

Développement d'applications mobiles : ANDROID

TP2 : Cycle de la vie d'une application

Elise JOFFRE

Nous allons générer les méthodes correspondant à chaque évènement du cycle de vie et bien comprendre les différentes étapes.

Lien GitHub de l'exercice 1 du TP : <https://github.com/EliseJoffre/CycleDeVie>

Application :

onCreate() : Elle est appelée lorsque l'activité est créée pour la première fois. Elle sert à effectuer l'ensemble de la configuration statique normale de l'activité (créer des vues, lier des données à des listes, etc). Elle peut également fournir un ensemble contenant l'état précédemment gelé de l'activité, s'il y en avait un. Elle est suivie de onStart().

Dans le programme, elle est la première fonction à apparaître car elle initialise l'activité, elle est la première étape du cycle de vie de l'application.

onStart () : Dans le programme, elle est la deuxième activité du cycle de vie, elle est appelée lorsque l'activité devient visible pour l'utilisateur. On remarque qu'elle est toujours appelée en deuxième lorsqu'on circule entre les 2 pages de notre application. Elle est suivie par onResume() si l'activité arrive au premier plan ou onStop() si elle est cachée.

onResume() : Elle est appelée lorsque l'activité commence à interagir avec l'utilisateur. Elle est toujours suivie de onPause(). Dans notre programme, ceci est confirmé lorsque l'on circule entre les 2 activités.

onSaveInstanceState () (ce qui a été remplacé onFreeze) : Lorsque l'activité commence à s'arrêter, le système appelle la méthode onSaveInstanceState () afin que l'activité enregistre les informations d'état. Elle doit être implémenter après la méthode onPause (), et avant onStop ().

onPause() : Elle est appelée lorsque le système est sur le point de commencer à reprendre une activité précédente. Elle est suivie par onResume () si l'activité revient au premier plan, ou onStop () si elle devient invisible pour l'utilisateur.

onStop() : Elle appelée lorsque l'activité n'est plus visible pour l'utilisateur, car une autre activité a été reprise et couvre celle-ci. Cela peut arriver soit parce qu'une nouvelle activité est commencée, qu'une autre est en train d'être mise en face de celle-ci, ou que celle-ci est détruite. Lorsque l'on passe à l'activité 2 le onStop() est alors appelée sur l'activité 1. Elle est suivie par onRestart() si on revient sur celle-ci ou onDestroy() si on ne souhaite plus utiliser l'activité.

onRestart() : Elle est appelée après qu'une activité ait été arrêtée et avant d'être redémarrée. Elle est toujours suivie par onStart ().

onDestroy() : Cette méthode est appelée lorsqu'on souhaite détruire l'activité ou lorsque le système la détruit.

Quels sont les paramètres d'entrée de onSaveInstanceState() et de onRestoreInstanceState() ?

La méthode `onSaveInstanceState` possède pour paramètre le bundle qui sera renvoyé en paramètre de la méthode `onRestoreInstanceState` (`savedInstanceState`). Bundle est une `HashMap` typée qui permet d'enregistrer les différents champs primitifs ou parcelables que l'on souhaite passer de l'activité mourante à l'activité recréée.

Ainsi, dans la méthode `onSaveInstanceState` on stocke les données à sauvegarder dans ce bundle et dans la méthode `onRestoreInstanceState` on les restaure à partir de ce bundle.

Lien GitHub de l'exercice 2 du TP : <https://github.com/EliseJoffre/Bibliotheque>