

TP N° 2**Exercice 1**

Une activité n'est pas linéaire, elle est soumise à plusieurs évènements. Chaque événement est représenté dans une méthode. Le cycle de vie d'une application Android passe par :

onCreate() : Cette méthode est appelée à la création d'une activité. Elle permet de l'initialiser. C'est ici que l'interface graphique est spécifiée.

onStart() : Cette méthode est appelée quand l'application est démarrée.

onResume() : Cette méthode est appelée quand l'application passe (ou repasse) en avant-plan.

onPause() : Appelée quand l'application passe en arrière-plan et qu'une autre application se met devant.

onStop() : Appelée quand l'application n'est plus visible.

onRestart() : Appelée quand l'application redevient visible.

onDestroy() : Appelée quand votre application est fermée par le système à cause d'un manque de ressources, ou par l'utilisateur à l'utilisation d'un *finish()*.

1. Rajouter les méthodes correspondantes (de la même manière que pour la méthode *onCreate*) déjà générée par ADT. Pour générer ces méthodes, commencer à taper le nom de la méthode dans le fichier de code, puis taper sur les touches *Ctrl+Espace*.
2. Nous désirons voir le comportement de l'activité pour chacun de ces évènements. Pour cela, chaque méthode va afficher son nom.

Exercice 2

Pour créer une nouvelle activité, il faut suivre les étapes suivantes :

- Créer une nouvelle classe dans votre package qui hérite de la classe *Activity*.
- Générer la méthode *onCreate()*.
- Créer un nouveau fichier *layout*, et y ajouter les éléments graphiques désirés.
- Associer ce fichier *layout* à votre activité dans la méthode *onCreate()*.
- Ajouter la nouvelle activité dans le fichier Manifest.

Remarque : Pour cette nouvelle activité, dans le manifest, ne pas garder la balise `<intent-filter>` : Cette balise permet, grâce à son action `android.intent.action.MAIN`, de spécifier que l'activité en cours est l'activité de départ de l'application.

Le passage entre deux activités requiert un *Intent*. Un *Intent* est un conteneur d'information. Il permet de passer des messages entre deux activités. L'activité appelante pourra ainsi transmettre des informations à l'application appelée, ainsi qu'au système Android.

Il existe plusieurs manières de créer un *Intent*. Nous allons choisir la suivante :

```
Intent myIntent = new Intent(<contexte>, <classe de l'activité cible>);
```

Pour démarrer une autre activité, il faut lancer la méthode *startActivity(Intent i)* de la classe *Activity* initiale.

Pour passer un message à une activité, on ajoute des *Extras* aux *Intents*. Par exemple :

```
myIntent.putExtra(<id>, <chaîne>);
```

va passer la chaîne de caractère *chaîne* dans une variable appelée *id* à l'activité cible de l'*Intent*.

Dans l'activité cible, on récupère cet *Extra* en utilisant :

```
getIntent().getStringExtra(<id>)
```

Exercice 3

Créer un projet Android qui permet de réaliser une calculatrice simplifiée. Votre travail doit respecter les règles suivantes :

- Le clic sur le bouton « +, -, *, / » permet de faire l'opération correspondante des deux valeurs dans les champs de saisie et d'afficher le résultat dans le champ de texte Res.
- L'application doit gérer les cas particuliers des opérations et les champs de saisie n'acceptent que des chiffres.