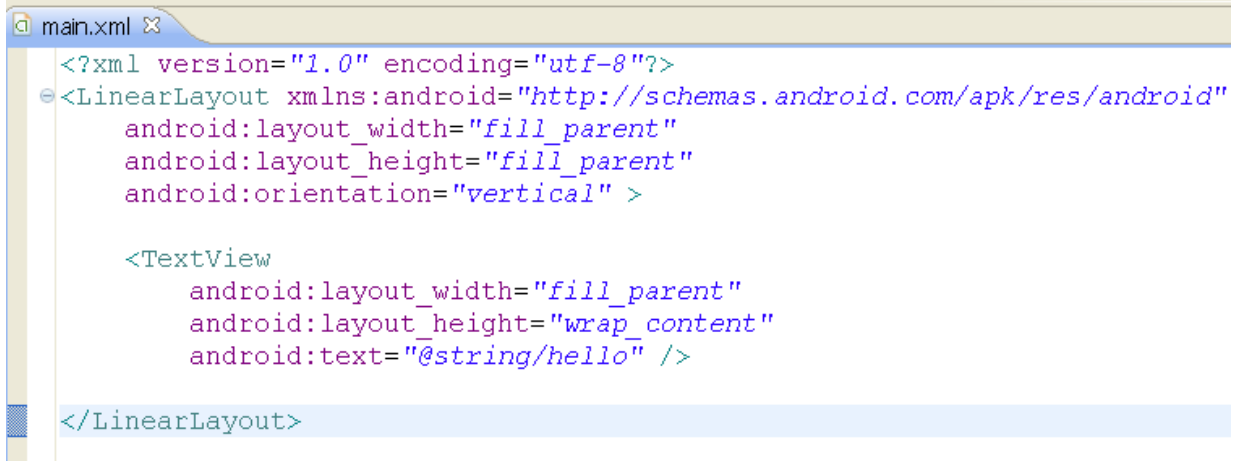


## Bài 2- Chương trình đầu tiên (Tăng / Giảm một con số)

Yêu cầu: phải thực hiện bài 1 để có được project

Đạt được: chương trình có 1 thông báo xuất kết quả ban đầu là 0. 1 Nút tăng nhấn vào sẽ tăng kết quả lên 1. 1 nút giảm nhấn vào sẽ giảm kết quả xuống.

### 1. Mở file main.xml trong đường dẫn myproject1/res/layout



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

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello" />

</LinearLayout>
```

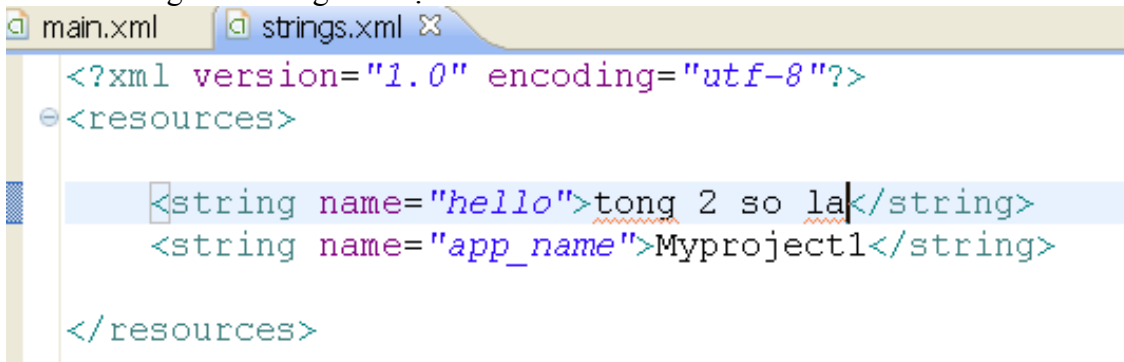
Ta thấy bọc bên ngoài là một LinearLayout (layout theo kiểu từng dòng) . Có thuộc tính:

- android:layout\_width và android:layout\_height : chiều rộng và cao của đối tượng. Nó có 3 giá trị là:
  - fill\_parent : to bằng cha của nó. (được đổi tên thành match\_parent từ API level 8)
  - match\_parent: giống fill\_parent
  - wrap\_content: to vừa đủ nội dung mà nó chứa.
- android:orientation: định hướng (vertical: thẳng đứng, horizontal: nằm ngang)

Bên trong là một TextView có thuộc tính là:

Android:text: là nội dung của TextView, ở đây có thể đánh 1 chuỗi trực tiếp hoặc dùng bằng tài nguyên như hình. Tức là tạo ra 1 chuỗi với tên là hello. Sau đó định nghĩa nội dung của hello trong file strings.xml trong thư mục values.

### 2. Mở file strings.xml trong thư mục values và sửa như sau:



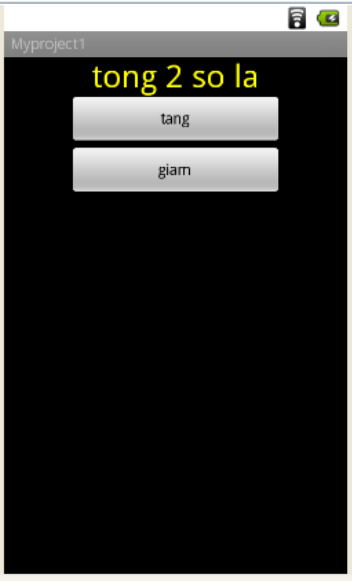
```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="hello">tong 2 so la</string>
    <string name="app_name">Myproject1</string>

</resources>
```

Chạy chương trình và bạn sẽ hiểu main.xml dùng tài nguyên từ strings.xml như thế nào.

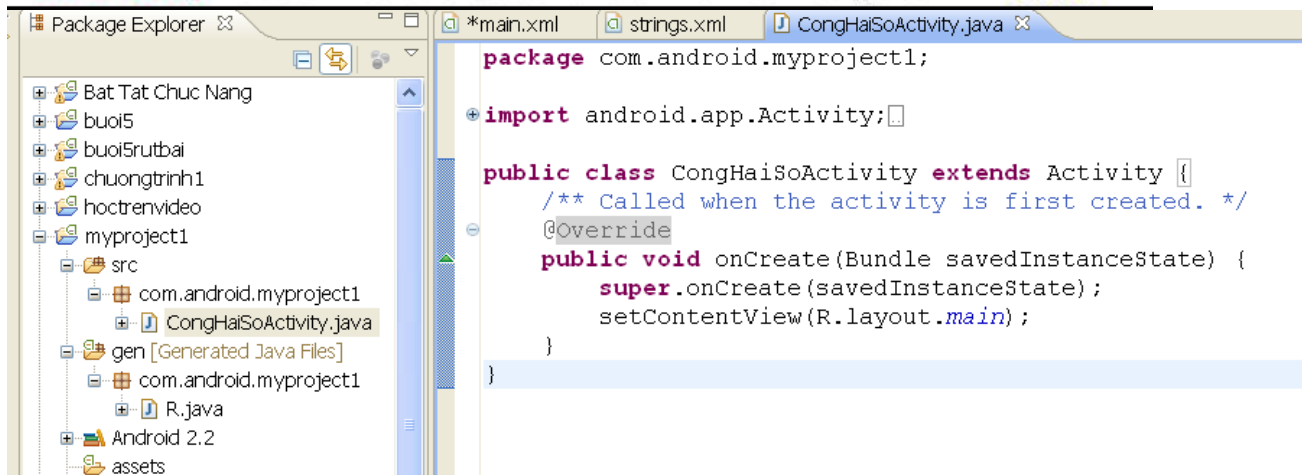
3. Mở file main.xml , sửa nội dung nằm trong LinearLayout như sau:

<pre> main.xml x strings.xml &lt;TextView     android:id="@+id/tvKetQua"     android:layout_width="fill_parent"     android:layout_height="wrap_content"     android:textSize="30dp"     android:textColor="#FFFF00"     android:layout_gravity="center"     android:gravity="center"     android:text="@string/hello" /&gt;  &lt;Button     android:id="@+id/btnTang"     android:layout_width="200dp"     android:layout_height="wrap_content"     android:layout_gravity="center"     android:text="tang" /&gt;  &lt;Button     android:id="@+id/btnGiam"     android:layout_width="200dp"     android:layout_height="wrap_content"     android:layout_gravity="center"     android:text="giam" /&gt; </pre>	<p>Và ta sẽ có giao diện như sau:</p> 
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------

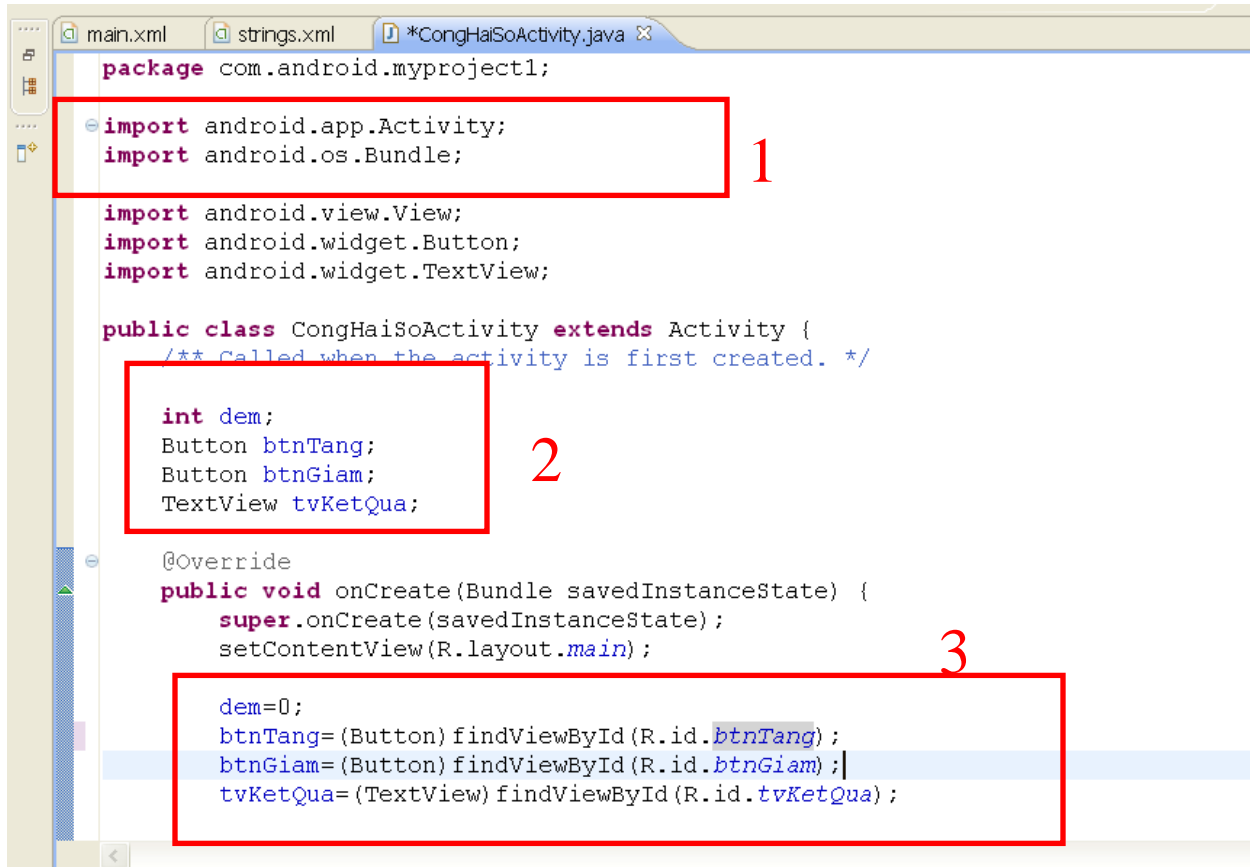
Các thuộc tính mới:

- Android:id : đặt id cho đối tượng. Bắt đầu phải là @+id/ rồi đến một cái tên (phần lập trình sẽ dùng cái tên đó để ánh xạ đến đối tượng)
- Chiều rộng, cao của layout có thể dùng đơn vị tính như dp,px, in, mm,pt, sp (dp đơn vị trừu tượng dựa vào mật độ vật lý của màn hình, 1dp tương đương 1px trên màn hình 160dpi)
- Android:layout\_gravity: canh nó (trên,dưới...) so với cha nó.
- Android:gravity: canh (trên dưới...) nội dung bên trong nó.

4. Tiếp theo ta bắt đầu lập trình. Double Click vào file CongHaiSoActivity.java trong thư mục myproject1/src/com.android.myproject1.



5. Bổ sung thêm các phần mã lệnh như sau:



- Phần 1: Nhập thư viện cần thiết cho các đối tượng.
- Phần 2: Tạo trước các biến, đối tượng.
- Phần 3: Ánh xạ từ giao diện vào các đối tượng.

6. Phát triển tiếp bằng cách bắt sự kiện Click cho 2 nút. Chú ý: khi ta gõ tendoituong rồi dấu “.” thì eclipse sẽ hỗ trợ bằng cách hiện ra hộp thoại liệt kê các thuộc tính, phương thức... của đối tượng đó, hoặc ta có thể dùng Ctrl+Space để hiện, hãy tận dụng nó để viết mã nhanh hơn như sau (xem hình) gõ đến View. hộp thoại hiện ra hãy chọn vào View.OnClickListener() để hoàn tất câu lệnh:

```
btnGiam=(Button) findViewById(R.id.btnGiam);
tvKetQua=(TextView) findViewById(R.id.tvKetQua);
```

```
btnTang.setOnClickListener(new View.)
```

public static interface  
**View.OnClickListener**  
android.view.View.OnClickListener  
Known Indirect Subclasses  
[CharacterPickerDialog](#), [KeyboardView](#),  
[QuickContactBadge](#)

#### Class Overview

Interface definition for a callback to be invoked when a view is clicked.

**View.OnClickListener() Anonymous Inner Type - android**  
● BaseSavedState(Parcel source) - android.view.View  
● BaseSavedState(Parcelable superState) - android.vi  
● MeasureSpec() - android.view.View.MeasureSpec  
① View.onCreateContextMenuListener() Anonymous  
① View.onFocusChangeListener() Anonymous Inner  
① View.onKeyListener() Anonymous Inner Type - an  
① View.onLongClickListener() Anonymous Inner Typ  
① View.onTouchListener() Anonymous Inner Type -

Press 'Tab' from proposal table or click for focus

Press 'Ctrl+Space' to show Template Proposals

Câu lệnh sau khi hoàn tất sẽ như sau:

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

    dem=0;
    btnTang=(Button) findViewById(R.id.btnGiam);
    btnGiam=(Button) findViewById(R.id.btnGiam);
    tvKetQua=(TextView) findViewById(R.id.tvKetQua);

    btnTang.setOnClickListener(new View.OnClickListener() {

        public void onClick(View v) {
            // TODO Auto-generated method stub

        }

    })
}
```

Chú ý: Bạn để ý thấy mã phát sinh vẫn còn thiếu 1 dấu “;” ; “nhớ bổ sung thêm.

Bên dưới phần ghi chú //TODO Auto-generated method stub là những gì sẽ làm khi nút btnTang được nhấn.

- Bạn làm tương tự cho nút btnGiảm và viết lệnh bên trong 2 nút để tăng/giảm biến đếm và dùng hàm setText để gán kết quả cho textview. Mã lệnh cuối cùng như sau:

```
main.xml | strings.xml | CongHaiSoActivity.java
dem=0;
btnTang=(Button)findViewById(R.id.btnTang);
btnGiảm=(Button)findViewById(R.id.btnGiảm);
tvKetQua=(TextView)findViewById(R.id.tvKetQua);

btnTang.setOnClickListener(new View.OnClickListener() {

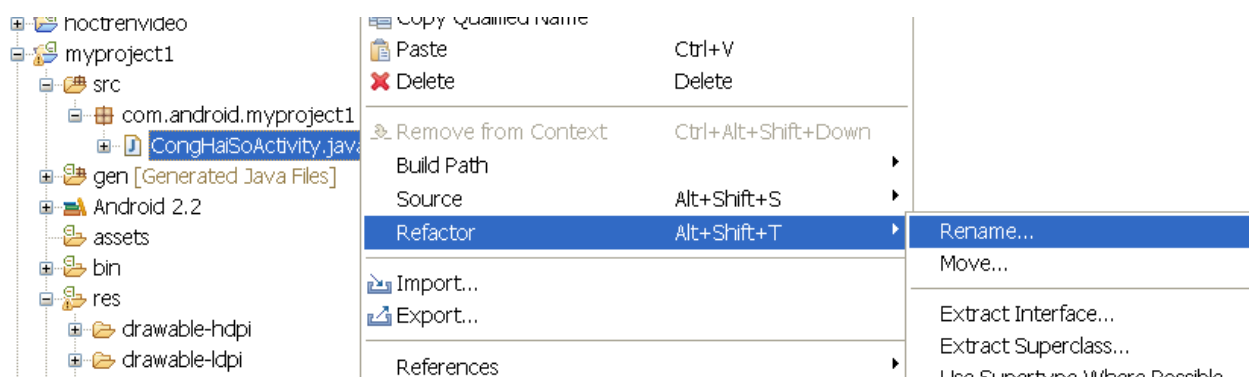
    public void onClick(View v) {
        // TODO Auto-generated method stub
        dem++;
        tvKetQua.setText("Ket qua la " + dem);
    }

});
btnGiảm.setOnClickListener(new View.OnClickListener() {

    public void onClick(View v) {
        // TODO Auto-generated method stub
        dem--;
        tvKetQua.setText("Ket qua la " + dem);
    }

});
```

- Đến đây bạn có thể chạy chương trình để thấy kết quả.
- Chú ý, bạn thấy rằng chuỗi mặc định của chương trình là “tong 2 so la” không đúng với chương trình, bạn có thể dễ dàng sửa nó trong file string.xml. Tên Activity là CongHaiSoActivity cũng không đúng. Bạn có thể sửa tên Activity như sau: Click phải file .java/Refactor/Rename:



Trong hộp thoại Rename Compilation Unit gõ vào tên mới: TangGiamMotSoActivity, check 3 dấu chọn đầu tiên Next/Next/Finish

