

Building User Interface

Session 3

Objective

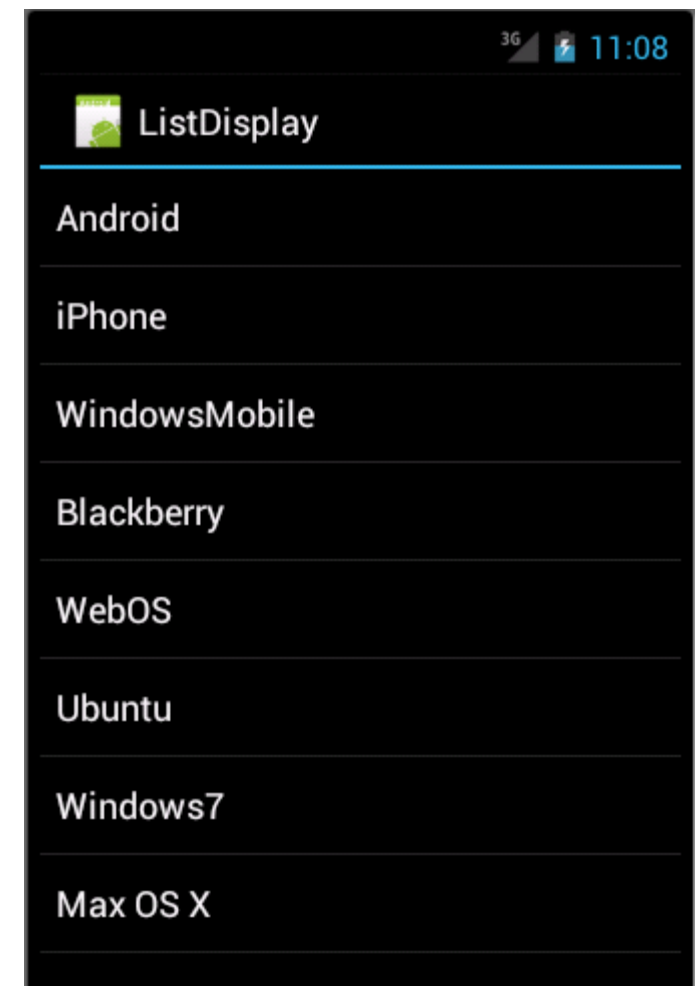
- Java Collections
- Listview, Spinner
- Gridview
- Adapter
- RecyclerView

Java Collections

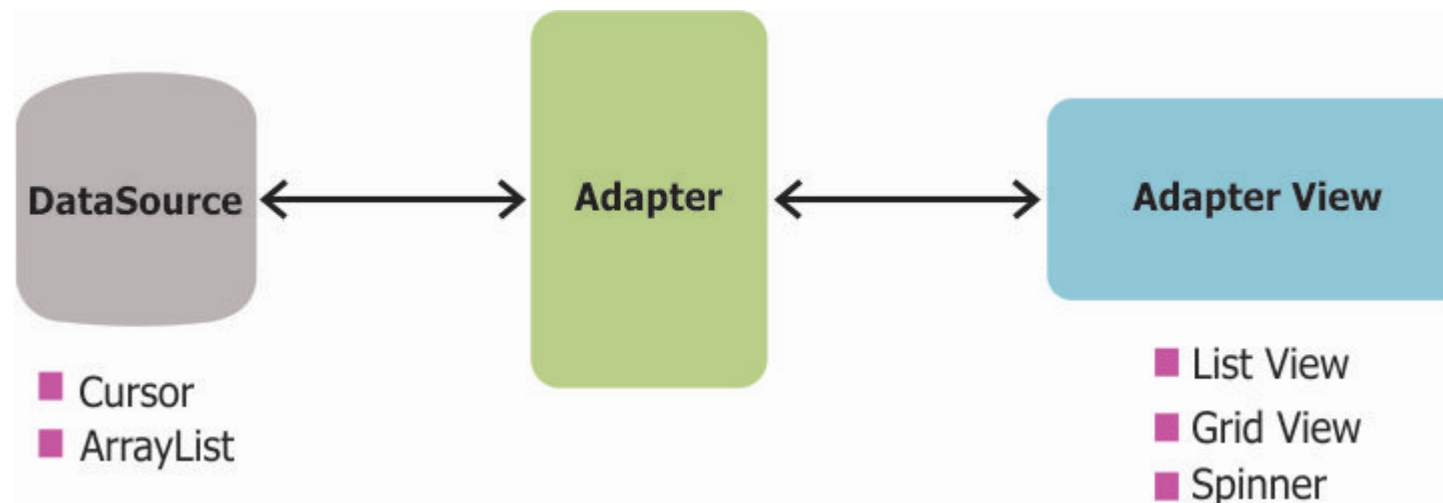
- Java cung cấp các lớp collection chuẩn để xây dựng giao diện collection: ArrayList, LinkedList, HashSet, HashMap...
- ArrayList Class:
 - The ArrayList class extends AbstractList và thực hiện List interface. ArrayList hỗ trợ mảng động (có thể thêm/bớt phần tử khi đang thực hiện)
- ArrayList Methods:
 - **void add(int index, Object element)**
 - **boolean add(Object o)**
 - **boolean contains(Object o)**
 - **Object get(int index)**
 - **Object remove(int index)**
 - **int size()**

Listview

- Android **Listview** là một view group hiển thị các thành phần theo một danh sách
 - Có thể cuộn theo chiều dọc
 - Có thể tự động chèn vào list thông qua Adapter (lấy nội dung từ nguồn khác như array hoặc database)
- Ví dụ
 - Danh bạ
 - Danh sách



ArrayAdapter



The **ArrayAdapter** là cầu nối giữa nguồn dữ liệu (**ArrayList**) và đối tượng biểu diễn trực quan (ListView) và cấu hình hai yếu tố:

- Mảng dữ liệu dùng làm nguồn cho danh sách
- Biến đổi các mục trong array thành mục tương ứng trong View object

ArrayAdapter

- Ta có thể sử dụng adapter khi nguồn dữ liệu là một mảng. Mặc định, ArrayAdapter tạo một view cho mỗi item trong mảng bằng việc gọi toString() trên mỗi item và đặt nội dung vào **TextView**
`ArrayAdapter<String> itemsAdapter = new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1, items);`

Trong Constructor này, các tham số là –

- Tham số đầu tiên **this** là ngữ cảnh của ứng dụng. Trong hầu hết các trường hợp, giữ nguyên **this**.
- Tham số thứ hai là layout được định nghĩa trong file XML và có **TextView** cho mỗi chuỗi trong mảng.
- Tham số cuối cùng là mảng các chuỗi sẽ được đưa vào trong Textview.

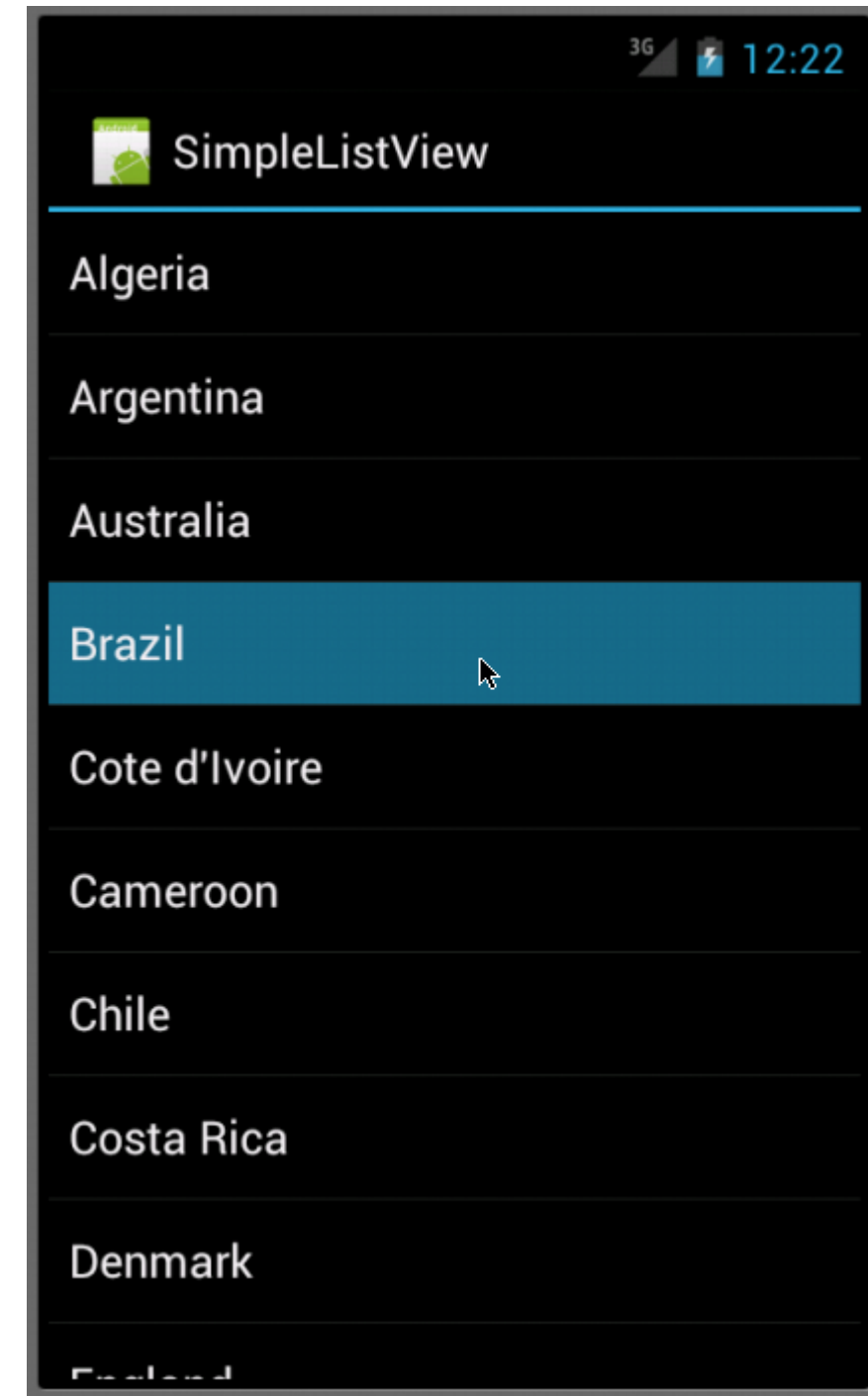
Sau khi tạo ArrayAdapter, ta kết nối adapter với ListView hiển thị dữ liệu:

```
ListView listView = (ListView) findViewById(R.id.listview);  
listView.setAdapter(adapter);
```

Listview Events

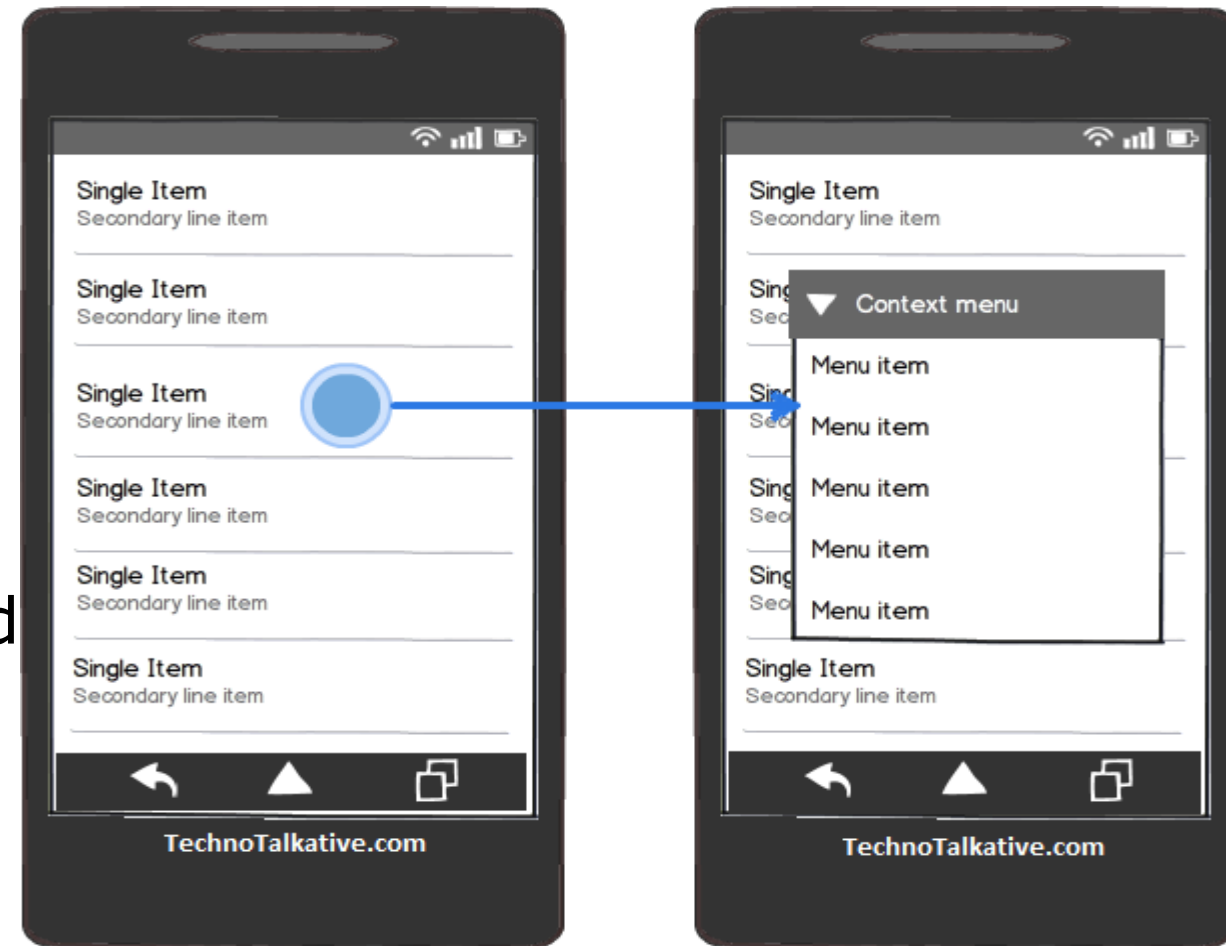
- Thêm sự kiện ClickListener cho Listview

```
listView.setOnItemClickListener(new  
OnItemClickListener() {  
    public void onItemClick(AdapterView<?> parent,  
        View view, int position, long id) {  
        // When clicked, show a toast with the TextView text  
        Toast.makeText(getApplicationContext(),  
            ((TextView) view).getText(),  
            Toast.LENGTH_SHORT).show(); }  
});
```



Listview Events

```
listview.setOnItemClickListener(new  
AdapterView.OnItemClickListener() {  
  
    @Override  
    public boolean  
onItemLongClick(AdapterView<?> arg0,  
View arg1,  
                    int arg2, long arg3) {  
    // TODO Auto-generated method  
stub  
  
    isclick = false;  
    islongclick = true;  
  
  
    return false;  
}});
```



Ví dụ1

- Tạo danh sách từ string.xml
- Mở file res/values/string.xml
- Thêm mảng string

```
}<resources>
    <string name="app_name">Listview1</string>
}    <string-array name="tut_android">
        <item>Bài 1: Hello World</item>
        <item>Bài 2: TextView, EditText, Button</item>
        <item>Bài 3: Bắt sự kiện click Button</item>
        <item>Bài 4: Thiết kế giao diện với RelativeLayout</item>
        <item>Bài 5: Thiết kế giao diện với LinearLayout</item>
        <item>Bài 6: Thực hành xây dựng ứng dụng Calculator</item>
        <item>Bài 7: Toast, CheckBox, RadioButton, Dialog</item>
        <item>Bài 8: Intent - Chuyển đổi giữa các màn hình</item>
        <item>Bài 9: Cấu trúc Project</item>
        <item>Bài 10: Vòng đời Activity</item>
    </string-array>
}</resources>
```

Ví dụ 2

- Tạo danh sách từ mảng string trong file java
- Thêm mảng string trong file java

```
private void loadData() {  
  
    listData = new ArrayList<>();  
    listData.add("Bài 1: Hello World");  
    listData.add("Bài 2: TextView, EditText, Button");  
    listData.add("Bài 3: Bắt sự kiện click Button");  
    listData.add("Bài 4: Thiết kế giao diện với RelativeLayout");  
    listData.add("Bài 5: Thiết kế giao diện với LinearLayoout");  
  
    // create adapter with listData and layout item  
    adapter = new ArrayAdapter<>(context, android.R.layout.simple_list_item_1, listData);  
  
    lvTut.setAdapter(adapter);  
}
```

- Sự kiện

```
listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
    @Override  
    public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {  
        Toast.makeText(context, listData.get(i), Toast.LENGTH_SHORT).show();  
        // do something  
    }  
});
```

Thêm phần tử vào list

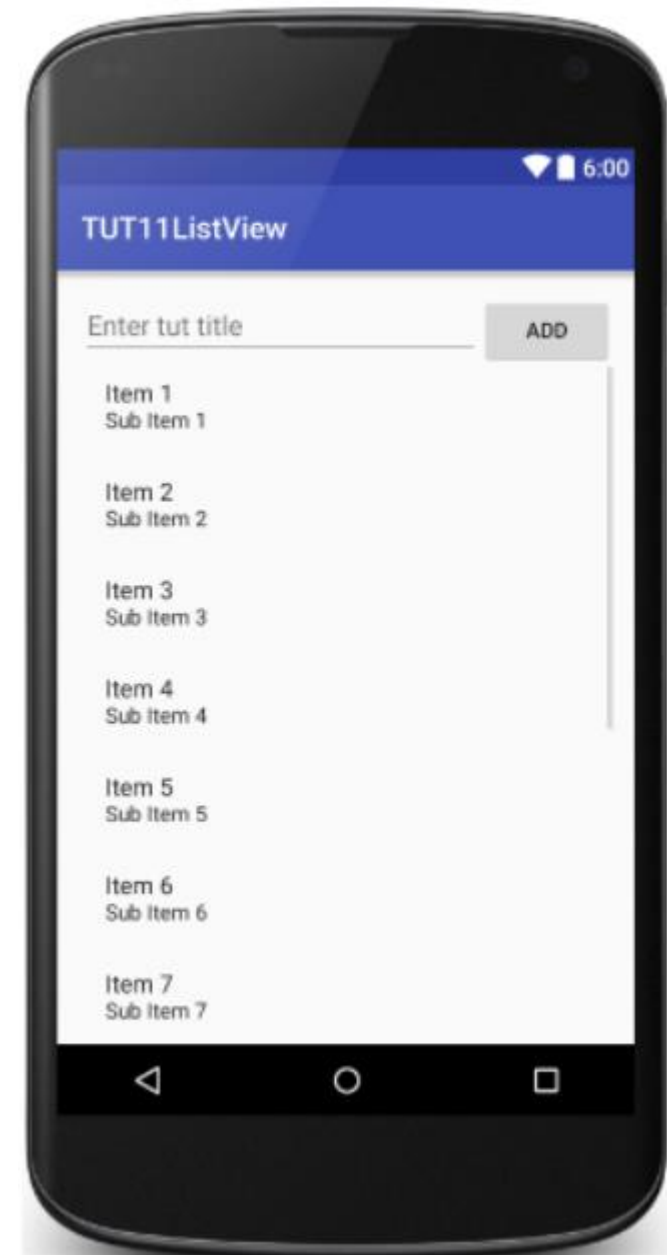
- Tạo editText và Button

```
private void addTut(){
    String tut = editTut.getText().toString().trim();

    if(TextUtils.isEmpty(tut)) {
        Toast.makeText(context, "Please enter tut title",
        Toast.LENGTH_SHORT).show();
        return;
    }

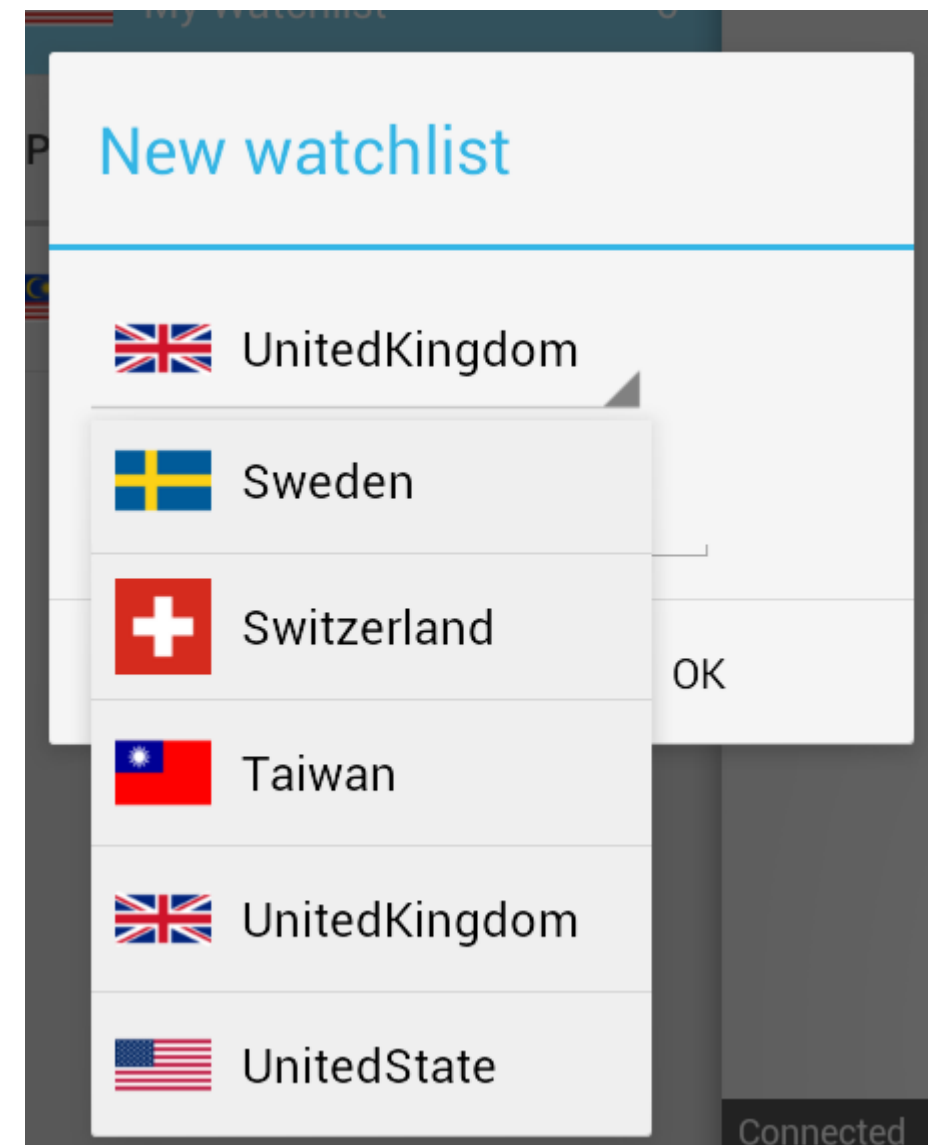
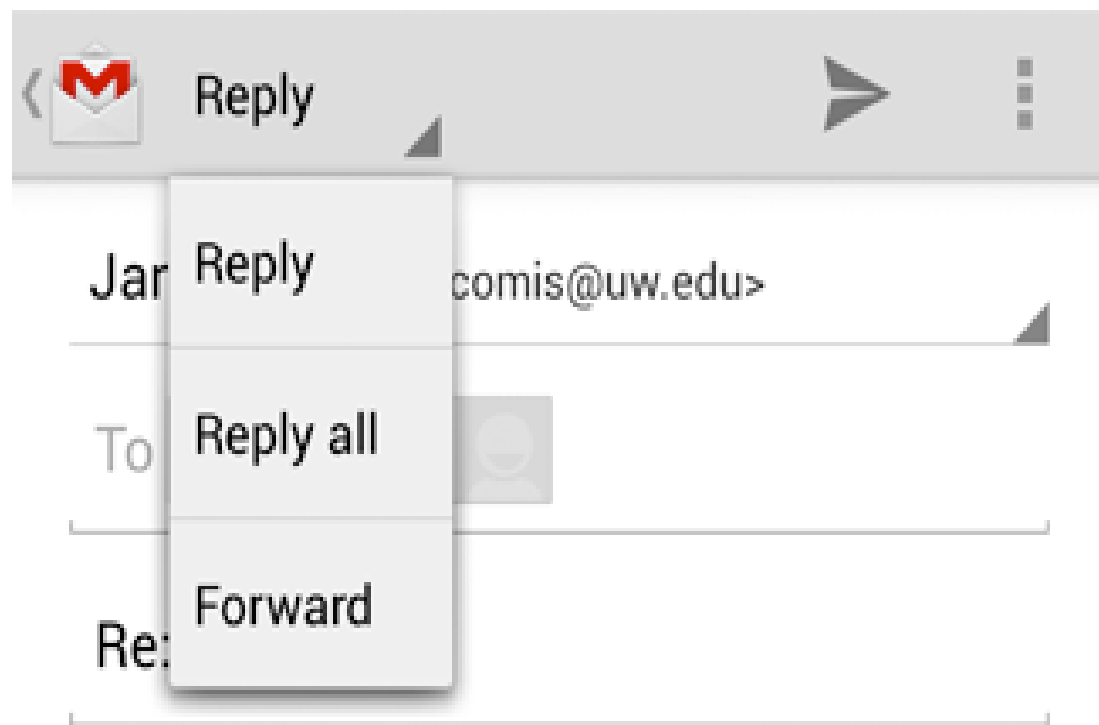
    listData.add(tut);

    // update data to show on listview
    adapter.notifyDataSetChanged();
}
```



Spinner

- Spinner allows you to select an item from a drop down menu
- For example. When you are using Gmail application you would get drop down menu as shown below, you need to select an item from a drop down menu.



Spinner

Thay đổi nội dung của **res/layout/activity_main.xml** theo layout dưới đây:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:padding="10dip"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content">

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dip"
        android:text="Category:"
        android:layout_marginBottom="5dp"/>

    <Spinner
        android:id="@+id/spinner"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:prompt="@string/spinner_title"/>

</LinearLayout>
```

Spinner

Code MainActivity.java:

```
Spinner spinner;  
List<String> categories;
```

```
// Spinner element  
Spinner spinner = (Spinner) findViewById(R.id.spinner);  
// Spinner click listener  
spinner.setOnItemSelectedListener(this); // Alt + enter → implement  
// Spinner Drop down elements  
List<String> categories = new ArrayList<String>();  
categories.add("Automobile");  
categories.add("Business Services");  
categories.add("Computers");  
categories.add("Education");  
categories.add("Personal");  
categories.add("Travel");  
// Creating adapter for spinner  
ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this,  
    android.R.layout.simple_spinner_item, categories);  
//dataAdapter.addAll(categories);  
// Drop down layout style - list view with radio button  
dataAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item); //  
attaching data adapter to spinner  
spinner.setAdapter(dataAdapter);
```

GridView

- GridView hiển thị các item trong mảng lưới hai chiều
 - Các item không cần định nghĩa trước, chúng tự động chèn vào layout bằng ListAdapter
 - Ứng dụng view ảnh, video player.



GridView

grid_layout.xml

```
<?xml version="1.0" encoding="utf-8"?>
<GridView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/grid_view"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:numColumns="auto_fit"
    android:columnWidth="90dp"
    android:horizontalSpacing="10dp"
    android:verticalSpacing="10dp"
    android:gravity="center"
    android:stretchMode="columnWidth" >

</GridView>
```


Xây dựng khuôn dạng cho từng item

- * Tạo file xml
- Tạo khuôn dạng

```
<ImageView  
    android:id="@+id/ivContent"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:adjustViewBounds="true" />
```

Tạo lớp Java

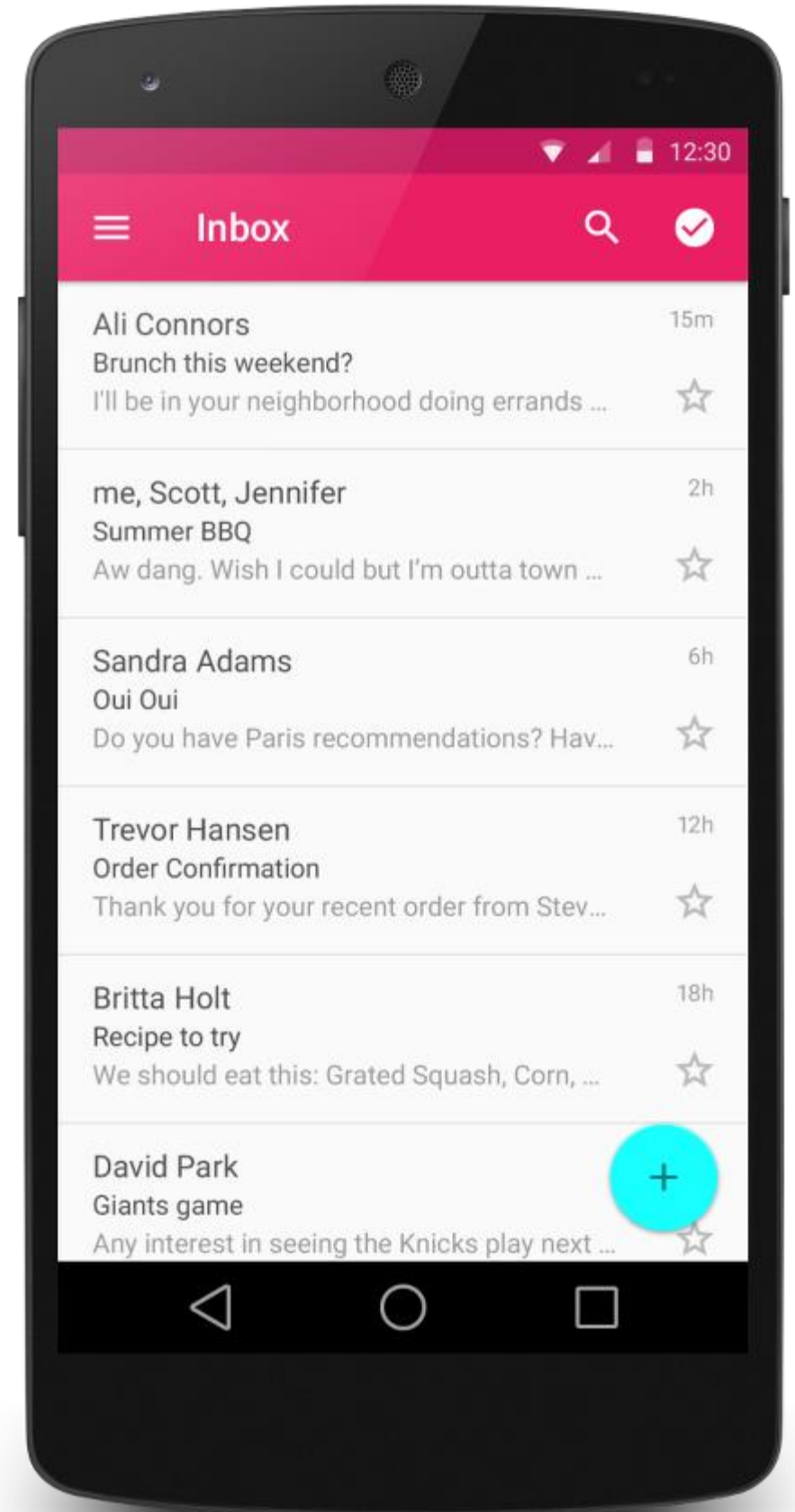
- Khai báo List (chỉ số các ảnh)
- Khai báo inflater
- Khai báo constructor
- Thực hiện @override

Gridview nâng cao

- Khi click vào item thì hiện ảnh ra màn hình mới
- Thêm một activity
- Xây dựng giao diện chứa ảnh hiển thị
- Xây dựng file java điều khiển việc hiện ảnh từ item (dùng intent)

RecyclerView

- Android RecyclerView là một view mới trong android giống ListView nhưng có nhiều ưu điểm hơn.
 - Cho phép load dữ liệu nhiều hơn ListView
 - Vuốt mượt hơn
 - Hỗ trợ đa dạng layout các phần tử trong danh sách.
 - Thường sử dụng kết hợp RecyclerView với CardView



RecyclerView

```
<android.support.v7.widget.RecyclerView  
    android:id="@+id/recycler_view"  
    android:scrollbars="vertical"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"/>
```

Build.gradle

```
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
    testCompile 'junit:junit:4.12'  
    compile 'com.android.support:appcompat-v7:23.1.1'  
    compile 'com.android.support:design:23.1.1'  
    compile 'com.android.support:recyclerview-v7:23.1.1'  
}
```

RecyclerView

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/a
ndroid"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_behavior="@string/appbar_scrolling_view_b
ehavior"
    tools:showIn="@layout/activity_main"
    tools:context=".MainActivity">

    <android.support.v7.widget.RecyclerView
        android:id="@+id/recycler_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:scrollbars="vertical" />

</RelativeLayout>
```

Custom Adapter

Khi cần hiển thị chuỗi item trong list theo cách của người dung, ta cần tạo XML layout cho mỗi item.

Tự tạo ArrayAdapter class.

Defining the Model

```
public class Movie {  
    private String title, genre, year;  
  
    public Movie() {  
    }  
  
    public Movie(String title, String genre, String year) {  
        this.title = title;  
        this.genre = genre;  
        this.year = year;  
    }  
}
```

Custom Adapter

Creating the View Template

Tạo XML layout biểu diễn view template cho mỗi item trong file: *movie_list_row.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">
    <TextView
        android:id="@+id/title"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:text="Titanic"
        android:textColor="@color/title"
        android:textSize="16dp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/genre"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/title"
        android:text="Romance" />

    <TextView
        android:id="@+id/year"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:text="1990"
        android:textColor="@color/year" />
</RelativeLayout>
```


Custom Adapter

Defining the Adapter

```
public class MoviesAdapter extends
RecyclerView.Adapter<MoviesAdapter.MyViewHolder> {

    private List<Movie> moviesList;

    public class MyViewHolder extends RecyclerView.ViewHolder {
        public TextView title, year, genre;

        public MyViewHolder(View view) {
            super(view);
            title = (TextView) view.findViewById(R.id.title);
            genre = (TextView) view.findViewById(R.id.genre);
            year = (TextView) view.findViewById(R.id.year);
        }
    }
}
```

Custom Adapter

Attaching the Adapter to a RecyclerView

```
public class MainActivity extends AppCompatActivity {  
    private ArrayList<Movie> arrayList;  
    private RecyclerView recyclerView;  
    private MoviesAdapter moviesAdapter;  
    ....  
  
    recyclerView = (RecyclerView) findViewById(R.id.recyclerview);  
    moviesAdapter = new MoviesAdapter(arrayList);  
    RecyclerView.LayoutManager manager = new  
LinearLayoutManager(getApplicationContext());  
    recyclerView.setLayoutManager(manager);  
    recyclerView.setItemAnimator(new DefaultItemAnimator());  
    recyclerView.setAdapter(moviesAdapter);  
  
}
```

Summary

- In this session, we learnt:
 - Java Collections
 - Listview, Spinner
 - GridView
 - Adapter
 - RecyclerView