



Android ListView

Cours 5

ListView, List item selector, Adapter, OnItemClickListener



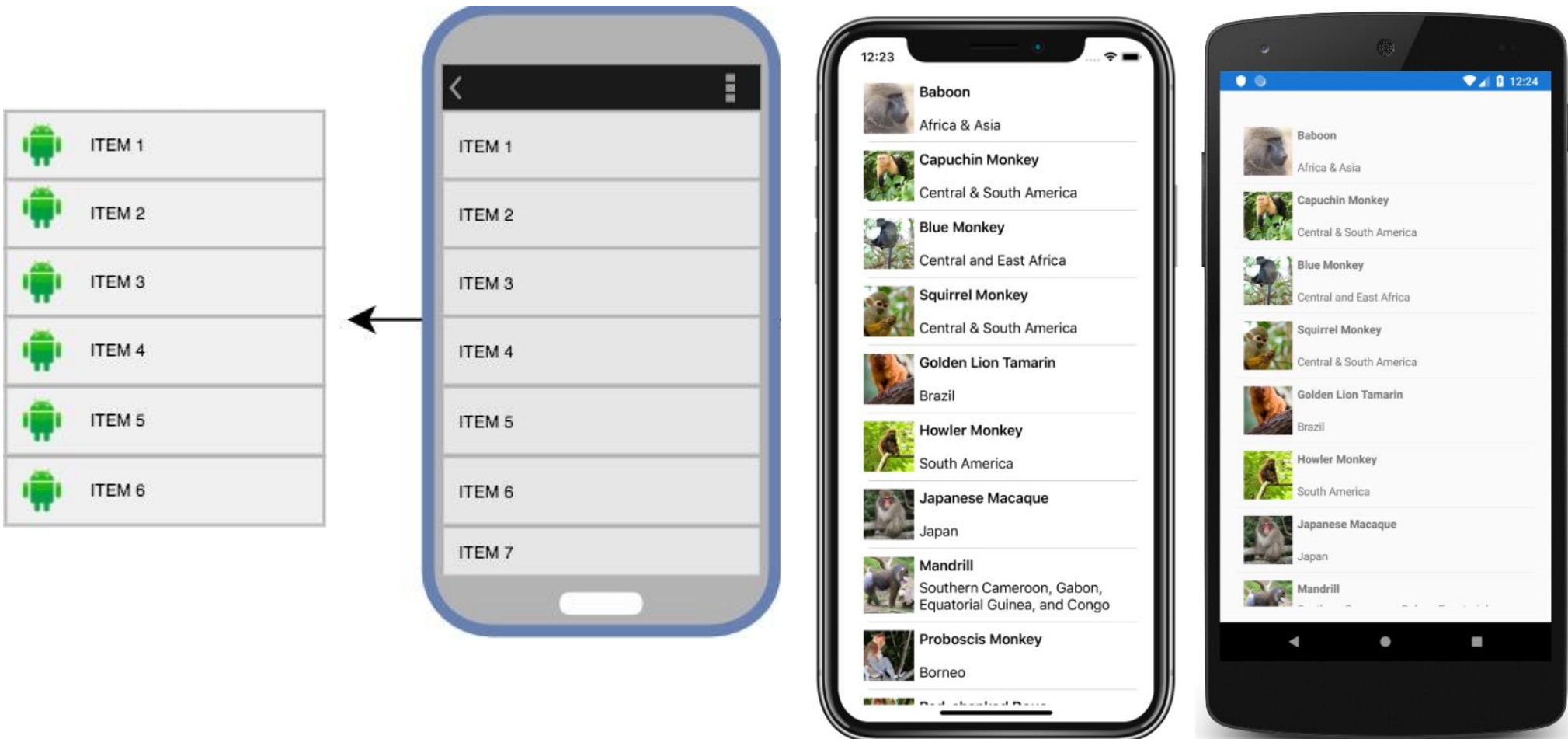


Programme de la séance

- ListView
- Adapter
- Listener

ListView

- `android.widget.ListView` est largement utilisé dans les applications mobiles Android.
- Il affiche tous les éléments de la liste sous la forme d'une liste verticale.





Comment obtenir un objet ListView?

- Il existe deux manières de créer un ListView.
 - Utilisez le widget `android.widget.ListView` directement.
 - Faites votre classe d'activité hérite de `android.app.ListActivity`. Vous pouvez ensuite appeler la méthode `setAdapter()` pour définir les données d'élément.

- La méthode `onListItemClick()` est utilisée pour répondre à une action de clic sur l'élément de la liste.



Adapter

- Après avoir obtenu l'objet ListView, vous devez y ajouter des données d'élément via **Adapter**.
- Les adapteurs permettent de faire le mapping entre les données et la ListView
- Les adaptateurs les plus utilisés sont:
 - **ArrayAdapter**
 - **SimpleAdapter**
 - **BaseAdapter**
- Il existe d'autres adapters, mais ces trois sont les plus utilisés.

Propriétés et méthodes des ListView.

- **android:choiceMode**: La valeur peut être singleChoice (ne peut sélectionner qu'un seul élément), multipleChoice (peut sélectionner plusieurs éléments) ou aucune.
- **android:divider**: Il s'agit d'un identifiant d'objet de couleur ou de dessin, utilisé pour définir la couleur du séparateur.
- **android:dividerHeight**: définissez la hauteur du diviseur.
- **android:entries**: la valeur est un identifiant de ressource de tableau défini dans android studio. Les données du tableau seront affichées dans le ListView.
- **android:headerDividersEnabled**: Valeur booléenne, true signifie que le diviseur est dessiné après la vue de l'en-tête.
- **android:footerDividersEnabled**: valeur booléenne, vrai signifie que le séparateur est tracé avant la vue du pied de page.
- **setOnItemClickListener()**: Définit l'objet OnItemClickListener qui répondra à l'événement click de l'élément ListView.

ArrayAdapter

❑ Les types de liste associés au constructeur **ArrayAdapter** :

1. **simple_list_item_1**: chaque élément de la liste est un objet `TextView`.
2. **simple_list_item_2**: chaque élément de la liste est un objet `TextView`, la taille du texte est plus grande.
3. **simple_list_item_checked**: Chaque élément de la liste est une case à cocher.
4. **simple_list_item_single_choice**: Affiche un bouton radio à la droite de chaque élément de la liste. Même si c'est un bouton radio, il peut être vérifié plusieurs fois si la valeur `android:choiceMode` de `ListView` est `multipleChoice`.
5. **simple_list_item_multiple_choice**: affiche une case à cocher à droite de chaque élément de la liste. Même si c'est une case à cocher, il est possible de cocher une case si la valeur `android:choiceMode` de `ListView` est `singleChoice`.

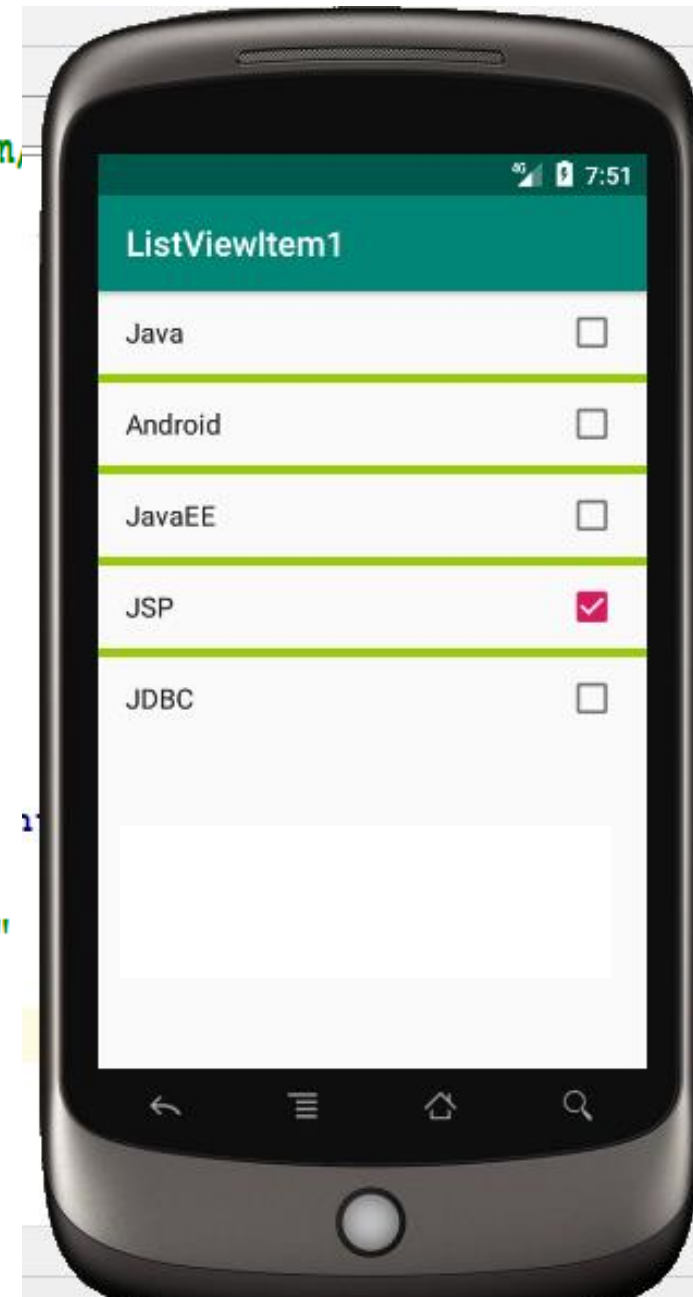
Exemple ArrayAdapter

- *activity_main.xml* :

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <ListView
        android:id="@+id/listViewExample"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:divider="@android:color/holo_green_light"
        android:dividerHeight="5dp"
        android:clickable="true"
        android:choiceMode="singleChoice" />

</RelativeLayout>
```



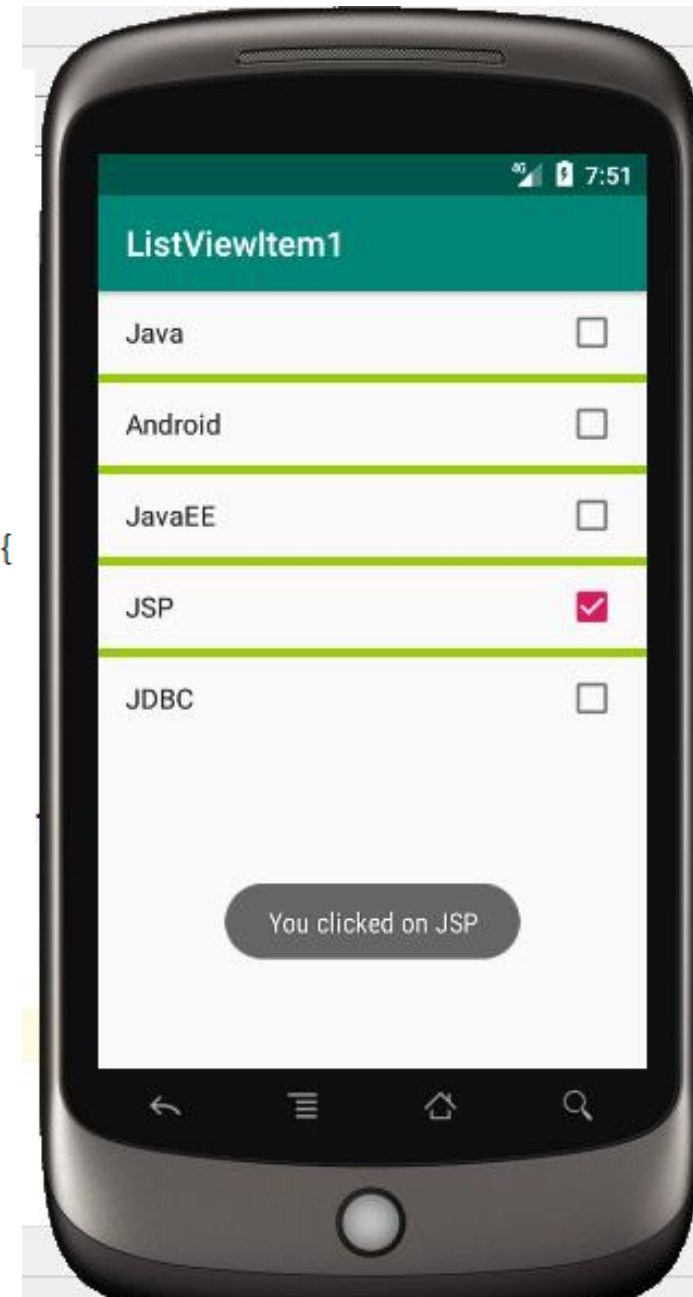
Exemple ArrayAdapter

- *MainActivity.java* 1/2:

```
package com.example.listviewitem1;
```

```
import ...
```

```
public class MainActivity extends AppCompatActivity {  
    ListView listView;  
    ArrayList<String> dataList = new ArrayList<String>();  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        dataList.add("Java"); dataList.add("Android");  
        dataList.add("JavaEE");  
        dataList.add("JSP");  
        dataList.add("JDBC");  
        listView = findViewById(R.id.listViewExample);  
    }  
}
```





Exemple ArrayAdapter

- *MainActivity.java* 2/2:

```
ArrayAdapter<String> arrayAdapter=new ArrayAdapter<String>(
    context: this,
    android.R.layout.simple_list_item_multiple_choice,
    dataList);
listView.setAdapter(arrayAdapter);
listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
        Object clickItem=adapterView.getAdapter().getItem(i);
        Toast.makeText(
            context: MainActivity.this,
            text: "You clicked on "+clickItem.toString(),
            Toast.LENGTH_LONG).show();
    }
});
}
```



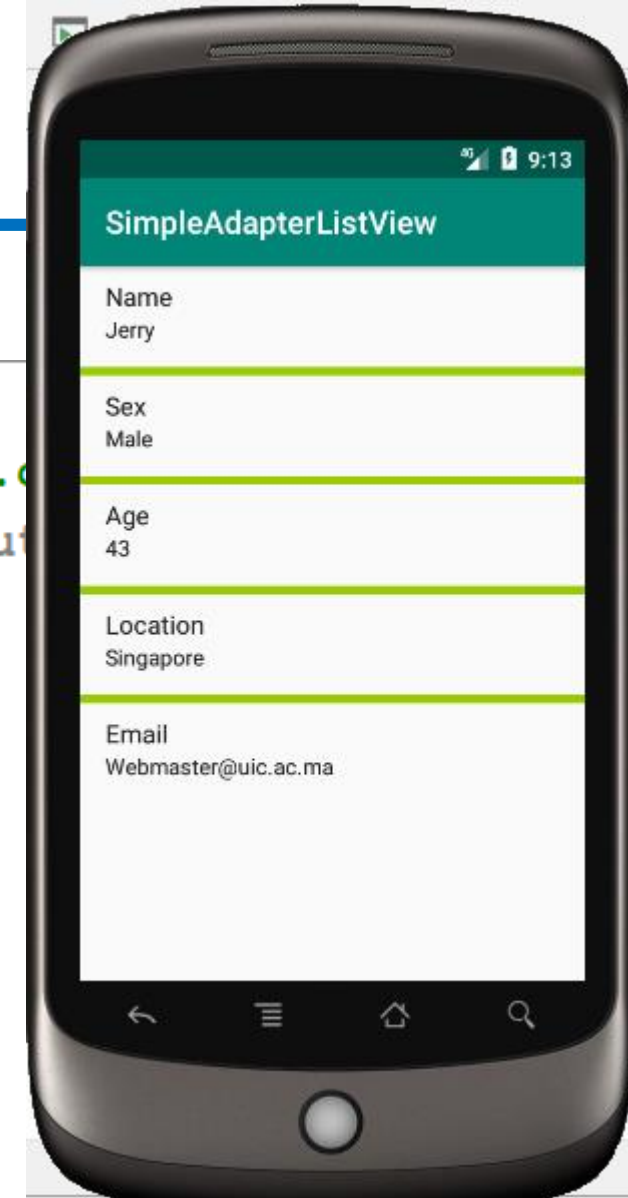
❑ Il y a 5 paramètres dans le constructeur `SimpleAdapter`.

- 1. 1er paramètre:** C'est l'objet de contexte avec lequel ce `SimpleAdapter` sera utilisé.
- 2. 2e paramètre:** il s'agit d'un objet `List<Map<String, Object>>`, chaque objet de la liste représentant une donnée d'élément `ListView`.
- 3. 3e paramètre:** il s'agit du xml de présentation du composant d'interface utilisateur de `ListView`. Il décide quels composants de l'interface utilisateur seront affichés dans chaque élément `ListView`.
- 4. 4eme paramètre:** ce tableau `String` contient les clés de l'objet `map`. Chaque valeur de carte associée à une clé sera affichée dans l'élément `ListView`.
- 5. 5eme paramètre:** ce tableau `int` contient l'identifiant de ressource de l'objet `view` dans le fichier xml d'agencement d'article (deuxième paramètre spécifié). Chaque objet de vue affichera la valeur associée extraite des clés d'utilisation des objets de la carte (quatrième paramètre spécifié) par ordre.

Exemple SimpleAdapter

- *activity_main.xml* :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <ListView
        android:id="@+id/listViewExample"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:divider="@android:color/holo_green_light"
        android:dividerHeight="5dp"
        android:clickable="true"
        android:choiceMode="singleChoice" />
</LinearLayout>
```



Exemple SimpleAdapter

- *MainActivity.java* 1/3:

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        String[] titleArr={"Name", "Sex", "Age", "Location", "Email"};
        String[] descArr={"Jerry", "Male", "43", "Singapore", "Webmaster@uic.ac.ma"};

        ArrayList<Map<String, Object>> itemDataList = new ArrayList<Map<String, Object>>();
        int titleLen=titleArr.length;
        for(int i =0; i < titleLen; i++) {
            Map<String,Object> listItemMap = new HashMap<String,Object>();
            listItemMap.put(k: "title", titleArr[i]);
            listItemMap.put(k: "description", descArr[i]);
            itemDataList.add(listItemMap);
        }
    }
}
```



Exemple SimpleAdapter

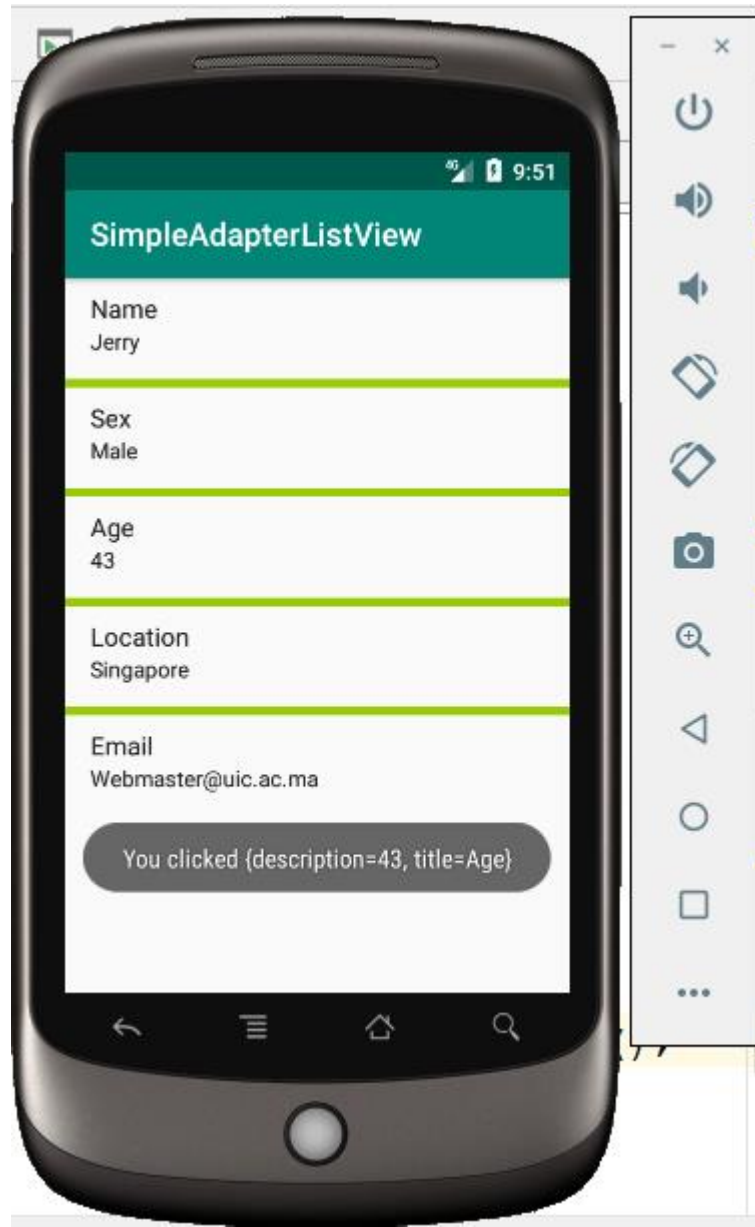
- *MainActivity.java* 2/3:

```
SimpleAdapter simpleAdapter = new SimpleAdapter(  
    context: this,  
    itemDataList,  
    android.R.layout.simple_list_item_2,  
    new String[]{"title", "description"},  
    new int[]{android.R.id.text1, android.R.id.text2}  
);
```

```
ListView listView = (ListView) findViewById(R.id.listViewExample);  
listView.setAdapter(simpleAdapter);  
listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
    @Override  
    public void onItemClick(AdapterView<?> adapterView, View view, int index, long l) {  
        Object clickItemObj = adapterView.getAdapter().getItem(index);  
        Toast.makeText(context: MainActivity.this,  
            text: "You clicked " + clickItemObj.toString(), Toast.LENGTH_SHORT).show();  
    }  
});
```

Exemple SimpleAdapter

- *MainActivity.java 3/3:*



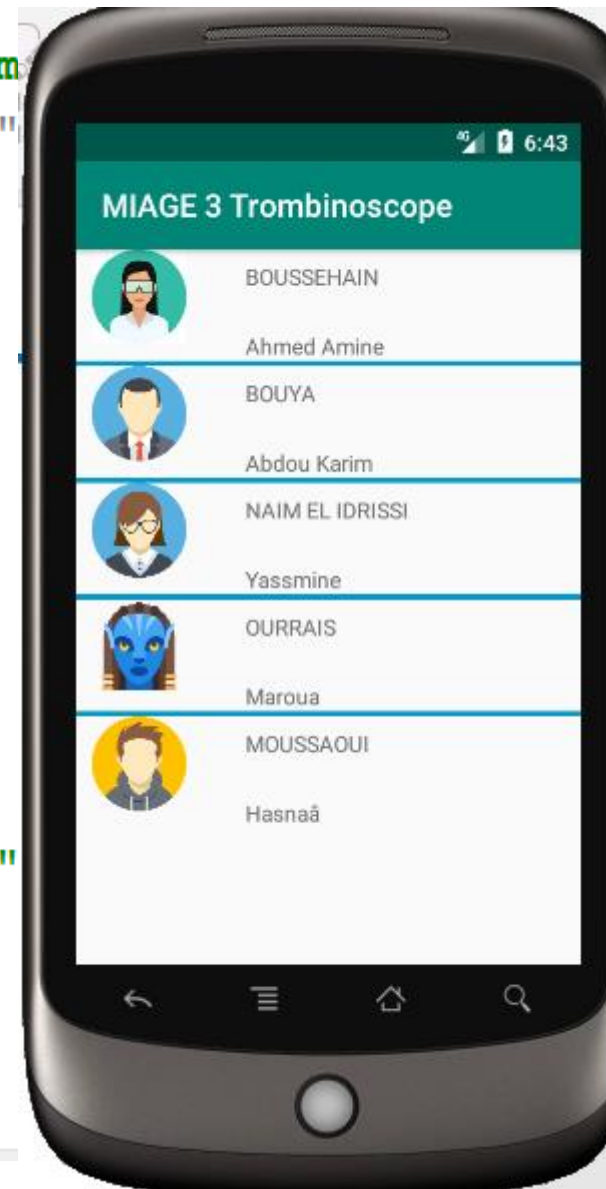
- ❑ Celui-ci personnalisera un objet **SimpleAdapter** qui utilise une disposition d'élément ListView auto-définie pour ajouter une image au début de chaque élément ListView.
- ❑ Vous devez définir un *selector* d'item dans le fichier layout. Le *selector* représente la structure des items personnalisés de la liste.

Exemple CustomAdapter

- *activity_main.xml* :

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout xmlns:android="http://schemas.android.com
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".MainActivity">

<ListView
    android:id="@+id/listViewExample"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:divider="@android:color/holo_blue_dark"
    android:dividerHeight="3dp"
    android:clickable="true"
    android:choiceMode="singleChoice" />
</LinearLayout>
```



Exemple CustomAdapter

- *List item selector.xml*:

```
<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <ImageView
        android:id="@+id/userImage"
        android:layout_width="80dp"
        android:layout_height="60dp" />
    <TextView
        android:id="@+id/userLastname"
        android:layout_width="304dp"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_marginStart="107dp"
        android:layout_marginLeft="107dp"
        android:layout_marginTop="7dp"
        android:text="TextView" />
    <TextView
        android:id="@+id/userFirstname"
        android:layout_width="304dp"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_marginStart="107dp"
        android:layout_marginLeft="107dp"
        android:layout_marginTop="51dp"
        android:text="TextView" />
</RelativeLayout>
```





Exemple CustomAdapter

- *MainActivity.java 1/4:*

```
import ...
```

```
public class MainActivity extends AppCompatActivity {  
    private ListView listView;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        setTitle("MIAGE 3 Trombinoscope");  
        listView = (ListView) findViewById(R.id.listViewExample);  
  
        String[] firstName = {"BOUSSEHAIN", "BOUYA", "NAIM EL IDRISSE", "OURRAIS", "MOUSSAOUI"};  
        String[] lastName = {"Ahmed Amine", "Abdou Karim", "Yassmine", "Maroua", "Hasnaâ"};  
        int[] avatar = {R.mipmap.image1, R.mipmap.image2, R.mipmap.image6, R.mipmap.image7, R.mipmap.  
        ArrayList<Map<String, Object>> itemDataList = new ArrayList<Map<String, Object>>();
```

Exemple CustomAdapter

- *MainActivity.java* 2/4:

```
int titleLen = firstName.length;
for(int i =0; i < titleLen; i++) {
    Map<String, Object> listItemMap = new HashMap<String, Object>();
    listItemMap.put(k: "imageId", avatar[i]);
    listItemMap.put(k: "lastname", lastName[i]);
    listItemMap.put(k: "firtname", firstName[i]);
    itemDataList.add(listItemMap);
}
```

```
SimpleAdapter simpleAdapter = new SimpleAdapter(
    context: this,
    itemDataList,
    R.layout.list_item_selector,
    new String[]{"imageId", "lastname", "firtname"},
    new int[]{R.id.userImage, R.id.userFirstname, R.id.userLastname});

listView.setAdapter(simpleAdapter);
```



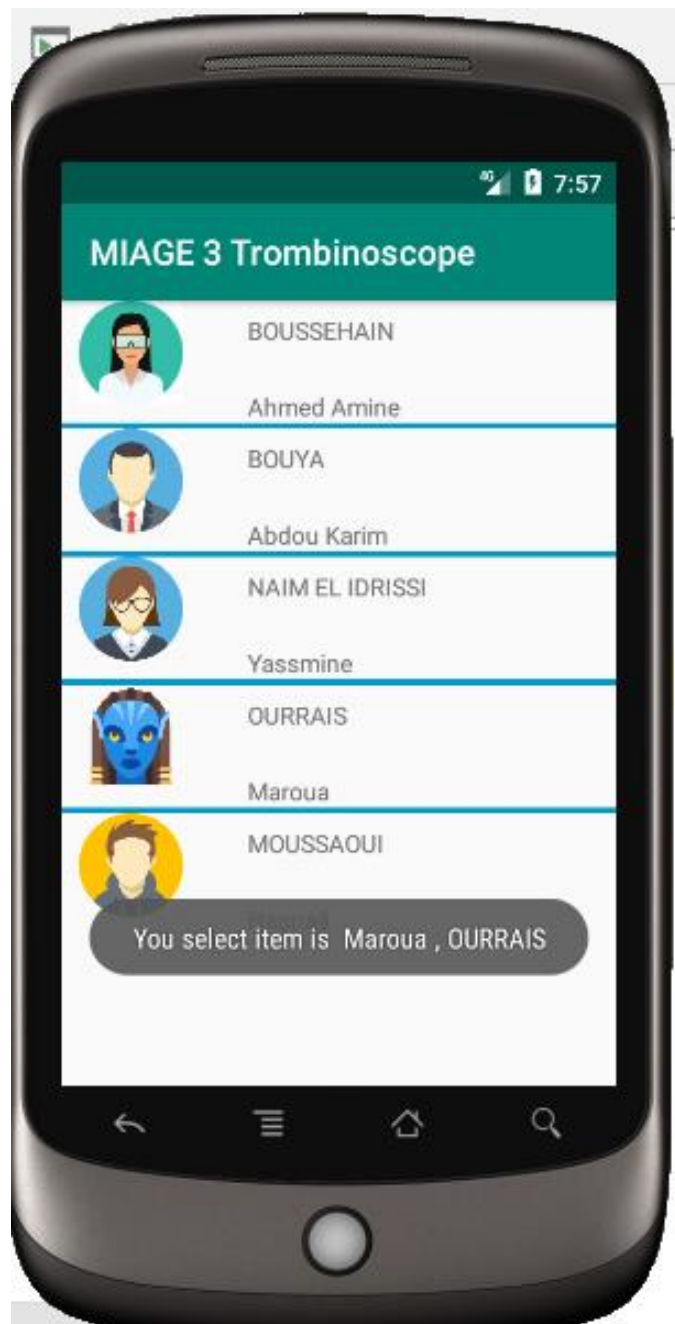
Exemple CustomAdapter

- *MainActivity.java* 3/4:

```
listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
    @Override  
    public void onItemClick(AdapterView<?> adapterView, View view, int index, long l) {  
        Object clickItemObj = adapterView.getAdapter().getItem(index);  
        HashMap clickItemMap = (HashMap)clickItemObj;  
        String itemLatname = (String)clickItemMap.get("lastname");  
        String itemFirstName = (String)clickItemMap.get("firstname");  
        Toast.makeText(  
            context: MainActivity.this,  
            text: "You select item is " + itemLatname + " , " + itemFirstName,  
            Toast.LENGTH_SHORT).show();  
    }  
});
```

Exemple CustomAdapter

- *MainActivity.java 4/4:*





- Documentation officielle :
 - ❑ <https://developer.android.com/index.html>
- Tutoriel Android :
 - ❑ <https://developer.android.com/studio/index.html>