

Exercice

Interface simple et communication entre Activity

Mars-Avril-Mai 2019

Formation Android – Tristan SALAUN



Dans cet exercice nous allons revoir les principes étudiés en cours :

- La définition d'interfaces simples.
- La communication entre Activity (dans les 2 sens : de A vers B et de B vers A).
- L'utilisation d'une listView

Objectif

- Le but de cet exercice est de gérer des contacts.
- L'application comportera 2 parties :
 - 1 : une partie édition du contact
 - 2 : sélection d'un contact

Le modèle de données

- Nous allons définir une classe qui représentera notre contact.
 - Créer une classe : Contact
 - Définissez les champs :
 - public String name;
 - public String firstName;
 - public int gender;
 - public String notes;

Mars-Avril-Mai 2019

Formation Android – Tristan SALAUN



Créer une classe Java qui contiendra les informations de notre contact, il s'agit de notre objet qui modélise notre donnée. C'est un POJO.

Gestion des données

- Nous allons créer un Singleton (File, New, Singleton) qui contiendra une ArrayList contenant la liste de nos contacts. Nous pourrons donc y accéder à tout endroit de notre application, afin de visualiser les informations et les modifier.

Le singleton est accessible de partout, il nous servira de « base de donnée » ultra simple.

Menu principal

- Le menu principal de notre application comporte uniquement 2 boutons :
 - Editer un contact
 - Choisir un contact

Le layout de cette activité comporte 2 boutons, qui lancent chacun une activité différente.

Editer un contact

- Le premier écran de cette partie liste les contacts (avec une recyclerView).
- Quand on clique sur un contact, une activity EditContact est lancé, avec l'id (le numéro dans la liste) du contact.
- La classe EditContact récupère l'id du contact et pré-remplis les champs pour éditer le contact :
 - Des EditText pour les champs de type String
 - Un Spinner pour choisir le genre du contact
 - Un bouton pour enregistrer les modifications

Mars-Avril-Mai 2019

Formation Android – Tristan SALAUN



La première activité contiendra le fragment qui permet de lister nos contacts. Sur le click, on lance une nouvelle activité avec l'id du contact à éditer en paramètre (dans les extras du bundle, contenu dans l'Intent).

La seconde Activity récupère l'id du contact, récupère via le singleton les informations du contact et pré-remplis les champs.

L'utilisateur peut donc mettre à jour les valeurs des champs.

Au click sur le bouton « Enregistrer », nous allons mettre à jour la liste contenue dans le Singleton,

Choisir un contact

- Cette activité PickContact, comporte les éléments suivants :
 - Des TextView pour afficher le nom, prénom, genre et note du contact.
 - Un bouton intitulé « Choisir un contact » qui lance une Activité qui listera les contact et au click renverra son id.
 - Au retour de l'activité, on remplit les informations correspondantes au contact en utilisant l'id reçu.

Mars-Avril-Mai 2019

Formation Android – Tristan SALAUN



Pour réaliser le picker de contact, il suffit de réutiliser le fragment qui liste les contacts et modifier le code du listener de click de l'Activity afin de retourner l'id de l'item sélectionné.

Pour aller plus loin

- Affichez une couleur de fond différente (ou une icône) en fonction du genre du contact.
- L'activity EditContact ne comporte plus de bouton enregistrer, mais affiche une boîte de dialogue (AlertDialog) quand on clique sur le bouton back et demande la confirmation à l'utilisateur d'enregistrer, ou non les modifications faites.

Mars-Avril-Mai 2019

Formation Android – Tristan SALAUN



Dans l'Adapter de notre RecyclerView, nous allons changer la couleur de fond de l'item en fonction du genre du contact.

Pour gérer le bouton back, nous allons redéfinir la méthode :

`@Override`

```
public void onBackPressed() {
```

```
}
```