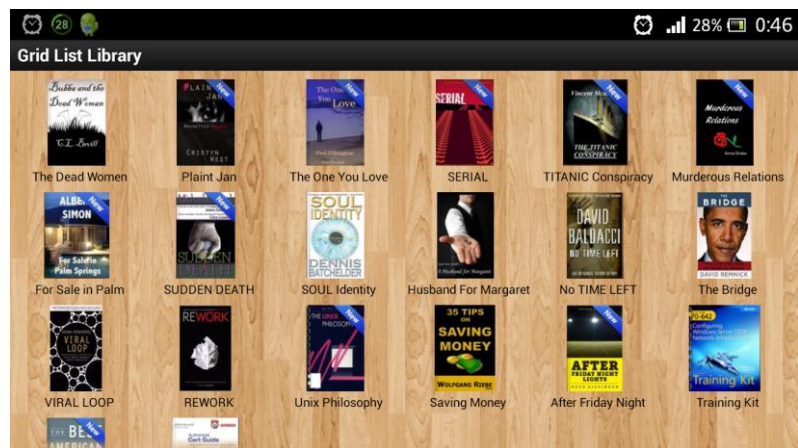
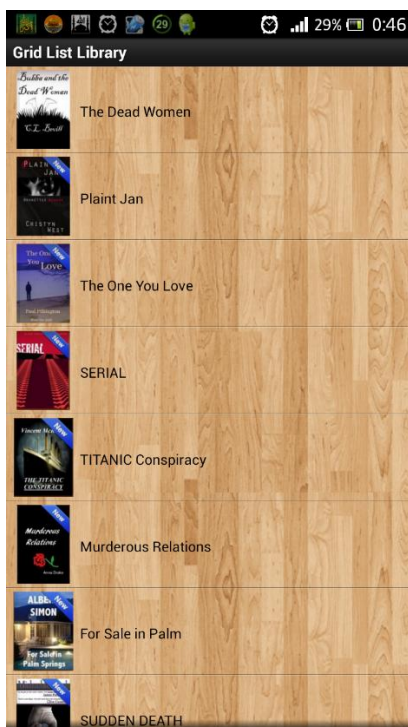
2. Méthode pour la détection de l'orientation (XML) :

On Change la disposition des éléments du fichier XML de l'Item de cette façon :

- `LISTVIEW` -> Texte à droite de l'image
- `GRIDVIEW` -> Texte au-dessous de l'image

NOTE: On doit prendre en considération l'état initial de l'orientation de l'appareil `PORTRAIT` ou `LANDSCAPE`

Résultat à atteindre :



Bon Travail