

LẬP TRÌNH ANDROID CĂN BẢN

Bài 4: Common widgets

Ths. Trần Xuân Thanh Phúc | Trường Đại học Công Nghiệp Thực Phẩm

Mục tiêu

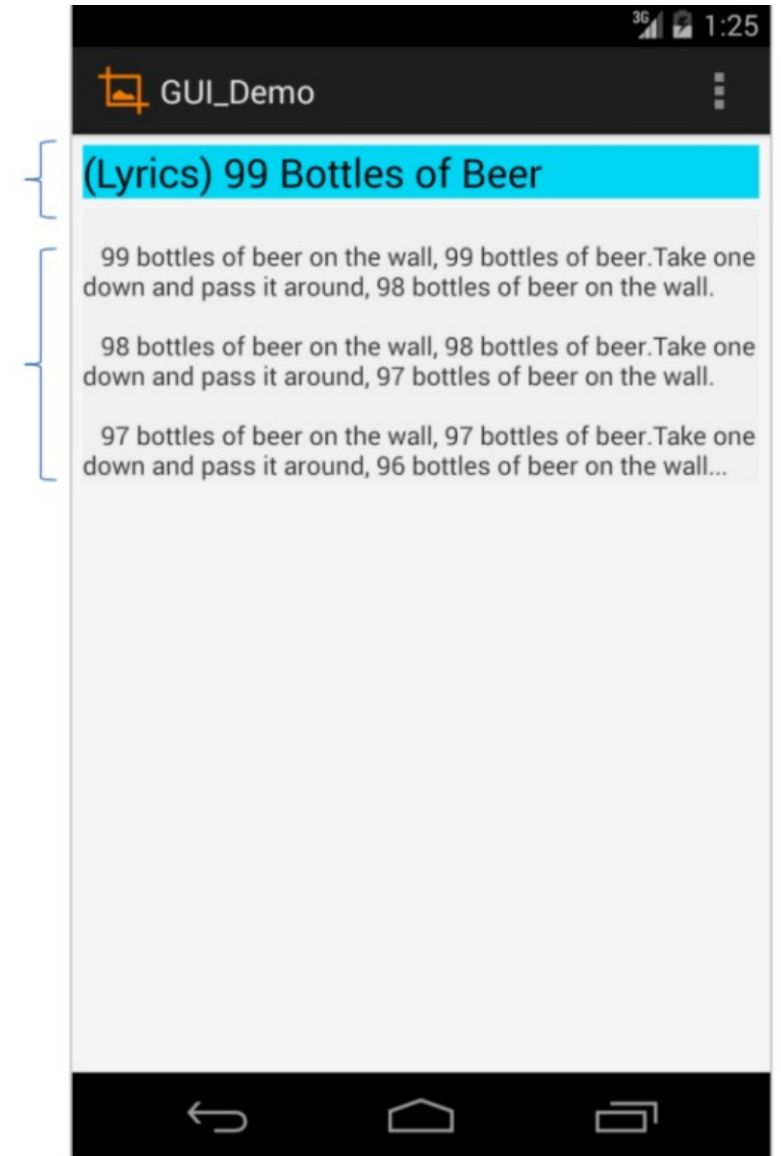
- Hiểu rõ về các widget thường dung nhất và vận dụng vào bài toán.

Nội dung

- Textview
- Button
- ImageView & ImageButton
- EditText
- Checkbox
- RadioButton
- ScrollView
- Soft Keyboard

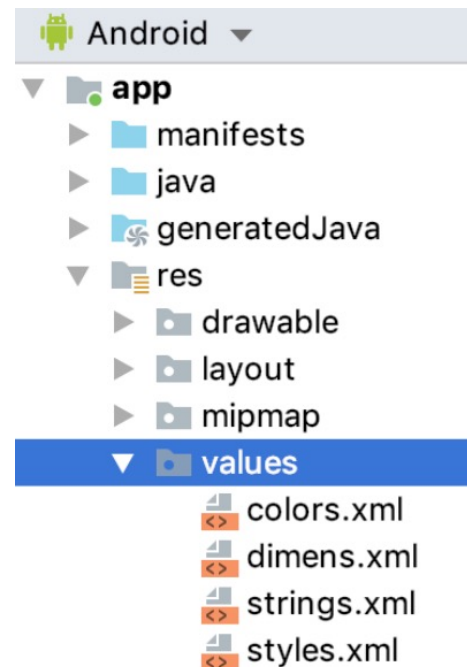
1. Textview

- Còn gọi là label, dùng để trình bày một đoạn văn bản
- Không chỉnh sửa được nội dung văn bản, nên không dùng để nhập liệu
- Có thể trình bày được các ký tự đặc biệt, ví dụ '\n' (xuống dòng)
- Có thể trình được nội dung HTML, định dạng bởi `Html.fromHtml("bold string")`



1. Textview - Các giá trị thường gặp

Thuộc tính	Ý nghĩa	Giá trị/Tham chiếu
text	Nội dung văn bản hiển thị	@string/... - strings.xml
textSize	Kích thước văn bản	@dimen/... - dimens.xml
textColor	Màu sắc văn bản	@color/... - colors.xml
textStyle	Kiểu văn bản	regular/bold/italic
background	Màu nền của textview	@color/... - colors.xml

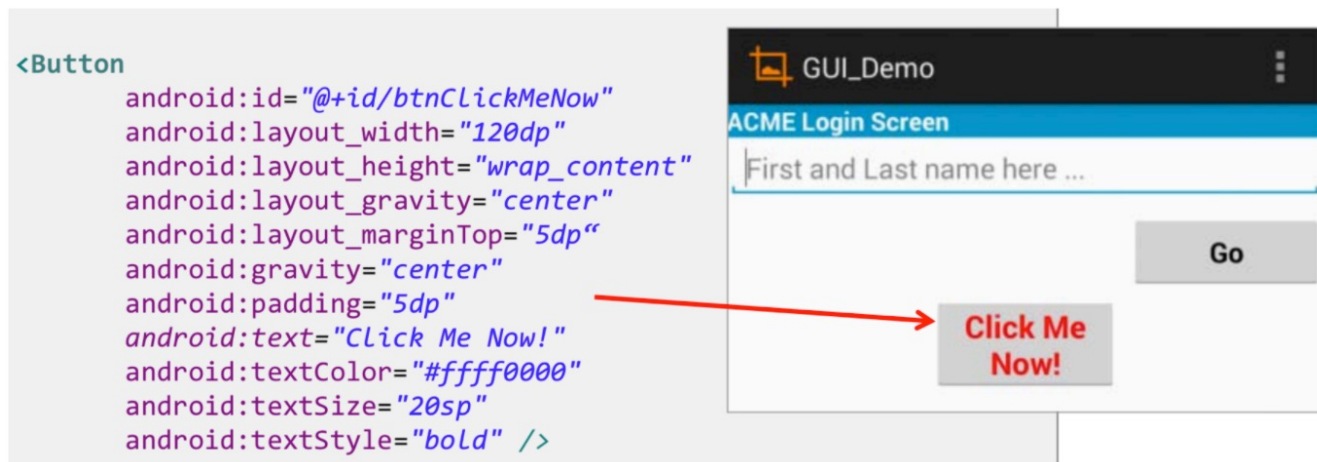


```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <color name="blue">#3498db</color>
  <color name="gray">#bdc3c7</color>
</resources>
```

```
<resources>
  <string name="app_name">HelloWorld</string>
  <string name="text_description">\n\t99 bottles of beer on the wall, 99 bottles of
  beer..</string>
</resources>
```

2. Button

- Là sub-class của TextView → có các thuộc tính giống TextView
- Cho phép người dùng click (tap) vào → sự kiện cơ bản: onClick
- Có thể thay đổi thuộc tính background của thông qua các file drawable.xml, thay đổi hành vi của button dựa trên các trạng thái (pressed, focused).



3. Imageview & ImageButton

- Giúp nhúng hình ảnh (png, jpg, gif...) vào ứng dụng
- Các thuộc tính **android:src** và **android:background** giúp thiết lập hình ảnh
- Hình ảnh được lưu trữ trong thư mục **res/drawable**. Các thư mục **drawable-mdpi**, **drawable-hdpi**, **drawable-xhdpi**... được tạo ra để chứa các hình ảnh tương ứng với các kích thước khác nhau của màn hình thiết bị



4. EditText

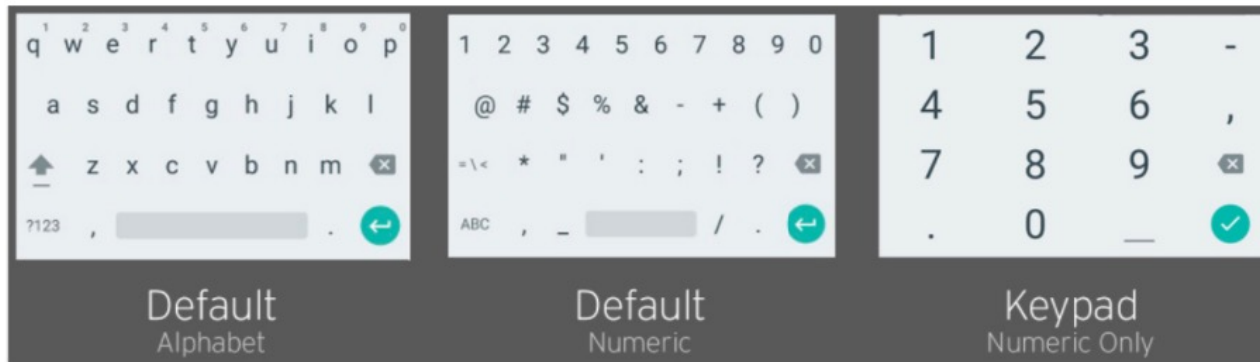
- EditText mở rộng từ TextView, cho phép người dùng nhập liệu
- Ngoài việc hiển thị plain text, EditText còn có thể định dạng văn bản theo HTML, dùng `Html.fromHtml(html_text)`
- Dùng `getText()` và `setText()` để get và set dữ liệu



4. EditText - InputType

- EditText có thể chấp nhận các chuỗi theo mẫu như: số, số điện thoại, ngày, giờ... thông qua **android:inputType**
- Có thể kết hợp 2 hay nhiều thuộc tính. Ví dụ: **textCapWords | textAutoCorrect** sẽ giúp nhập chuỗi sao cho viết hoa ký tự đầu tiên của mỗi từ, đồng thời nếu từ nhập bị sai chính tả sẽ tự động sửa lỗi.
- Bàn phím ảo (soft keyboard) cũng tự động điều chỉnh tương ứng với inputType.

4. Edittext - InputType



- Ⓐ "none"
- Ⓐ "text"
- Ⓐ "textCapCharacters"
- Ⓐ "textCapWords"
- Ⓐ "textCapSentences"
- Ⓐ "textAutoCorrect"
- Ⓐ "textAutoComplete"
- Ⓐ "textMultiLine"
- Ⓐ "textImeMultiLine"
- Ⓐ "textNoSuggestions"
- Ⓐ "textUri"
- Ⓐ "textEmailAddress"
- Ⓐ "textEmailSubject"
- Ⓐ "textShortMessage"
- Ⓐ "textLongMessage"
- Ⓐ "textPersonName"
- Ⓐ "textPostalAddress"
- Ⓐ "textPassword"
- Ⓐ "textVisiblePassword"
- Ⓐ "textWebEditText"
- Ⓐ "textFilter"
- Ⓐ "textPhonetic"
- Ⓐ "number"
- Ⓐ "numberSigned"
- Ⓐ "numberDecimal"
- Ⓐ "phone"
- Ⓐ "datetime"
- Ⓐ "date"
- Ⓐ "time"

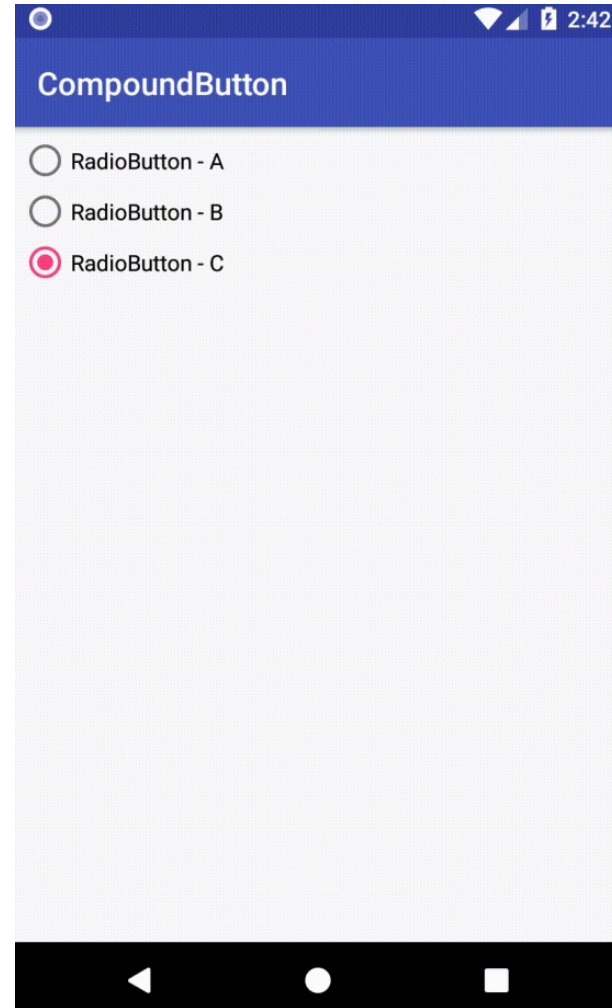
5. Checkbox

- CheckBox mở rộng từ CompoundButton tạo là một loại Button cho phép hiện thị hộp kiểm, hiện thị thông tin trạng thái checked hay unchecked, mã để tùy biến CheckBox
- Một số thuộc tính thường dùng
 - **android:checked** thiết lập trạng thái checked, unchecked với giá trị "true" hay "false"
 - **android:button** để gán Drawable vào View (vẽ trạng thái cho CheckBox, RadioButton ...)
 - **android:buttonTint** để gán màu Tint



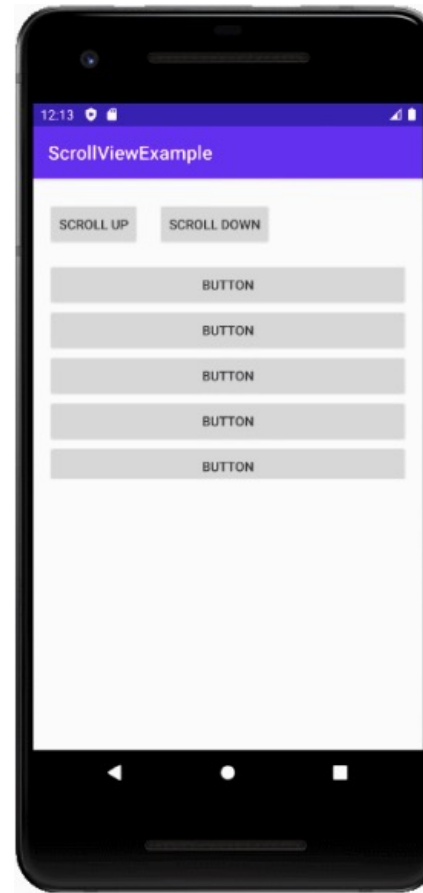
6. RadioButton

- RadioButton cũng là loại control biểu diễn trạng thái checked/unchecked.
- Thường RadioButton sẽ sử dụng cùng với RadioGroup, lúc đó sẽ có vài RadioButton bên trong RadioGroup và ở một thời điểm, người dùng chỉ có thể chọn một.
- Một số thuộc tính quan trọng
 - Để thay đổi layout RadioGroup theo hướng ngang hay đứng dùng thuộc tính **android:orientation** (vertical hoặc horizontal)
 - **android:buttonTint** : thay đổi màu sắc



7. ScrollView

- **ScrollView** là một kiểu **Layout**, nó là một bộ chứa (container) hình chữ nhật có thanh cuộn thẳng đứng và có thể chứa một thành phần khác có kích thước lớn hơn nó. Tương tự với **ScrollView**, **HorizontalScrollView** là một bộ chứa với thanh cuộn nằm ngang.



ScrollView



HScrollView

8. Soft Keyboard

- Ẩn / Hiện soft keyboard trên 1 view cụ thể (thông thường là EditText và các widget dùng để nhập văn bản)

```
public void showSoftKeyboard( View view) {  
    if (view.requestFocus()) {  
        InputMethodManager imm = (InputMethodManager)  
            getSystemService(Context.INPUT_METHOD_SERVICE);  
        imm.showSoftInput(view, InputMethodManager.SHOW_IMPLICIT);  
    }  
}
```

```
public void hideSoftKeyboard(View view) {  
    InputMethodManager imm = (InputMethodManager)  
        getSystemService(Context.INPUT_METHOD_SERVICE);  
    imm.hideSoftInputFromWindow(view.getWindowToken(), 0);  
}
```

- Thêm “Done” key

– XML

```
<EditText android:imeOptions="actionDone" />
```

– Java

```
myEditText.setImeOptions(EditorInfo.IME_ACTION_DONE);
```

8. Soft Keyboard

- Hiện keyboard khi vừa mở activity (Manifest)

```
<activity android:name=".MainActivity"  
    android:windowSoftInputMode="stateVisible">  
</activity>  
# stateUnchanged, stateHidden
```

- Điều chỉnh UI để dành không gian cho keyboard

```
<activity android:name=".MainActivity"  
    android:windowSoftInputMode="adjustResize">  
</activity>  
#adjustPan, adjustUnspecified
```

- Kết hợp thuộc tính bằng toán tử |

LẬP TRÌNH ANDROID CĂN BẢN

Kết thúc 🤗

Ths. Trần Xuân Thanh Phúc | Trường Đại học Công Nghiệp Thực Phẩm