

# Application MyListViewSQL

Il s'agit dans ce TP de manipuler les objets de type ListView afin que les informations stockées dans la base de données manipulée lors du précédent TP soient affichées sous forme de liste avec la possibilité d'effectuer une opération particulière lorsque l'on clique sur un item de la liste. Le point de départ de votre application sera donc l'application MyLifeCycleSQL développée précédemment (cf. Fig. 1).



*Fig.1*

## PARTIE 1

La différence se situe au niveau de l'action du bouton « Show » qui devra lancer une nouvelle activité chargée d'afficher dans une ListView le contenu de la base de données (cf. Fig.2). Ce contenu sera trié en fonction du numéro d'ID des éléments de la base.

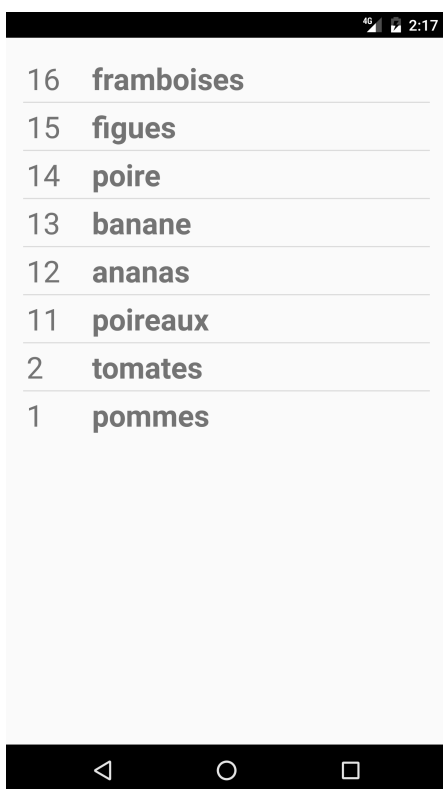
Cette nouvelle activité sera une extension de la classe ListActivity :

```
public class Showlist extends ListActivity
```

Pour plus d'informations sur ce type d'activité, consultez :

<https://developer.android.com/reference/android/app/ListActivity.html>

Votre activité principale se chargera d'initier la connexion avec la BD et d'insérer des éléments dans la base de données (bouton Save). L'activité secondaire se chargera de lister les éléments contenus dans cette base et éventuellement d'en effacer (ceci sera mentionné plus loin dans le sujet).



*Fig.2*



*Fig.3*

Quoiqu'il en soit, vos 2 activités travaillant sur la même base de données, il faudra gérer via une classe singleton l'objet étendant la classe SQLiteOpenHelper et l'objet de type SQLiteDatabase nécessaires aux interactions avec votre BD. Vous vous assurerez de plus que toutes les opérations effectuées sur la BD (ouverture, consultation et modification) sont toutes réalisées via une tâche détachée (inutile de mettre en place une barre de progression en ce qui concerne les opérations de lecture et de modification).

## PARTIE 2

Modifiez votre code de façon à ce que lorsque l'on clique sur l'un des items de la liste, le contenu de l'item soit barré (cf. Fig. 3). Utilisez pour cela la méthode setPaintFlags de l'objet TextView. Les options disponibles pour cette méthode sont des constantes de l'objet Paint, cf :

<https://developer.android.com/reference/android/graphics/Paint.html>

Attention il s'agit de rajouter l'attribut « Texte barré » à l'item en cours et non pas de modifier complètement son style, il faut donc penser à récupérer les attributs courants (getPaintFlags) et à les combiner avec le nouvel attribut.

## PARTIE 3

Modifiez votre code de façon à ce que, lorsque l'on effectue un clic long sur un item de la liste, ce dernier soit supprimé de la BD et que la liste soit mis à jour automatiquement.