



FPT POLYTECHNIC

THIẾT KẾ GIAO DIỆN TRÊN
ANDROID

**Bài 4: Các widget lựa
chọn**

www.poly.edu.vn

Nội dung bài học

1. Tổng quan
2. Listview
3. Spinner
4. Gridview
5. Autocomplex TextView

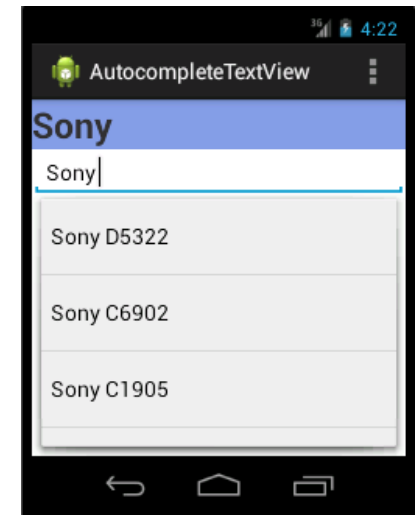
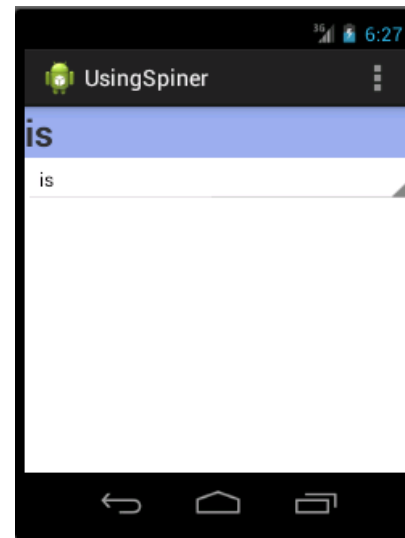
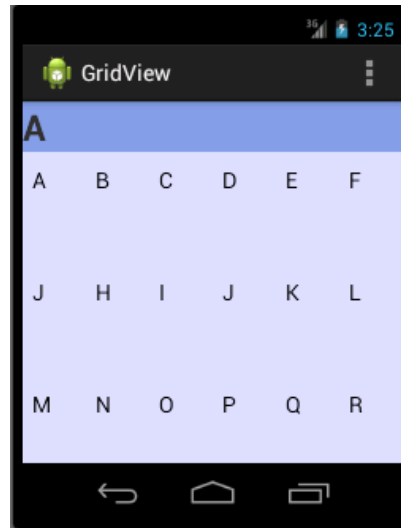
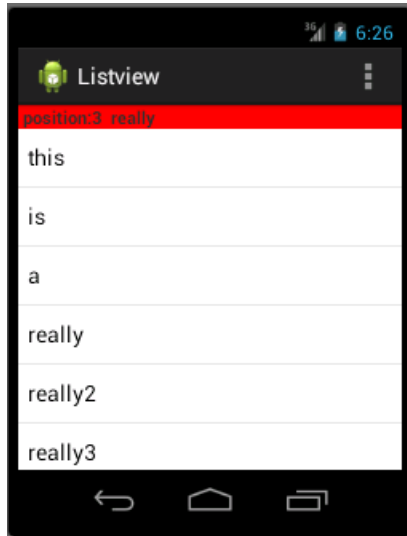


1. Tổng quan về các Collection Widget

- RadioButton và CheckButton thích hợp cho việc chọn từ một số ít các tùy chọn.
- Khi số lựa chọn lớn hơn, có thể dùng các widget khác kiểu như: *listbox*, *combobox*, *drop-down list*, *picture gallery*, v.v...
- Android ***data adapter*** cung cấp một giao diện chung cho các *selection list* (các danh sách cho phép chọn) từ các mảng nhỏ cho đến các nội dung từ CSDL.
- ***Selection view*** – các widget hiển thị một danh sách các lựa chọn mà một adapter cung cấp để người dùng thực hiện chọn lựa tại đây: ListView, Spinner, GridView, AutoCompleteTextView, Gallery

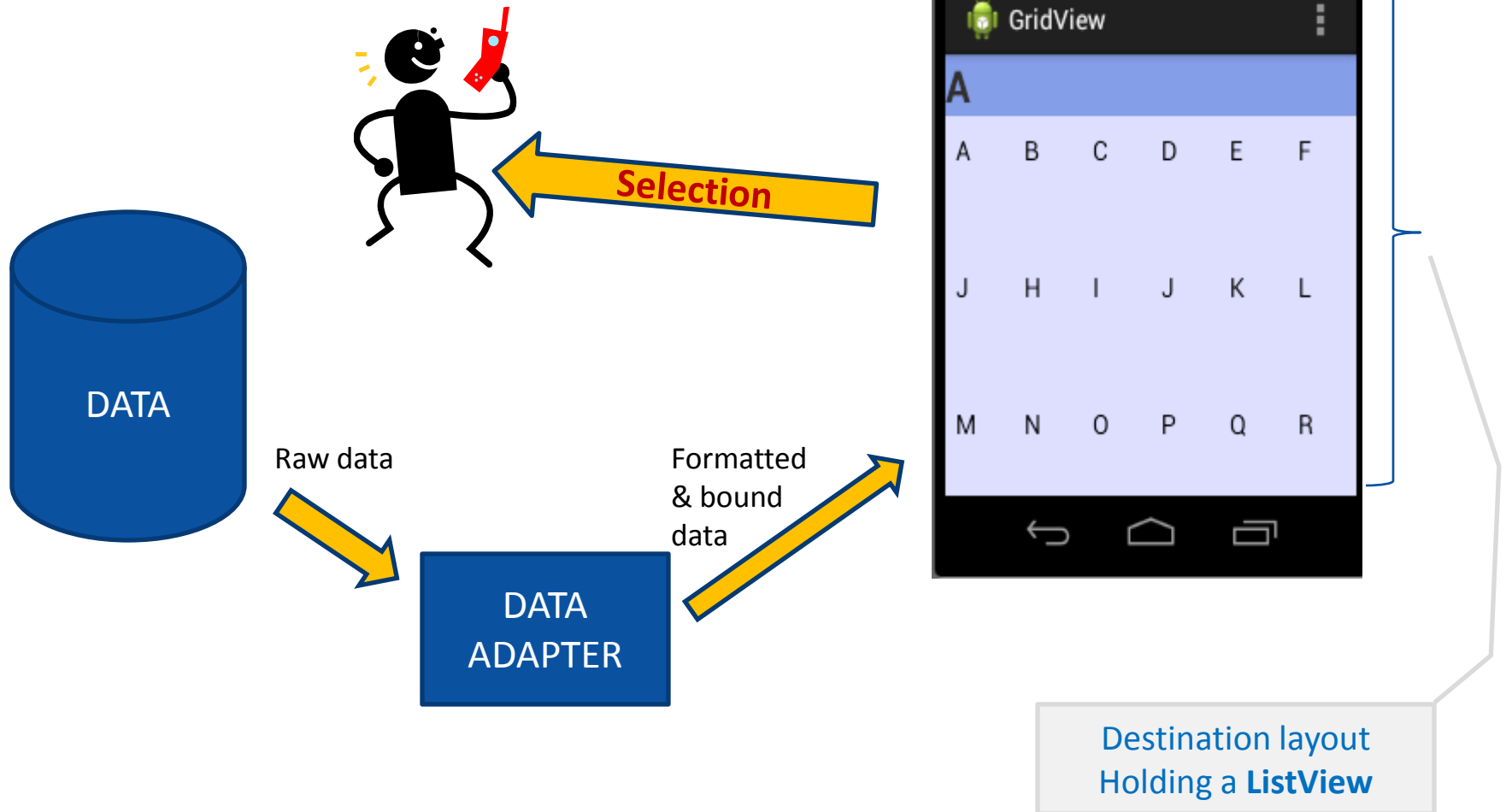
1. Tổng quan về các Selection Widget

ListView, GridView, AutoCompleteTextView, Spinner



1. Tổng quan về các Collection Widget

Displaying/Selecting Options



1. Tổng quan về các Collection Widget

Sử dụng ArrayAdapter

- Loại adapter đơn giản nhất là **ArrayAdapter**. Ta chỉ bọc một đối tượng loại này ra ngoài một mảng Java hoặc java.util.List instance từ bên trong một **ListActivity** (Chú ý: không phải một Activity...).
- Sử dụng mảng dữ liệu định sẵn ArrayAdapter.

```
String [] arr={"this", "is", "a", "really", "silly", "list"}  
ArrayAdapter<String> adapter = new  
    ArrayAdapter<String>(this, android.R.layout.simple_list_item_1, arr);
```
- Dữ liệu từ Data source (arr) sẽ được gắn vào ArrayAdapter, ArrayAdapter sẽ gắn vào ListView.

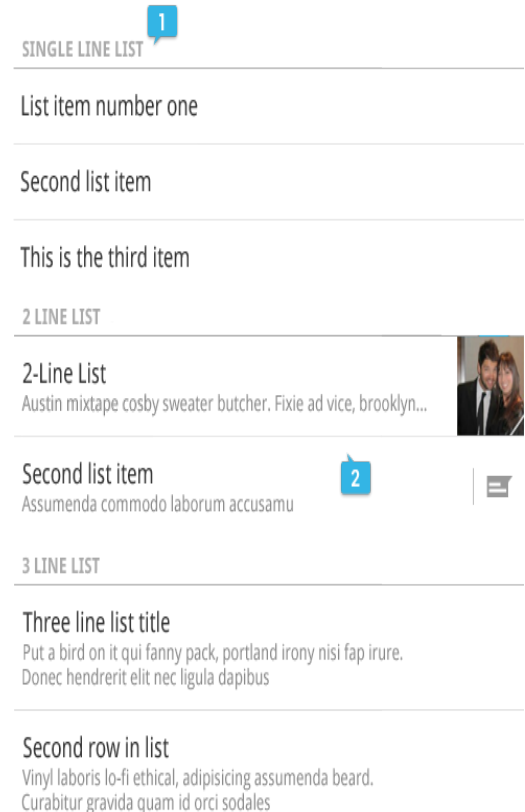
1. Tổng quan về các Collection Widget

Sử dụng ArrayAdapter

- Tham số 1 của constructor ArrayAdapter : **this**, chính là context của Activity hiện tại, bạn có thể viết **MainActivity.this**.
- Tham số thứ 2 resource ID của view sẽ hiển thị danh sách (thường là một ListView , chẳng hạn cái đã được định nghĩa trong resource : ***android.R.layout.simple_list_item_1*** ở trên). Trong đó **android** chính là layout Listview mà được Android xây dựng sẵn. Còn **simple_list_item_1.xml** được lưu trong thư mục /res/layout/
- Tham số thứ 3: chính là **arr** (data source), bạn có thể truyền vào ArrayList.

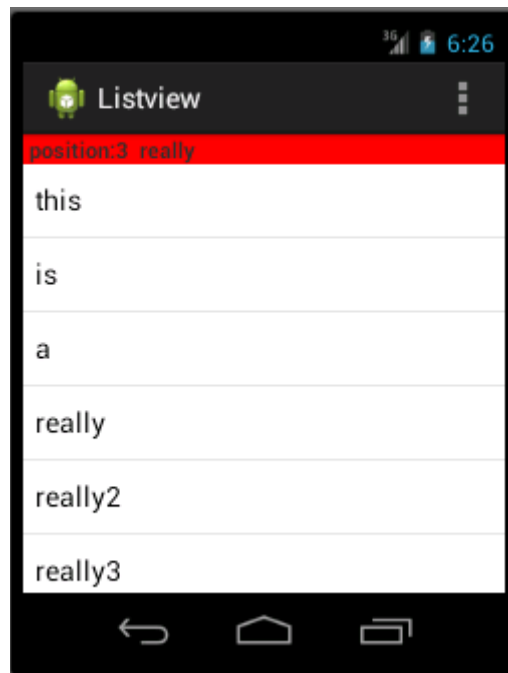
2. ListView

- List biểu diễn nhiều dòng dữ liệu theo hàng dọc, chia làm 2 phần:
 - 1. Section Divider: để tổ chức các item thành một nhóm dễ quản lý.
 - 2. Line Items: có thể chứa nhiều loại dữ liệu khác nhau như checkbox, icon, action buttons...
- ListView là một ví dụ đơn giản nhất về List



2. ListView

- **Ví dụ 1:** giao diện dưới có 2 control:
 - **ListView** : dùng để hiển thị mảng dữ liệu
 - **TextView** có màu đỏ : Dùng để hiển thị vị trí và giá trị của phần tử được chọn trong ListView khi được click vào.



2. ListView

- **Ví dụ 1: ListView** (1 of 4) Dưới đây là nội dung của **activity_main.xml**: Đặt id cho Listview là **list** bạn có thể định dạng thêm một số đặc tính khác nữa cho thẻ nếu cần.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <TextView
        android:id="@+id/selection"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:background="#ff0000cc"
        android:textStyle="bold" />
    <ListView
        android:id="@android:id/list"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:drawSelectorOnTop="false"
        android:scrollbars="none" />

    <TextView android:id="@android:id/empty"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Empty set" />
</LinearLayout>
```

2. ListView

- **Ví dụ 1: ListView (2 of 4)** Bây giờ bạn mở **MainActivity.java** lên để viết code.

```
package selectionwidgets;
import android.app.ListActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;

public class ArrayAdapterDemo extends ListActivity {

    TextView selection;
    String[] items = { "this", "is", "a", "really",
                      "really2", "really3", "really4",
                      "really5", "silly", "list" };

    // next time try an empty list such as:
    // String[] items = {};
```



2. ListView

● Ví dụ 1: ListView (3 of 4)

@Override

```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.main);  
  
    setListAdapter(new ArrayAdapter<String>(  
        this,  
        android.R.layout.simple_list_item_1,  
        items));  
  
    selection=(TextView)findViewById(R.id.selection);  
}
```

 Selection listener

@Override

```
protected void onItemClick(ListView l, View v,  
                            int position, long id) {  
    super.onItemClick(l, v, position, id);  
    String text = " position:" + position + " " + items[position];  
    selection.setText(text);  
}  
  
}
```

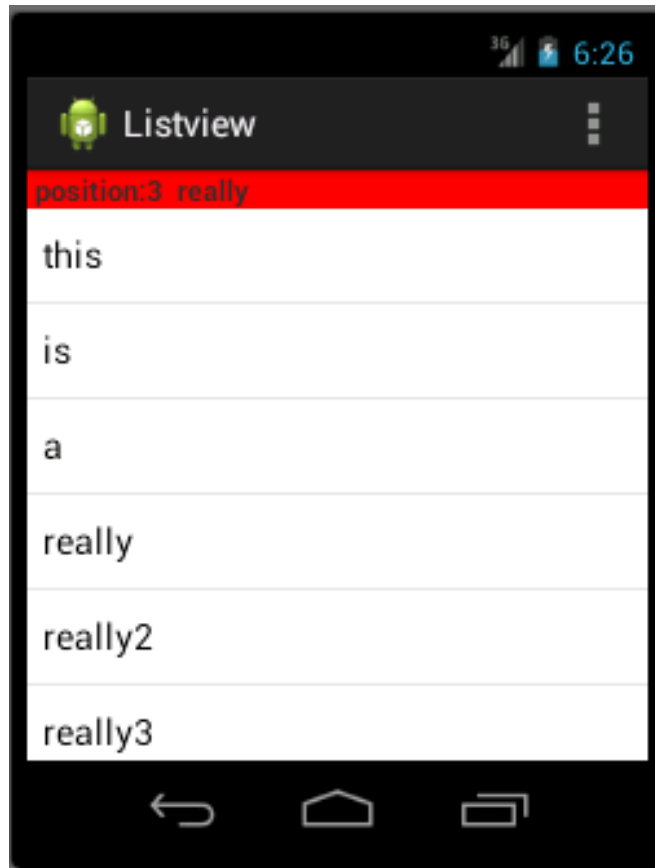
 List adapter

2. ListView

- **Giải thích ví dụ 1:**
- Ví dụ này dùng một số component dựng sẵn của Android
 - Trong XML layout ta gọi một widget *ListView* dựng sẵn với id ***android:id/list*** (trong đó: multiple lines, black /orange background, light-gray separator line, horiz/vert. scroll-bar)
 - Trong khi đặt cấu hình của ArrayAdapter ta dùng ***android.R.layout.simple_list_item_1***

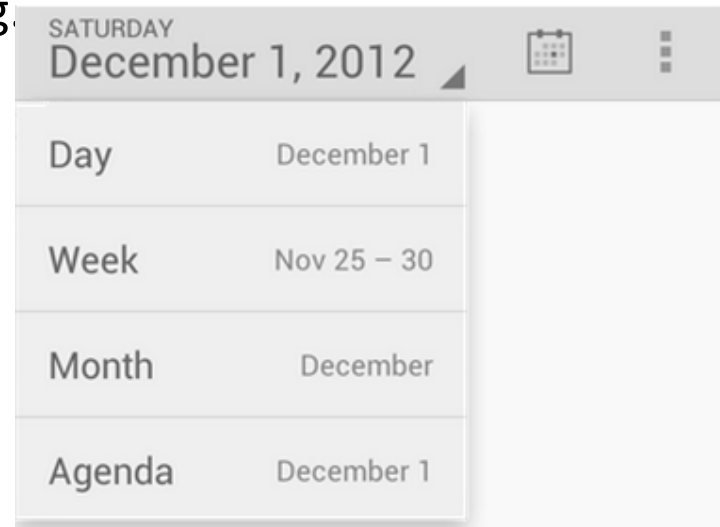
2. ListView

- Demo → ListView



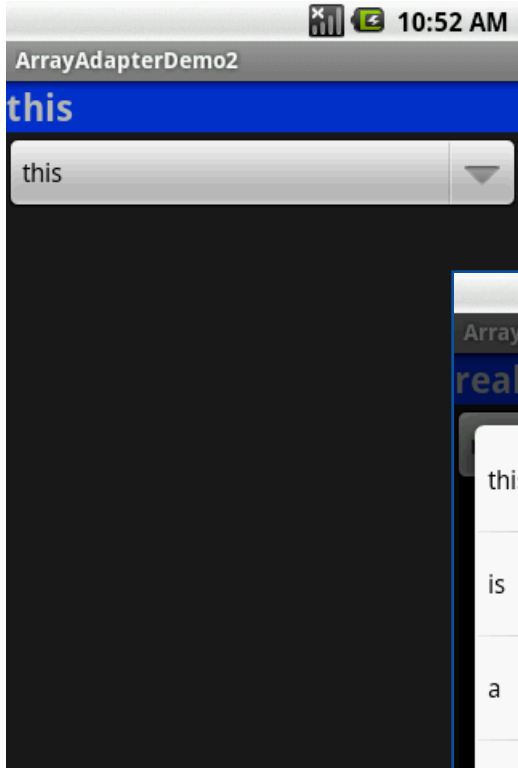
3. Spinner

- Trong Android, **Spinner** là widget chọn dạng *drop-down*.
- Spinner có cùng chức năng với ListView nhưng chiếm ít không gian hơn.
- Cũng như với ListView, ta dùng adapter để liên kết dữ liệu với Spinner, dùng hàm ***setAdapter()***
- Thêm một đối tượng listener để nhận chọn lựa từ danh sách (hàm ***setOnItemSelectedListener()***).
- Dùng ***setDropDownViewResource()*** để cung cấp resource ID của multi-line selection list view cần dùng.

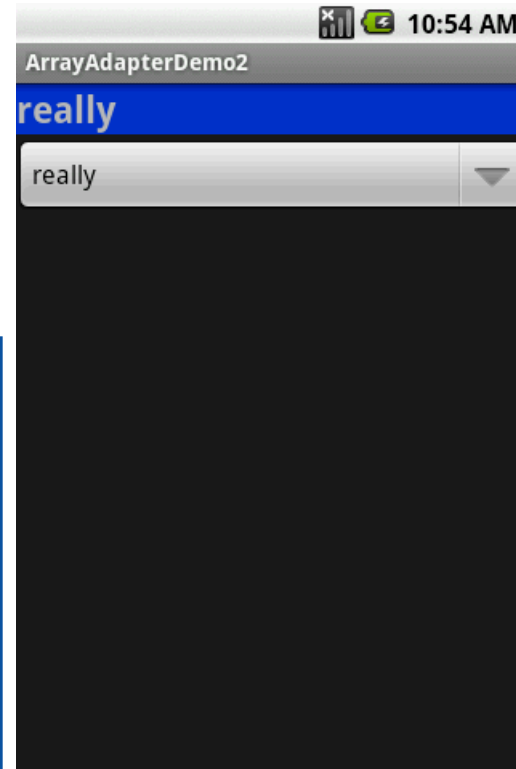
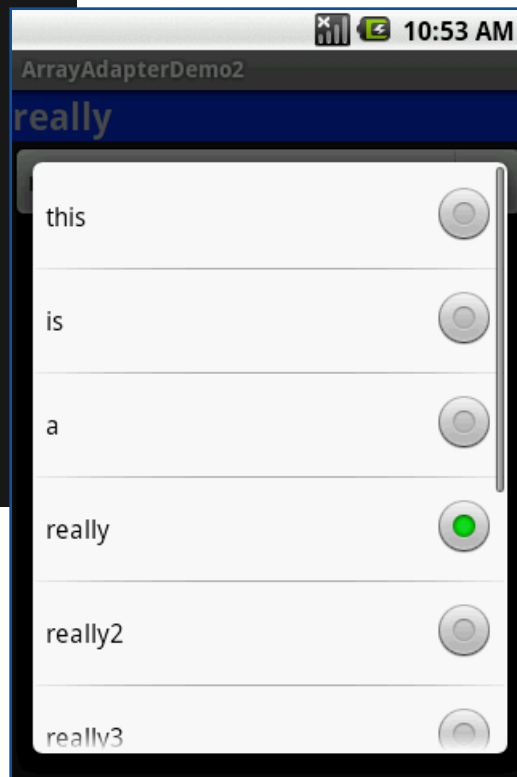


3. Spinner

- Ví dụ 2. Sử dụng Spinner



1. Click đây



3. Hiện thị giá trị chọn

2. Chọn một dòng tương ứng

3. Spinner

● Ví dụ 2. Sử dụng Spinner

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    android:id="@+id/myLinearLayout"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    xmlns:android="http://schemas.android.com/apk/res/android"
>
    <TextView
        android:id="@+id/selection"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:background="#ff0033cc"
        android:textSize="14pt"
        android:textStyle="bold"
    >
    </TextView>
    <Spinner
        android:id="@+id/spinner"
        android:prompt="gender"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content">
    </Spinner>
</LinearLayout>
```

3. Spinner

● Ví dụ 2. Sử dụng Spinner

```
package selectionwidgets;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.TextView;

public class ArrayAdapterDemo2 extends Activity
    implements AdapterView.OnItemClickListener {

    TextView selection;
    String[] items = { "this", "is", "a",
                      "really", "really2", "really3",
                      "really4", "really5", "silly", "list" };
}
```

3. Spinner

● Ví dụ 2. Sử dụng Spinner

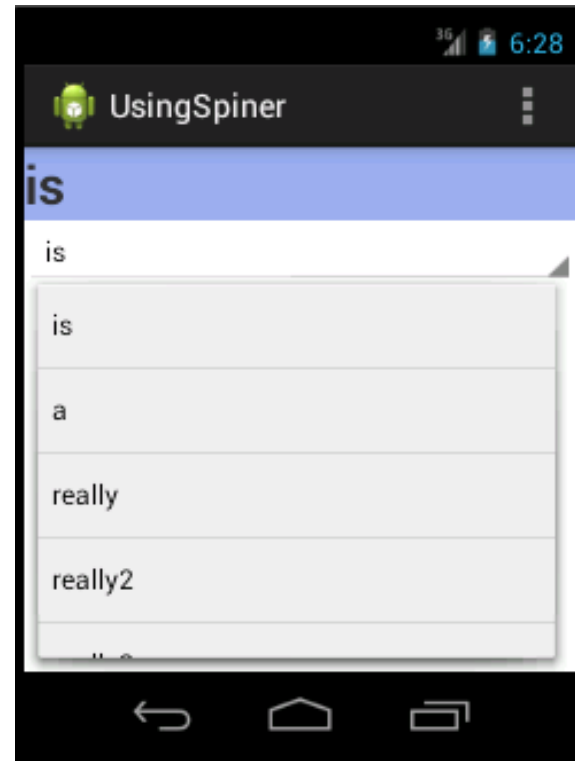
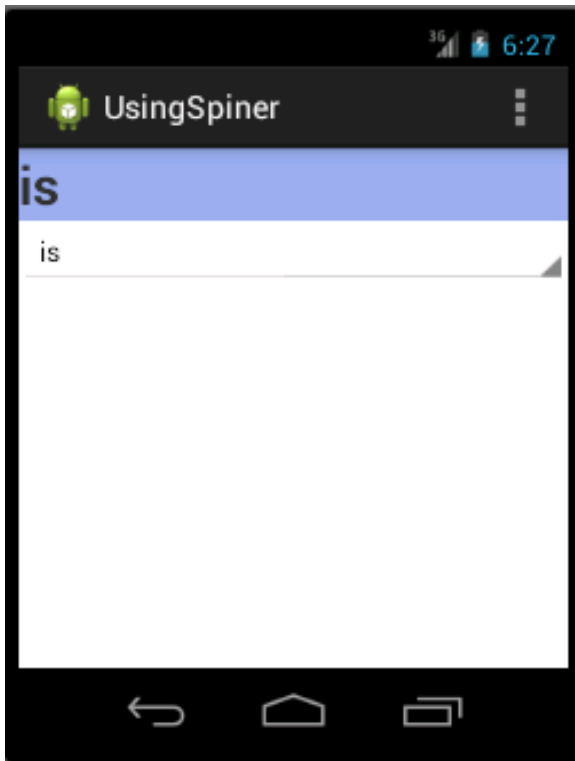
```
@Override
public void onCreate(Bundle icle) {
    super.onCreate(icle);
    setContentView(R.layout.main);

    selection = (TextView) findViewById(R.id.selection);

    Spinner spin = (Spinner) findViewById(R.id.spinner);
    spin.setOnItemSelectedListener(this);
    // bind array to UI control to show one-line
    ArrayAdapter<String> aa = new ArrayAdapter<String>(
        this, android.R.layout.simple_spinner_item, items);
    // showing the drop-down multi-line window
    aa.setDropDownViewResource(
        android.R.layout.simple_spinner_dropdown_item);
    // associate GUI spinner and adapter
    spin.setAdapter(aa);
}
// //////////////////////////////////////
public void onItemSelected(
    AdapterView<?> parent, View v, int position, long id) {
    selection.setText(items[position]);
}
public void onNothingSelected(AdapterView<?> parent) {
    selection.setText("");
}
}
```

3. Spinner

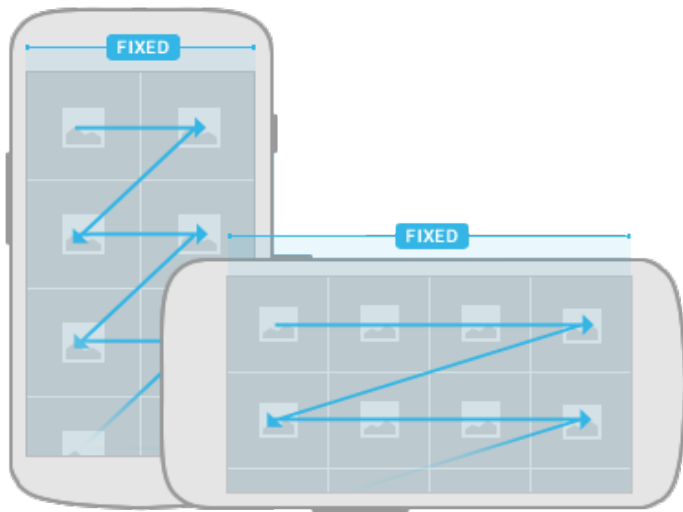
- Demo → Spinner



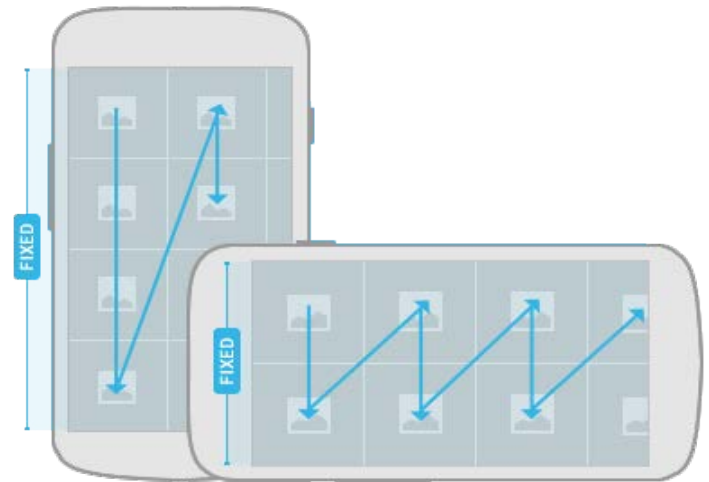
4. GridView

- **GridView**

- GridView là một ViewGroup hiển thị các phần tử trong một lưới chiều, có thể thiết kế thanh cuộn theo chiều dọc hoặc ngang. Thông thường một chiều sẽ được cố định, chiều còn lại sẽ thiết kế thanh cuộn.
- Các item được chèn tự động vào lưới bằng một ListAdapter.



Cố định chiều ngang, Cuốn theo chiều dọc



Cố định chiều dọc, Cuốn theo chiều ngang

4. GridView

- **GridView**
- Một vài thuộc tính để xác định số column và size là:
 - **android:numColumns** chỉ ra số cột sẽ hiển thị.
 - **android:verticalSpacing** khoảng trống giữa các item với lưới theo hàng.
 - **android:horizontalSpacing** khoảng trống giữa các item với lưới trong cột
 - **android:columnWidth** chỉ ra độ rộng của cột
 - **android:stretchMode** chỉ ra độ dẫn của cột nếu để thuộc tính *auto_fit* cho *android:numColumns*, thì độ rộng của cột sẽ tự động dẫn đều.

4. GridView

- Ví dụ 3 . GridView

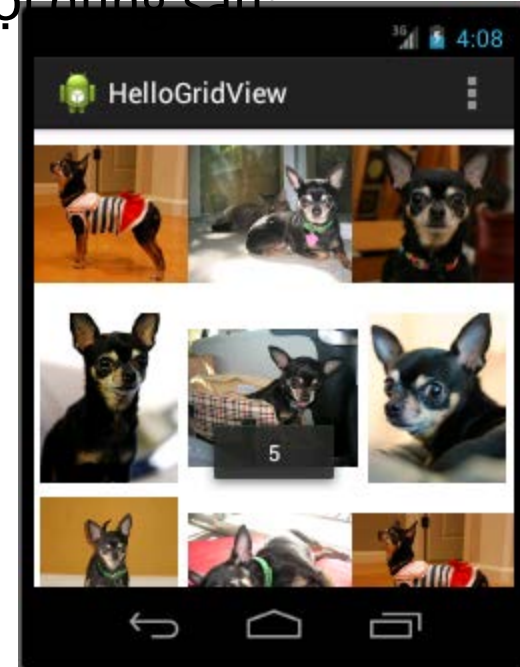
- Tạo một ứng dụng chứa các lưới hình ảnh như ở bên, khi chọn vào mỗi ảnh, một thông báo hiển thị vị trí hình ảnh sẽ hiển thị. Các bước tạo:

- Tạo một project: *HelloGridView*.

- Chèn ảnh vào thư mục res/drawable/

- Mỗi tệp res/layout/main.xml và chèn nội dung sau:

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



4. GridView

- Ví dụ 3 . GridView

- Mở tệp HelloGridView.java và chèn đoạn code sau vào phương thức onCreate()

@Override

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

```
  
    GridView gridview = (GridView) findViewById(R.id.gridview);  
    gridview.setAdapter(new ImageAdapter(this));
```

```
  
    gridview.setOnItemClickListener(new OnItemClickListener() {  
        public void onItemClick(AdapterView<?> parent, View v, int position, long id) {  
            Toast.makeText(MainActivity.this, "" + position,  
Toast.LENGTH_SHORT).show();  
        }  
    });  
}
```


4. GridView

● Ví dụ 3 . GridView

- Tạo một class tên ImageAdapter kế thừa BaseAdapter:

```
public class ImageAdapter extends BaseAdapter {  
    private Context mContext;  
    public ImageAdapter(Context c) { mContext = c; }  
    public int getCount() { return mThumbIds.length;}  
    public Object getItem(int position) { return null;}  
    public long getItemId(int position) { return 0; }  
    // create a new ImageView for each item referenced by the Adapter  
    public View getView(int position, View convertView, ViewGroup parent) {  
        ImageView imageView;  
        if (convertView == null) { // if it's not recycled, initialize some attributes  
            imageView = new ImageView(mContext);  
            imageView.setLayoutParams(new GridView.LayoutParams(85, 85));  
            imageView.setScaleType(ImageView.ScaleType.CENTER_CROP);  
            imageView.setPadding(8, 8, 8, 8);  
        } else {  
            imageView = (ImageView) convertView;  
        }  
        imageView.setImageResource(mThumbIds[position]);  
        return imageView;  
    }  
}
```

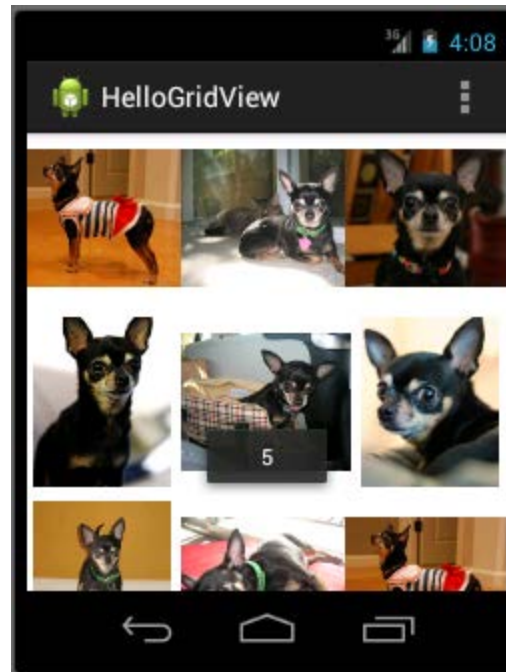
4. GridView

● Ví dụ 3 . GridView

```
// references to our images
private Integer[] mThumbIds = {
    R.drawable.sample_2, R.drawable.sample_3,
    R.drawable.sample_4, R.drawable.sample_5,
    R.drawable.sample_6, R.drawable.sample_7,
    R.drawable.sample_0, R.drawable.sample_1,
    R.drawable.sample_2, R.drawable.sample_3,
    R.drawable.sample_4, R.drawable.sample_5,
    R.drawable.sample_6, R.drawable.sample_7,
    R.drawable.sample_0, R.drawable.sample_1,
    R.drawable.sample_2, R.drawable.sample_3,
    R.drawable.sample_4, R.drawable.sample_5,
    R.drawable.sample_6, R.drawable.sample_7
};
}
```

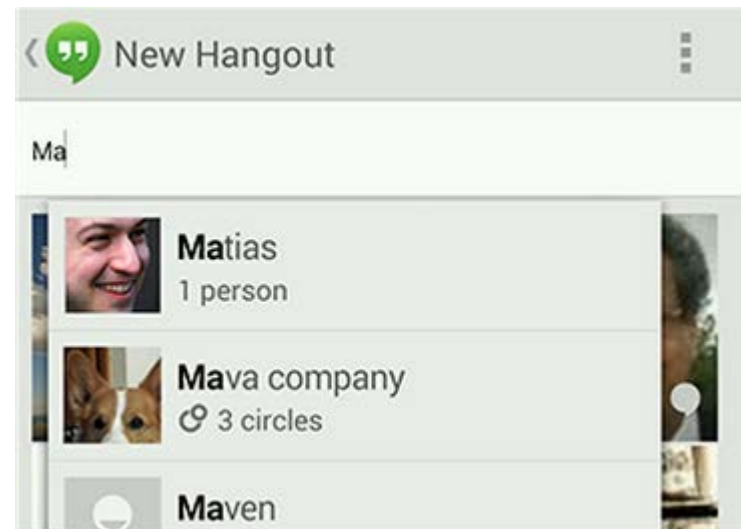
4. GridView

- Demo → Gridview



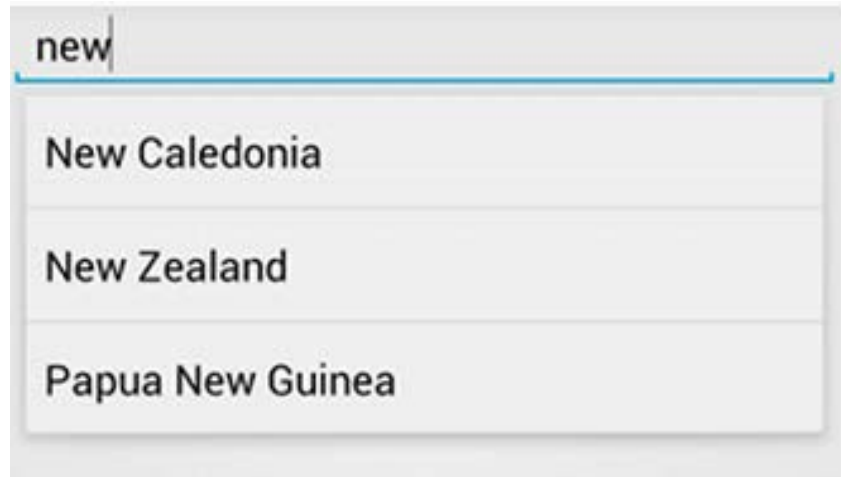
5. AutocompleteTextView

- Đối với các thiết bị di động, việc hỗ trợ nhập dữ liệu nhanh cho người sử dụng là điều rất cần thiết.
- Với **auto-completion**, trong khi người dùng gõ, phần text đã nhập được dùng làm tiết đầu tổ để lọc dữ liệu, so sánh phần text đã nhập với một danh sách các lựa chọn có thể.
- Từ phù hợp sẽ hiển thị ra trong danh sách lựa chọn đó, giống như Spinner, ta sẽ chọn từ cần lấy.
- Người sử dụng có thể nhập từ mới (không có trong danh sách) hoặc có thể lựa chọn từ ở trong danh sách hiển thị.



5. AutoCompleteTextView

- AutoCompleteTextView là một lớp con của EditText, vì vậy nó sẽ kế thừa một số thuộc tính của EditText như: color, font...
- Như hình dưới: ta chỉ cần nhập ký tự **new** đầu tiên, nó sẽ lọc ra các từ có ký tự đầu là **new**. Đây **không phải là Spinner** mà nó là **AutoCompleteTextView**.
- Các bước sau đây mô tả cách thiết lập một AutoCompleteTextView cung cấp các từ gợi ý từ một mảng có sẵn, sử dụng ArrayAdapter:



5. AutoCompleteTextView

- (1) Thêm thẻ AutoCompleteTextView vào layout.
- Trong `AutoCompleteTextView` thuộc tính ***android:completionThreshold*** chỉ ra số ký tự tối thiểu mà người dùng có thể nhập để hiển thị ra danh sách từ gợi ý tương tự.

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<AutoCompleteTextView
```

```
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:id="@+id/autocomplete_country"  
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:completionThreshold="3"/>
```

5. AutocompleteTextView

- (2) Định nghĩa mảng dữ liệu chứa từ gợi ý. Ví dụ tạo một mảng dữ liệu chứa tên các thành phố được định nghĩa ở tệp .xml trong thư mục (res/values/strings.xml):

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<resources>
```

```
<string-array name="countries_array">
```

```
<item>Afghanistan</item>
```

```
<item>Albania</item>
```

```
<item>Algeria</item>
```

```
<item>American Samoa</item>
```

```
<item>Andorra</item>
```

```
<item>Angola</item>
```

```
<item>Anguilla</item>
```

```
...
```

```
</string-array>
```

```
</resources>
```

5. AutocompleteTextView

- (3) Sử dụng đoạn code sau để hiển thị những từ gợi ý lên AutoCompleteTextView:

```
// Get a reference to the AutoCompleteTextView in the layout
```

```
AutoCompleteTextView textView = (AutoCompleteTextView)
```

```
findViewById(R.id.autocomplete_country);
```

```
// Get the string array
```

```
String[] countries = getResources().getStringArray(R.array.countries_array);
```

```
// Create the adapter and set it to the AutoCompleteTextView
```

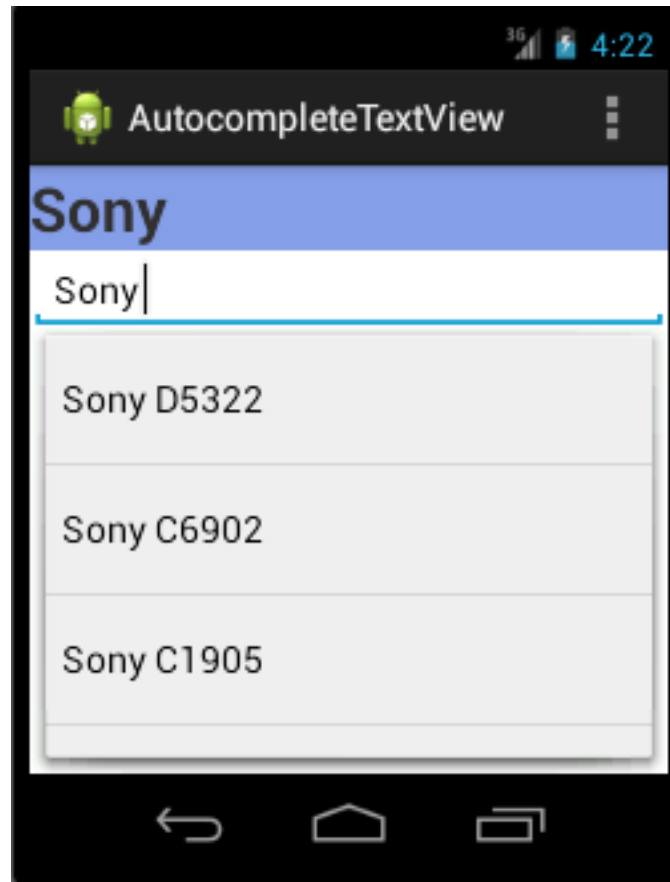
```
ArrayAdapter<String> adapter =
```

```
    new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1, countries);
```

```
textView.setAdapter(adapter);
```


5. AutocompleteTextView

- Demo → AutocompleteTextview



Tổng kết bài học

- Tổng quan về selection widget
- Listview
- Spinner
- Gridview
- Autocomplex TextView

Kết thúc!

