

Trimestre Hiver, 2017

Mohamed Lokbani

IFT1155 – Examen Intra –

Inscrivez tout de suite : votre nom et le code permanent.

Nom : _____ | Prénom(s) : _____

Signature : _____ | Login : _____

Date : Mercredi 22 février 2017

Durée : 2 heures (de 18h30 à 20h30)

Local : Z-317, Pavillon Claire McNicoll

Directives :

- Toute documentation est permise.
- Calculatrice **non** permise.
- Répondre directement sur le questionnaire.
- Les réponses **doivent être brèves, précises, claires et nettement présentées.**

1. _____ /20 (1.1)

2. _____ /20 (2.1, 2.2, 2.3, 2.4)

3. _____ /20 (3.1)

4. _____ /20 (4.1)

5. _____ /20 (5.1)

Total : _____ /100

Directives officielles

* Interdiction de toute communication verbale pendant l'examen.

* Interdiction de quitter la salle pendant la première heure.

* L'étudiant qui doit s'absenter après la première heure remettra sa carte d'étudiant au surveillant, l'absence ne devant pas dépasser 5 minutes. Un seul étudiant à la fois peut quitter la salle.

* Toute infraction relative à une fraude, un plagiat ou un copiage est signalée par le surveillant au directeur de département ou au professeur qui suspend l'évaluation.

F.A.S

Exercice 1 (20 points) trouvez la meilleure correspondance pour les descriptions ci-dessous en y associant un nombre de 1 à 10. Le nombre ne peut-être utilisé qu'une seule fois. Nous avons complété une description à titre d'exemple.

A. Un projet Android (1) définit un nom de _____ (2) unique dans son fichier _____ (3).

B. Une _____ (4) Android est implémenté en créant une sous-classe de la classe _____ (5) définie dans l'API Android.

C. La méthode _____ (6) est définie dans la classe Activity, mais doit en général être redéfinie dans notre sous-classe.

D. L'annotation _____ (7) est utilisée ici pour s'assurer que l'on est bien en train de redéfinir la bonne méthode avec le bon prototype.

E. On initialise l'interface graphique à l'aide de la description faite dans le fichier _____ (8).

F. La classe _____ (9) contient des sous-classes définissant des références de ressources.

G. La méthode _____ (10) de la classe Activity permet de retrouver un objet java représentant un objet graphique de l'interface à partir d'un identifiant fourni dans la description _____ (11) de l'interface.

Activité	
Activity	
Android	1
AndroidManifest.xml	
findViewById	
main.xml	
onCreate	
Override	
Package	
R	
XML	

Exercice 2 (20 points) expliquez ce qui suit (ces définitions sont correctes) :

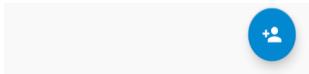
2.1 L'objet Bundle a la caractéristique de pouvoir « survivre » à la destruction du processus de l'application.

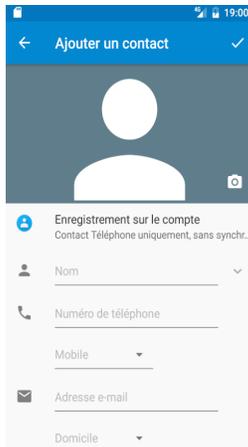
2.2 La méthode onDestroy() représente en quelque sorte les dernières volontés de l'activité.

2.3 Dans un intent, les catégories permettent de raffiner la demande d'action et la sélection du composant susceptible de réaliser l'action. La catégorie est optionnelle dans un intent implicite, mais elle est souvent nécessaire pour obtenir le résultat désiré.

2.4 Si on utilise un Intent implicite, il est souhaitable lorsque le composant n'est pas standard de s'assurer que ce composant est bien disponible sur l'appareil.

Exercice 3 (20 points) « Contacts » est une application Android présente sur la plupart des appareils. Elle permet de rassembler sur un même toit tous les contacts de l'utilisateur, qu'ils proviennent de Google+, Hangouts, Gmail...

	
<p>On commence par lancer l'application « Contacts » en cliquant sur le raccourci « Contacts ».</p>	<p>Dans l'application, on clique par la suite sur le signe « + » pour ajouter un nouveau contact.</p>



Ajouter un nouveau contact consiste à compléter les différents champs comme le nom, numéro de téléphone, etc. Par la suite, nous allons associer une photo à ce contact. Nous cliquons pour cela sur l'icône appareil photo. Une nouvelle fenêtre est affichée pour nous demander de choisir entre « Prendre une photo » ou « Sélectionner une photo » déjà disponible.



Nous allons sélectionner « Prendre une photo ». L'application « Caméra » est lancée. Nous allons prendre la photo, valider la prise, pour finalement l'enregistrer avec ce nouveau contact. Au moment de l'enregistrement, l'application détecte qu'il faut recadrer l'image pour avoir les dimensions appropriées. Android va lancer l'application par défaut responsable de ce traitement, la seule

disponible sur l'appareil. On dimensionne la photo et on valide. La photo est enregistrée et elle fait maintenant partie du contact.

Expliquer comment Android va réaliser ces différentes opérations en arrière-plan. Comment les intents ainsi que les composantes fondamentales sont-ils utilisés dans un tel scénario?

On ne vous demande pas d'écrire du code.

Exercice 4 (20 points) dessiner l'interface graphique à partir du fichier XML ci-dessous dont la syntaxe est correcte, en prenant soin d'inclure un résumé de votre démarche.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent" >
  <TextView
    android:id="@+id/TextView01"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="#00f"
    android:gravity="center"
    android:padding="25dp"
    android:text="@string/rouge"
    android:textColor="#fff" />
  <TextView
    android:id="@+id/TextView02"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:background="#ffff00"
    android:gravity="center"
    android:padding="25dp"
    android:text="@string/vert"
    android:textColor="#000" />
  <TextView
    android:id="@+id/TextView03"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:layout_toLeftOf="@+id/TextView02"
    android:background="#0f0"
    android:gravity="center"
    android:padding="25dp"
    android:text="@string/jaune"
    android:textColor="#000" />
  <TextView
    android:id="@+id/TextView04"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:layout_toRightOf="@+id/TextView03"
    android:background="#ff0000"
    android:gravity="center"
    android:padding="25dp"
    android:text="@string/noir"
    android:textColor="#fff" />
  <TextView
    android:id="@+id/TextView05"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:background="#000"
    android:gravity="center"
    android:padding="25dp"
    android:text="@string/bleu"
    android:textColor="#fff" />
</RelativeLayout>
```

Code des couleurs

bleu : #00f
blanc : #fff
noir : #000
vert : #0f0
jaune : #ffff00
rouge : #ff0000

Exercice 5 (20 points) soit les fichiers « Java » de l'application « LifeCycle » :

<pre>1 public class MainActivity extends TracerActivity { 2 @Override 3 protected void onCreate(Bundle savedInstanceState) { 4 super.onCreate(savedInstanceState); 5 setContentView(R.layout.activity_main); 6 } 7 public void onClick(View view) { 8 Intent intent = new Intent(this, SecondActivity.class); 9 startActivity(intent); 10 } 11 }</pre>	MainActivity.java
<pre>1 public class SecondActivity extends TracerActivity { 2 @Override 3 protected void onCreate(Bundle savedInstanceState) { 4 super.onCreate(savedInstanceState); 5 setContentView(R.layout.activity_second); 6 } 7 }</pre>	SecondActivity.java
<pre>1 public class TracerActivity extends Activity { 2 public static final String APP_TAG = "Garfield"; 3 public static String nom; 4 public static String tableau[]; 5 @Override 6 protected void onCreate(Bundle savedInstanceState) { 7 super.onCreate(savedInstanceState); 8 affiche("onCreate"); 9 } 10 @Override 11 protected void onPause() { 12 super.onPause(); 13 affiche("onPause"); 14 } 15 @Override 16 protected void onResume() { 17 super.onResume(); 18 affiche("onResume"); 19 } 20 @Override 21 protected void onStop() { 22 super.onStop(); 23 affiche("onStop"); 24 } 25 @Override 26 protected void onDestroy() { 27 super.onDestroy(); 28 affiche("onDestroy"); 29 } 30 @Override 31 protected void onRestoreInstanceState(Bundle savedInstanceState) { 32 super.onRestoreInstanceState(savedInstanceState); 33 affiche("onRestoreInstanceState"); 34 } 35 @Override 36 protected void onSaveInstanceState(Bundle outState) { 37 super.onSaveInstanceState(outState); 38 affiche("onSaveInstanceState"); 39 } 40 private void affiche(String NomMethode) { 41 String nom = this.getClass().getName(); 42 String [] tableau = nom.split("\\."); 43 Log.i(APP_TAG, NomMethode + " " + tableau[tableau.length - 1]); 44 } 45 }</pre>	TracerActivity.java

Il n'y a aucune erreur dans le code!

Quand l'application est lancée ...

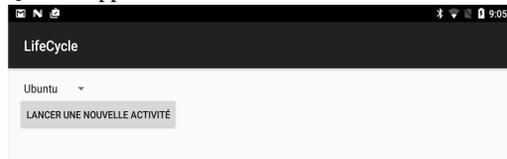


Figure -1-

Après avoir cliqué sur le bouton « LANCER UNE ... »



Figure -2-

Donnez la séquence de messages affichés dans le LogCat pour les situations suivantes :

1 on est en mode portrait.

- a- on lance l'application « LifeCycle ». On obtient la « Figure -1- ».
- b- on ne fait rien, après quelques secondes l'écran est verrouillé et puis il s'éteint.
- c- on déverrouille l'écran (l'écran va s'allumer donc).
- d- on clique sur le bouton « applications récentes » (carré) pour ouvrir une liste d'applications récemment utilisées.
- e- on sélectionne de nouveau notre application.
- f- on clique sur le bouton « Maison » (« Home »).
- g- on relance notre application.
- h- on clique sur le bouton « retour » (« Back »).

2 on efface tout et on commence à zéro. On est en mode portrait.

a- on lance l'application « LifeCycle ». On obtient la « Figure -1- ».

b- on clique sur le bouton « LANCER UNE NOUVELLE ACTIVITÉ ». On obtient la « Figure -2- ».

c- on ne fait rien, après quelques secondes l'écran est verrouillé et puis il s'éteint.

d- on déverrouille l'écran (l'écran va s'allumer donc).

e- on clique sur le bouton « retour » (« Back »).

3 on efface tout et on commence à zéro. On est en mode portrait.

- a- on lance l'application « LifeCycle ». On obtient la « Figure -1- ».
- b- on tourne l'écran en mode paysage.
- c- on clique sur le bouton « LANCER UNE NOUVELLE ACTIVITÉ ».
- d- on ne fait rien, après quelques secondes l'écran est verrouillé et puis il s'éteint.
- e- on déverrouille l'écran (l'écran va s'allumer donc).
- f- on tourne l'écran en mode portrait.

4. En conclusion, quel est l'effet du :

- bouton « Home » :

- bouton « Back » :

- bouton « applications récentes » :

- bouton « la mise en veille de l'écran » :