Annexe 3– Sérialisation d’objets

Pour faire cette annexe…

* Copier / coller votre projet des mémos que nous avons fait avec des fichiers texte
* Retirer les parties relatives à ces mêmes fichiers texte

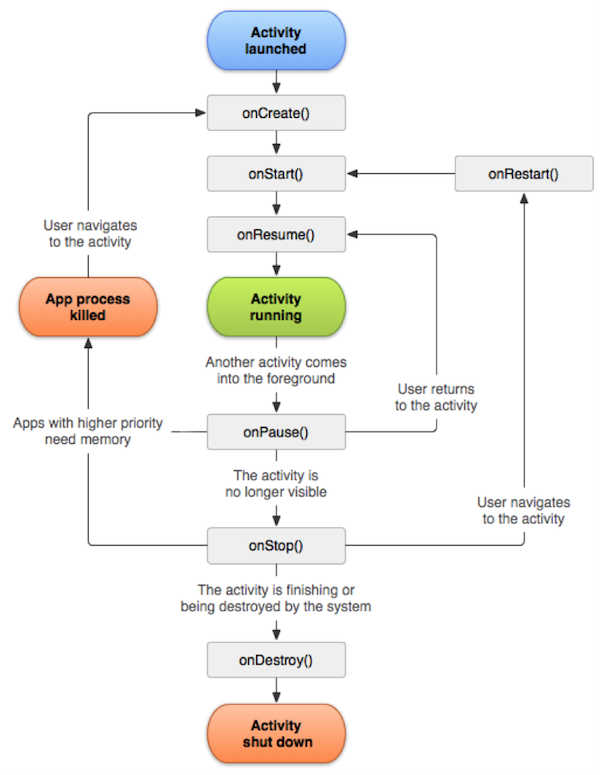
Nous avons vu à l’annexe 1 une façon de conserver des données au-delà de la fermeture de l’app, en utilisant des fichiers texte en mémoire interne

Certains problèmes existent avec cette façon :

* Cliquez ou appuyez ici pour entrer du texte.
* Cliquez ou appuyez ici pour entrer du texte.

Pour résoudre le premier défi, on peut avoir recours à un Singleton de même qu’aux méthodes préétablies du cycle de vie d’une app Android

# cycle de vie



onRestoreInstanceState est appelé

Tourner le téléphone

retour d'avoir changé la langue du téléphone, …

Retour du bouton Recents, du bouton Home, "retour du boomerang"

onSaveInstanceState est appelé

Recoit un appel

Ferme L’app ou quand tu fais back

Attention ! onDestroy n’est pas tjrs appelé selon la doc d’Android

<https://developer.android.com/guide/components/activities/activity-lifecycle>

<https://medium.com/@theabhishekavi/android-activity-lifecycle-9bc7de812dff>

1. Codons un singleton ayant comme variable d’instance un ArrayList de Strings étant nos mémos
2. Codons les opérations ajouter et voir les mémos localement d’abord ( sans persistance ) en travaillant sur l’ArrayList
3. Lorsqu’on quitte quelle activité pourrait-on écrire l’ArrayList dans un fichier de sérialisation ?

## Sérialisation d’objets :

1. Dans le singleton, codons une méthode permettant de sauvegarder l’état de l’ArrayList dans un fichier de sérialisation et appelons-la dans la méthode du cycle de vie onStop de l’activité choisie en 3
2. Dans le singleton, codons une méthode permettant de récupérer l’objet ArrayList provenant d’un fichier de sérialisation. Où / Comment devrait-t-on appeler cette méthode ?

Cliquez ou appuyez ici pour entrer du texte.