

Android – TP2 : les premières intentions

- Récupérez sur l'intratek le projet sur lequel vous allez travailler : SIL1-TP2-projetAndroidStudio.zip, placez le dans un dossier M4104C.
- Pour le lancement de l'émulateur Android reportez vous au TP1. **ATTENTION : l'émulateur ne sera à lancer qu'une seule et unique fois pendant la séance de travail.**
- Vous trouverez sur le web toute la documentation nécessaire pour ce module
 - Documentation Java : <http://docs.oracle.com/javase/7/docs/api>
 - Guide User Interface Android : <http://developer.android.com/guide/topics/ui/index.html>
 - Android référence : <http://developer.android.com/reference/android/package-summary.html>

Prise en main des intentions

Exercice 4 : Premières intentions

Pour réaliser cet exercice, vous aurez à modifier Exercice4Activity.java et activity_exercice4.xml (respectivement la classe représentant une activité et sa vue associée) et à créer une nouvelle activité HelloActivity.java avec sa vue associée.

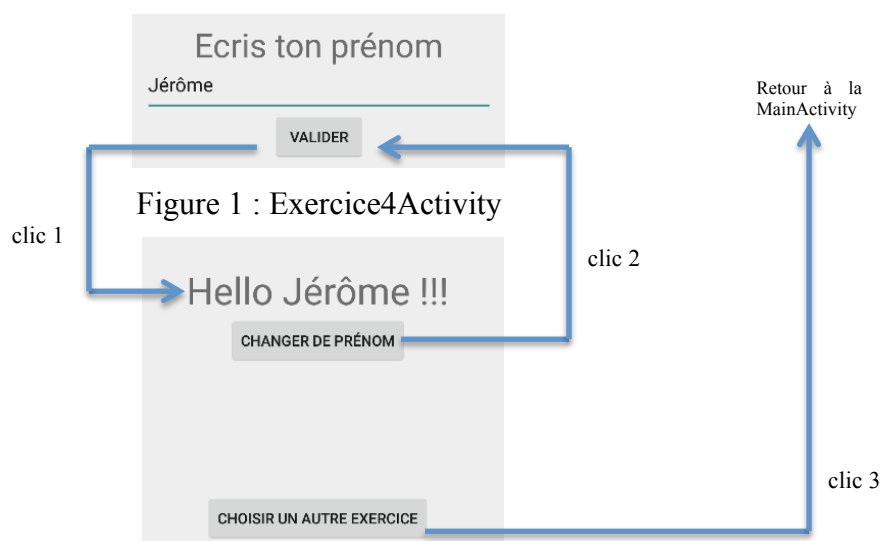


Figure 2 : HelloActivity

Scénario : un utilisateur écrit son prénom (par exemple Jérôme) dans un objet de type EditText (voir figure 1). Lors du clic sur le bouton valider (clic 1), un texte « Hello Jérôme ! » apparaîtra sur une nouvelle activité (voir figure 2). Cette nouvelle activité proposera de changer de nom (clic 2 - retour à l'activité précédente) ou de choisir un autre exercice (clic 3 - retour à l'activité principale MainActivity).

Etape 1 : créer l'interface graphique de la figure 1 dans le fichier activity_exercice4.xml.

Etape 2 : créer une nouvelle activité HelloActivity : *File > New... > Activity > Empty Activity*. Dans *Activity name*, mettre HelloActivity. La vue XML associée (*Layout name*) aura comme nom activity_hello. Cliquer sur le bouton Finish. **Créer** l'interface graphique de la figure 2 dans le fichier activity_hello.xml.

Etape 3 : réaliser la demande d'intention dans la classe Exercice4Activity permettant le lancement de la nouvelle activité HelloActivity. Pour ce faire :

1. Réaliser la partie événementielle du bouton Valider. Associer un événement onClick à ce bouton.
2. Dans la méthode onClick, récupérer la chaîne de caractères donnant le prénom dans le composant graphique de type EditText :

```
String prenom = composantEditText.getText().toString() ;
```

3. Créer une intention pour la nouvelle activité HelloActivity :

```
Intent intent = new Intent(this, HelloActivity.class);
```

4. Ajouter la chaîne à l'intention pour transférer le prénom à la nouvelle activité :

```
intent.putExtra(EXERCICE_4_PRENOM, prenom);
```

EXERCICE_4_PRENOM est une constante de l'activité représentant le nom du paramètre. Ajouter l'instruction suivante comme attribut de classe :

```
public static final String EXERCICE_4_PRENOM = "prenom";
```

5. Lancement de la demande d'intention en utilisant la méthode startActivity

```
startActivity(intent);
```

Vérifier à chaque étape que votre programme fait bien ce que vous voulez. Par exemple, dans cette étape valider le changement de fenêtre (la récupération du prénom se fera dans l'étape suivante).

Rappel : pour tester votre application en lançant la commande de l'IDE *Run > Run 'app'*. Cette commande chargera votre application (.apk) sur votre émulateur.

Etape 4 : récupérer le prénom passé par l'intention dans l'activité HelloActivity. Pour ce faire, ajouter l'instruction suivante dans la méthode onCreate. Pour rappel, cette méthode crée l'activité et affiche la vue associée.

```
String prenom = getIntent().getStringExtra(Exercice4Activity.EXERCICE_4_PRENOM);
```

Ensuite, ajouter ce prénom dans un composant de type TextView pour afficher « Hello prénom !!! ».

Remarque : noter l'usage de la constante EXERCICE_4_PRENOM de la classe Exercice4Activity représentant le nom du paramètre envoyé.

Etape 5 : réaliser le retour à l'activité Exercice4Activity. L'activité HelloActivity propose à l'utilisateur un bouton « Changer de prénom ». Lorsque ce bouton est cliqué, l'application revient à l'activité Exercice4Activity. Pour réaliser ce retour, l'activité HelloActivity est arrêtée et enlevé de la pile d'activité ce que provoque la réapparition de l'activité Exercice4Activity. Pour ce faire, ajouter à la méthode onClick du bouton l'instruction suivante : `super.finish()`;

Etape 6 : réaliser le retour à l'activité MainActivity. L'activité HelloActivity propose à l'utilisateur un bouton « Choisir un autre exercice ». Lorsque ce bouton est cliqué, l'application revient à l'activité MainActivity. Pour réaliser ce retour, l'activité HelloActivity fait une demande d'intention pour l'activité MainActivity :

```
Intent intent = new Intent(this, MainActivity.class);
```

```
intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
```

```
startActivity(intent);
```

Vous remarquerez que le flag `Intent.FLAG_ACTIVITY_CLEAR_TOP` est utilisé pour éviter l'empilement des activités dans la pile. Enlever ce flag et tester votre application pour comprendre son intérêt : Cliquer plusieurs fois sur « Choisir un autre exercice » et ensuite cliquer sur le bouton retour ◀ ou ↶ pour vérifier l'empilement des activités

Etape 7 : Informer sur le retour HelloActivity vers Exercice4Activity : Changer de prénom ou bouton retour ◀ ou ↶. Prendre exemple sur les transparents 11, 12 et 13 du TD2.

1. Modifier le lancement de la demande d'intention de la classe Exercice4Activity en utilisant la méthode `startActivityForResult`. Cette méthode fournira à l'intention un code de requête qui sera renvoyé lors de l'arrêt de l'activité HelloActivity (voir transparent 11 du TD2). Remplacer l'instruction `startActivity(intent);` de l'activité Exercice4Activity par l'instruction :

```
startActivityForResult(intent, EXERCICE_4_HELLO_REQUEST);
```

EXERCICE_4_HELLO_REQUEST est une constante représentant le code de requête. Ajouter l'instruction suivante comme attribut de classe :

```
public final static int EXERCICE_4_HELLO_REQUEST = 0;
```

2. Ajouter une méthode `onActivityResult` à la classe Exercice4Activity. Cette méthode sera appelée lors de l'arrêt de l'activité HelloActivity. On pourra vérifier à l'aide du code de requête que c'est bien cette activité qui redonne la main (remarque : dans notre cas c'est peu utile vu que l'on a une seule activité pouvant dépendre de Exercice4Activity mais utile dans le cas de MainActivity par exemple – à voir). Ajouter les instructions du transparent 12 du TD2 à la classe Exercice4Activity.
3. Le `resultCode` de la méthode `onActivityResult` permettra de déterminer le choix utilisateur lors de l'arrêt de l'activité HelloActivity soit bouton retour soit ◀ ou ↶ soit le bouton « changer prénom ». Pour ce dernier cas, ajouter à la classe HelloActivity avant l'instruction arrêtant l'activité `super.finish();` l'instruction : `setResult(RESULT_OK);` (voir transparent 13 du TD2).